Proceedings of the 2010 International Conference on Computational and Mathematical Methods in Science and Engineering

Almería (Andalucía), Spain

June 26-30 2010

Editor:
J. Vigo-Aguiar (Spain)

Associate Editors:
**H. Adeli (USA), J. A. López-Ramos (Spain), S. Oharu (Japan),
J. Ranilla (Spain), J. Rosenthal (Switzerland), N. Stollenwerk (Germany),
Ezio Venturino (Italy) and Bruce A. Wade (USA)**

ISBN 978-84-613-5510-5

Register Number: 09/9851

Front Cover: “The Mathematician”, Diego Rivera (1918)

Printed on acid-free paper

Preface:

We are honoured to bring you this collection of articles and extended abstracts from the *"**10th International Conference on Computational and Mathematical Methods in Science and Engineering"*** (CMMSE 2010), held at Almería, Spain, from 26 to June 30, 2010.

Society is a human product plenty of works, product of the human action and Mathematics is, probably, one of these most wonderful works. Mathematics arises as a response to problematic situations that may or not have a mathematical root and the task of those dedicated to mathematics is to find answers to this type of questions. Mathematics is a dynamic organization since its procedures generate new problems and appeal to new results that in turn lead to new results that may tackle and propose new questions. Traditionally, Engineering has got strong connections with basic sciences in general and, in particular, with Mathematics, being a fundamental tool in all processes of analysis and calculus that an engineer has to carry out. In the last decades, the idea of Mathematics as a tool whose use allows to create models and solve real problems has been strengthened at the same time that research on many fields of Science and Engineering has growth rapidly due to interaction of these disciplines and computational and mathematical methods. Challenges that Science and Engineering face of are so complex that only an interdisciplinary relation, where Mathematics plays a principal role, will let to solve them. On the other hand, the influence of a general computerization of a part of our culture is leading to a great increasing of interest in computational methods. Computation has joint as a third crucial component to the two classical elements of the scientific method, experimentation and theory. Computing that only a few years ago was intractable today is carried out routinely. Many people expect to control size and complexity with the help of more powerful computers but this is a vain hope without the existing of an adequate development of Mathematics. CMMSE aims to detach how these computational and mathematical methods are, both together, crucial in the development of many disciplines.

CMMSE 2010 is a forum where experts of many different scientific fields present their latest advances and share ideas and experiences in order to explore new directions in Science and Engineering. CMMSE 2010 special sessions represent some of these emerging disciplines: from differential equations treated from different points of view, with applications to propagation of sound or heat, or electrodynamics to Mathematical Biology; from signal or image processing to from Computational Chemistry or Information Theory. These special sessions involve numerical solution of differential equations, mathematical models in artificial intelligence, computational science education, algorithms and computation for complex networks, bio-mathematics, computational chemistry, asymptotic preserving methods for kinetic and hyperbolic equations, numerical solutions of PDE's error estimation and reliability, applications of

algebra to cryptography and coding theory, high performance computing, sampling theory and meshfree methods, orthogonal polynomials and applications and COMSOL: multiphysics for modelling.

Today the resolution of scientific problems is unthinkable without High performance computing techniques. For second year, we have the pleasure to work with the Spanish Network CAPAP-H "High Performance Computing on Heterogeneous Parallel Architectures." We would like to give a special mention to José Ranilla, Esther Garzón and Enrique S. Quintana-Ortí for their fabulous and very organized work.

An essential issue in the society of information that we are living in is privacy and integrity of communications. We are grateful to the managers of the Spanish Network MatSI for their collaboration in the promotion and organization of the mini symposium “Applications of Algebra to Cryptography and Coding Theory”. We also would like to mention Consuelo Martinez for her support in this organization.

We would like to thank the plenary speakers for their excellent contributions in research and leadership in their respective fields. We express our gratitude to special session organizers and to all members of the Scientific Committee, who have been a very important part of the conference, and, of course, to all participants.

These four volumes contain all the results of the conference. For a question of style, volumes I, II and III contain the articles written in LaTeX and volume IV contains the articles written in Word and short-abstracts.

We cordially welcome all participants. We hope you enjoy this conference.

Almería, Andalucía, Spain, June 26, 2010

J. Vigo-Aguiar, H. Adeli, Juan A. López-Ramos, S. Oharu, J. Ranilla,
J. Rosenthal, N. Stollenwerk, Ezio Venturino, Bruce Wade

Acknowledgements:

We would like to express our gratitude to our sponsors:

- Addlink/COMSOL
- Departamento de Matemática Aplicada de la Universidad de Salamanca,
- Facultad de Ciencias Experimentales de la Universidad de Almería,
- Hanscan Spain, S.A.,
- Junta de Andalucía,
- Ministerio de Ciencia e Innovación,
- Universidad de Almería.

Finally, we would like to thank all of the local organizers:

J. Peralta (Universidad de Almería), and J. Ranilla, P. Alonso, (Universidad de Oviedo) for their support to make possible the conference. M.T. Bustos, A. Fernández (Universidad de Salamanca) for his invaluable help in the LaTeX and database preparation

CMMSE 2010 Plenary Speakers :

- Hojjat Adeli, Ohio State University, USA
- S. Jin, University of Wisconsin Madison, USA
- Shinnosuke Oharu, Chuo University, Tokyo, Japan
- Joachim Rosenthal, University of Zurich, Switzerland
- Ian P. Hamilton, Wilfrid Laurier University, Canada
- Denis Trystam, Grenoble Institute of Technology, France
- Guido Vanden Berghe, University of Gent, Belgium
- Ezio Venturino, Dipartimento di Matematica, University Torino, Italy
- John Whiteman, Brunel University, UK,
- Yang Chen, Imperial College London, UK

The organization of the "Minisymposium on High Performance Computing applied to Computational Problems in Science and Engineering" was part of the activities of the Spanish network "CAPAP-H 2: Red de Computación de Altas Prestaciones sobre Arquitecturas Paralelas Heterogéneas", supported by the Spanish "Ministerio de Innovación y Ciencia" (Acciones Complementarias TIN2009-08058-E).

The organization of the Minisymposium “Applications of Algebra to Cryptography and Coding Theory” has been supported partially in its organization and financial support with the Spanish network “Matemáticas en la Sociedad de la Información”, supported by Ministerio de Innovación y Ciencia (Acciones Complementarias MTM2008-03268-E).

Contents:

Volume I

Contents:

Volume II

Contents:

Volume III

Contents:

Volume IV

Short Abstracts

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Inverses of regular Hessenberg matrices

J. Abderramán Marrero[1] and Venancio Tomeo[2]

[1] *Department of Mathematics Applied to Information Technologies, Telecommunication Engineering School, U.P.M. Tech. University of Madrid*

[2] *Department of Algebra, School of Statistics, U.C.M. University Complutense*

emails: jc.abderraman@upm.es, tomeo@estad.ucm.es

Abstract

A new proof of the general representation for the entries of the inverse of any unreduced Hessenberg matrix of finite order is found. Also this formulation is extended to the inverses of reduced Hessenberg matrices. Those entries are given with proper Hessenbergians from the original matrix. It justifies both the use of linear recurrences for such computations and some elementary properties of the inverse matrix. As an application of current interest in the theory of orthogonal polynomials on the complex plane, the resolvent matrix associated to a finite Hessenberg matrix in standard form is calculated. The results are illustrated with two examples on the unit disk.

Key words: General orthogonal polynomials, Hessenberg matrix, Hessenbergian, inverse matrix, lower semiseparable (plus diagonal) matrix, resolvent matrix
MSC 2000: 11B83, 15A09, 15A15, 33C45, 47A10, 65F05

1 Introduction

The significance of matrix inversion in many parts of science and engineering and the methods used for its resolution are well known. The Cayley formula for the entries of the inverse matrix in terms of the adjoint matrix involves determinants. Both the computation and the expansion as a sum of products of these determinants present difficulties. These problems can be avoided by taking advantage of the special structure of certain matrices, for example tridiagonal, band, or Hessenberg matrices, to develop less costly algorithms or to identify properties invariant under matrix inversion.

In this direction, algorithms for the inversion of unreduced symmetric tridiagonal matrices were introduced in [3]. These algorithms were generalized to unreduced Hessenberg matrices in [10] and to banded unreduced matrices in [21]. In all of them the entries of the matrix inverse were represented as a product of two linear recurrences. The relation between certain elements of the inverse matrix of an unreduced Hessenberg

 ISBN 13: 978-84-613-5510-5

matrix and the product of two linear recurrences was proven in [6]. These recurrences were obtained from a closed formula for the entries on and above the diagonal of the inverse matrix of a lower Hessenberg matrix in terms of the Hessenbergians, [20], of its proper principal submatrices. In parallel, the low rank properties of submatrices of the inverse matrices of tridiagonal and banded matrices were outlined in [1], based on explicit representations of their minors. The results of [1, 6] were closely related with the nullity theorem. The subsequent development of this theorem and its implications for the invariance of low rank properties in the inversion of semiseparable matrices, [4, 7, 8, 14, 19], have dominated research up to the present time. A first closed and general representation for all entries of the inverse of any unreduced Hessenberg matrix was given by one of the authors, V. Tomeo, in [17]. Later, analogous expressions are obtained in [22]. In addition, there is an abundant literature related to general or specialized algorithms for the inverse of structured matrices. These only work for unreduced Hessenberg matrices. Recent algorithms for the inversion of Hessenberg matrices can be found in [2, 5] .

Without loss of generality we work with upper Hessenberg matrices. The compact expression for the entries below and on the diagonal is straightforward when using the Cayley formula and the Sylvester theorem on determinants,

$$(H^{-1})_{i,j} = \frac{A_{j,i}}{\det\ H} = (-1)^{i+j} \left(\prod_{k=0}^{i-j-1} h_{i-k,i-k-1} \right) \frac{\det\ H_{j-1} \det H_{n-i}^{(i)}}{\det H} \qquad (1)$$

The submatrix H_{j-1} is the left principal one of order $j-1$. The submatrix $H_{n-i}^{(i)}$ is the right principal one of order $n-i$, which begins in the $i+1$-th row and column and finishes in the n-th row and column. This formula is equivalent those given in [6] for the entries on and above the diagonal, $i \leq j$, for the inverse matrix of a lower Hessenberg matrix.

The validity of the representation in closed form for all entries of the inverse matrix H^{-1}, in terms of proper Hessenbergians given in [17, 22] for unreduced Hessenberg matrices is extended here to the reduced case. In addition, a new and more compact proof is introduced. This class of expressions allows us to solve for all the entries of the matrix using homogeneous linear recurrences, [11, 12], with well defined coefficients for each Hessenberg matrix. This approach has been applied in the case of tridiagonal matrices. A solution for the elements of the inverse matrix using a set of linear recurrences was introduced in [9]. A more sophisticated method was given in [13], where the solutions of second order linear difference equations were used in a boundary value problem. Thus, a compact representation for the inverse matrix of any unreduced tridiagonal matrix was obtained via combinatorial expressions, equivalent to the Leibniz formula for determinants.

In Section 2 we introduce a new proof for the representation of all entries of the inverse matrix of any unreduced Hessenberg matrix, in terms of proper Hessenbergians. The representation is extended to reduced Hessenberg matrices, although we must consider the avoidable indeterminacies that could arise. Section 3 is devoted to the linear recurrences involved in the computation of Hessenbergians and recalls some of

 ISBN 13: 978-84-613-5510-5

the elementary properties of the inverse of a Hessenberg matrix. As an interesting application of our results, in Section 4 a closed formula is given for the elements of the finite sections of the resolvent matrix associated to any sequence of monic orthogonal polynomials on a bounded region of the complex plane. It is illustrated with two examples on the unit disk.

2 Inverses of regular Hessenberg matrices

To begin with there is proved a preliminary lemma which will simplify the later proofs. For $1 \leq k \leq m-1 < n$, we define the upper Hessenberg submatrix H^C_{m-1-k} of order $m-1-k$ associated to a left principal submatrix H_{m-1}. Its first $m-2-k$ columns are equal to those of H_{m-1-k}, while the last column comprises the elements of the last column, $m-1$, of H_{m-1}. For example, for $m=8$, $k=3$, the resulting matrix H^C_4 is

$$H^C_4 = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{17} \\ h_{21} & h_{22} & h_{23} & h_{27} \\ 0 & h_{32} & h_{33} & h_{37} \\ 0 & 0 & h_{43} & h_{47} \end{bmatrix}.$$

Lemma 1 *The proper Hessenbergians,* $\det H^C_{i-1}$, $\det H^{(i)}_{n-i}$, *and first order minors* $M_{j;i}$ *with* $1 \leq j < i$, *of an upper Hessenberg matrix* H *of order* n *satisfy the following equations*

$$(-1)^{m-i} h_{i,i-1} \det H^C_{i-1} \det\ H^{(i)}_{n-i} = \sum_{j=1}^{i-1} h_{j,m-1} (-1)^{m-1-j} M_{j;i} \tag{2}$$

The submatrix H^C_{i-1} *is defined relative to the left principal submatrix* H_{m-1}, $i < m$.

Proof. Expanding $\det H^C_{i-1}$ along its last row and using (1), we have

$$(-1)^{m-i} h_{i,i-1} \det H^C_{i-1} \det H^{(i)}_{n-i} =$$

$$= (-1)^{m-1-(i-1)} h_{i-1,m-1} M_{i-1;i} + (-1)^{m-i-1} h_{i,i-1} h_{i-1,i-2} \det H^C_{i-2} \det H^{(i)}_{n-i}$$

Iterating the procedure, the left side of (2) is equal to

$$\sum_{j=i-2}^{i-1} h_{j,m-1} (-1)^{m-1-j} M_{j;i} + (-1)^{m-i-2} \left(\prod_{k=0}^{2} h_{i-k,i-k-1} \right) \det H^C_{i-3} \det H^{(i)}_{n-i}$$

After $i-2$ iterations, with the convention that $\det H_0 = 1$, there results

$$(-1)^{m-i} h_{i,i-1} \det H^C_{i-1} \det H^{(i)}_{n-i} =$$

$$= \sum_{j=2}^{i-1} h_{j,m-1} (-1)^{m-1-j} M_{j;i} + (-1)^{m-2} \left(\prod_{k=0}^{i-2} h_{i-k,i-k-1} \right) \det H^C_1 \det H^{(i)}_{n-i}$$

 ISBN 13: 978-84-613-5510-5

$$= \sum_{j=1}^{i-1} h_{j,m-1}(-1)^{m-1-j} M_{j;i}$$

■

The centered principal submatrices $H^{(i)}_{j-i-1}$ or order $j-i-1$ appear in the next theorem in a role analogous to that played by the submatrices of H in the definitions given in (1). These matrices are formed from the matrix H, taking the elements from $(i+1)$-st to the j-th rows and columns .

Theorem 1 *Any upper Hessenberg matrix H of order n with complex coefficients satisfies the equations*

$$\det H^{(i)}_{j-i-1} \det H = \det H_{j-1} \det H^{(i)}_{n-i} - \left(\prod_{k=0}^{j-i-1} h_{j-k,j-k-1} \right) M_{j;i} \tag{3}$$

for $1 \le i < j \le n$, where $M_{j;i}$ is the corresponding first order minor of the matrix H.

Proof. For fixed i, $1 \le i < n$, we proceed by induction on j, $i < j \le n$.

If $j = i+1$, the result follows straightforwardly from expanding $\det H_n$ along its j-th row:

$$1 \cdot \det H = \det H_{j-1} \det H^{(j-1)}_{n-j-1} - h_{j,j-1} M_{j;i} \tag{4}$$

We suppose the statement is true for $i < j \le m-1 < n$. Then, for $j = m \le n$, expanding the Hessenbergians in (3) for the matrices depending on m along their last rows and using the induction hypothesis, we have,

$$\det H_{m-1} \det H^{(i)}_{n-i} - \det H^{(i)}_{m-i-1} \det H =$$

$$= \left(\prod_{k=0}^{m-i-2} h_{(m-1)-k,(m-1)-k-1} \right) h_{m-1,m-1} M_{m-1;i}$$

$$-h_{m-1,m-2} \left(\det H^{C}_{(m-1)-1} \det H^{(i)}_{n-i} - \det H^{(i)C}_{(m-1)-i-1} \det H \right)$$

The Hessenberg matrices $H^{C}_{(m-1)-1}$ and $H^{(i)C}_{(m-1)-i-1}$ are evident in this context and they are associated to the left principal submatrix H_{m-1} and the centered principal submatrix $H^{(i)}_{(m-1)-i}$, respectively.

If we expand the determinants indexed by m along their last rows one more time and use the induction hypothesis,

$$\det H_{m-1} \det H^{(i)}_{n-i} - \det H^{(i)}_{m-i-1} \det H =$$

$$= \left(\prod_{k=0}^{m-i-2} h_{(m-1)-k,(m-1)-k-1} \right) \left(\sum_{l=m-2}^{m-1} h_{l,m-1}(-1)^{m-1-l} M_{l;i} \right) +$$

$$+h_{m-1,m-2} h_{m-2,m-3} \left(\det H^{C}_{(m-2)-1} \det H^{(i)}_{n-i} - \det H^{(i)C}_{(m-2)-i-1} \det H \right)$$

 ISBN 13: 978-84-613-5510-5

After $m-i$ iterations and using the induction hypothesis,

$$\det H_{m-1} \det H_{n-i}^{(i)} - \det H_{m-i-1}^{(i)} \det H =$$

$$= \left(\prod_{k=0}^{m-i-2} h_{(m-1)-k,(m-1)-k-1} \right) \times$$

$$\times \left(\sum_{l=i}^{m-1} h_{l,m-1}(-1)^{m-1-l} M_{l;i} + (-1)^{m-i} h_{i,i-1} \det H_{i-1}^{C} \det H_{n-i}^{(i)} \right) \tag{5}$$

The induction hypothesis can be used up to here. We make the convention that any Hessenbergian of negative order is null. Thus $\det H_{-1}^{(i)C} = 0$.

If we invoke Lemma 1, then (5) yields,

$$\det H_{m-1} \det H_{n-i}^{(i)} - \det H_{m-i-1}^{(i)} \det H =$$

$$= \left(\prod_{k=0}^{m-i-2} h_{(m-1)-k,(m-1)-k-1} \right) \left(\sum_{l=1}^{m-1} h_{l,m-1}(-1)^{m-1-l} M_{l;i} \right) \tag{6}$$

In order conclude the proof, it is sufficient to show that any upper Hessenberg matrix H of order n satisfies:

$$\sum_{l=1}^{m-1} h_{l,m-1}(-1)^{m-1-l} M_{l;i} = h_{m,m-1} M_{m;i}$$

For this purpose we give the sum with cofactors of the matrix H,

$$\sum_{l=1}^{m-1} h_{l,m-1}(-1)^{m-1-l} M_{l;i} = (-1)^{m-1-i} \sum_{l=1}^{m-1} h_{l,m-1}(-1)^{l+i} M_{l;i}$$

Because $m-1 \neq i$, the sum of alien cofactors, [20], of the matrix H is null. Taking into consideration that we are working with an upper Hessenberg matrix of order n, we have,

$$(-1)^{m-1-i} \sum_{l=1}^{n} h_{l,m-1}(-1)^{l+i} M_{l;i} = (-1)^{m-1-i} \sum_{l=1}^{m} h_{l,m-1}(-1)^{l+i} M_{l;i} = 0$$

The induction step is verified, after an appropriate change in the index k of the product from equation (6),

$$\det H_{m-1} \det H_{n-i}^{(i)} - \det H_{m-i-1}^{(i)} \det H = \left(\prod_{k=0}^{m-i-1} h_{m-k,m-k-1} \right) M_{m;i}.$$

This concludes the proof. ■

The general representation for entries of the inverse of an upper Hessenberg matrix H as products of proper Hessenbergians, [17, 22], is also a consequence of Theorem 1.

 ISBN 13: 978-84-613-5510-5

Corollary 1 *A general expression for the elements $(H^{-1})_{i,j}$ of the inverse matrix of an upper Hessenberg regular matrix H, is*

$$(H^{-1})_{i,j} = \frac{(-1)^{i+j}(\prod_{k=2}^{i} h_{k,k-1}) \left(\det H_{j-1} \det H^{(i)}_{n-i} - \det H^{(i)}_{j-i-1} \det H\right)}{(\prod_{k=2}^{j} h_{k,k-1}) \det H} \tag{7}$$

That is, the element $(H^{-1})_{i,j}$ of the inverse matrix can be represented as, [17, 22]

$$(H^{-1})_{i,j} = \begin{cases} (-1)^{i+j} \frac{\left(\prod_{k=0}^{i-j-1} h_{i-k,i-k-1}\right) \det H_{j-1} \det H^{(i)}_{n-i}}{\det H} & if \quad i \geq j, \\ (-1)^{i+j} \frac{\det H_{j-1} \det H^{(i)}_{n-i} - \det H^{(i)}_{j-i-1} \det H}{\left(\prod_{k=0}^{j-i-1} h_{j-k,j-k-1}\right) \det H} & if \quad i < j. \end{cases} \tag{8}$$

The cases with $i \geq j$ are equation (1). For $i < j$, when the matrix H is not unreduced, the result is also valid as a consequence of Theorem 1. Indeed, for $i < j$,

$$\det H_{j-1} \det H^{(i)}_{n-i} - \det H^{(i)}_{j-i-1} \det H = \left(\prod_{k=0}^{j-i-1} h_{j-k,j-k-1}\right) M_{j;i}$$

Then, (7) results in,

$$(H^{-1})_{i,j} = (-1)^{i+j} \frac{\left(\prod_{k=0}^{j-i-1} h_{j-k,j-k-1}\right) M_{j;i}}{\left(\prod_{k=0}^{j-i-1} h_{j-k,j-k-1}\right) \det H} = \frac{A_{j;i}}{\det H},$$

the Cayley formula for $(H^{-1})_{i,j}$.

3 Recurrences for the computation and some elementary properties of the inverse matrix

Although fast numerical algorithms can be used for the computation of Hessenbergians from (8), we concentrate on the homogeneous linear recurrences found for them.

Determinants of left principal submatrices, $\det H_i$, $i = 1, \cdots, n$ and $\det H_n = \det H$, of any upper Hessenberg matrix of order n satisfy the following large recurrence relations, with $\det H_0 = 1$,

$$\det H_i = \sum_{m=1}^{i} (-1)^{m-1} \left(\prod_{k=1}^{m-1} h_{i-k+1,i-k}\right) h_{i-m+1,i} \det H_{i-m} \tag{9}$$

For Hessenbergians of right principal submatrices, $\det H^{(i)}_{n-i}$, for $i < j \leq n$, the recurrences are similar, now with the initial conditions $\det H^{(i)}_0 = 1$ for $j = i$.

$$\det H^{(i)}_{j-i} = \sum_{m=1}^{j-i} (-1)^{m-1} \left(\prod_{k=1}^{m-1} h_{j-k+1,j-k}\right) h_{j-m+1,j} \det H^{(i)}_{j-i-m} \tag{10}$$

 ISBN 13: 978-84-613-5510-5

The recurrences for Hessenbergians of the centered principal submatrices, $\det H^{(i)}_{j-i-1}$, can be obtained as particular cases of (10). Therefore, when the matrix H is unreduced, the computation of the elements of its inverse presents no difficulty.

When the matrix H is reduced, the numerical computation of the elements of its inverse presents no difficulty if $i \geq j$, or if $i < j$ and the null elements from the subdiagonal do not appear in the product of the denominator of (8). The computational difficulty appears when $i < j$ and one or more null elements of the subdiagonal of the matrix H appear more than once in the product of the Hessenbergians of the numerator of (8). This can happen when the minor associated to this element of the inverse matrix is not a Hessenbergian and it has in its second subdiagonal null and non-null elements from the subdiagonal of the matrix H.

We overcome indeterminacies by introducing auxiliary parameters in place of the zeros that can appear in the product of the denominator of (8). We use as an illustration a reduced Hessenberg matrix of order 6, obtained in a random way

$$H = \begin{bmatrix} -4 & 2 & -4 & 1 & -4 & -1 \\ -1 & 3 & 2 & 0 & -1 & 4 \\ 0 & 0 & 1 & 2 & 4 & 1 \\ 0 & 0 & 4 & -4 & 2 & -2 \\ 0 & 0 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 2 \end{bmatrix} \tag{11}$$

Elements h_{32} and h_{54} of the subdiagonal are null. The element $(H^{-1})_{2,5} = \frac{331}{60}$ has associated the minor $M_{5;2}$, which is not a Hessenbergian. In this case, by (8), using parameters instead of the zeros in h_{32} and h_{54} in the Hessenbergians of the numerator and in the product of the denominator, we have

$$(H^{-1})_{2,5} = (-1)\frac{\det H_4 \det H_4^{(2)} - \det H_2^{(2)} \det H}{\left(\prod_{k=0}^{2} h_{4-k,3-k}\right) \det H} =$$

$$= \frac{2648\alpha\beta}{480\alpha\beta} = \frac{331}{60}.$$

We can obtain the right numerical result using the previous recurrences, replacing the parameters α and β by the value 1.

To calculate the element $(H^{-1})_{1,6} = -\frac{129}{20}$, we work with the minor $M_{6;1}$ of the matrix H, which is also not a Hessenbergian. We proceed in a similar way,

$$(H^{-1})_{1,6} = (-1)\frac{\det H_5(\alpha,\beta) \det H_5^{(1)}(\alpha,\beta) - \det H_4^{(1)}(\alpha,\beta) \det H(\alpha,\beta)}{(-4\alpha\beta)\,(120)}$$

$$(H^{-1})_{1,6} = \frac{-256\alpha^2\beta^2 + 768\alpha^2\beta - 1016\alpha\beta^2 + 3096\alpha\beta}{-480\alpha\beta} = \frac{-32\alpha\beta + 96\alpha - 127\beta + 387}{-60}$$

If we give now to the parameters their null values,

$$(H^{-1})_{1,6} = -\frac{129}{20}$$

 ISBN 13: 978-84-613-5510-5

we obtain the right result using the recurrences associated to the Hessenbergians and elementary symbolic computations. Were we to solve numerically as for the element previously obtained, replacing the parameters α and β in the recurrences involved in (8) by the value 1, it is obvious that the result would be inaccurate.

3.1 Some elementary properties of the inverse matrix

It is well known that the inverse of an upper Hessenberg matrix is a lower semiseparable (plus diagonal) matrix, as can be derived easily from (8) for $i \geq j + k$,

$$\det \begin{bmatrix} (H^{-1})_{i,j} & (H^{-1})_{i,j+k} \\ (H^{-1})_{i+l,j} & (H^{-1})_{i+l,j+k} \end{bmatrix} = 0$$

It is also known that the inverse matrix is semiseparable if and only if the matrix H is tridiagonal. In the unreduced case, more important in applications, if there are zeros in the diagonal of the inverse matrix, some principal, left or right, submatrices have non-maximal rank, with null associated Hessenbergians.

Moreover, the low rank property of some of the principal submatrices involved is a sufficient condition for the nullity of some element above the diagonal for the inverse matrix of a unreduced Hessenberg matrix H. The next illustrative matrix has an inverse with null diagonal elements and some elements above the diagonal are also null. This can be checked using (8).

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad ; \quad H^{-1} = \begin{bmatrix} 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 1 \\ -1 & 1 & 0 & 0 \end{bmatrix}$$

The entry $(H^{-1})_{1,3}$ is 0, because $\det H_2$ and $\det H_1^{(1)}$ are null. Also the entry $(H^{-1})_{2,4} = 0$ because $\det H_3$, $\det H_2^{(2)}$, and $\det H_1^{(2)}$ are null Hessenbergians.

4 General Orthogonal Polynomials: Hessenberg matrices in standard form

The computation of the resolvent matrices associated to orthogonal polynomials on the real line and on the unit circle and their associated tridiagonal and pentadiagonal Green matrices, are of current interest. Our results have application to the more general case of finite sections of the resolvent matrix associated with any sequence of orthogonal polynomials on an arbitrary and bounded domain in the complex plane. We will give two particular examples on the unit disk.

Given an infinite HPD (Hermitian positive definite) matrix, $M = (c_{ij})_{i,j=0}^{\infty}$, whether it comes from a measure or not, we denote by M' the matrix obtained by eliminating from M its first column. Let M_n and M'_n be the corresponding sections of order n of M and M', respectively, i.e., the corresponding left principal submatrices. As M is an HPD matrix, an infinite upper Hessenberg matrix $D = (d_{ij})_{i,j}^{\infty}$ can be built. Matrix

 ISBN 13: 978-84-613-5510-5

D is in standard form. That is, it has a positive subdiagonal. Its sections of order n satisfy, [18],

$$D_n = T_n^{-1} M_n' T_n^{-H}, \tag{12}$$

where $M_n = T_n T_n^H$ is the Cholesky decomposition of M_n, and $\{\widetilde{P}_n(z)\}_{n=0}^{\infty}$ is its associated orthogonal sequence of monic polynomials, [16]. From the properties of the matrix D_n, we have a determinantal expression for $\widetilde{P}_n(z)$. That is, the zeros of the orthogonal polynomials are the eigenvalues of the Hessenberg matrix:

$$\widetilde{P}_n(z) = det\,(zI_n - D_n) \tag{13}$$

If this Hessenbergian is expanded along the last row and the procedure is iterated, we obtain, as a particular case of (9), the *large recurrence relation* for monic orthogonal polynomials:

$$\widetilde{P}_n(z) = (z - d_{n,n})\widetilde{P}_{n-1}(z) - \sum_{k=1}^{n-1} d_{k,n} \left[\prod_{m=k}^{n-1} d_{m+1,m}\right] \widetilde{P}_{k-1}(z), \tag{14}$$

with initial condition $\widetilde{P}_0(z) = 1$.

The matrix obtained when deleting from D its first i rows and columns is denoted $D^{(i)}$. From $D^{(i)}$ we can build the infinite HPD matrix $M(i)$, for all $i \in \mathcal{Z}_+$. It defines an inner product. Then, the associated monic polynomials are defined, for $n \geq i$, as

$$\widetilde{P}_{n-i}^{(i)}(z) = det\left(zI_{n-i} - D_{n-i}^{(i)}\right) \tag{15}$$

with $\widetilde{P}_0^{(i)}(z) = 1$. They are orthogonal with respect to the inner product defined by $M(i)$. When expanding this Hessenbergian, we obtain, as a particular case of (10), the *large recurrence relation* for the associated monic polynomials,

$$\widetilde{P}_{n-i}^{(i)}(z) = (z - d_{n,n})\widetilde{P}_{n-i-1}^{(i)}(z) - \sum_{k=i+1}^{n-1} d_{k,n} \left[\prod_{m=k}^{n-1} d_{m+1,m}\right] \widetilde{P}_{k-i-1}^{(i)}(z), \tag{16}$$

Corollary 2 *The elements for finite sections $(I_n z - D_n)^{-1}$, $n \geq 1$, of the resolvent matrix related to the monic orthogonal polynomials coming from the matrix D are*

$$\left((I_n z - D_n)^{-1}\right)_{i,j} = \begin{cases} (-1)^{i+j} \left(\prod_{k=0}^{i-j-1} d_{i-k,i-k-1}\right) \frac{\widetilde{P}_{j-1}(z)\widetilde{P}_{n-i}^{(i)}(z)}{\widetilde{P}_n(z)} & if \quad j \leq i \\ \frac{(-1)^{i+j}}{\left(\prod_{k=0}^{j-i-1} d_{j-k,j-k-1}\right)} \left[\frac{\widetilde{P}_{j-1}(z)\widetilde{P}_{n-i}^{(i)}(z)}{\widetilde{P}_n(z)} - \widetilde{P}_{j-i-1}^{(i)}(z)\right] & if \quad i < j \end{cases} \tag{17}$$

If there are known expressions in closed form for the orthogonal polynomials of the sequence under analysis and those of its associated sequences, then the closed form expressions for the finite sections of the resolvent matrix are easily obtained. When expressions in closed form for the monic polynomials are not known, the entries of the resolvent matrix, for any complex number z, can be obtained numerically using the preceding recurrences.

 ISBN 13: 978-84-613-5510-5

4.1 A measure with radial symmetry on the unit disk

Let μ be a measure on the unit disk with a radially symmetric weight function constant on every circle centered on the origin. We suppose also that μ is a probability measure, i.e. $c_{00} = 1$. We have $\omega(z) = \omega(|z|)$. In this case, writing $r = |z|$, the moments are

$$c_{ij} = \int\limits_{|z|<1} z^i \overline{z}^j \omega(r) dx dy = \int_0^1 \omega(r) r^{i+j+1} dr \int_0^{2\pi} e^{I(i-j)\theta} d\theta,$$

where the imaginary unit is denoted by I, to avoid confusion with the i index. By symmetry if $i \neq j$ then $c_{ij} = 0$. We have

$$c_{ii} = 2\pi \int_0^1 \omega(r) r^{2i+1} dr, \qquad i > 1, \quad with \quad 2\pi \int_0^1 \omega(r) dr = 1,$$

The moment matrix $M = (c_{ij})_{i,j=0}^{\infty}$ is diagonal and the associated Hessenberg matrix $D = (d_{ij})_{i,j=1}^{\infty}$ satisfies $d_{i+1,i} = \sqrt{\frac{c_{ii}}{c_{i-1,i-1}}}$ and $d_{ij} = 0$ if $i \neq j+1$. The monic polynomials are $\widetilde{P}_n(z) = z^n$ and the associated polynomials, for $n > i$, are $\widetilde{P}_{n-i}^{(i)}(z) = z^{n-i}$, with $\widetilde{P}_n^{(0)}(z) = 1$. Using Corollary 2, we obtain the resolvents of the finite sections

$$(I_n z - D_n)^{-1} = \begin{bmatrix} 1/z & 0 & 0 & \cdots & 0 \\ \sqrt{\frac{c_{11}}{c_{00}}}\frac{1}{z^2} & 1/z & 0 & \cdots & 0 \\ \sqrt{\frac{c_{22}}{c_{00}}}\frac{1}{z^3} & \sqrt{\frac{c_{22}}{c_{11}}}\frac{1}{z^2} & 1/z & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sqrt{\frac{c_{n-1,n-1}}{c_{00}}}\frac{1}{z^n} & \sqrt{\frac{c_{n-1,n-1}}{c_{11}}}\frac{1}{z^{n-1}} & \sqrt{\frac{c_{n-1,n-1}}{c_{22}}}\frac{1}{z^{n-2}} & \cdots & 1/z \end{bmatrix} \tag{18}$$

4.2 A measure without radial symmetry on the unit disk

Now, we give an example partially treated in [15] with a full Hessenberg matrix in standard form. We consider the density function on the unit closed disk given by $\omega(z) = |z-1|^2$ with $|z| \leq 1$. The density function is null for $z = 1$ and positive in the rest of the disk. The moments are obtained by applying Green's formula,

$$c_{ij} = \frac{1}{2I} \int\limits_{|z|=1} \left[-\frac{z^{i-j}}{j+1} + \left(\frac{1}{j+1} + \frac{1}{j+2} \right) z^{i-j-1} - \frac{z^{i-j-2}}{j+2} \right] dz$$

Therefore, the matrix of moments is

$$M = \begin{bmatrix} \pi\left(1+\frac{1}{2}\right) & -\frac{\pi}{2} & 0 & 0 & \cdots \\ -\frac{\pi}{2} & \pi\left(\frac{1}{2}+\frac{1}{3}\right) & -\frac{\pi}{3} & 0 & \cdots \\ 0 & -\frac{\pi}{3} & \pi\left(\frac{1}{3}+\frac{1}{4}\right) & -\frac{\pi}{4} & \cdots \\ 0 & 0 & -\frac{\pi}{4} & \pi\left(\frac{1}{4}+\frac{1}{5}\right) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{19}$$

 ISBN 13: 978-84-613-5510-5

The elements of the matrix D are given by

$$d_{ij} = \begin{cases} \frac{-2\sqrt{i}}{\sqrt{(i+1)(i+2)j(j+1)(j+2)}} & \text{if} \quad i \leq j \\ \frac{\sqrt{j(j+3)}}{j+2} & \text{if} \quad i = j+1 \\ 0 & \text{if} \quad i > j+1 \end{cases} \tag{20}$$

The monic polynomials are obtained from (13)-(14),

$$\widetilde{P}_n(z) = \frac{1}{(n+1)(n+2)} \sum_{k=0}^{n} (k+2)(k+1) z^k \tag{21}$$

and the associated monic polynomials, if $n > i$, from (15)-(16),

$$\widetilde{P}^{(i)}_{n-i}(z) = z^{n-i} + \frac{2}{(i+2)(n+1)(n+2)} \sum_{k=i}^{n-1} (k+2)(k+1-i) z^{k-i} \tag{22}$$

In the particular case $n = j-1 > i$, the monic polynomials $\widetilde{P}^{(i)}_{j-1-i}(z)$ are obtained from (22).

The resolvent matrices of D_n are readily obtained using Corollary 2, with the subdiagonal entries of D_n given in (20). The monic polynomials and their associated polynomials are obtained from (21) and (22), respectively.

Acknowledgements

This work was partially supported by a research grant of the UPM-CAM in Madrid, Spain.

References

[1] W.W. Barrett and P.J. Feinsilver, *Inverses of banded matrices*, Linear Algebra Appl., **41** (1981), 111–130.

[2] T. Bella, Y. Eidelman, I. Gohberg, I. Koltracht and V. Olshevsky, *A fast Björck-Pereyra type algortihm for solving Hessenberg-quasiseparable-Vandermonde systems*, SIAM J. Matrix Anal. Appl., **31** (2009), 790–815.

[3] B. Bukhberger and G.A. Emel'yanenko, *Methods of inverting tridiagonal matrices*, USSR Computational Math. and Math. Phys., **13** (1973), 10–20.

[4] S. Delvaux and M. Van Barel, *Structures preserved by matrix inversion*, SIAM J. Matrix Anal. Appl., **28**:1 (2006), 213–228.

[5] M. Elouafi and D.A. Hadj, *A new recursive algorithm for inverting Hessenberg matrices*, Appl. Math. Comput., **214** (2009), 497–499.

 ISBN 13: 978-84-613-5510-5

[6] D.K. Faddeev, *Properties of a matrix, inverse of a Hessenberg matrix* , Journal Mathematical Sciences, **24** (1984), 118-120.

[7] M. Fiedler and T.L. Markham, *Completing a matrix when certain entries of its inverse are specified* , Linear Algebra Appl., **74** (1986), 225–237.

[8] W.H. Gustafson, *A note on matrix inversion*, Linear Algebra Appl., **57** (1984), 71–73.

[9] Y. Huang and W.F. McColl, *Analytical inversion of general tridiagonal matrices*, J. Phys. A: Math. Gen., **30** (1997), 7919–7933.

[10] Y. Ikebe, *On inverses of Hessenberg matrices*, Linear Algebra Appl., **24** (1979), 93–97.

[11] R.K. Kittappa, *A representation of the solution of the nth order linear difference equation with variable coefficients*, Linear Algebra Appl., **193** (1993), 211–222.

[12] R.K. Mallik, *On the solution of a Linear Homogeneous Difference Equation with variable coefficients*, SIAM J. Math. Anal., **31**:2 (2000), 375–385.

[13] R.K. Mallik, *The inverse of a tridiagonal matrix*, Linear Algebra Appl., **325** (2001), 109–139.

[14] G. Strang and T. Nguyen, *The interplay of ranks of submatrices*, SIAM Review, **46**:4 (2004), 637–646.

[15] P.K. Suetin, *Polynomials Orthogonals over a Region and Beierbach Polynomials*, Amer. Math. Soc., Providence, Rhode Island, 1974.

[16] G. Szegö, *Orthogonal Polynomials*, Colloq. Pub., 23, Amer. Math. Soc., Providence, Rhode Island, (fourth edition), 1975.

[17] V. Tomeo, *Subnormalidad de la matriz de Hessenberg asociada a los P.O. en el caso hermitiano*, Ph. D. thesis, Facultad de Informática, U.P.M Madrid, in Spanish 2004.

[18] E. Torrano and R. Guadalupe, *On the Moment Problem in the Bounded Case*, J. Comp. App. Math., **49** (1993), 263–269.

[19] R. Vanderbril and M. Van Barel, *A note on the nullity theorem*, J. Comp. App. Math., **189** (2006), 179–190.

[20] R. Vein and P. Dale, *Determinants and their applications in Mathematical Physics*, Springer Verlag, New York, 1999.

[21] T. Yamamoto and Y. Ikebe, *Inversion of band matrices*, Linear Algebra Appl. **24** (1979) 105–111.

[22] J. M. Zemke, *Hessenberg eigenvalues-eigenmatrix relations*, Linear Algebra Appl. **414** (2006) 589–606.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Load Balancing on non-dedicated heterogeneous systems with the ALBIC library

A Acosta[1], F Almeida[1], V Blanco[1], EM Garzón[2] and JA Martínez[2]

[1] *Dpt Statistics and Computer Science, La Laguna University*

[2] *Dpt Computer Architecture and Electronics, Almeria University*

emails: aacostad@ull.es, falmeida@ull.es, vblanco@ull.es, gmartin@ual.es, jmartine@ual.es

Abstract

Adapting parallel codes to state of the art parallel computers composed of heterogeneous multinode-multicore processors is a fundamental problem in parallel computing. The strong dependence on the parallel architectures means that applications must be tailored with a high programming effort. We have devoloped a lightweighted library that allows the dynamic load balancing of iterative codes in heterogeneous dedicated and non-dedicated systems. The library eases porting homogeneous parallel codes to heterogeneous platforms, since the code intrusion is low the programming effort is quite reduced. The preliminary tests developed on iterative programs show that the overhead introduced by the library is negligible.

Key words: Heterogeneous computing, Dynamic load balancing, Non-dedicated System

1 Introduction

High performance computing is the field of computational science dealing with engineering and scientific applications of cluster-based computing. It concerns with the development of models and strategies which allow hard computing applications to be solved using the most advanced computing platforms. Up to now most of the applications requiring large computing resources have been solved on supercomputers or clusters composed by a large number of identical processors. Nowadays, the concept of cluster computing has been enlarged. Currently, a cluster can be seen as a flexible reconfigurable computing system which is composed of nodes with different characteristics and performance, which can be simultaneously used by multiple users and processes. These computing environments are known as non dedicated heterogeneous computing systems, and there exists a strong demand for developing new strategies for

 ISBN 13: 978-84-613-5510-5

adapting the software to this kind of heterogeneous environments. The performance of this kind of system is very conditioned by the strong dependence that exists between parallel code and architecture [6] and the process of allocating tasks to processors often becomes a problem requiring considerable programmer effort [7].

Specifically, we set out to solve the problem of synchronizing parallel programs in heterogeneous architectures. Given a program developed for a homogeneous system, we hope to obtain a version that makes use of the system's heterogeneous abilities by allocating tasks according to the computational ability of each processing element. The simplest way to approach the problem consists on manually adapting the code as required by the architectural characteristics[10]. This approach usually implies at least a knowledge of said characteristics, such that the parallel program's tasks can be allocated according to the computational capacity of each processor. A more general approach can be obtained in the context of self-optimization strategies based on a run time model [8, 11]. In this approach, an analytical model that parametrizes the architecture and the algorithm is instantiated for each specific case so as to optimize the program execution. This strategy is considerably more general than the previous one, though more difficult to apply since the modeling process is not trivial [12], nor is its subsequent instantiation and minimization for each case. A search of the literature yields some generic tools such as mpC [14, 13] and HeteroMPI [14, 15] that provide the mechanisms that allow algorithms to be adapted to heterogeneous architectures, but which also demand some input from the user and are quite code intrusive. Adaptive strategies have been also proposed in AMPI [16] and Dyn-MPI [17]. AMPI is built on Charm++ [18] and allows automatic load balancing based on process virtualization. Although it is an interesting generic tool, it involves a complex runtime environment. DynMPI has been implemented as a MPI extension and has a wider range of applicability. However, it is highly code intrusive since data structures, code sections and communication calls must be instrumented.

Our interest is focussed on the efficient implementation of iterative algorithms on non dedicated heterogeneous computing systems. This type of algorithms can be found in a wide set of scientific and engineering problems such as partial differential equations solvers (PDEs) or image processing algorithms, among others; their common feature is its iterative behavior [5, 9]. As main goal, we pursue the design of an approach to be able of dynamically adapt the computational burden related to each processor according to the computational power supplied by the non dedicated heterogeneous nodes, where the overhead of the node is also a source of heterogeneity. Additionally, under this approach the managment of the dynamic heterogeneity of the system must be hidden to the programmer.

The `ULL_Calibrate` library presented in [1] facilitates the programmer the task of tailoring parallel code developed for homogeneous systems to heterogeneous ones, reducing the runtime on dedicated heterogenous systems. However, this library does not collect information about the dynamic load on the system, so the approach to load balance described in [1] is not useful on non dedicated systems. In the same direction, recently, the ADITHE approach has also been proposed to adapt the iterative computation on dedicated heterogeneous clusters of multicore nodes [4]. It has been

 ISBN 13: 978-84-613-5510-5

shown its ability to automatically load balance on these systems, however it is relevant the programmer's effort in order to implement ADITHE in the context of every specific application.

The goal of this work is to extend the `ULL_Calibrate` library to non dedicated systems, without losing its advantages, that is: (1) its use does not require changing any line of code in existing programs, thus minimizing code intrusion; and (2) it allows dynamic task balancing within a parallel program running on a non dedicated heterogeneous system, while adapting to system conditions during execution. The new library proposed in this paper facilities the automatic Adaptive Load Balancing of Iterative Computation (ALBIC) on non dedicated heterogeneous systems. Hereinafter it will be referred as ALBIC library.

We validated our proposal using as test problem the resource allocation optimization via dynamic programming algorithms [9]. The preliminary computational results show that the benefits yielded by using our balancing library offer substantial time reductions in every case. The efficiency level obtained, considering the minimum code intrusion, and the reduced extraoverhead introduced by ALBIC in the load balancing process makes this library a useful tool in the context of

heterogeneous non-dedicted platforms.

This paper is structured as follows: in Section 2 we introduce the goals and background of this work; Section 3 shows how to use ALBIC library and the advantages our approach yields; in Section 4 a model of computational power of non-dedicated processor and the load balancing algorithm; next, Section 5 shows the preliminary validation performed on the selected problem (RAP) ; and finally, we close with some conclusions and future research directions.

2 Background

Our objective is to develop a simple and efficient dynamic adaptation strategy of the code for heterogeneous systems that minimizes code intrusion, so that the program can be adapted without any prior knowledge of the architecture and without the need to develop analytical models. We intend to apply the technique to a wide variety of problems, specifically to parallel programs which can be expressed as a series of synchronous iterations. To accomplish this, we have developed ALBIC library with which to instrument specific sections in the code. The instrumentation required is minimal, as it is the resulting overhead. Using this instrumentation, the program will dynamically adapt itself to the destination architecture. This approach is particularly effective in SPMD applications with replicated data.

Our library's design is directed at solving the time differences obtained when executing the parallel code. It is based on an iterative scheme, such as that appearing in **Listing 1,** which shows a parallel version of the dynamic programming approach to the resource allocation problem considered as an iterative procedure. The code involves a main loop that executes N iterations where a amount of calculation operation (work load) is performed for each iteration. Each processor performs calculations in

 ISBN 13: 978-84-613-5510-5

accordance with the size of the task allocated, $M/nproc$. Following this calculation, a collective communication operation is carried out during which all the processors synchronize by gathering collecting data before proceeding to the next iteration. Note that each iteration of the inner loop (the loop in j) has a complexity order of $O(j)$. That means that when a block data distribution with blocks of the same is used by the parallel code the load is not balanced among the processors.

```
// nprocs = Number of processors
// myid  = Process ID
// M = Number of columns
// N = Number of rows (iterations)
...
despl = (int *) malloc( nprocs * sizeof(int));
count = (int *) malloc( nprocs * sizeof(int));

// Static column distribution with blocks
// of fixed size M/nprocs for myid process
despl[0] = 0; ncols = M/nprocs;
for (i = 0; i< nprocs; i++) {
   count[i] = ncols;
   if (i) despl[i] = despl[i-1] + count[i-1];
}

for (i = 0; i <= N; i++) {
/*========begin iterative section=========*/
fin = despl[myid] + count[myid];
for (j = despl[myid]; j < fin; j++) {
   G[i][j] = (*f)(i, 0);
   for (x = 0; x <= j; x++) {
      fij = G[i - 1][j - x] + (*f)(i, x);
      if (G[i][j] < fij)
         G[i][j] = fij;
   }
}
/*========end iterative section=========*/
MPI_Allgatherv(&G[i][despl[myid]], ... );
}
```

Listing 1: Basic algorithm of an iterative scheme.

The load balancing problem consists of allocating on each processor a static load proportional to its computational capacity. The allocation of tasks according to the computational power of the processors depends on the processors and also on the application. When the processors are non-dedicated it is necessary to devise specific approaches to estimate dynamically this parameter.

In Section 4 we propose the model of computational power that is key for the ALBIC library to be able to dynamically balance the load. Before to analyze this model we describe the more relevant characteristics of ALBIC.

 ISBN 13: 978-84-613-5510-5

```
// nprocs = Number of processors
// myid  = Process ID
// M = Number of columns
// N = Number of rows (iterations)

...
despl = (int *) malloc( nprocs * sizeof(int));
count = (int *) malloc( nprocs * sizeof(int));

despl[0] = 0; ncols = M/nprocs;
for (i = 0; i< nprocs; i++) {
   count[i] = ncols;
   if (i) despl[i] = despl[i-1] + count[i-1];
}
for (i = 0; i <= N; i++) {
  ALBIC_MPI_calibrate (ALBIC_MPI_INIT, i, &counts,
                       &displ, threshold, 1, M+1);
  fin = despl[myid] + count[myid];
  for (j = displ[myid]; j < fin; j++) {
    G[i][j] = (*f)(i, 0);
    for (x = 0; x <= j; x++) {
      fij = G[i - 1][j - x] + (*f)(i, x);
      if (G[i][j] < fij)
        G[i][j] = fij;
    }
  }
  ALBIC_MPI_calibrate(ALBIC_MPI_FIN , i, &counts,
                      &displ, threshold, 1, M+1);
  MPI_Allgatherv (&G[i][displ[myid]], count[myid], ... );

}
```

Listing 2: Calibrated version of the basic algorithm of an iterative scheme.

3 Dynamic task allocation on non dedicated systems based on ALBIC

The library we developed allows for dynamic balancing with the introduction of just two calls to the `ALBIC_MPI_calibrate()` function in the section of code that is to be balanced, as shown by the code in Listing 2. A call is introduced at the beginning and end of the section to be balanced, so that each processor can know on runtime how long it will take to execute the assigned task. The balanced load results from a comparison of this execution time for each processor and the subsequent task redistribution.

It is worth noting that the collective communication at the end of each iteration acts as a sort of barrier that forces a high degree of synchronization between all the processes. Listing 3 shows the interface of the calibrating function. The following are input arguments to the balancing function:

 ISBN 13: 978-84-613-5510-5

```
int ALBIC_MPI_calibrate (ALBIC_MPI_Section section, int iteration,
                         int **counts, int **displs,
                         int threshold,
                         int size_object, int size_problem);
```

Listing 3: Prototype of the ALBIC calibrating function.

- **section:** The section is used to determine the entry point where the routine is used. It can take the following two values:
  - **ALBIC_MPI_Section ALBIC_MPI_INIT:** Indicates the beginning of the section to be balanced.
  - **ALBIC_MPI_Section ALBIC_MPI_END:** Indicates the end of the section to be balanced.
- **iteration:** Indicates the iteration to be balanced. A 0 value indicates whether the program is on its first or subsequent iterations. The first iteration has a particular treatment.
- **counts[], displs[]:** Indicates the task size to be computed by each processor. `counts[]` is an integer array containing the amount of work, W_i, that is processed by i-th processor. `displs[]` specifies the distance (relative to the work data vector) at which to place the data processed by each processor.
- **threshold:** Corresponds to a number of microseconds which indicate whether to balance or not. The semantics for this parameter in one iteration are as follows:
  - Let T_i be the time processor i takes to execute the task assigned.
  - $T_{max} = Maximum(T_i)$
  - $T_{min} = Minimum(T_i)$
  - If $(T_{max} - T_{min}) > threshold$ then balance. If not, the system has already balanced the workload.
- **size_objects:** The size of the data type manipulated during computation expressed as the number of elements to be communicated in the communication routine.
- **size_problem:** Corresponds to the total problem size to be computed in parallel, so the calculations of the new task sizes are consistent with the tasks allocated to each processor `counts[]`, `displs[]`.

Note the library's ease of use and the minimum code intrusion. The only change necessary is to add calls to the functions at the beginning (`ALBIC_MPI_init_calibratelib()`) and end of the code, (`ALBIC_MPI_shutdown_calibratelib()`).

 ISBN 13: 978-84-613-5510-5

4 The balancing algorithm on non-dedicated systems

Before to analyze the details of the balancing algorithm, we describe the model of computational power which is key on this algorithm.

4.1 Model of computational power on non-dedicated systems

Although a large number of balancing algorithms can be found in the literature [19], we opted for a simple and efficient strategy that yielded satisfactory results. The methodology chosen, however, allows for the implementation of balancing algorithms which may be more efficient.

The computational power of i-th node, denoted by α_i, is proportional to the ratio between the size of task to be computed by the node ($counts_i$) and the run-times (T_i) to compute the task.

On the other hand, it is well known that to exploit an heterogeneous system composed by a set of *procs* nodes, the workload on each node i has to be balanced. For a heterogeneous system this condition means that the workload of every node has to be proportional to its computational power or speed [2, 3]. So, to balance the workload on the heterogeneous system, the load distribution has to verify the relation

$$\frac{counts_1}{\alpha_1} = \ldots = \frac{counts_{procs}}{\alpha_{procs}} = \frac{N}{\sum_{j=1}^{procs} \alpha_j} \Rightarrow \quad counts_i = \alpha_i \cdot \frac{N}{\sum_{j=1}^{procs} \alpha_j} \tag{1}$$

where $N = size\_problem = \sum_{j=1}^{procs} counts_j$ is the total load to compute the algorithm. Then, to adapt the parallel implementation of an algorithm on heterogeneous multiprocessors, it is necessary to know the computational power or speed of each node of multiprocessor and the total load.

On non dedicated systems the computational power available to every task is a time depending parameter, then $\alpha_i(t)$ has to be periodically measured to dynamically adapt the load attached to every node. ALBIC library is based on the accurate estimation of $\alpha_i(t)$ by means two timing mechanisms: (1) the information from `/proc` file updated every 10 ms by the Linux Operative System and (2) the user-time provided by the C library `getrusage()`. The `/proc` file supplies information about the node load taking account all the processes computed by the i node, and on the other hand, the time during the specific process loads the node is get by means `getrusage()` and it is denoted by $T_i(t)$. Then, $\alpha_i(t)$ can be estimated by the following relation:

$$\alpha_i(t) = \frac{counts_i(t)}{\frac{T_i(t)}{1-L_i(t)}} \tag{2}$$

where $L_i(t)$ denotes the load percentage of i-th node at instant t, $0 \leq L_i(t) < 1$, $L_i(t) = 0$ when the i-th node is absolute free of charge and $L_i(t)$ is close to one when the i-th node is fully overloaded. Thus, the run-time $T_i(t)$ dedicated to the load $counts_i(t)$ is divided by the factor $1 - L_i(t)$ to model the load in the i-th node due to other processes.

 ISBN 13: 978-84-613-5510-5

It is relevant to stress that to obtain the value of parameter $L_i(t)$ it is necessary to compute the exponential moving average of instant loading factor over the l last time samples [20], to warranty that short temporal high loads do not excessively penalize the estimation of computational power of the node. In order to measure the load of processors, $L_i(t)$, according this model, an operative system kernel module for Linux, called Non Dedicated system Load Monitor (NDLM-module), has been developed. NDLM-module monitories the parameter $L_i(t)$ and supplies it to `ALBIC_MPI_calibrate`() function on fly by means the `/proc` file.

4.2 Keys of the balancing algorithm

The call to the `ALBIC_MPI_calibrate`(...) function must be made by all the processors and implements the balancing algorithm. All processors perform the same balancing operations as follows:

- The time required by each processor to carry out the computation in each iteration has to be given to the algorithm. Since each processor needs the times of the other processors, the exchange is performed through a collective operation.
  - $T[]$ = vector where each processor gathers all the times (T_i).
  - $L[]$ = vector where each processor gathers all the processors loads (L_i).
  - $size\_problem$ = the size of the problem to be computed in parallel.
  - $counts[]$ = vector holding the sizes of the tasks to be computed by each processor.
- The first step is to verify that the threshold is not being exceeded

  if (MAX($T[]$) - MIN($T[]$)) $> THRESHOLD$ then, BALANCE
- The computational power $\alpha_i(t)$ is calculated for each i processor according the model before described:
- Finally, the sizes of the new *counts* are calculated for each processor.

$$counts[i] = size\_problem * \frac{\alpha_i}{\sum_{i=0}^{procs-1} \alpha_i}$$

Once the *counts* vector is computed, the *displs* vector is also updated. Using this method, each processor fits the size of the task allocated according to its own computational capacity. The system could be extended to run on heterogeneous non dedicated systems and on systems with dynamic load. For that purpose, the array $T[]$ must be fed not only with the execution times but with the loading factor on each processor.

 ISBN 13: 978-84-613-5510-5

5 Evaluation

To evaluate the ALBIC library we have used a multicore dual node composed of two nodes of 8 Core shared memory system (2 AMD Opteron processors with 4 cores each), 16 cores in total, and use the resource allocation problem as benchmarking test. Although the platform is heterogeneous, the irregular inner loop of the test applications introduces the desired heterogeneity when testing in dedicated mode. To test the library in non-dedicated mode we artificially introduce extra overload on the cores. The results can be extrapolated to an heterogeneus platform without loss of generality. We develop several experiments. First we try to analyze the overhead introduced by ALBIC when performing the load balance. Then, we check that ALBIC balances properly when working in non-dedicated clusters and finally the increase of performance obtained from ALBIC when developing the dynamic load balance in a non-dedicated cluster. To fulfill it, three versions of the parallel code are considered, the *parallel* version, a block homogeneous code, the $ULL\_Calibrate$ a parallel version where the dynamic load balance is obtained through the use of the $ULL\_Calibrate$ library, and the $ALBIC$ code that performs the dynamic load balance using ALBIC.

Figure 1 shows the running times and efficiency of the three parallel codes when executed on the homogeneous dedicated system. We can see that as a consecuence of the dynamic load balance, $ULL\_calibrate$ and $ALBIC$ outperform to the *parallel* code. The natural parallelization by homogeneous blocks produces an unbalanced execution than can be corrected with our dynamic load balance approach. We can also see that the overhead introduced by $ALBIC$ is negligible and the improvement obtained from the dynamic load balance is observable. When the larger number of processors is used with the bigest problem the curves $ULL\_calibrate$ and $ALBIC$ almost overlap, what means that the overhead introduced by our module is compensated by a better balancing that includes the load of the processors. We also observe in this figure the gain of the efficiency when using $ALBIC$. The overhead introduced by $ALBIC$ relative to $ULL\_calibrate$ is also quantified in Figure 2.

The lightweighted NDLM introduces a very slow burden that tends to vanish. When the size of the problem is large enough for the number of processors used, $ALBIC$ performs even better than $ULL\_calibrate$ due to the use of the system load on the metrics. The negative value observed in the relative benefit presented in Figure 2-right denotes that $ULL\_calibrate$ performs better than $ALBIC$ in this case, since the size of the problem per processor decreases when the number of processors increase.

In Figure 3-left a diagram of iterations is presented on a non-dedicated system where the artificial load is introduced on the first of three processor. We observe as starting with an unbalanced distribution due to the nature of the problem, after a few iterations $ALBIC$ converges to a distribution where the first processor (50% artificially overloaded) receives the half of the work, in terms of execution time. The other two processors receive double amount of work. Figure 3-right shows the benefit of using $ALBIC$ on a non-dedicated system, where the 50% of the processors are overloaded on each execution, the execution with 8 processors overloads only 3 of them. The gain of $ALBIC$ is clearly stated.

 ISBN 13: 978-84-613-5510-5

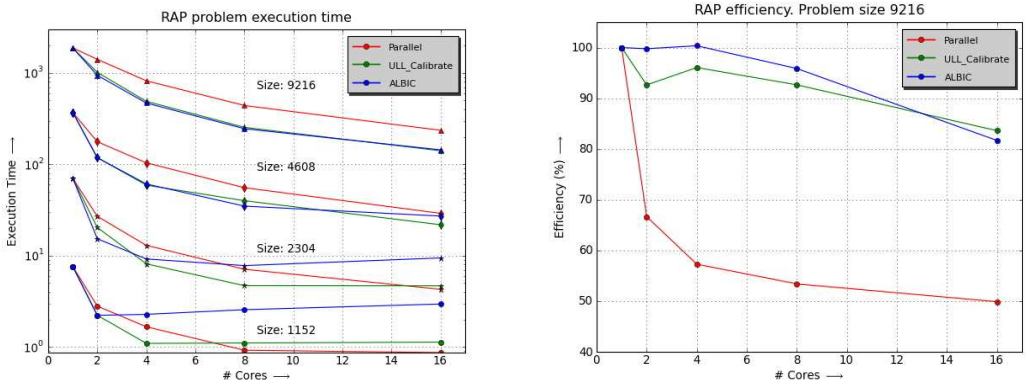


Figure 1: Left - Execution time. Right - Efficiency.eps

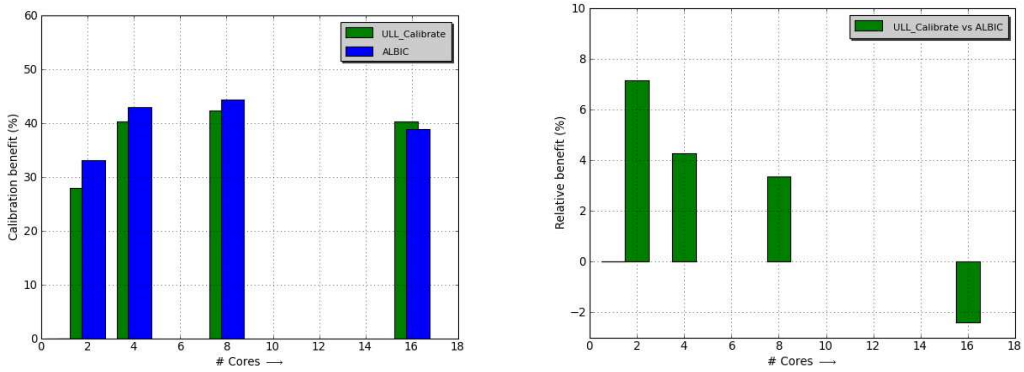


Figure 2: The overhead of the NDLM module. Left - Benefit of $ULL\_calibrate$ and $ALBIC$ relative to the *parallel* code. Right - Relative benefit of $ULL\_calibrate$ versus $ALBIC$

6 Conclusions and future works

We have developed the ALBIC library that allows for the dynamic load balancing of iterative parallel codes on heterogeneous dedicated and non-dedicated systems. The preliminary tests described in this work show that the tool achieves a good performance, the overhead introduced by the library and the input required from the user are minimum. In the near future we plan to extend the library to other problem and architectural domains, considering the dynamic thread allocation in shared memory architectures.

 ISBN 13: 978-84-613-5510-5

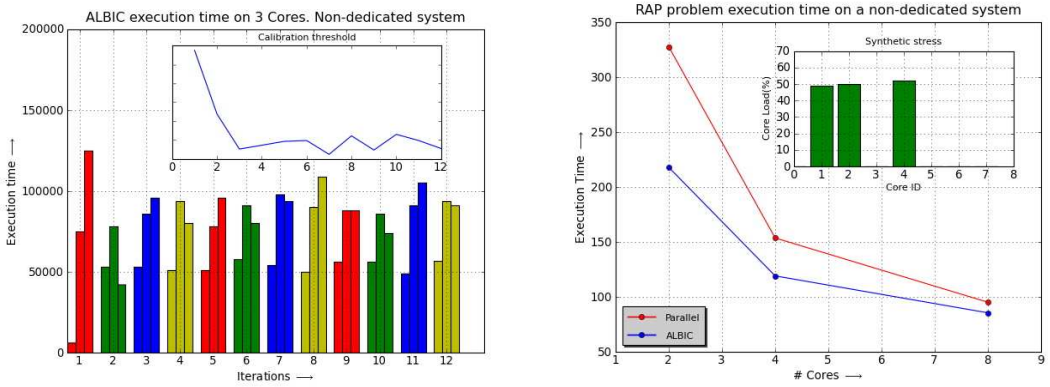


Figure 3: Left - A traced exection of *ALBIC* on a non dedicated system, the first processor is overloaded. Right - *parallel* vs *ALBIC* on a non dedicated system the 50% of the processors are artificially overloaded up to the 50%.

Acknowledgment

This work has been funded by grants from the Spanish Ministry of Science and Innovation TIN2008-01117, TIN2008-06570-C04, Junta de Andalucia (P08-TIC-3518) and Canary Government SolSubC200801000307. Moreover, it has been developed in the framework of the network (CAPAP-H) TIN2009-08058-E.

References

[1] I. Galindo, F. Almeida, V. Blanco, J.M. Badía-Contelles *Dynamic Load Balancing on Dedicated Heterogeneous Systems.* Recent Advances in Parallel Virtual Machine and Message Passing Interface. LNCS 5205, Springer, p. 64–74, (2008).

[2] J. L. Bosque and L. Pastor, *A Parallel Computational Model for Heterogeneous Clusters*, IEEE Transactions on Parallel and Distributed Systems. **Vol. 17**, Issue 12 (2006) 1390–1400.

[3] Y. Chen; S. Xian-He and M. Wu, *Algorithm-system scalability of heterogeneous computing*, Journal of parallel and distributed computing. **Vol 68**, num 11, (2008) 1403–1412.

[4] J.A. Martínez, E.M. Garzón, A. Plaza, and I. García *ADITHE: An approach to optimise iterative computation on heterogeneous multiprocessors.* In Proceedings of the 2009 International Conference on Computational and Mathematical Methods in Science and Engineering, **2** (2009) 665676.

[5] S. Tabik, E.M. Garzón, I. García, and J.J. Fernández *High Performance Noise Reduction for Biomedical Multidimensional Data.* Digital Signal Processing. 17(4):724-736 (2007).

[6] Dongarra, J., Bosilca, G., Chen, Z., Eijkhout, V., Fagg, G.E., Fuentes, E., Langou, J., Luszczek, P., Pjesivac-Grbovic, J., Seymour, K., You, H., Vadhiyar, S.S.: Self-adapting numerical software (sans) effort. IBM Journal of Research and Development **50**(2-3) (2006) 223–238

 ISBN 13: 978-84-613-5510-5

[7] Kalinov, A., Lastovetsky, A.L., Robert, Y.: Heterogeneous computing. Parallel Computing **31**(7) (2005) 649–652

[8] Cuenca, J., Giménez, D., Martinez, J.P.: Heuristics for work distribution of a homogeneous parallel dynamic programming scheme on heterogeneous systems. Parallel Comput. **31**(7) (2005) 711–735

[9] Alba, E., Almeida, F., Blesa, M.J., Cotta, C., Díaz, M., Dorta, I., Gabarró, J., León, C., Luque, G., Petit, J.: Efficient parallel lan/wan algorithms for optimization. the mallba project. Parallel Computing **32**(5-6) (2006) 415–440

[10] Aliaga, J.I., Almeida, F., Badía-Contelles, J.M., Barrachina-Mir, S., Blanco, V., Castillo, M.I., Dorta, U., Mayo, R., Quintana-Ortí, E.S., Quintana-Ortí, G., Rodríguez, C., de Sande, F.: Parallelization of the gnu scientific library on heterogeneous systems. In: ISPDC/HeteroPar, IEEE Computer Society (2004) 338–345

[11] Almeida, F., González, D., Moreno, L.M.: The master-slave paradigm on heterogeneous systems: A dynamic programming approach for the optimal mapping. Journal of Systems Architecture **52**(2) (2006) 105–116

[12] Al-Jaroodi, J., Mohamed, N., Jiang, H., Swanson, D.R.: Modeling parallel applications performance on heterogeneous systems. In: IPDPS, IEEE Computer Society (2003) 160

[13] mpC: parallel programming language for heterogeneous networks of computers. `http://hcl.ucd.ie/Projects/mpC`

[14] Lastovetsky, A., Reddy, R.: Heterompi: Towards a message-passing library for heterogeneous networks of computers. Journal of Parallel and Distributed Computing **66** (2006) 197–220

[15] HeteroMPI: Mpi extension for heterogeneous networks of computers. `http://hcl.ucd.ie/Projects/HeteroMPI`

[16] Huang, C., Lawlor, O., Kale, L.: Adaptive mpi. (2003)

[17] Weatherly, D., Lowenthal, D., Lowenthal, F.: Dyn-mpi: Supporting mpi on non dedicated clusters. (2003)

[18] charm++ System. `http://charm.cs.uiuc.edu/research/charm/index.shtml#Papers`

[19] Bosque, J.L., Marcos, D.G., Pastor, L.: Dynamic load balancing in heterogeneous clusters. In Hamza, M.H., ed.: Parallel and Distributed Computing and Networks, IASTED/ACTA Press (2004) 37–42

[20] D. P. Bovet and M. Cesati *Understanding the Linux Kernel.* O'Reilly & Assoc. Inc., 2003.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

The influence of seasonality on dengue epidemiology, modelling and data analysis

Maíra Aguiar[1], Sebastien Ballesteros[1] and Nico Stollenwerk[1]

[1] *Centro de Matemática e Aplicações Fundamentais, Universidade de Lisboa, Portugal*

emails: mairа@ptmat.fc.ul.pt, sebastien@ptmat.fc.ul.pt, nico@ptmat.fc.ul.pt

Abstract

We investigate dengue fever epidemiology via a multi-strain model including seasonality and compare with empirical data from Thailand. The empirically observed fluctuations suggests a crucial role of determinstcal chaos in understanding the system dynamics.

Key words: dengue fever, seasonality, parameter estimation, deterministic chaos

1 Introduction

For two strain model to capture primary and secondary infection, we have the following SIR-type model, now labelling the SIR classes for the hosts that have seen the individual strains. Susceptibles to both strains (S) get infected with strain 1, (I_1), or strain 2, (I_2), with force of infection β_1 and β_2 respectively. They recover from infection with strain 1 (becoming R_1) or from strain 2 (becoming R_2), with recovery rate γ. In this recovered class, people have full immunity against the strain that they were exposed to and infected, and also, temporary immunity against the other strain (called the period of temporary cross-immunity). After this, with rate α, they enter again in the susceptible classes (S_1 respectively S_2), where the index represents the first infection strain. Now, S_1 can be reinfected with strain 2, to become (I_{12}), meeting I_2 with infection rate β_2 or meeting I_{12} with infection rate $\phi_2\beta_2$, and S_2 can be reinfected with strain 1 (becoming I_{21}) meeting I_1 or I_{21} with infections rates β_1 and $\phi_1\beta_1$ respectively.

The parameter ϕ in our model also acts decreasing the infectivity of secondary infection, where people are more likely to be hospitalized because of the severity of the disease (DHF/DSS), and do not contributed to the force of infection as much as people with first infection do. Finally, I_{12} and I_{21} go to the recovered class (R), immune against all strains. We include demography of the host population by denoting the birth and death rate by μ, assuming constant population size N, and seasonality by η. For

 ISBN 13: 978-84-613-5510-5

simplicity, we consider $\beta_1 = \beta_2 =: \beta_0$ and $\phi_1 = \phi_2$, i.e, no epidemiological asymmetry between strains.

The complete mean field ODE system for the two-strain epidemiological system is given by

$$\begin{aligned}
\frac{dS}{dt} &= -\frac{\beta}{N}S(I_1 + \phi_1 I_{21}) - \frac{\beta}{N}S(I_2 + \phi_2 I_{12}) + \mu(N - S) \\
\frac{dI_1}{dt} &= \frac{\beta}{N}S(I_1 + \phi_1 I_{21}) - (\gamma + \mu)I_1 \\
\frac{dI_2}{dt} &= \frac{\beta}{N}S(I_2 + \phi_2 I_{12}) - (\gamma + \mu)I_2 \\
\frac{dR_1}{dt} &= \gamma I_1 - (\alpha + \mu)R_1 \\
\frac{dR_2}{dt} &= \gamma I_2 - (\alpha + \mu)R_2 \\
\frac{dS_1}{dt} &= -\frac{\beta}{N}S_1(I_2 + \phi_2 I_{12}) + \alpha R_1 - \mu S_1 \\
\frac{dS_2}{dt} &= -\frac{\beta}{N}S_2(I_1 + \phi_1 I_{21}) + \alpha R_2 - \mu S_2 \\
\frac{dI_{12}}{dt} &= \frac{\beta}{N}S_1(I_2 + \phi_2 I_{12}) - (\gamma + \mu)I_{12} \\
\frac{dI_{21}}{dt} &= \frac{\beta}{N}S_2(I_1 + \phi_1 I_{21}) - (\gamma + \mu)I_{21} \\
\frac{dR}{dt} &= \gamma(I_{12} + I_{21}) - \mu R
\end{aligned} \tag{1}$$

where $\beta = \beta_0 * (1.0 + \epsilon * cos(\omega * t))$.

As initial conditions we take: For a constant population size $N = 100.0$, $S = 70, I_1 = 20, I_2 = 10, R_1 = 0, R_2 = 0, S_1 = 0, S_2 = 0, I_{12} = 0, I_{21} = 0, R = 0$. We fix the transition rates of the model as far as is known, as it follows, $\mu = 65y^{-1}, \gamma = 52, \alpha = 2, and \beta = 2\gamma$, and vary the most unknown parameter ϕ. The seasonality rate is $\epsilon = 0.1$.

We investigate the influence of the seasonal forcing in the infection rate on the dymanics, which was previously investigated giving deterministic chaos just with the multi-strain aspect of the dengue model [1, 2, 3].

Empirical data

Empirical data in form of time series of dengue incidences per month are in good quality avaliable e.g. from Thailand. The time series of provinces in the north of Thailand, here for the province of Chiang Mai, show irregular epidemics each year

 ISBN 13: 978-84-613-5510-5

with a smooth increase and decrease, Fig. 1. On the otehr hand, the times series of Bangkok, on which much research attention has been focused, presents rather erratic und uncorrelated dynamics, see Fig. 2, much more noisy than Fig. 1.

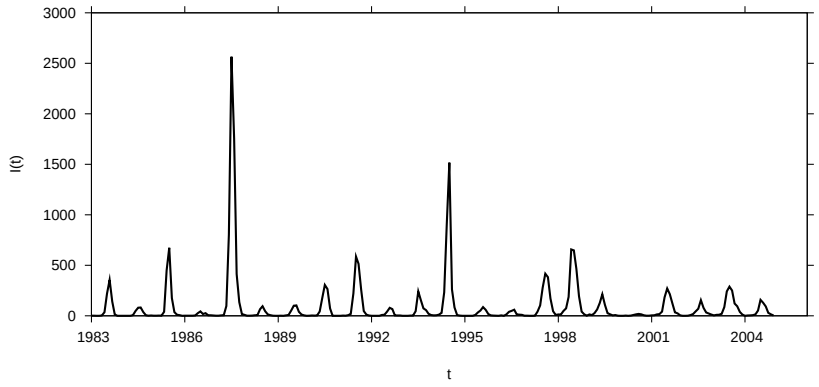


Figure 1: *Time series of dengue cases in Chiang Mai.*

The time series in Fig. 1 shows clear seasonality, but irregular maxima of dengue outbreaks each year. On these observations the modelling in terms of ODEs with possibly deterministically chaotic dynamics might have a chance to at least describe qualitatively the dynamics of the epidemiological system.

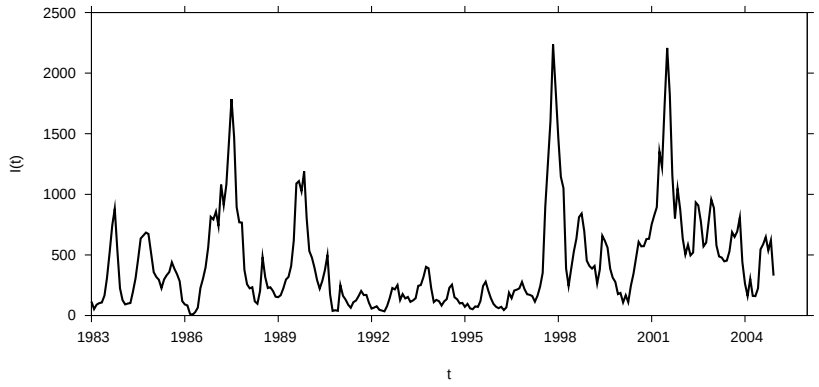


Figure 2: *Time series of dengue cases in Bangkok.*

To investigate the case further we add seasonality to the previously investigated dengue model with temporary cross-immunity which showed bifurcations and deterministic chaos in wide parameter regions. Whereas technical parameter estimation is notoriously difficult for chaotic time series but temporally local approaches possible (e.g. the Ionides approach [5] which will be described and applied below), we first invetigate which parameter regions are most likely to give seasonal outbreak with chaotic maxima, based on our previous experience on chaotic parameter regions of the non-seasonal model.

 ISBN 13: 978-84-613-5510-5

Simulations with seasonality

For the simulations we use the parameters as given above and vary the parameter of difference in force of infection between primary and secondary infection, the parameter ϕ.

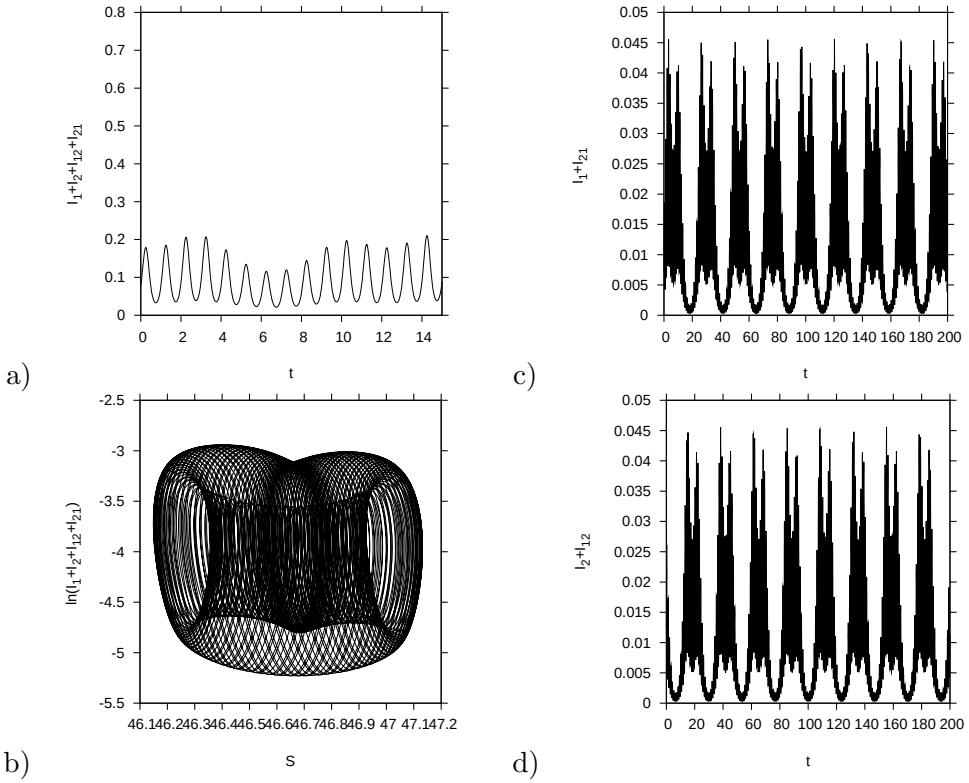


Figure 3: *Simulations for $\phi = 0.2$, for detailed description see main text.*

In Fig. 3 we show for $\phi = 0.2$ in a) a time series of the total number of infected with any strain, in b) the state spcae plot projection for susecptibles and logarithm of total infected, in c) and d) time series for infected with strain one respectively strain two. Plotting the individual strains on top of each other shows the phase relation between the two, in most cases the strains are out of phase. For $\phi = 0.2$ the non-seasonal model shows a limit cycle after a Hopf bifurcation from a stable fixed point. Hence here including seasonality we observe a torus in state space which is densely filled.

The following Figures are in the same formate, but for different ϕ-values, moving through the bifurcation diagram given before for the non-forced case. The bifurcation diagram with seasonality is not very informative since we have many local extrema even for the most simple case of the torus, as observed in Fig. 3 b).

 ISBN 13: 978-84-613-5510-5

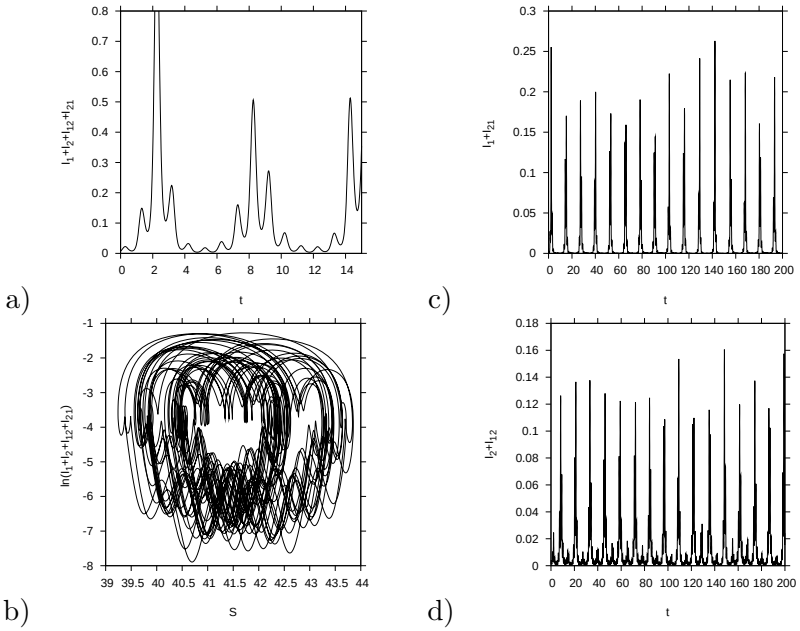


Figure 4: *Simulations for* $\phi = 0.5$.

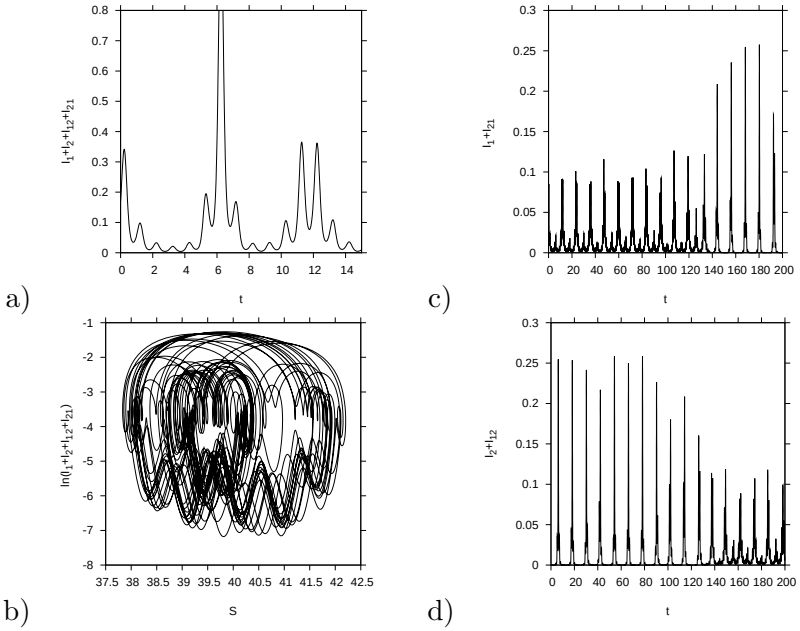


Figure 5: *Simulations for* $\phi = 0.6$.

 ISBN 13: 978-84-613-5510-5

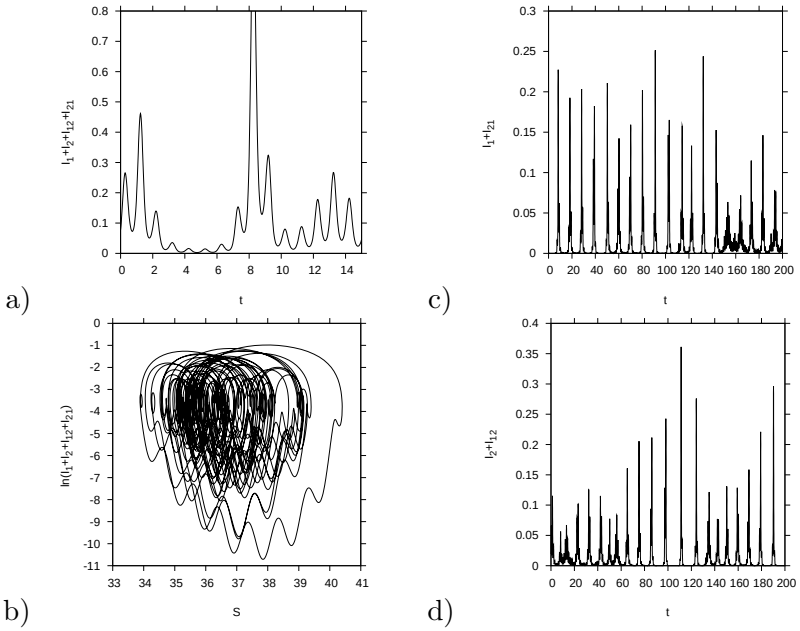


Figure 6: *Simulations for* $\phi = 0.8$.

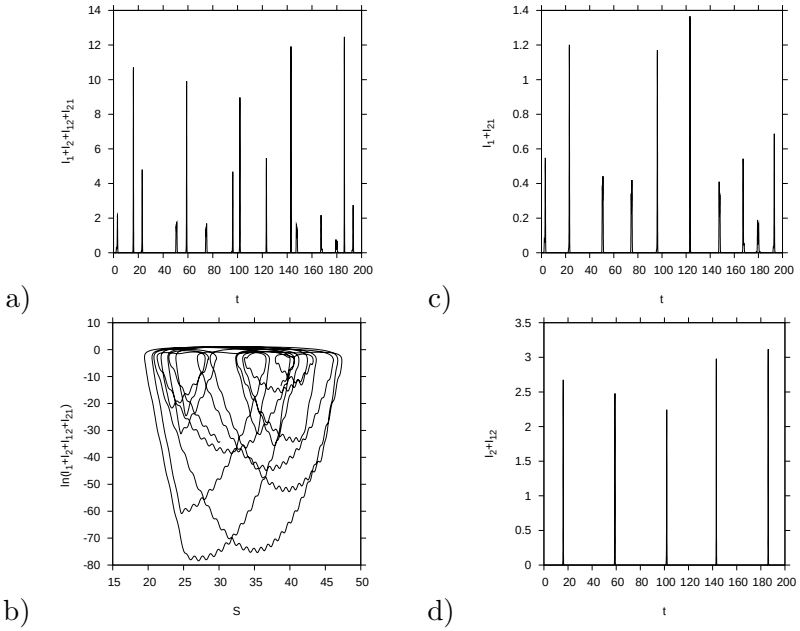


Figure 7: *Simulations for* $\phi = 1.0$.

 ISBN 13: 978-84-613-5510-5

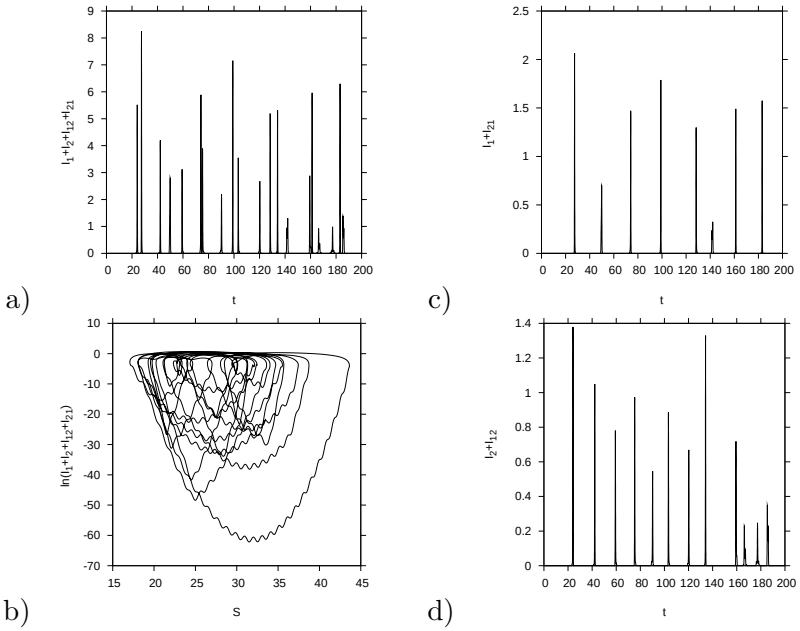


Figure 8: *Simulations for* $\phi = 1.5$.

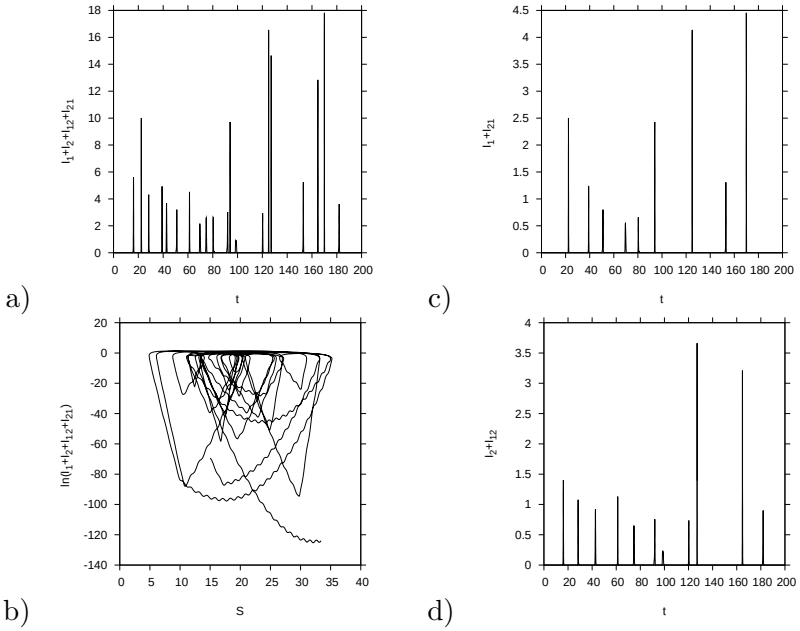


Figure 9: *Simulations for* $\phi = 2.5$.

 ISBN 13: 978-84-613-5510-5

For ϕ values up to nearly 1.0 we observe oscillations from the seasonal forcing on top of the chaotic outbreaks every years observed already for the non-forced model, whereas for $\phi = 1.0$ and further the troughs become so low that there are no outbreak for many years observed, making the interplay between non-seasonal chaotic signal and seasonality less plausible to describe the fluctuations observed in the data time series of northern Thailand, e.g. with much higher incidence rates than the noisy Bangkok data.

To further investigate the dynamics of the system, all other pareameters have to be varied, biologically most unknown are the seasonality and the temporary cross-immunity. But also the contact rate is not precisely known. A first attempt is shown in the next section.

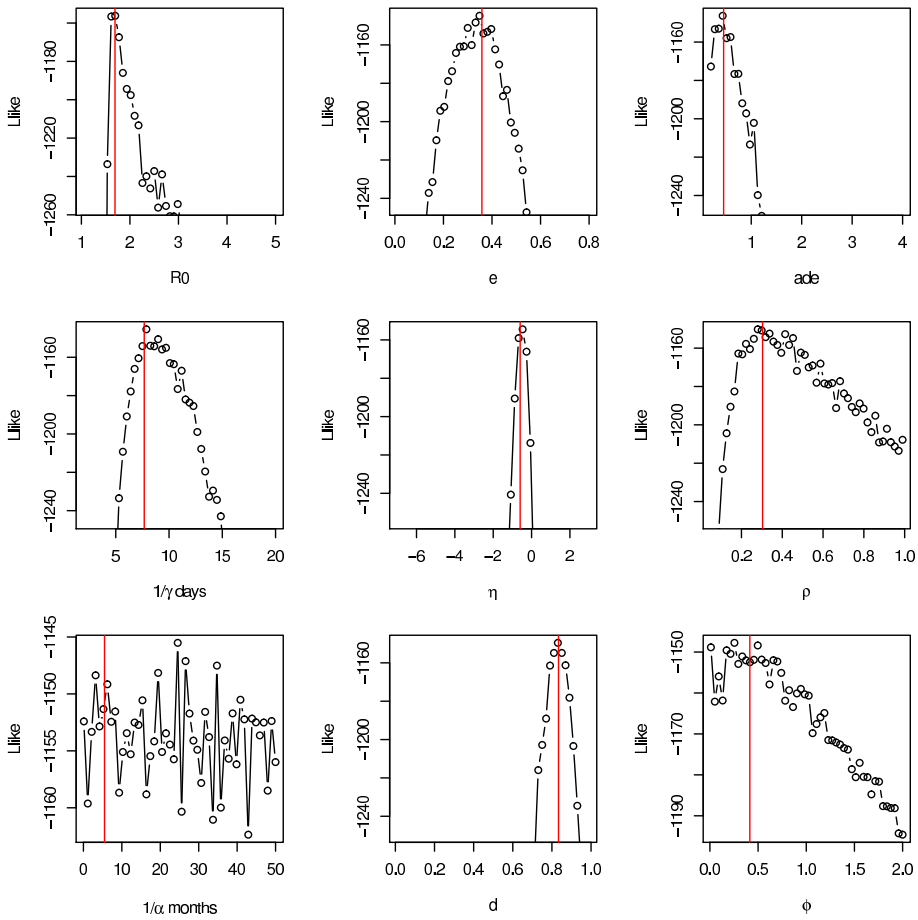


Figure 10: *Likelihood slices from the parameter estimation using the Ionides approach.*

 ISBN 13: 978-84-613-5510-5

First parameter estimation

To contrast our model Eq. (1) to data we have used monthly incidence of Dengue Hemorrhagic Fever (DHF) for the province of Chiang Mai in Thailand, see Fig. 1. As DHF is expected to occur mostly after secondary infections, we computed monthly incidence rates as $x_t = \int_{t-1}^{t} \gamma(I_{12}(t) + I_{21}(t))dt$ therefore considering only secondary infections. The obtained predicted values (x_t) were contrasted to the real monthly reported incidence, y_t, with an observation process characterized by a reporting rate (ρ). We set $Y_t|\rho_t, x_t \sim P(\rho_t, x_t)$ (where P stands for the Poisson distribution) and, to account for overdispersion, we allow variability in the reporting rate by assuming that $\rho_t \sim \Gamma(\frac{1}{\phi}, \rho\phi)$, where Γ is the gamma distribution. The observation process can then be fully described by a negative binomial distribution (NB): $Y_t|x_t \sim NB(\text{mean} = \rho x_t, \text{size} = \frac{1}{\phi})$.

For better agreement with the data, we have implemented a stochastic version of Eq. (1) using an Euler-multinomial approximation to the continuous-time Markov process. To avoid definitive extinctions, we add immigration to the model by introducing an extra parameter (η) in the force of infection, now defined by $\beta(I + \eta)$. As extra-demographic stochasticity is to be expected in data we added white noise to the transmission rate of the Markov chain compartment model using the general framework described in [4]. As DHF is strongly seasonal, see Fig. 1, seasonal forcing was also added on the transmission rate with the following parametric choice: $\beta(t) = \beta_0(1 + \epsilon \sin(2\pi(t + \delta)$), with R_0 roughly being β_0/γ. Whereas in bifurcation analysis the phase factor d does not play any decisive role, the parameter estimation process is very sensitive to it, and d has to be estimated from the data as well.

Parameter inference was achieved through an implementation of maximum likelihood via iterated filtering (MIF), the [5] approach written in C. Essentially, one starts with an number of simulations with ensembles of parameter values and intial conditions for the first part of the time series and compared the performance with the data part, then taking the best performers, one goes along the next piece of time series etc., hence a temporarilly local approach. On top there is a simulated annealing type procedure, while going several times over the whole time series. Finally the best performing sets of parameters, called particles, give the best estimate in this approach. Results were obtained with 1000 particles and an integration time step of 1 days.

Fig. 10 shows the best performing parameters in red, and in addition calculated likelihood slices to give an idea of the relyability of the estimates, confidence intervals. E.g. for some parameters the likelihood slice shows a well defined maximum, hence the corresponding parameter being well estimated, and for other parameters, namely here α. Finally, Fig. 11 shows a comparison of a realization of the model with the best parameter set and the actual empirical time series of dengue cases in Chiang Mai. Remember that the dynamics shows deterministic chaos, hence realizations and data set never can coinside, but describe the qualitative dynamic features of the system. Jus short term predictability can be obtained, limited by the prediction horizon given by the largest Lyapunov exponent.

Further investigations will be needed to obtain definite insight into the realistic

 ISBN 13: 978-84-613-5510-5

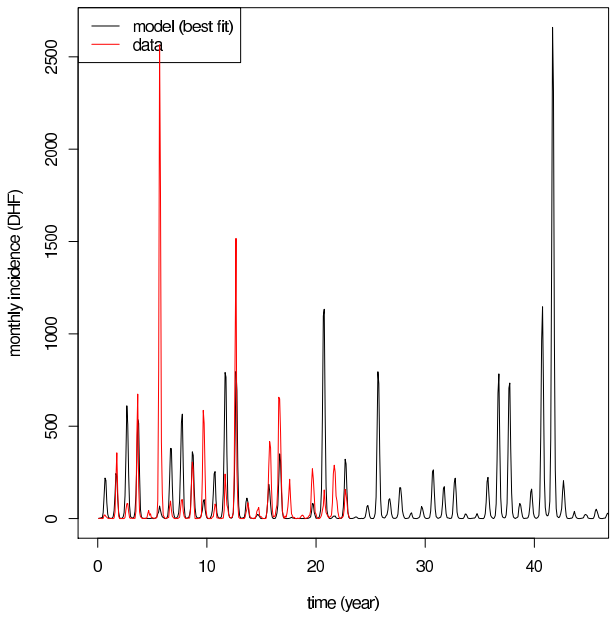


Figure 11: *Typical trajectory (corrected by reporting rate) of model of Eq. (1) for the maximum likelihood estimates of the parameters.*

parameters and the interplay between noise levels, as well dynamic as observation noise, with the deterministically chaotic dynamics.

Acknowledgements

This work has been supported by the European Union under FP7 in the EPIWORK project and by FCT, Portugal.

References

[1] M. Aguiar and N. Stollenwerk, *A new chaotic attractor in a basic multi-strain epidemiological model with temporary cross-immunity*, arXiv:0704.3174v1 [nlin.CD] (2007) (accessible electronically at http://arxive.org).

[2] M. Aguiar, B.W. Kooi and N. Stollenwerk, *Epidemiology of dengue fever: A model with temporary cross-immunity and possible secondary infection shows bifurcations and chaotic behaviour in wide parameter regions*, Math. Model. Nat. Phenom. **3** (2008) 48–70.

[3] M. Aguiar, N. Stollenwerk and B.W. Kooi *Torus bifurcations, isolas and chaotic attractors in a simple dengue model with ADE and temporary cross immunity,* in Proceedings of 8th Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2008, ISBN 978-84-612-1982-7 (2008).

[4] He, D., Ionides, E. L., and King, A. a. (2010). Plug-and-play inference for disease dynamics: measles in large and small populations as a case study. *Journal of the Royal Society, Interface / the Royal Society*, 7(43):271–83.

[5] Ionides, E., Breto, C., and King, A. (2006). Inference for nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 103(49):18438.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Sufficiency and duality in nondifferentiable multiobjective programming under generalized d_I- univexity

I. Ahmad and S. Al-Homidan

Department of Mathematics and Statistics, King Fahd University of Petroleum and Minerals, Dhahran-31261, Saudi Arabia

e-mails: *izharmaths@hotmail.com*, *homidan@kfupm.edu.sa*

Abstract

A nondifferentiable multiobjective programming problem is considered. We introduce a new class of generalized d_I-univexity in which each component of the objective and constraint functions is directionally differentiable in its own direction d_i. Based upon these generalized functions, sufficient optimality conditions are established for a feasible point to be efficient and properly efficient under the generalised d_I-univexity requirements, Moreover, weak, strong and strict converse duality theorems are also derived for Mond-Weir type dual programs.

Key words: Multiobjective programming, Efficient solutions, Properly efficient solutions Generalized d_I-univexity, Sufficiency, Duality .
AMS Subject Classification: 90C26, 90C29, 90C46, 49J52

1. Introduction

The field of multiobjective programming, also known as vector programming, has grown remarkably in different directions in the setting of optimality conditions and duality theory. It has been enriched by the applications of various types of generalizations of convexity theory, with and without differentiability assumptions, and in the framework of continuous time programming, fractional programming, inverse vector optimization, saddle point theory, symmetric duality and vector variational inequalities etc.

Hanson [1] introduced a class of functions by generalizing the difference vector $x - \overline{x}$ in the definition of a convex function to any vector function $\eta(x, \overline{x})$. These functions were named invex by Craven [2] and η-convex by Kaul and Kaur [3]. Hanson and Mond [4] defined two new classes of functions called Type I and Type II functions, which were further generalized to pseudo Type I and quasi Type I functions by Rueda and Hanson [5]. Zhao [6] established optimality conditions and duality in nonsmooth scalar programming problems assuming Clarke[7] generalized subgradients under Type I functions.

Kaul et al. [8] extended the concept of type I functions from a single objective to a multiobjective programming problem by defining the type I and its various generalizations. They investigated necessary and sufficient optimality conditions and derived Wolfe

 ISBN 13: 978-84-613-5510-5

type and Mond-Weir type duality results. Suneja and Srivastava [9] introduced generalized d-type I functions in terms of directional derivative for a multiobjective programming problem and discussed Wolfe type and Mond-Weir type duality results. In [10], Kuk and Tanino derived optimality conditions and duality theorems for non-smooth multiobjective programming problems involving generalized Type I vector valued functions. Gulati and Agarwal [11] discussed sufficiency and duality results for nonsmooth multiobjective problems and $(F, \alpha, \rho, d-$ type I functions.

Antczak [12] studied d-invexity is one of the generalization of invex function, which is introduced by [13]. In [12], Antczak established, under weaker assumptions than Ye, the Fritz John type and Karush-Kuhn-Tucker type necessary optimality conditions for weak Pareto optimality and duality results which have been stated in terms of the right differentials of functions involved in the considered multiobjective programming problem. Many authors [14, 15, 16] proved that the Karush-Kuhn-Tucker type necessary conditons [12] are sufficient conditions under various generalized d-invex functions. Recently, Antczak [17] corrrected the Karush-Kuhn-Tucker necessary conditions in [17] and discussed the sufficiency and duality under $d-r-$ Type I functions. More recently, Silmani and Radjef [18] introduced generalzed d_I-invexity in which each component of the objective and constraint functions is directionally differentiable in its own direction and established the necessary and sufficient conditions for efficient and properly efficient solutions. They also observed the Karush-Kuhn-Tucker sufficient conditions [14, 15, 16] are not applicable. The duality results for a Mond-Weir type dual are derived in [18].

In this paper, we introduce d_I-V- univexity and generalized $d_I - V-$ univexity in which each component of the objective and constraint functions of a multiobjective programming problem is directionally differentiable in its own direction d_i. Various Karush-Kuhn-Tucker sufficient optimality conditions for efficient and properly efficient solutions to the problem are establish involving new classes of semidirectionally differentiable generalized type I functions. Moreover, usual duality theorems are discussed for a Mond-Weir type dual involving aforesaid assumptions. The results in this paper extend many earlier work appeared in the literature.

2 Preliminaries and definitions

The following conventions for equalities and inequalities will be used. If $x = (x_1....,x_n), y = (y_1....,y_n) \in \mathbb{R}^n$, then $x = y \Leftrightarrow x_i = y_i, \quad i = 1,...,n;$, $x < y \Leftrightarrow x_i < y_i, \quad i = 1,...,n;$, $x \leqq y \Leftrightarrow x_i \leqq y_i, \quad i = 1,...,n;$, $x \leq y \Leftrightarrow x \leqq y \quad and\, x \neq y.$, We also note $\mathbb{R}^q_{\leqq}$ (resp.$\mathbb{R}^q_{\geq}$ or $\mathbb{R}^q_{\geq}$) the set of vectors $y \in \mathbb{R}^q$ with $y \geqq 0$(resp. $y \geq 0$ or $y > o$)

Definition 1 [19]. *Let D be a nonempty subset of $\mathbb{R}^n$, $\eta : D \times D \to \mathbb{R}^n$ and let x_0 be an arbitrary point of D.The set D is said to be invex at x_0 with respect to η, if for each $x \in D$,*

$$x_0 + \lambda\eta(x, x_0) \in D, \forall\lambda \in [0, 1].$$

D is said to be an invex set with respect to η, if D is invex at each $x_0 \in D$ with respect to the same η.

 ISBN 13: 978-84-613-5510-5

Definition 2 [20]. *Let $D \subseteq \mathbb{R}^n$ be an invex set with respect to $\eta : D \times D \to \mathbb{R}^n$.A function $f : D \to \mathbb{R}$ is called pre-invex on D with respect to η, if for all $x, x_0 \in D$,*

$$\lambda f(x) + (1-\lambda) f(x_0) \geqq f(x_0 + \lambda\eta(x, x_0)), \forall \lambda \in [0,1].$$

Definition 3 [12]. *Let $D \subseteq \mathbb{R}^n$ be an invex set with respect to $\eta : D \times D \to \mathbb{R}^n$. A m-dimensional vector valued function $\Psi : D \to \mathbb{R}^m$ is pre-invex with respect to η, if each of its components is pre-invex on D with respect to the same function η.*

Definition 4[7]. *Let D be a nonempty open set in $\mathbb{R}^n$. A function $f : D \to \mathbb{R}$ is said to be locally Lipschitz at $x_0 \in D$, if there exist a neighborhood $\upsilon(x_0)$ of x_0 and a constant $K > 0$ such that*

$$|f(y) - f(x)| \leqq K\|y - x\|, \quad \forall \quad x, y \in \upsilon(x_0),$$

where $\|.\|$ denotes the Euclidean norm. We say that f is locally Lipschitz on D if its locally Lipschitz at any point of D.

Definition 5 [7]. *If $f : D \subseteq \mathbb{R}^n \to R$ is locally Lipschitz at $x_0 \in D$, the clarke generalized directional derivative of f at x_0 in the direction $d \in \mathbb{R}^n$,denoted by $f^0(x_0; d)$ is given by*

$$f^0(x_0; d) = \lim_{y \to x_0} \sup_{t \to 0^+} \frac{f(y + td) - f(y)}{t}.$$

And the usual one-sided directional derivative of f at x_0 in the direction d is defined by

$$f'(x_0; d) = \lim_{\lambda \to 0^+} \frac{f(x_0 + \lambda d) - f(x_0)}{\lambda},$$

whenever this limit exists. Obviously, $f^0(x_0; d) \geqq f'(x_0; d)$.

We say that f is directionally differentiable at x_0 if its directional derivative $f'(x_0; d)$ exists finite for all $d \in \mathbb{R}^n$.

Definition 6[13]. *Let $f : D \to \mathbb{R}^n$ be a function defined on a nonempty open set $D \subset \mathbb{R}^n$ and directionally differentiable at $x_0 \in D$. f is called d-invex at x_0 on D with respect to η, if there exists a vector function $\eta : D \times D \to \mathbb{R}^n$,* such that for any $x \in D$,

$$f_i(x) - f_i(x_0) \geqq f_i^{'}(x_0; n(x, x_0)), \text{ for all } i = 1, ..., N, \tag{1}$$

where $f_i^{'}(x_0; n(x, x_0))$ denotes the directional derivative of f_i at x_0 in the direction $n = (x, x_0)$: $f_i^{'}(x_0; n(x, x_0)) = \lim_{\lambda \to 0^+} \frac{f_i(x_0 + \lambda\eta(x, x_0)) - f_i(x_0)}{\lambda}$.
If inequalities (1) are satisfied at any point $x_0 \in D$, then f is said to be d-invex on D with respect to η.

Definition 7[18]. *Let D be a nonempty set in $\mathbb{R}^n$ and $\phi : D \times D \to \mathbb{R}^n$ a function.*

 ISBN 13: 978-84-613-5510-5

- *We say that $f : D \to \mathbb{R}$ is a semi-directionally differentiable at $x_0 \in D$,if there exist a nonempty subset $S \subset \mathbb{R}^n$ such that $f'(x_0; d)$ exists finite for all $d \in S$*

- *We say that f is a semi-directionally differentiable at $x_0 \in D$ in the direction $\phi(x, x_0)$, if its directional derivative $f'(x_0; \phi(x, x_0))$ exists finite for all $x \in D$.*

Definition 8[18]. Let $f : D \to \mathbb{R}^n$ be a function defined on a nonempty open set $D \subset \mathbb{R}^n$ and for all $i = 1 \cdots N, f_i$ is semi-directionally differentiable at $x_0 \in D$ in the direction $\eta_i : D \times D \to \mathbb{R}^n$. f is called d_I-invex at x_0 on D with respect to $(\eta_i)_{i=\overline{1,N,}}$ if for any $x \in D$,

$$f_i(x) - f_i(x_0) \geqq f_i^{'}(x_0; \eta_i(x, x_0)), for all\, i, ..., N, \tag{2}$$

where $f_i^{'}(x_0; \eta_i(x, x_0))$ denotes the directional derivative of f_i at x_0 in the direction $\eta_i(x, x_0)$: $f_i^{'}(x_0; \eta_i(x, x_0)) = \lim\limits_{\lambda \to 0^+} \dfrac{f_i(x_0 + \lambda \eta_i(x, x_0)) - f_i(x_0)}{\lambda}$.
If inequalities (2) are satisfied at any point $x_0 \in D$, then f is said to be d_I-invex on D with respect to $(\eta_i)_{i=\overline{1,N,}}$

Consider the following multiobjective programming problem

$$\begin{aligned} \text{(MP)} \quad & \text{Minimize } f(x) = (f_1(x), f_2(x), ..., f_N(x)) \\ & \text{Subject to } g(x) \leqq 0, \\ & x \in D, \end{aligned}$$

where $f : D \to \mathbb{R}^N$, $g : D \to \mathbb{R}^k$, D is a nonempty open subset of $\mathbb{R}^n$. Let $X = \{x \in D : g(x) \leqq 0\}$ be the set of feasible solutions of (MP). For $x_0 \in D$,we denote by $J(x_0)$ the set $\{j \in \{1, ..., k\} : g_j(x_0) = 0\}, J = |J(x_0)|$ and by $\widetilde{J}(x_0)$(resp .$\widetilde{J}(x_0)$)the set $\{j \in \{1, ..., k\} : g_j(x_0) < 0$(resp.$g_j(x_0) > 0)\}$. we have $J(x_0) \cup \widetilde{J}(x_0) \cup \overline{J}(x_0) = \{1, ..., k\}$ and if $x_0 \in X, \overline{J}(x_0) = \varnothing$.

We recall some optimality concepts, the most often studied in the literature, for the problem (MP).

Definition 9 *A point $x_0 \in X$ is said to be a local weakly efficient solution of the problem (MP),if there exists a neighborhood $N(x_0)$ around x_0 such that*

$$f(x) \not< f(x_0) \text{ for all } x \in N(x_0) \cap X$$

Definition 10 *A Point $x_0 \in X$ is said to be a weakly efficient (an efficient) solution of the problem (MP), if there exists no $x \in X$ such that*

$$f(x) < f(x_0) (f(x) \leq f(x_0)).$$

Definition 11 *An efficient solution $x_0 \in X$ of (MP) is said to be properly efficient,if there exists a positive real number M such that inequality*

$$f_i(x_0) - f_i(x) \leqq M[f_j(x) - f_j(x_0)]$$

 ISBN 13: 978-84-613-5510-5

is verified for all $i \in \{1, \cdots, N\}$ and $x \in X$ such that $f_i(x) < f_i(x_0)$, and for a certain $j \in \{1, \cdots, N\}$ such that $f_i(x) > f_j(x_0)$.

Following Jeyakumar and Mond [21] ,Kaul et al.[8]and Slimani and Radjef [18],we give the following definitions.

Definition 12 *(f, g) is d_I-V-univex type I at $x_0 \in D$ if there exist positive real valued functions α_i and β_j defined on $X \times D$, nonnegative functions b_0 and b_1,also defined on $X \times D, \phi_0 : R \to R, \phi_1 : R \to R; \eta_i : X \times D \to R^n$, and $\theta_j : X \times D \to R^n$ such that*

$$b_0(x, a)\phi_0[f_i(x) - f_i(x_0)] \geqq \alpha_i(x, a) f_i^{'}(x_0; \eta_i(x, x_0)) \tag{3}$$

and

$$-b_1(x_1 x_0)\phi_1[g_j(x_0)] \geqq \beta_j(x, x_0) g_j^{'}(x_0; \theta_j(x, x_0)) \tag{4}$$

for every $x \in X$ and for all $i = 1, 2, \cdots, p \quad$ and $j = 1, 2, \cdots, m$.

If the inequality in (3) is strict (whenever $x \neq x_0$), we say that (MP) is of semistrictly d_I-V-univex type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}} \quad and (\theta_j)_{j=\overline{1,k}}$.

Definition 13 *(f, g) is quasi-d_I-V-univex type I at $x_0 \in D$ if there exist positive real valued functions α_i and β_j, defined on $X \times D$, nonnegative functions $b_0$and b_1, also defined on $X \times D, \phi_0 : R \to R, \phi_1 : R \to R$ and $(N + k)$ dimensional vector functions $\eta_i : X \times D \to R^n, i = \overline{1, N}$ and $\theta_j : X \times D \to R^n, j = \overline{1, k}$ such that for some vectors $\lambda \in R^N_{\geqq}$ and $\mu \in R^k_{\geqq}$:*

$$b_0(x, 0)\phi_0 \left[\sum_{i=1}^{N} \lambda_i \alpha_i(x, x_0)(f_i(x) - f_i(x_0)\right] \leqq 0 \Rightarrow \sum_{i=1}^{N} \lambda_i f_i^{'}(x_0; \eta_i(x, x_0)) \leqq 0 \quad \forall x \in X \tag{5}$$

and

$$b_1(x, 0)\phi_1 \left[\sum_{j=1}^{k} \mu_j \beta_j(x, x_0) g_j(x_0)\right] \geqq 0 \Rightarrow \sum_{j=1}^{k} \mu_j g_j^{'}(x_0; \theta_j(x, x_0)) \leqq 0 \quad \forall x \in X \tag{6}$$

If the second inequality in (5) is strict ($x \neq x_0$), we say that (MP) is of semi-strictly quasi d_I-V-univex type I at $X_o$with respect to $(\eta_i)_{i=\overline{1,N}} \quad and (\theta_j)_{j=\overline{1,k}}$.

Definition 14 *(f, g) is pseudo-d_I-V-univex type I at $x_0 \in D$ if there exist positive real valued functions α_i and β_j, defined on $X \times D$, nonnegative functions b_0 and b_1,also defined on $X \times D$, $\phi_0 : R \to R, \phi_1 : R \to R$ and $(N + k)$ dimensions vector functions $\eta_i : X \times D \to R^n, i = \overline{1, N}$ and $\theta_j : X \times D \to R^n, j = \overline{1, k}$ such that for some vectors $\lambda \in R^N_{\geqq}$ and $\mu \in R^k_{\geqq}$:*

$$\sum_{i=1}^{N} \lambda_i f_i^{'}(x_0; \eta_i(x, x_0)) \geqq 0 \Rightarrow b_0(x, x_0)\phi_0 \left[\sum_{i=1}^{N} \lambda_i \alpha_i(x, x_0)(f_i(x) - f_i(x_0))\right] \geqq 0 \forall x \in X \tag{7}$$

 ISBN 13: 978-84-613-5510-5

and

$$\sum_{j=1}^{k} \mu_j g_j^{'}(x_0; \theta_j(x, x_0)) \geqq 0 \Rightarrow b_1(x, x_0)\phi_1 \left[\sum_{j=1}^{k} \mu_j \beta_j(x, x_0) g_j(x_0) \right] \leqq 0 \forall x \in X \qquad (8)$$

Definition 15 (f, g) *is quasi pseudo-d_I-V-univex type I at $x_0 \in D$ if there exist positive real valued functions α_i and β_j, defined on $X \times D$, nonnegative functions b_0 and b_1,also defined on $X \times D$,$\phi_0 : R \to R, \phi_1 : R \to R$ and $(N + k)$ dimensions vector functions $\eta_i : X \times D \to R^n, i = \overline{1, N}$ and $\theta_j : X \times D \to R^n, j = \overline{1, k}$ such that the relation (5) and (8) are satisfied. If the second inequality in (8) is strict ($x \neq x_0$, we say that (VP) is of quasi strictly-pseudo $d_I - V - typeI$ at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$*

Definition 16 (f, g) *is pseudoquasi -d_I-V-univex type I at $x_0 \in D$ if there exist positive real valued functions α_i and β_j, defined on $X \times D$, nonnegative functions b_0 and b_1,also defined on $X \times D$,$\phi_0 : R \to R, \phi_1 : R \to R$ and $(N + k)$ dimensions vector functions $\eta_i : X \times D \to R^n, i = \overline{1, N}$ and $\theta_j : X \times D \to R^n, j = \overline{1, k}$ such that $\mu \in R^k_{\geqq}$ the relations (7) and (6) are satisfied. If the second inequality in (7) is strict ($x \neq x_0$),*we say that (VP) is of strictly-pseudo quasi $d_I - \mathrm{V} - typeI$ at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$

3.Optimality conditions

In this section , we discuss some sufficient conditions for a point to be an efficient or properly efficient for (MP) under generalized $d_I - V-$ univex type I assumptions.

Theorem 3.1. Let x_0 be a feasible solution for (MP) and suppose that there exist $(N + J)$ vector functions $\eta_i : X \times X \to R^n, i = \overline{1, N}, \theta_j : X \times X \to R^n, j \in J(x_0)$ and scalars $\overline{\lambda}_i \geqq 0,\ i = \overline{1, N}, \sum_{i=1}^{N} \lambda_i = 1;\ \overline{\mu}_j \geqq 0,\ j \in J(x_0)$ such that

$$\sum_{i=1}^{N} \overline{\lambda}_i f_i^{'}(x_0; \eta_i(x, x_0) + \sum_{j \in J(x_0)} \overline{\mu}_j g_j^{'}(x_0; \theta_j(x, x_0)) \geqq 0 \quad \forall x \in X \qquad (9)$$

Further, assume that one of the following conditions is satisfied:

(a) (i) (f, g) is quasi strictly-pseudo $d_I - V-$ univex type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j \in J(x_0)}, \lambda, \mu$ and for some positive functions $\alpha_i, i = \overline{1, N}, \beta_j, j \in J(x_0)$,

(ii) for any $u \in R$, $u \leqq 0 \Rightarrow \phi_0(u) \leqq 0; \phi_1(u) < 0 \Rightarrow u < 0; b_0(x, x_0) > 0, b_1(x, x_0) > 0$;

(b) (i) (f, g) is strictly-pseudo $d_I - V-$ univex type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j \in J(x_0)},\ \lambda,\ \mu$ and for some positive functions $\alpha_i,\ i = \overline{1, N},\ \beta_j, j \in J(x_0)$,

(ii) for any $u \in R$, $\phi_0(u) > 0 \Rightarrow u > 0; u \geqq 0 \Rightarrow \phi_1(u) \geqq 0, b_0(x, x_0) > 0, b_1(x, x_0) \geqq 0$;

Then u is an efficient solution for (MP).

 ISBN 13: 978-84-613-5510-5

Proof: Condition (a). Suppose that x_0 is not an efficient solution of (MP). Then there exists an $x \in X$ such that

$$f(x) \leq f(x_0),$$

which implies that

$$\sum_{i=1}^{N} \overline{\lambda}_i \alpha_i(x, x_0)[f_i(x) - f_i(x_0)] \leqq 0. \tag{10}$$

Since $b_0(x, x_0) > 0$; $u \leqq 0 \Rightarrow \phi_0(u) \leqq 0$, the above inequality gives

$$b_0(x, x_0)\phi_0[\sum_{i=1}^{N} \overline{\lambda}_i \alpha_i(x, x_0)[f_i(x) - f_i(x_0)] \leqq 0.$$

From the above inequality and Hypothesis (a)(i), we have

$$\sum_{i=1}^{N} \overline{\lambda}_i f_i^{'}(x_0; \eta_i(x, x_0)) \leqq 0.$$

By using the inequality (9) we deduce that

$$\sum_{j \in J(x_0)} \overline{\mu}_j g_j^{'}(x_0; \theta_j(x, x_0)) \geqq 0,$$

which implies from the condition a(ii) that

$$b_1(x, x_0)\phi_1[\textstyle\sum_j \overline{\mu}_j \beta_j(x, x_0) g_j(x_0)] < 0.$$

Since $b_1(x, x_0) > 0; \phi_1(u) < 0 \Rightarrow (u) < 0$, we get

$$\sum_{j \in J(x_0)} \overline{\mu}_j \beta_j(x, x_0) g_j(x_0) < 0. \tag{11}$$

As $\overline{\lambda} \geqq 0$ and $g_j(x_0) = 0; \forall j \in J(x_0)$, it follows that $\overline{\lambda}_j g_j(x_0) = 0, \forall j \in J(x_0)$, which implies that

$$\sum_{j \in J(x_0)} \overline{\mu}_j \beta_j(x, x_0) g_j(x_0) = 0.$$

The above equation contradicts inequality (11) and hence the conclusion of the theorem follows:

Condition (b) : Since $g_j(x_0) = 0, \mu_j \geqq 0, \forall j \in J(x_0)$, and $\beta_j(x, x_0) > 0, j \in J(x_0)$, we obtain

$$\sum_{j \in J(x_0)} \mu_j \beta_j(x, x_0) g_j(x_0) = 0, \quad \forall x \in X.$$

By Hypothesis (b)(ii), we get

$$b_1(x, x_0)\phi_1[\sum_{j \in J(x_0)} \mu_j \beta_j(x, x_0)] \geqq 0.$$

 ISBN 13: 978-84-613-5510-5

From the above inequality and the Hypothesis (b) (i)(in view of reverse implication in (8), if follows that

$$\sum_{j\in J(x_0)} \mu_j g_j^{'}(x_0;\theta_j(x,x_0)) < 0, \quad \forall x \in X\backslash\{x_0\}.$$

By using inequality (9), we deduce that

$$\sum_{i=1}^{N} \lambda_i f_i^{'}(x_0;\eta_i(x,x_0)) > 0, \quad \forall x \in X\backslash\{x_0\}. \tag{12}$$

which by virtue of relation (7) implies that

$$b_0(x,x_0)\phi_0[\sum_{i=1}^{N} \lambda_i\alpha_i(x,x_0)(f_i(x)-f_i(x_0))] > 0, \quad \forall x \in X\backslash\{x_0\}.$$

The above inequality along with Hypothesis (b)(ii) gives

$$\sum_{i=1}^{N} \lambda_i\alpha_i(x,x_0)(f_i(x)-f_i(x_0)) > 0 \quad \forall x \in X\backslash\{x_0\}. \tag{13}$$

Since (17) and (18) contradicts each other,and hence the conclusion follows:

Theorem 3.2. Let x_0 be a feasible solution for (MP) and suppose that there exist $(N+J)$ vector functions $\eta_i : X\times X \to R^n,\ i=\overline{1,N},\ \theta_j : X\times X \to R^n j \in J(x_0)$ and scalars $\overline{\lambda}_i \geqq 0,\ i=\overline{1,N},\ \sum_{i=1}^{N}\lambda_i = 1,\ \overline{\mu}_j \geqq 0,\ j\in J(x_0)$ such that (9) of Theorem 3.1 is satisfied.
Moreover, assume that one of the following conditions is satisfied.

(a)(i) (f,g) is pseudo quasi $d_I - V-$ univex type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j\in J(x_0)}, \lambda, \mu$ and for some positive functions $\alpha_i, i=\overline{1,N}$ and $\beta_j, j\in J(x_0)$,

(ii) for any $u\in R$, $u\geqq 0 \Rightarrow \phi_1(u)\geqq 0,\ \phi_0(u)\geqq 0 \Rightarrow u\geqq 0$, $b_0(x,x_0)>0, b_1(x,x_0)\geqq 0$;

(b)(i) (f,g) is strictly pseudo $d_I - V-$ univex type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j\in J(x_0)}, \lambda, \mu$ and for positive functions $\alpha_i = \overline{1,N}$ and $\beta_j, j\in J(x_0)$,

(ii) for any $u\in R\ u\leqq 0 \Rightarrow \phi_0(u)\leqq 0; u\geqq 0 \Rightarrow \phi_1(u)\geqq 0; b_0(x,x_0) > 0, b_1(x,x_0)\geqq 0$;

Then x_0 is an efficient solution for (MP). Further Suppose that these exist positive real numbers n_i, m_i such that $n_i < \alpha_i(x,x_0) < m_i, i=\overline{1,N}$ for all feasible x.Then x_0 is a properly efficient solution for (MP).

 ISBN 13: 978-84-613-5510-5

Proof: (a): Suppose that x_0 is not an efficient solution of (MP).Then there exists an $x \in X_0$ such that $f(x) \leq f(x_0)$ which implies that

$$\sum_{i=1}^{N} \lambda_i \alpha_i(x, x_0)(f_i(x) - f_i(x_0)) < 0. \tag{14}$$

Since $g_j(x_0) = 0, \mu_j \geqq 0$ and $\beta_j, (x, x_0) > 0 \quad \forall j \in J(x_0)$ we obtain

$$\sum_{j \in J(x_0)} \mu_j \beta_j(x, x_0) g_j(x_0) = 0.$$

From the above inequality and hypothesis a(ii), we have

$$b_1(x, x_0)\phi_1[\sum_{j \in J(x_0)} \mu_j \beta_j(x, x_0) g_j(x_0)] \geqq 0.$$

Using hypothesis a(i), we deduce that

$$\sum_{j \in J(x_0)} \mu_j \beta_j(x, x_0) g_j^{'}(x_0; \theta_j(x, x_0)) \leqq 0. \tag{15}$$

The inequalities (9) and (14) yield that

$$\sum_{i=1}^{N} \lambda_i f_i^{'}(x_0; \eta_i(x, x_0)) \geqq 0,$$

which by Hypothesis (a)(i), we obtain

$$b_0(x, x_0)\phi_0[\sum_{i=1}^{N} \lambda_i \alpha_i(x, x_0)(f_i(x) - f_i(x_0))] \geqq 0, \tag{16}$$

The inequality (16) and Hypothesis (a)(ii) give

$$\sum_{i=1}^{N} \lambda_i \alpha_i(x, x_0)(f_i(x) - f_i(x_0) \geqq 0 \tag{17}$$

Since (14) and (17) contradict each other,we conclude that x_0 is not an efficient solution of (MP). The properly efficient solution follows as in Hanson et al. [22]. For the proof of part (b), we proceed as in part (b) of Theorem (3.1), we get inequality (17). Thus complete the proof.

4. Mond-Weir type duality

Consider the following multiobjective dual to problem (MP)

(MD) Maximize $f(y) = (f_1(y), f_2(y), \ldots f_N(y))$

 ISBN 13: 978-84-613-5510-5

subject to

$$\sum_{i=1}^{N} \lambda_i f_i^{'}(y;\eta_i(x,y)) + \sum_{j=1}^{k} \mu_j g_j^{'}(y;\theta_j(x,y)) \geqq 0, \quad \forall x \in X$$

$$\mu_j g_j(y) \geqq 0,\ j = 1,2,....,k, \quad y \in D, \lambda \in R^N_{\geqq}, \mu \in R^k_{\geqq}$$

$$\eta_i : X \times D, \quad \forall i = 1,2,...,N, \quad \theta_j : X \times D \to R^n, j = 1,2,...,k.$$

Let Y be the set of feasible solutions of problem (MD); that is,

$$Y = \{(y,\lambda,\mu,(\eta_i)_i,(\theta_j)_j) : \sum_{i=1}^{N} \lambda_i f_i^{'}(y;\eta_2(x,y)) + \sum_{j=1}^{k} \mu_j g_j^{'}(y;\theta_j(x,y)) \geqq 0,$$

$$\mu_j g_j(y) \geqq 0, \forall x \in X; y \in D, \lambda \in R^N_{\geqq}, \mu \in R^k_{\geqq}; \eta_i : X \times D \to R^n \quad \forall 1,2,....,N;$$

$$\theta_j : X \times D \to R^n \quad \forall j = 1,2,...,k\}.$$

We denote by $P_{rD}Y$, the projection of set Y on D.

We state the following duality theorems.

Theorem 4.1 (Weak Duality). Let x and $(y,\lambda,\mu,(\eta_i)_{i=\overline{1,N}},(\theta_j)_j = \overline{1,k})$ be feasible solution for (MP) and (MD) respectively. Moreover, assume that one of the following conditions is satisfied:

(a)(i) (f,g) is pseudo quasi d_I-V-univex type I at y with respect to $\lambda > 0,\ \mu,\ (\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j=\overline{1,k}}$ and for some positive functions α_i, β_j for $i = 1,2,..,N$ and $j = 1,2,..,k$,

(ii) for any $u \in R\ \phi_0(u) \geqq 0 \Rightarrow u \geqq 0;\ u \geqq 0 \Rightarrow \phi_1(u) \geqq 0; \quad b_0(x,y) > 0,\ b_1(x,y) \geqq 0;$

(b)(i) (f,g) is strictly-pseudo quasi d_I-V-univex type I at y with respect to $\lambda,\ \mu,\ (\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j=\overline{1,k}}$ and for some positive function α_i, β_j for $i = 1,2,...,N$ and $J = 1,2,...,k$,

(ii) for any $u \in R$, $\phi_0(u) \geqq 0 \Rightarrow u > 0;\ u \geqq 0 \Rightarrow \phi_1(u) \geqq 0; b_1(x,y) \geqq 0,\ b_0(x,y) > 0;$

(c)(i) (f,g) is quasi strictly-pseudo $d_I - V$-univex type I at y with respect to $\lambda, \mu, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j=\overline{1,k}}$ and for some positive functions α_i, β_j for $i = 1,2,..,N$ and $j = 1,2,...k$,

(ii) for any $u \in R$,$\phi_0(u) > 0 \Rightarrow u > 0;\ u > 0 \Rightarrow \phi_1(u) > 0; \quad b_0(x,y) > 0,\ b_1(x,y) > 0.$

Then $f(x) \nleqq f(y)$.

Remark 1: If we omit the assumption $\lambda > 0$ in the condition (a)(i) or the word "strictly" in the condition (b),we obtain, for this part of theorem,$f(x) \not< f(y)$.

Theorem 4.2 (Strong Duality).Let x_0 be a weakly efficient solution for (MP).Assume that the function g satisfies the d_I- constraint qualification at x_o with respect to $(\theta_j)_{j=\overline{1,k}}$.

 ISBN 13: 978-84-613-5510-5

Then there exist $\lambda \in R^N_{\overline{\geq}}$ and $\mu \in R^K_{\overline{\geq}}$ such that $(x_0, \lambda, \mu, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j=\overline{1,k}}) \in Y$ and objective functions of (MP) and (MD) have the same values at x_0 and $(x_0, \lambda, \mu, (\eta_i)_{i=\overline{1,N}}, (\theta_J)_{J=\overline{1,k}})$, respectively. If, further, the weak duality between (MP) and (MD) in theorem holds with the condition (a) without $\lambda > 0$ (resp. with the condition (b) or (c)), then $(x_0, \lambda, \mu, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j=\overline{1,k}}) \in Y$ is a weakly efficient (resp. an efficient) solutions of (MD).

Theorem 4.3 (Strict Converse Duality). Let x_0 and $(y_0, \lambda, \mu, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j=\overline{1,k}})$ be feasible solutions for (MP) and (MD) respectively, such that

$$\sum_{i=1}^{N} \lambda_i f_i(x_0) = \sum_{i=1}^{N} \lambda_i f_i(y_0). \tag{18}$$

(ii) (f, g) is strictly pseudo quasi $d_I - V-$ type I at y_o with respect to $(\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j=\overline{1,k}}$ and for λ and μ. Then $x_0 = y_0$

4. Acknowledgement

This reseach is supported by Fast Track Project No FT090018 of King Fahd University of Petroleum and Minerals, Saudi Arabia.

References

[1] Hanson, M.A. , On sufficiency of the Kunn-Tucker conditions, Journal of Mathematics Analysis and Applications 1981; **80**; 445-550

[2] Craven, B.D. , Invex functions and constrained local minima, Bulletin of Australian Mathematical Society 1981;**24**, 357-366.

[3] Kaul,R.N. and Kaur K. Optimality criteria in nonlinear programming involving non convex functions, Journal of Mathematical Analysis and Applications, 1985;**105**, 104-112

[4] Hanson, M.A. and Mond, B., Necessary and sufficient conditions in constrained optimization, Mathematical Programming 1987; **37**; 51-58

[5] Ruedo, N.G. and Hanson, M.A., Optimality criteria in Mathematical programming involving generalized invexity, Journal of Mathematical Analysis and Applications 1988;**130**; 375-385.

[6] Zhoa, Fo, On sufficiency of the Kunn-Tucker conditions in non differentiable programming, Bulletin Australian Mathematical society 1992;46;385-389.

[7] Clarke, F.H., Optimization and Nonsmooth Analysis, John Wiley and Sons, New York 1983.

[8] Kaul, R. N ; Suneja, S.K. and Srivastava, M.K, Optimality criteria and duality in multi objective optimization involving generalized invexity, Journal of Optimization Theory and Applications 1994; **80**,465-482

[9] Suneja, S.K; Srivastava, M.K; Optimality and duality in non differentiable multi objective optimization involving d-Type I and related functions, Journal of Mathematical Analysis and Applications 1997;**206**,465-479.

[10] Kuk, H., Tanino, T.,Optimality and duality in nonsmooth multi objective optimization involving generalized Type I functions, Computers and Mathematics with Applications 2003;**45**,1497-1506.

[11] Gulati, T. R. and Agarwal, D., Sufficiency and duality in nonsmooth multiobjective optimization involving generalized $(F, \alpha, \rho, d)-$type I functions, Computers and Mathematics with Applications 2006;**52**, 81-94.

[12] Antczak, T., Multiobjective programming under d-invexity, European Journal of operational Research 2002;**137**,28-36.

[13] Ye, Y.L., d-invexity and optimality conditons; Journal of Mathematical Analysis and Applications 1991;**162**,242-249.

[14] Mishra, S. K, Wang, S.Y and Lai,K. K, optimality and duality in non differentiable and multi objective programming under generalized d-invexity, Journal of Global Optimization 2004;**29**, 425-438.

[15] Mishra, S.K, Wang, S.Y and Lai,K.K, Nondifferentiable multiobjective programming under generalized d-univexity, European journal of operational Research 2005;**160**,218-226.

[16] Mishra, S.K and Noor, M.A., Some nondifferentiable multi objective programming problems, Journal of Mathematical Analysis and Applications, 2006;**316**,472-482.

[17] Antczak, T., Optimality conditions and duality for nondifferentiable multi objective programming problems involving $d-r-$ Type I functions, Journal of Computational and Applied Mathematics, 2009;**225**, 236-250.

[18] Silmani, H., and Radjef, M.S., Non differentiable multi objective programming under generalized d_I- invexity, European Journal of Operational Research,202;2010, 32-41.

[19] Antczak, T., Mean value in invexity analysis, Nonlinear Analysis: Theory, Methods and Applications, 2005; **60**, 1473-1484.

[20] Ben-Israel, A. and Mond, B., What is invexity ?, Journal of Australian Mathematical Society Series B, 1986; 28, 1-9.

[21] Jeyakumar, V. and Mond, B., On generalized convex mathematical programming, Journal of the Australian Mathematical Society Series B, 1992; 34,43-53.

[22] Hanson, M.A., Pini, R. and Singh, C., Multiobjective programming under generalized type I invexity, Journal of Mathematical Analysis and Applications, 2001;**261**,562-577.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Lagrange interpolation in Banach spaces for normalized three-layers neural networks

Giampietro Allasia[1] **and Cesare Bracco**[1]

[1] *Department of Mathematics "G. Peano", University of Turin, Italy*

emails: giampietro.allasia@unito.it, cesare.bracco@unito.it

Abstract

The Lagrange interpolation problem in Banach spaces is a crucial point for some applications of the artificial neural networks. The interpolation problem is approached by cardinal basis interpolation. A localizing scheme is then applied and some error estimates are given. Finally, the results of several numerical tests are reported in order to show the approximation performances of the proposed interpolants.

Key words: neural networks, interpolation in Banach spaces, cardinal basis functions

1 Introduction

Three-layers networks are one of the basic structures in the study of neural networks; they consists of an input layer, a hidden layer, and an output layer. In the present paper we consider a linear output F of the type

$$F(z) = \sum_{i=1}^{N} a_i l_i(z), \tag{1}$$

where N is the number of neurons in the hidden layer and a_i are the coefficients of the linear output, belonging to $\mathbb{R}$. An important case occurs when the l_i are determined by an *activation function* α, that is

$$F(z) = \sum_{i=1}^{N} a_i \alpha(z, c_i), \tag{2}$$

where each c_i is the center vector for the i-th neuron in the hidden layer. In particular, α can be taken as a radial function, that is

$$\alpha(z, c_i) = \beta(\|z - c_i\|), \tag{3}$$

 ISBN 13: 978-84-613-5510-5

so to obtain a so-called *radial basis function network* (see, e.g., [8]). As the output F is often a multivariate function $F : \mathbb{R}^p \to \mathbb{R}$, the norm considered in (3) is usually the Euclidean distance.

Networks of type (2), and in particular RBF networks (3), can be also normalized. In this case the output is

$$F(z) = \frac{\sum_{i=1}^{N} a_i \alpha(z, c_i)}{\sum_{k=1}^{N} \alpha(z, c_k)} = \sum_{i=1}^{N} a_i g_i(z), \tag{4}$$

where

$$g_i(z) = \frac{\alpha(z, c_i)}{\sum_{k=1}^{N} \alpha(z, c_k)}, \qquad i = 1..., n.$$

If we assume that the ouput is known at a finite number of points, that is, N couples of input-output data $\left\{(z_k, y_k)\right\}_{k=1}^{N}$ are given, the parameters a_i and c_i are determined such that the fit between the artificial output F and the known values is optimized. Though there are many criteria one can use to choose the parameters, here we focus our attention on the case in which we require that F satisfy the interpolation conditions

$$F(z_i) = y_i, \qquad i = 1, ..., n.$$

What we obtain is, from a mathematical point of view, a multivariate Lagrange interpolation problem, which can be solved by many well-known methods (see, e.g., [5]).

However, in many applications of the artificial neural networks, such as learning theory (see, e.g., [4]), the classical input domain $\mathbb{R}^p$ is unsuited, since the input data may belong to more general spaces, namely Banach spaces. Then, in those cases it is essential to develop suitable Lagrange interpolation schemes in Banach space, which is the topic of the paper.

In the case of Hilbert spaces the interpolation problem can be solved, for instance, by a sort of generalization of the Lagrange formula (see [9]) constructed with scalar products, which is a particular case of polynomial operator interpolant ([6], [7], see [10]). This interpolant can be modified so to get a cardinal basis solution to the same problem (see [1]), obtaining acceptable approximation performances. In a Banach space setting the constructions made using the inner product cannot be easily generalized, while it seems quite natural to use cardinal basis interpolation, both because Shepard-type functions can be constructed wherever a norm is present, and because the corresponding interpolant would be a particular case of normalized neural network.

Section 2 introduces the Lagrange interpolation problem in the Banach space setting, some basic properties of the cardinal basis interpolants, and Cheney's construction of the cardinal basis functions. Then, in Section 3, we apply a localizing scheme to the interpolant and we are then able to make more accurate error estimates, especially for the noteworthy case of Shepard-type functions. In Section 4 we provide examples and numerical tests in the space $\mathcal{C}[-\pi, \pi]$ equipped with the infinity norm.

 ISBN 13: 978-84-613-5510-5

2 Cardinal Basis Interpolation in Banach Spaces

Let X and Y be Banach spaces on the field $\mathbb{R}$, let Ω be a bounded open subset of X, and let $S = \{z_1, ..., z_n\} \subset \Omega$ be a given set of distinct points. We consider the Lagrange interpolation problem which consists of determining a continuous function $F : X \to Y$ such that

$$F(z_i) = y_i, \qquad i = 1, ..., n,$$

where $y_i \in Y$, $i = 1, ..., n$. It is convenient to suppose the existence of an underlying continous function $f : X \to Y$, whose values at the nodes are the data values, that is $y_i = f(z_i)$, $i = 1, ..., n$. A simple way to determine a solution of (2) is by considering

$$F(z) = \sum_{i=1}^{n} l_i(z) f(z_i), \tag{5}$$

where $l_i(z)$, for $i = 1, ..., n$, are functions from X into $\mathbb{R}$ satifying $l_i(z_j) = \delta_{ij}$, $i, j = 1, ..., n$. This approach has been studied, in the particular case $X = Y$ and l_i polynomials, by Prenter [9], who proved the existence of a (non-unique) solution but did not succeeded in giving a constructive formula, since the proof is based on the existence of the projections of X on certain subspaces, which cannot be explicitly constructed in general. Here, we attempt to solve the problem by a different and more constructive approach, using as l_i the cardinal basis functions.

We define the interpolant

$$F(z) = \sum_{i=1}^{n} g_i(z) f(z_i), \tag{6}$$

where $g_i : \Omega \to \mathbb{R}$, $i = 1, ..., n$, are cardinal basis functions, that is, satisfy

$$g_i \in C^0(\Omega), \qquad g_i(z) \geq 0, \qquad \sum_{i=1}^{n} g_i(z) = 1, \qquad g_i(z_j) = \delta_{ij}, \tag{7}$$

Note that for a given $z \in \Omega$, each $g_i(z)$ takes a scalar value, while $f(z_i) \in Y$, and so $F(z) \in Y$. From now on we will consider the underlying function f bounded on Ω.

If $f(z_i) = c,\ i = 1, ..., n$, we have from (6)

$$F(z) = \sum_{i=1}^{n} g_i(z) c = c. \tag{8}$$

Another noteworthy property is

$$\|F(z)\| \leq \max_i \|f(z_i)\|. \tag{9}$$

 ISBN 13: 978-84-613-5510-5

In fact

$$\|F(z)\| \leq \sum_{i=1}^{n} g_i(z)\|f(z_i)\| \leq \sum_{i=1}^{n} g_i(z) \max_i \|f(z_i)\| = \max_i \|f(z_i)\|,$$

where both the norms in X and Y are represented for simplicity by the same symbol.

The interpolant (6) has also some good approximation properties. In particular we can give the rough error bound

$$\|f(z) - F(z)\| \leq \|\sum_{i=1}^{n} g_i(z)[f(z) - f(z_i)]\| \leq$$
$$\max_i \|f(z) - f(z_i)\| \leq \sup_{z\in\Omega} \max_i \|f(z) - f(z_i)\|. \tag{10}$$

The dependence of the error bound from the underlying function, the distribution of the nodes and the diameter of Ω can be better shown by the inequality

$$\|f(z) - F(z)\| \leq \omega[f](\max_i \|z - z_i\|) \leq \omega[f]\big(\mathrm{diam}(\Omega)\big), \tag{11}$$

where

$$\omega[f](\delta) = \sup_{u,v\in\Omega} \{\|f(u) - f(v)\|, \|u - v\| \leq \delta\}$$

is the modulus of continuity of f. However, these are just rough error bounds, which does not provide information about the error behaviour when the nodes get closer and closer to each other.

An important way to construct cardinal basis functions defined on $\mathbb{R}^p$ is Cheney's method (see [3], pp. 67-68), which can be extended to a Banach space X.

Definition 2.1. *Let $\alpha : X \times X \to \mathbb{R}$ be a continuous function such that*

$$\alpha(z_i, z_i) = 0, \qquad \alpha(z, z_i) > 0, \qquad i = 1, ..., n.$$

Then, setting

$$g_i(z) = \frac{\prod_{j=1, j\neq i}^{n} \alpha(z, z_j)}{\sum_{k=1}^{n} \prod_{j=1, j\neq k}^{n} \alpha(z, z_j)}, \qquad i = 1, ..., n,$$

we get Cheney's cardinal basis functions.

These functions can be also represented in the *barycentric form*

$$g_i(z) = \frac{1/\alpha(z, z_i)}{\sum_{k=1}^{n} 1/\alpha(z, z_k)}, \qquad g_i(z_i) = 1, \qquad i = 1, ..., n,$$

 ISBN 13: 978-84-613-5510-5

which is usually more suitable from a computational point of view. In the following, for simplicity, we will omit to specify $g_i(z_i) = 1$, $i = 1, ..., n$.
The $g_i(z)$, $i = 1, ..., n$, are continuous functions on X, since they are ratios of continuous functions and the denominators are non-vanishing. So, the interpolant $F(z)$ in (6) is continuous too, being a linear combination of the values $f(z_i)$ with coefficients $g_i(z)$.
The barycentric form of the interpolant clearly shows that Cheney'costruction corresponds to a normalized network (4).
A natural choice for α is

$$\alpha(z, w) = \|z - w\|^\mu, \qquad \mu \in \mathbb{R}^+.$$

In this case, we have

$$g_i(z) = \frac{1/\|z - z_i\|^\mu}{\sum_{k=1}^{n} 1/\|z - z_k\|^\mu}, \qquad i = 1, ..., n, \tag{12}$$

that is, a version of Shepard basis functions for Banach spaces.

3 Localizing scheme

In practice, in many cases it is convenient to use a localized version of Cheney's construction, that is

$$\tilde{F}(z) = \sum_{i=1}^{n} \tilde{g}_i(z) f(z_i) = \sum_{i=1}^{n} \frac{\tau_i(z)(1/\alpha(z, z_i))}{\sum_{k=1}^{n} \tau_k(z)(1/\alpha(z, z_k))} f(z_i), \quad \delta > 0, \tag{13}$$

where $\tau_i : \Omega \longrightarrow \mathbb{R}^+$ is a continuous function such that

$$\tau(z_i) = 1, \quad \tau(z) > 0 \quad \text{for } z : \|z - z_i\| < \delta, \quad \text{and} \quad \tau(z) = 0, \quad \text{for } z : \|z - z_i\| \geq \delta,$$

and δ is suitably chosen. Hence the interpolant $\tilde{F}$, when evaluated at any $z \in \Omega$, considers only the nodes closest to z, that is, the nodes z_i such that $\|z - z_i\| < \delta$. We note that $\tilde{F}$ is a continuous operator like F. We also note that the cardinal basis functions in (13) are still a partition of unity, and therefore the constant functions are reproduced by the interpolant. As a consequence of this localization, the error estimate (11) can be improved:

$$\|f(z) - \tilde{F}(z)\| \leq \omega[f](\delta).$$

If we consider the Shepard-type cardinal basis functions (12), in the sum (6) only the terms corresponding to the nodes z_i closest to z are significantly different from 0. It seems natural to localize taking, for instance,

$$\tau_i(z) = \big(1 - \|z - z_i\|/\delta\big)_+, \quad z \in \Omega, \quad i = 1, ..., n,$$

 ISBN 13: 978-84-613-5510-5

so to get the interpolant

$$\tilde{F}(z)=\sum_{i=1}^{n}\tilde{g}_i(z)f(z_i)=\sum_{i=1}^{n}\frac{\big(1-\|z-z_i\|/\delta\big)_+(1/\|z-z_i\|^{\mu})}{\sum_{k=1}^{n}\big(1-\|z-z_k\|/\delta\big)_+(1/\|z-z_k\|^{\mu})}f(z_i),\quad \delta>0. \tag{14}$$

The localized version of the Shepard-type interpolant in (14) gives us the opportunity to obtain more meaningful error estimates involving the so-called *fill distance*

$$h_{S,\Omega}:=\sup_{z\in\Omega}\min_{z_i\in S}\|z-z_i\|.$$

To get the following error bound we will assume Ω convex. Let $c\in Y$ be any constant vector and let I_δ be the ball of radius δ centered at z, taking $\delta=C_1h_{S,\Omega}$ with $C_1\geq 1$ real constant. Moreover, let us cosider the neighborhood of z $J_\delta=I_\delta\cap\Omega$; then we have for $z\in\Omega$

$$\|f(z)-\tilde{F}(z)\|\leq\|f(z)-c\|+\|c-\tilde{F}(z)\|=\|f(z)-c\|+\left\|\sum_{i=1}^{n}\tilde{g}_i(z)[f(z_i)-c]\right\|$$

$$\leq\|f(z)-c\|+\sum_{i=1}^{n}\tilde{g}_i(z)\|f(z_i)-c\|\leq\Big(1+\sum_{i=1}^{n}\tilde{g}_i(z)\Big)\sup_{w\in J_\delta}\|f(w)-c\|$$

and, then,

$$\|f(z)-\tilde{F}(z)\|\leq 2\sup_{w\in J_\delta}\|f(w)-c\|,\qquad z\in\Omega.$$

This relation shows that the local errors (and the global error as well) depends on the distance of the interpolated function f from the set of constant vectors. Then, taking $c=f(z_0)$, where z_0 is the node (or one of the nodes) closest to z, we obtain the bound

$$\|f(z)-\tilde{F}(z)\|\leq 2\sup_{w\in J_\delta}\|f(w)-f(z_0)\|,$$

and, since $\|w-z_0\|<C_1h_{S,\Omega}$ for any $w\in J_\delta$,

$$\|f(z)-\tilde{F}(z)\|\leq 2\,\omega[f](2C_1h_{S,\Omega}).$$

Finally, if we suppose that f is Gâteaux-differentiable in Ω, then, by the finite increment formula applied on the segment $[z_0,w]\subset J_\delta$, we have

$$\|f(z)-\tilde{F}(z)\|\leq 2\sup_{w\in J_\delta}\sup_{0\leq\theta\leq 1}\|f'(z_0+\theta\Delta w)\|\|\Delta w\|,$$

with $\Delta w=w-z_0$, and, if the derivative is uniformly bounded on Ω, then

$$\|f(z)-\tilde{F}(z)\|\leq 2C_1h_{S,\Omega}\sup_{w\in J_\delta}\|f'(w)\|,$$

where $\|f'(w)\|$ stands for the norm of the operator $f'(w)$ in the space of linear and continuous operators from X into Y. This shows that this method has approximation order $\mathcal{O}(h_{S,\Omega})$.

 ISBN 13: 978-84-613-5510-5

4 Numerical tests

Now we consider a numerical example. Let $X = \mathcal{C}[-\pi, \pi]$ with norm

$$\|z\|_\infty = \max_{t\in[-\pi,\pi]} |z(t)|.$$

and let

$$\Omega = \{\alpha \sin t + \beta \sin(2t); \alpha, \beta \in [1, 1 + 1/20]\},$$

and $Y = \mathbb{R}$. Let the nodes be $z_i = \alpha_i \sin t + \beta_i \sin(2t)$, $(\alpha_i, \beta_i) \in [1, 1 + 1/20] \times [1, 1 + 1/20]$, $i = 1, ..., N$, and

$$f(z_i) = \int_{-\pi}^{\pi} t z_i(t) dt, \qquad i = 1, ..., N.$$

The operator

$$z(t) \to \int_{-\pi}^{\pi} t z(t) dt$$

plays here the role of a test function. We choose $\mu = 2$ in Shepard-type cardinal basis functions because, in general, a small value of μ avoids "flat spots "near the nodes (see, e.g., [2]). The cardinal basis interpolant, in the barycentric form, is

$$F(z) = \sum_{i=1}^{N} \frac{1 \Big/ \Big(\max\limits_{t\in[-\pi,\pi]} |z(t) - z_i(t)| \Big)^2 dt}{\sum\limits_{k=1}^{n} 1 \Big/ \Big(\max\limits_{t\in[-\pi,\pi]} |z(t) - z_k(t)| \Big)^2} \int_{-\pi}^{\pi} t z_i(t) dt.$$

Here we tested the interpolant taking nodes with coefficients (α_i, β_i) first on a regular $r \times r$ grid over $[1, 1 + 1/20] \times [1, 1 + 1/20]$ ($r = 5, 10, 15, 20, 30, 40$), and then constructing Halton points (with generating primes 2 and 3) on the same square [5]. Note that this choice for the nodes makes sense, because we have the inequality

$$\big\|[\alpha_1 \sin t + \beta_1 \sin(2t)] - [\alpha_2 \sin t + \beta_2 \sin(2t)]\big\|_\infty \leq$$
$$|\alpha_1 - \alpha_2| + |\beta_1 - \beta_2| \leq 2\big\|(\alpha_1, \beta_1) - (\alpha_2, \beta_2)\big\|_2.$$

The interpolant has been evaluated, using the barycentric formula, at 1089 points taking the coefficients (α, β) on a regular 33×33 grid over $[1, 1 + 1/20] \times [1, 1 + 1/20]$. The results are reported in Table 1 and Table 2.

nodes	RMSE	nodes	RMSE	nodes	RMSE
4	$1.988 \cdot 10^{-2}$	25	$2.037 \cdot 10^{-2}$	400	$1.387 \cdot 10^{-2}$
9	$2.315 \cdot 10^{-2}$	100	$1.667 \cdot 10^{-2}$	900	$1.227 \cdot 10^{-2}$
16	$2.172 \cdot 10^{-2}$	225	$1.493 \cdot 10^{-2}$	1600	$1.161 \cdot 10^{-2}$

Table 1: *Evaluation of test operator by cardinal basis interpolant (nodes on a regular grid) in $\mathcal{C}[-\pi, \pi]$ with infinity norm*

 ISBN 13: 978-84-613-5510-5

nodes	RMSE	nodes	RMSE	nodes	RMSE
4	$5.617 \cdot 10^{-2}$	25	$3.418 \cdot 10^{-2}$	400	$1.857 \cdot 10^{-2}$
9	$4.103 \cdot 10^{-2}$	100	$2.313 \cdot 10^{-2}$	900	$1.632 \cdot 10^{-2}$
16	$3.644 \cdot 10^{-2}$	225	$2.018 \cdot 10^{-2}$	1600	$1.510 \cdot 10^{-2}$

Table 2: *Evaluation of test operator by cardinal basis interpolant (Halton points as nodes) in $\mathcal{C}[-\pi, \pi]$ with infinity norm*

Then we tested the localized version of the formula considering the same example. Note that in this slightly different framework the choice of the parameter δ in (14) is a crucial point. Here we used a *stationary approach*, that is, δ decreases proportionally to the fill distance $h_{S,\Omega}$; in particular, for these numerical experiments we set $\delta = 2h_{S,\Omega}$. The results, reported in Tables 3-4, show significantly better approximation errors.

nodes	RMSE	nodes	RMSE	nodes	RMSE
4	$2.887 \cdot 10^{-2}$	25	$8.398 \cdot 10^{-3}$	400	$1.844 \cdot 10^{-3}$
9	$1.606 \cdot 10^{-2}$	100	$3.819 \cdot 10^{-3}$	900	$1.204 \cdot 10^{-3}$
16	$1.102 \cdot 10^{-2}$	225	$2.480 \cdot 10^{-3}$	1600	$8.953 \cdot 10^{-4}$

Table 3: *Evaluation of test operator by localized cardinal basis interpolant (gridded points as nodes) in $\mathcal{C}[-\pi, \pi]$ with infinity norm*

nodes	RMSE	nodes	RMSE	nodes	RMSE
4	$5.168 \cdot 10^{-2}$	25	$1.853 \cdot 10^{-2}$	400	$3.599 \cdot 10^{-3}$
9	$2.938 \cdot 10^{-2}$	100	$7.348 \cdot 10^{-3}$	900	$2.150 \cdot 10^{-3}$
16	$2.356 \cdot 10^{-2}$	225	$4.906 \cdot 10^{-3}$	1600	$1.665 \cdot 10^{-3}$

Table 4: *Evaluation of test operator by localized cardinal basis interpolant (Halton points as nodes) in $\mathcal{C}[-\pi, \pi]$ with infinity norm*

References

[1] G. Allasia and C. Bracco, *Two interpolation operators on irregularly distributed data in inner product spaces*, J. Comp. Appl. Math., in press, 2010.

[2] R.E. Barnhill, R.P. Dube and F.F. Little, *Properties of Shepard's surfaces*, Rocky Mountain J. Math. **13** (1983) 2, 365–382.

[3] W. Cheney and W. Light, *A course in approximation theory*, Brooks/Cole, Pacific Grove, 2000.

[4] F. Cucker and D.X. Zhou, *Learning theory: an approximation theory viewpoint*, Cambridge University Press, New York, 2007.

 ISBN 13: 978-84-613-5510-5

[5] G.E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, World Scientific Publishers, Singapore, 2007.

[6] E.F. Kashpur, V.L. Makarov and V.V. Khlobystov, *On the problem of Hermite Interpolation of Operators in a Hilbert Space*, Journal of Mathematical Sciences, vol. 84, **4** (1997), 1233–1244.

[7] V.V. Khlobystov and T.N. Popovicheva, *Interpolation and identification problems*, Cybernetics and Systems Analysis, Vol. 42 **3** (2006), 392–397.

[8] T. Poggio and F. Girosi, *Networks for approximation and learning*, Proc. IEEE, 78 **9** (1990), 1482–1497.

[9] P.M. Prenter, *Lagrange and Hermite Interpolation in Banach spaces*, J. Approx. Theory **4** (1971), 419–432.

[10] A. Torokhti and P. Howlett, *Computational Methods for modelling of nonlinear systems*, Elsevier, Amsterdam, 2007.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A parameterised shared-memory scheme for parameterised metaheuristics

Francisco Almeida[1], Domingo Giménez[2] and Jose-Juan López-Espín[3]

[1] *Departamento de Estadística, Investigación Operativa y Computación, University of La Laguna*

[2] *Departamento de Informática y Sistemas, University of Murcia*

[3] *Centro de Investigación Operativa, University Miguel Hernández of Elche*

emails: falmeida@ull.es, domingo@um.es, jlopez@umh.es

Abstract

This paper presents a parameterised shared-memory scheme for parameterised metaheuristics. The use of a parameterised metaheuristic facilitates experimentation with different metaheuristics and hybridation/combinations to adapt them to the particular problem we are working with. Due to the large number of experiments necessary for the metaheuristic selection and tuning, parallelism should be used to reduce the execution time. To obtain parallel versions of the metaheuristics and to adapt them to the characteristics of the parallel system, an unified parameterised shared-memory scheme is developed. Given a particular computational system and being fixed the parameters for the sequential metaheuristic, the appropriate selection of parameters in the unified parallel scheme eases the development of parallel efficient metaheuristics.

Key words: parallel metaheuristics, shared-memory

1 Introduction

Currently most of the computational parallel systems are formed by multicore components. Laptops and personal computers are multicore, and clusters and supercomputers are built by connecting multicore nodes. So, the development of efficient multicore versions of our algorithms is compulsory if we want to efficiently use the systems we have access to. Multicore systems can be programmed with the shared-memory paradigm, using OpenMP [10], which we use to develop the unified parameterised parallel shared-memory scheme of an unified parameterised scheme of metaheuristics.

Given that most of the interesting and attractive combinatorial problems belong to the NP class, exact methods are not very useful except for small sized problems. For

 ISBN 13: 978-84-613-5510-5

this reason many approximation methods have been provided that allow high quality solutions to be obtained in acceptable running times. In recent decades, metaheuristics have emerged as an advantageous technology for approximation algorithms [2]. They bring together methods and ideas from very different fields, such as artificial intelligence, mathematics and biology. The main concern here lies in their easy and immediate applicability to hard problems.

The use of a unified parameterised scheme for metaheuristics [9] facilitates the easy development of new metaheuristics or hybrid metaheuristics for experiment and adaptation to a particular problem. However, in the process of obtaining a well-tailored metaheuristic for a problem, it is necessary to experiment with a large number of metaheuristics and their parameters, and so the time dedicated to the experiments is very large.

To alleviate this problem, parallel versions of the methods can be developed. There are a large number of studies on the parallelization of metaheuristics [1]. Each metaheuristic may have a different parallel scheme, and some of them could follow a different paradigm. In our approach, and as main contribution, we consider the common development of parallel versions by using a unified metaheuristic scheme to obtain a unified parallel scheme for metaheuristics.

In addition, the parallel scheme is parameterised, and the values of some algorithmic parameters can be selected to optimise the execution of the parallel metaheuristic obtained from the sequential parameterised scheme of metaheuristics. The optimum values of the algorithmic parameters will depend of those of the metaheuristic parameters and of the characteristics of the computational system.

The rest of the paper is organised as follows. Section 2 presents the parameterised metaheuristic scheme. The corresponding parameterised shared-memory scheme for metaheuristics is presented in seciton 3. Section 4 presents some experimental results obtained when applying the parameterised parallel scheme to a particular problem. These results confirm the validity of our proposal. Finally, in section 5 the conclusions are summarised and some future research lines are outlined.

2 Parameterised metaheuristic scheme

The use of a general scheme for metaheuristics (algorithm 1) allows us to quickly develop and experiment with different metaheuristics to decide which metaheuristic, combination/hybridation of metaheuristics and tuning parameters are the most suitable for solving a particular problem. With such a scheme, some of the functions can be reused for different methods, so facilitating the development of metaheuristics.

Each basic function in this unified metaheuristic scheme can be parameterised so that different values of the parameters give different metaheuristics, hybridation/combination of metaheuristics or different versions of a particular metaheuristic. In that way, scheme 1 changes by making the basic functions in it parameterised functions, as shown in algorithm 2.

Different sets of parameters can be established for the different functions, and it

 ISBN 13: 978-84-613-5510-5

```
Initialize(S)
while (not EndCondition(S)) {
    SS = Select(S)
    if(|SS| > 1) SS1 = Combine(SS)
    else SS1 = SS
    SS2 = Improve(SS1)
    S = Include(SS2)
}
```

Algorithm 1: General scheme of a metaheuristic method.

```
Initialize(S,ParamInit)
while (not EndCondition(S,ParamEndCond)) {
    SS = Select(S,ParamSelec)
    if(|SS| > 1) SS1 = Combine(SS,ParamComb)
    else SS1 = SS
    SS2 = Improve(SS1,ParamImpr)
    S = Include(SS2,ParamIncl)
}
```

Algorithm 2: General parameterised scheme of a metaheuristic method.

is not the objective of this paper to study the best selection of the parameters. As an example, in the `Initialize` function `ParamInit` could be composed by four parameters: one corresponding to the size of an initial reference set; another for the number of elements to be improved with some improvement function (which also could be used in the improvement function inside the `while` loop); a third parameter to indicate the intensity of the improvement, as for example how big the neighborhood in a local search is; and the number of elements in the reference set. Similarly, a set of parameters should be determined for each function, and for a particular problem different metaheuristics and combinations of them are obtained by giving different values to those parameters.

3 Parameterised shared-memory scheme for metaheuristics

The parameterised metaheuristic scheme in algorithm 2 can be used to develop the corresponding unified parameterised shared-memory scheme. To do so, the functions in the scheme are parallelised independently, and different parallel patterns should be identified in the basic functions of the scheme. Two basic parallel schemes are identified for the functions in algorithm 2.

In the first scheme (algorithm 3) the elements in a set are treated independently. The set of metaheuristic parameters (`MetaheurParam`) is passed to the function, and the number of threads to be used in the parallel loop (`one-loop-threads`) is obtained as a function of the values of the metaheuristic parameters. This scheme can be used for example when crossing elements in a genetic algorithm, when randomly generating the initial set of elements... For different functions, the optimum value of `one-loop-threads`

 ISBN 13: 978-84-613-5510-5

depends of the values of the parameters of the metaheuristic and also of the cost of the processing function (cost of crossing function, random generation function...), which depends of the metaheuristic and the computational system.

```
one-loop(MetaheurParam):
    omp_set_num_threads(one-loop-threads(MetaheurParam))
    #pragma omp parallel for
        loop in elements
            treat element
```

Algorithm 3: Parallel scheme for independent treating elements (`scheme 1`).

The second scheme is a two-level parallelism scheme (algorithm 4), and a number of threads should be determined for each level. The number of threads to work in the first level (`first-level-threads`) is obtained as a function of the parameters of the metaheuristic (also of its functions, and consequently of the cost of them in the computational system). Once this number of threads has been determined, the number of threads to work in the second level (`second-level-threads`) is obtained as a function of the metaheuristic parameters and the number of threads working in the first level. Of course, the first scheme is a particular case of this second scheme when the number of threads in the second level is fixed to one, but it is better to consider two different schemes because the number of parameters to obtain and how they are obtained are different. This type of parallelism appears for example in the improvement functions, when a number of elements is selected to improve them (which gives a loop in the number of elements to improve) and each element is improved by analysing some elements in its neighbourhood (loop in the second level). It can appear in other parts, as for example when crossing elements if the crossing function is not a simple one and to parallelize it can contribute to reduce the execution time.

```
two-level(MetaheurParam):
    omp_set_num_threads(first-level-threads(MetaheurParam))
    #pragma omp parallel for
        loop in elements
            second-level(MetaheurParam,first-level-threads)

second-level(MetaheurParam,first-level-threads):
    omp_set_num_threads(second-level-threads(MetaheurParam,first-
       level-threads))
    #pragma omp parallel for
        loop in elements
            treat element
```

Algorithm 4: Two-level parallel scheme (`scheme 2`).

 ISBN 13: 978-84-613-5510-5

4 Computational results

To validate our proposal, some experiments have been carried out. Here, the results of some experiments with Simultaneous Equation Models (SEM) are presented. We will not explain what they are, but the interested reader could refer to some well known books [4, 5]. SEMs shave been traditionally used in econometrics, but they have begun to be used in other fields (networks simulation [6], medicine [7]...), so it is a problem of great interest. Normally they are developed by people with a wealth of experience in the particular problem represented by the model, but the use of automatic tools to provide the experts with satisfactory models is interesting in some cases, as for example when the dependence of the variables is not clear or when experiments are being carried out to determine variables to be included in the model. To automatically obtain satisfactory models, it is necessary to evaluate a large amount of candidate models and to measure their quality according to some criteria, like for example the Akaike Information Criterion (AIC) [3]. Genetic algorithms have been applied to this problem [8], and using the parameterised sequential scheme here presented the application of different metaheuristics to the problem is facilitated. Furthermore, the use of the parallel scheme allows us to make in a reasonable time the experiments with different metaheuristics to decide a satisfactory one for the specific model we are working with.

The problem consists of, given a set of values (obtained by experimentation, survey...), to obtain the variables which appear in each equation in the system, which means, the model which best represents the variables dependences.

To apply metaheuristic methods to obtain SEMs, a set of models is explored. Each element in the set is a candidate to be the best model. An element is defined as a matrix. In each row, an equation is represented using ones and zeros. If variable j appears in equation i, the value for the (i, j) position is one, and zero if not.

Experiments have been carried out in the supercomputer `BenArabi` of the Supercomputing Centre of Murcia. The part `Ben` is a HP Integrity Superdome SX2000 with 128 cores of the processor Intel Itanium-2 dual-core Montvale, and `Arabi` is a cluster of 102 nodes, each one with 8 cores of the processor Intel Xeon Quad-Core L5450. So, experiments have been made in systems with 8 and 128 cores.

Experiments have been carried out by randomly generating a system and the values of the exogeneous variables, and obtaining from them the values of the endogenous variables. By selecting the values of the parameter in algorithm 2, different metaheuristics have been applied. The methods considered to parallelize are: GRASP, a genetic algorithm (Genet), a scatter search (Scatt), two hybrid methods with a GRASP followed by genetic (GRA+Gen) and scatter search (GRA+Sca), a combination of genetic and scatter search (Gen+Sca) and a GRASP followed by this combination (GR+Ge+Sc). The parameters used for each method are shown in table 1. Note the advantage of using a unified metaheuristic scheme, both for sequential and parallel development, that allows for the testing of numerous methods and parallelizations. The meaning of the parameters in the table are: numbers of elements in the initial set (Init. num. elem.), numbers of elements in the reference set (Num. elem. iter.), percentage of elements of the initial set to be improved (Perd. impr. init.), intensification of the initial im-

 ISBN 13: 978-84-613-5510-5

provement (Int. impr. init.), number of best elements selected for combination (Num. best elem.), number of worst elements selected for combination (Num. worst elem.), number of combinations of best elements (Num. best-best), number of combinations of best elements with worst elements (Num. best-worst), number of combinations of worst elements (Num. worst-worst), percentage of elements generated by combination which are improved (Perc. impr. elem.), intensification of the improvement (Int. impr. elem.), percentage of elements to be mutated (Perc. elem. mut.), intensification in the improvement of elements obtained by mutation (Int. impr. mut.), and number of best elements included in the reference set (Num. best elem.). To compare the parallelism, the number of iterations in genetic and scatter search (or combinations) has been fixed at 50.

	GRASP	Genet	Scatt	GRA+Gen	GRA+Sca	Gen+Sca	GR+Ge+Sc
Init. num. elem.	200	500	100	200	200	100	200
Num. elem. iter.	-	500	20	200	20	50	50
Perc. impr. init.	100	0	100	100	100	100	100
Int. impr. init.	10	-	10	10	10	10	10
Num. best elem.	-	500	10	200	10	25	25
Num. worst elem.	-	0	10	0	10	25	25
Num. best-best	-	250	90	100	90	90	90
Num. best-worst	-	-	100	-	100	100	100
Num. worst-worst	-	-	90	-	90	90	90
Perc. impr. elem.	-	0	100	0	100	100	100
Int. impr. elem.	-	-	5	-	5	5	5
Perc. elem. mut.	-	10	0	10	0	10	10
Int. impr. mut.	-	0	-	0	-	5	5
Num. best elem.	-	500	10	200	10	25	25

Table 1: Values of the parameters for the different combinations of metaheuristics considered in the experiments.

Figure 1 shows the speed-up achieved in the two systems with the seven metaheuristics, when using the maximum number of cores in the systems (8 in `Arabi` and 128 in `Ben`) and without nested parallelism (it is called 8 or 128 in the figure), with the configuration (number of cores in each parallelism level) which gives the lowest execution time (lowest), and with the best combination of threads in the initialisation part and in the iteration part (parts).

In `Arabi` the speed-up is close to the number of cores, and normally the best configuration is to use non nested parallelism and 8 cores. Only in two cases do other combinations give better results. In the metaheuristics with lowest execution time per iteration the speed-up is lower because the sequential time of the parallelised parts is very low.

The situation is different in `Ben`. To use the maximum number of cores is not a good option, and some strategy to select the number of threads to work on the solution of the problem is preferable. So, the speed-up is always far from the maximum achievable (it may be otherwise with bigger problems). Furthermore, the randomness in the execution in the metaheuristics makes it difficult to draw definitive conclusions, but experiments with other problem sizes and configurations confirm this behaviour.

 ISBN 13: 978-84-613-5510-5

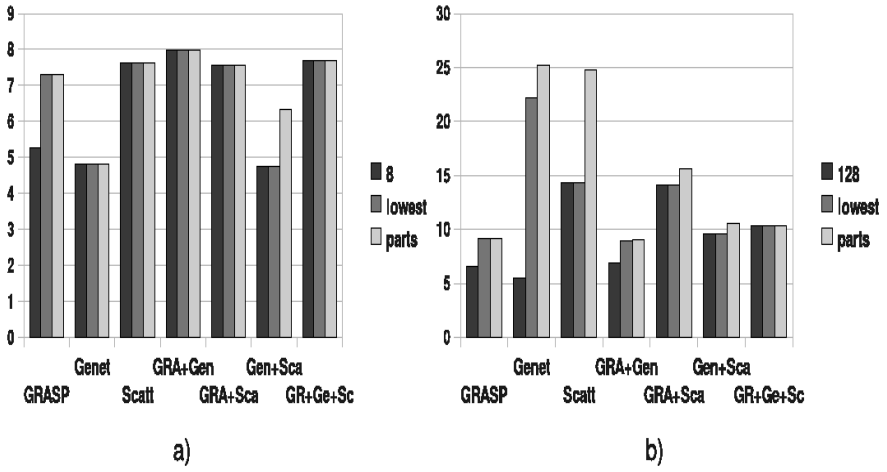


Figure 1: Speed-up of different metaheuristics, with the maximum number of cores, the maximum achieved speed-up and that obtained with different numbers of threads: a) `Arabi`, b) `Ben`.

5 Conclusions and future research

The use of a parameterised sequential scheme allow us to obtain different metaheuristics and hybridation/combination of metaheuristics by selecting different values for the parameters in the unified scheme. Furthermore, the parameterised metaheuristic scheme in algorithm 2 has been parallelized by parallelizing each basic function, with common parallelization strategies for functions with the same structure. So, a parameterised shared-memory scheme of metaheuristics has been obtained, so that parallel versions of different metaheuristics and combinations are obtained by simply selecting the values of the parameters of the metaheuristic. In addition, the use of the parameterised parallel scheme allows us to adapt the scheme to the metaheuristic (determined by the values of the parameters in the sequential metaheuristic scheme), the specific problem to be solved and the characteristics of the parallel system where it is solved. The applicability of the proposal has been tested with experiments with the problem of automatically obtaining satisfactory Simultaneous Equation Models from a set of values of variables.

A satisfactory selection of the values of the parallelism parameters in the different functions of the parallel produces a reduction in the parallel execution time. At present, we are working on the inclusion of a decision engine in the parallel scheme to decide the optimum parallel execution parameters.

 ISBN 13: 978-84-613-5510-5

Acknowledgement

The experiments have been carried out in the systems of the Supercomputer Centre of the Fundación Parque Científico of Murcia.

Partially supported by Fundación Séneca, Comunidad Autónoma de la Región de Murcia, 08763/PI/08, and Ministerio de Educación of Spain, TIN2008-06570-C04.

References

[1] Enrique Alba. *Parallel Metaheuristics: A New Class of Algorithms.* Wiley-Interscience, 2005.

[2] F. Glover and G. A. Kochenberger. *Handbook of Metaheuristics.* Kluwer, 2003.

[3] A. Gorobets. The optimal prediction simultaneous equations selection. *Economics Bulletin*, 36(3):1–8, 2005.

[4] W. Greene. *Econometric Analysis.* Prentice Hall, third edition, 1998.

[5] D. Gujarati. *Basic Econometrics.* McGraw Hill, 1995.

[6] M. Harchol-Balter and P. E. Black. Queueing analysis of oblivious packet-routing networks. In *SODA: ACM-SIAM Symposium on Discrete Algorithms*, 1993.

[7] R. Henry, I. Lu, L. Beightol, and D. Eckberg. Interactions between CO2 Chemoreflexes and Arterial Baroreflexes. *Am. Journal of Physiology*, 274(43):H2177–H2187, 1998.

[8] J. J. López-Espín and D. Giménez. Genetic algorithms for simultaneous equation models. In *DCAI*, pages 215–224, 2008.

[9] G. R. Raidl. A unified view on hybrid metaheuristics. In *Hybrid Metaheuristics, Third International Workshop, LNCS*, volume 4030, pages 1–12, October 2006.

[10] OpenMP web page. `http://openmp.org/wp/`.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Introducing High Performance Software Tools in Cloud Computing with PyACTS-PyOpenCF

F. Almeida[1], V. Blanco[1], V. Galiano[2], H. Migallón[2] and A. Santos[1]

[1] *Dpto. Estadstica, I.O. y Computacin, Universidad de La Laguna, S/C de Tenerife, Spain*

[2] *Dpto. Fsica y Arq. Computadores, Universidad Miguel Hernndez, Elche, Spain*

emails: falmeida@ull.es, vblanco@ull.es, vgaliano@umh.es, hmigallon@umh.es, asmarre@ull.es

Abstract

Many computational applications rely heavily on numerical linear algebra operations. Part of these applications are data and computation intensive that need to run in high performance computing environments. On the other hand, Cloud Computing is emerging as a new computing paradigm which aims to provide reliable, customized and QoS guaranteed dynamic computing environments for end users. For research groups, while the ACTS Collection brings robust and high-end software tools to the hands of application developers, cloud Computing provides convenient access to reliable, high-performance clusters and storage without the need to purchase and maintain sophisticated hardware. In this paper we propose to join these two paradigms of scientific computing in a framework that allows executing high performance tools included in ACTS in a heterogeneous and dynamic system.

Key words: Cloud Computing, high performance, Software Tools, ACTS, Python, Web Services, instructions

1 Introduction

Scientists and engineers in virtually every field are turning to high performance parallel computers to simulate and solve some of their problems. One reason for this trend resides in the fact that the models, algorithms, and phenomena are becoming more complex. Unfortunately, parallel architectures are expensive and hard to configure and administrate. Only major research centers have the necessary financial and human resources to manage a center of High Performance Computing.

 ISBN 13: 978-84-613-5510-5

In last two years, a new concept is emerging: Cloud Computing [2]. Clouds are hinting at a future in which we will not compute on local computers, but on centralized facilities operated by third-party computing and storage utilities. We can relate it to other similar technologies, especially grid-computing, but there are significant differences show in Table 1. In resume, Cloud computing describes a new supplement, consumption and delivery model for IT services based on Internet, and it typically involves the provision of dynamically scalable and often virtualized resources as a serviceover the Internet.

	Grid	Cloud
Underlying Concept	Utility Computing	Utility Computing
Main Benefit	Solve computationally complex problems	Provide a scalable standard environment for network-centric application development, testing and deployment
Resource distribution /allocation	Negotiate and manage resource sharing schedulers	Simple user Provider model pay-per-use
Domains	Multiple domains	Single domain
Character/history	Non-commercial, publicly funded	Commercial

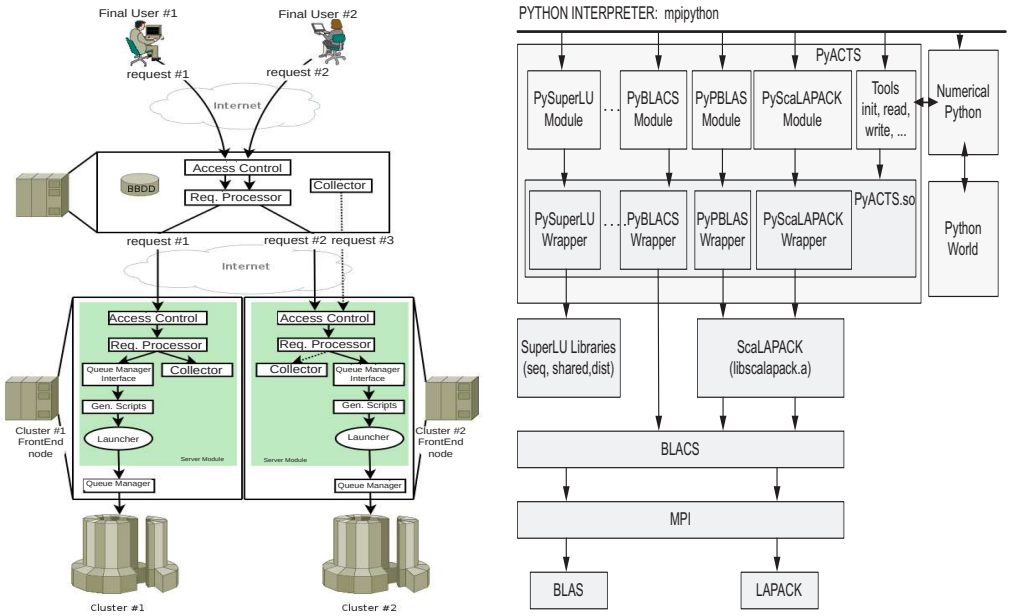
Figure 1: Main differences between Cloud and Grid Computing

In previous work, OpenCF [8, 11] has been presented as a framework on a Cloud Computing infrastructure, where users can access to the computing facilities on demand according to their needs. A significant difference between OpenCF and others Cloud Computing projects is that OpenCF does not revolve around the creation of virtual machines as a way to offer services to users. Rather, it allows for direct instances of applications to be run on the computing servers(SaaSapproach).This, which at first glance might seem like a disadvantage, is proposed as a way to facilitate access to this type of tool to end users with limited knowledge of programming or high performance computing.

This paper proposes the use of scripts such as computing service to users of cloud computing. That is, in addition to providing access to applications compiled on computing servers, the user can program interpreted Python code programming its own high performance application to be executed on any platform in the cloud. Users can make use of precompiled libraries in high performance computing servers through distributions like PyACTS [5]. Thus, the code can make use of high performance libraries to the lowest level and to be independent of the platform on which to run.

This paper is arranged as follows. Section 2 and 3 introduces OpenCF and PyACTS respectively. In section 4, we present the framework that integrates both architectures and we show several examples that illustrates the advantages of these new paradigm.

 ISBN 13: 978-84-613-5510-5



(a) OpenCF Architecture (b) PyACTS Architecture

Figure 2: OpenCF and PyACTS's models

2 OpenCF

This section describes the architecture of OpenCF. OpenCF software architecture is shown in Figure 2(a). As introduced in [9], OPenCF is highlighted by a modular design: module server and client module. The modules can be replaced independently and even extended to provide new functionality without disturbing the rest of the system components. The client and server implement the three lower layers of the stack that describes the Web service: Description of Services, XML Messaging and Transport. The fourth level, Service Discovery has not been implemented for security reasons. Therefore, system administrators still control access by of customers to parallel platforms through traditional techniques authentication.

The client provides an interface for the end user and translates the requests queries to the server. The server receives requests from clients Authenticated and transforms them into jobs for the queue manager. These modules, in turn, are also modularized. The module `Control Access`, `Submission Process` and `Collector` can be found on both the server and client. The client also maintains a database to manage information generated by the system. The server includes elements for generating scripts and work release the queuing system.Briefly, we should describe the features of the modules listed.

1. Client Module: The client is the interface between the end user and the system, where users are registered through a form. Below is a list of sub-modules.

 ISBN 13: 978-84-613-5510-5

- The `Database` stores information about users, servers, work, input and output files, etc.. It has been implemented as a base MySQL relational data [6] and is accessed through PHP scripts.
- The `Request processor` is via a Web interface which the user can access the list of available applications. Each entry in the list shows a short description of the routine. Tasks grouped according to the servers that support them. It also manages dynamically generate XHTML form input data according to the job description.
- The `Collector` manages the output generated by launched work on the server. We can also check the status of submissions through the Web interface, and download the results.

2. Server Module: The server handles all matters related to the work, making them available through the service and monitor its status and implementation.

   - The `Request Processor` consists of a set of PHP scripts that are responsible for analyzing the requests received from the client. In addition, it is also responsible to generate and export the Web service, to maintain updated the WSDL document [13] (*Web Service Description Language*) encapsulated with the protocol messages SOAP generated by NuSOAP [12].
   - The `Queue Manager Interface` manages the queue of the HPC system. The server needs to know how to run a work and how to check its status on the server is installed. Additionally, we need an XML description of each of the routines available to specify the job. In section 4 we will show an integrated example with the PyACTS library. In current version of OpenCF, once the user sends a job request, the server executes the code binary associated with the supplied arguments. In this way, we can incorporate new services by adding the file with the XML description with compiled code in the server module. This is the main difference with the proposed system in 4, where the application are programmed in a scripting language and not need to be compiled.
   - The `Script Generator` produces the necessary scripts for execution of work in different systems of queues. It is composed of a set of templates. We need a different template for each one of the queue managers supported.
   - The `Launcher` is the interface between OpenCF and operating system. For security reasons, you need a non-privileged user created to run the OpenCF code.

3. The `collector` is the interface that delivers the output data produced by execution of a job. Once work is complete, the queuing system automatically sends an email to the user, and moves output files to a temporary directory until they are downloaded by the `client collector`.

 ISBN 13: 978-84-613-5510-5

PyOpenCF born from the idea of developing OpenCF in a single programming language, so that it is more portable and independent as well as more efficient in upgrading the platform. For this, the language used was *Python* [7]. Python is an scripting language widely used by scientific and academic community. Same functions and characteristics of OpenCF have been implemented in this version. In this way, PyOpenCF offers a platform to submit precompiled jobs to the computing servers. However, if we would program our algorithm we could do it with an scripting language like Python, and submit this script with PyOpenCF. The main disadvantage is the bad performance in scripting languages, but according to [3], an application can be written in Python but the hard computational tasks can be executed in tunned libraries to each HPC system, without significant penalty in performance. In this sense, we will introduce PyACTS concepts in the next section.

3 PyACTS

The Advanced CompuTational Software (ACTS) Collection [4] is a set of software tools for computational sciences that helps programmers write high performance scientific codes for high-end computers. ACTS tools are mostly libraries (some are C libraries, some C++ class libraries, and some are Fortran libraries). They are primarily designed to run on distributed memory parallel computers. Portability and performance were both considerations in their design and implementation. The ACTS tools use the standard Message Passing Interface (MPI) [10] for communication. The computational model of ScaLAPACK, BLACS and PBLAS (included in the ACTS tools) consists of a one or two-dimensional process grid, where each process stores pieces of matrices and vectors. The prime beneficiaries of ACTS tools are developers of parallel engineering and scientific applications. Many areas of scientific computing are covered by ACTS tools, and can potentially make use of them. Nevertheless, parallel software can be more complex than serial software and significantly more expensive to implement.

In this context, we developed PyACTS as a set of modules which can be imported from the Python interpreter to enlarge the number of users that can make use of the routines included in ACTS. PyBLACS, PyPBLAS and PyScaLAPACK were our first steps to achieve our goal: an easy and integrated set of tools that can be used from Python but offers all the performance of the libraries in the original development environment (Fortran and C).

In Figure 3, we present a script used to test the interface to the PBLAS level 3 routine (*pvgemm*, $\alpha AB + \beta C$, $\alpha, \beta \in \mathbb{R}$, $A, B, C \in \mathbb{R}^{n \times n}$). This example reads the data from three text files and stores them in *PyACTS Arrays*. Note that this reading is completed by a single process (usually, [0,0] in the process grid) and it sends the data to the rest of the processes using PyBLACS to obtain a two-dimensional block-cyclic distribution. After executing *Txt2PyACTS* in Figure 3, the variables a, b and c are *PyACTS Arrays* and can be used as parameters in PyACTS routines. Once the matrix multiplication is done, the routine *PyACTS2Text* collects the (distributed) results and writes them into the text file. It is interesting to compare this script of with a Fortran or

 ISBN 13: 978-84-613-5510-5

```
from PyACTS import *
import PyACTS.PyPBLAS as PyPBLAS
ACTS_lib=PyACTS.ScaLAPACK_ID      # ScaLAPACK ID
PyACTS.gridinit()                 # Grid initialization
alpha=Scal2PyACTS(1.2,ACTS_lib)   # Distribute scalars
beta=Scal2PyACTS(2,ACTS_lib)
a=Txt2PyACTS("data_a.txt",ACTS_lib) # Read Text files and
b=Txt2PyACTS("data_b.txt",ACTS_lib) # store in PyACTS Arrays
c=Txt2PyACTS("data_c.txt",ACTS_lib)
result=PyPBLAS.pvgemm(alpha,a,b,beta,c) # Call routine
PyACTS2Text("data_result.txt",result)   # Write results
PyACTS.gridexit()
```

Figure 3: Example of PyPBLAS: *pvgemm*

C implementation with same functionality. The implementation with python is usually more readable and easily, allowing faster development. Performance tests demonstrated that the Python interfaces do not involve a significant performance penalty. In sum, PyACTS is an intuitive, handy, and powerful tool to access ACTS tools from Python in a parallel setting.

4 A well-matched couple

We present an evolution of both tools (and PyACTS PyOpenCF) which is precisely the union and interaction to achieve high performance platform for cloud computing philosophy. For this purpose, a new service called *PyOpenCF&PyACTS Web Client* was added to the PyACTS' distribution web (http://pyacts.umh.es). Thus, a user can log into the portal and submit their papers through the web browser, without need for compilation or libraries linking. The same code can be executed by Python different computing platforms without being rewritten. The work environment OpenCF management control processes and their results in the various computer servers as explained in section 2. The innovation introduced by the union of both tools are the programming flexibility and power in performance we achieved by making use of ACTS library routines from the Python language. In Figure 4, the pyacts.umh.es web client is shown.

The features of the application that has been developed in this version are the following:

- List of servers: the list of registered computing servers in the system, including server name and the address on the same.
- List of scripts: shows the user a list of previously existing scripts or stored on a remote system.

 ISBN 13: 978-84-613-5510-5

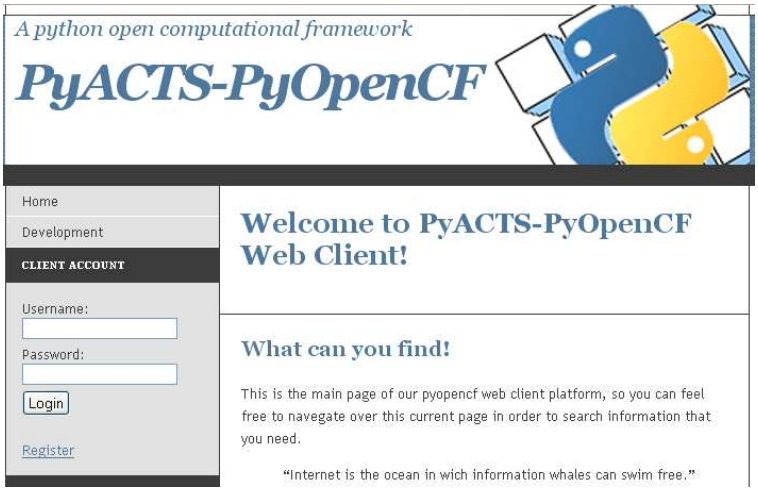


Figure 4: PyACTS-PyOpenCF Web Client

- Creation of new scripts / applications: allows user to define their own algorithms and programs using the Python scripting language.
- Launch a job from a script: From a Python uploaded code , user can launch a job selecting a computing server.
- Job Status: You can view the status of a particular job. It shows both the ID of the job, as the status in the remote server.
- Download results: if the job produces results as a file, we could it at any time by downloading from the web service.
- Deleting scripts: You can remove applications you no longer need to use.

In short, it seeks to achieve a more comfortable computing services work without worrying about the implementation of where or how the computation is done. Computing as a service is achieved with this architecture.

5 Conclusions

In this work we have presented a new web service which provides a framework to execute user applications in high performance servers in a comfortable and simple way. Python example has demonstrated that PyACTS is a user friendly interface that hides the challenges of parallel programming from non professional users, and PyOpenCF has illustrated a integrated framework for managing jobs in a set of remote servers. Both architectures allows users writing and submitting their own codes in available computing servers without worrying about compiling, linking, queue management, etc. The proposed web client is available for scientific community at `pyacts.umh.es`.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This research was partially supported by the Spanish Ministry of Science and Education under grant number TIN2008-06570-C04-04.

References

[1] Blackford LS, Choi J, Cleary A, D'Azevedo E, Demmel JW, Dhillon I, Dongarra J, Hammarling S, Henry G, Petitet A, Stanley K, Walker D, Whaley RC *ScaLAPACK User's Guide.* SIAM, Philadelphia, Pennsylvania(1997)

[2] Foster IT, Zhao Y, Raicu I, Lu S, *Cloud computing and grid computing 360-degree compared*, CoRR abs/0901.0131 (2009).

[3] L. A. Drummond and V. Galiano and V. Migallón and J. Penadés *PyACTS: A High-Level Framework for Fast Development of High Performance Applications* Lectures Notes in Computer Science Vol.4395: 417–425, 2007

[4] L. A. Drummond ,O. A. Marques *The ACTS Collection. Robust and high-performance tools for scientific computing: Guidelines for tool inclusion and retirement.* Tech. Rep. LBNL/PUB-3175, Computational research division, Lawrence Berkeley National Laboratory, Berkeley

[5] L. A. Drummond, V. Galiano, V. Migallón José Penadés, *PyACTS: A Python Based Interface to ACTS Tools and Parallel Scientific Applications*, International Journal of Parallel Programming, ISSN 0885-7458. DOI 10.1007/s10766-008-0083-4 (2008).

[6] MySQL Database Manager, *http://www.mysql.org/*

[7] Python Programming Languaje, *http://www.python.org/*

[8] A. Santos, F. Almeida , V. Blanco and J. C. Castillo, *Web services based scheduling in OpenCF*, The Journal of Supercomputing, ISSN 0920-8542. DOI 10.1007/s11227-009-0352-z (2009).

[9] A. Santos, F. Almeida, V. Blanco, *Lightweight Web Services for High Performance Computing", European Conference on Software Architecture* Madrid. Spain.(2007)

[10] Snir M, Otto S, Huss-Lederman S, Walker D, Dongarra J *MPI: The complete reference.* The MIT Press, Cambridge, MA (1998)

[11] A. Santos, F. Almeida, V. Blanco, *The OpenCF: an Open Source Computational Framework based on Web Services technologies* Seventh International Conference on Parallel Processing and Applied Mathematics PPAM2007, Gdansk, Poland.(2007)

 ISBN 13: 978-84-613-5510-5

[12] NuSOAP. Librera SOAP PHP, *http://dietrich.ganx4.com/nusoap/*

[13] WSDL (WS Description Language), *http://www.w3.org/TR/wsdl/*

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Matrices with maximal growth factor for Neville elimination

Pedro Alonso[1], Jorge Delgado[2], Rafael Gallego[1] and Juan Manuel Peña[2]

[1] *Departamento de Matemáticas, Universidad de Oviedo*

[2] *Departamento de Matemática Aplicada, Universidad de Zaragoza*

emails: palonso@uniovi.es, jorgedel@unizar.es, rgallego@uniovi.es, jmpena@unizar.es

Abstract

Neville elimination is a direct method for the resolution of linear systems of equations alternative to Gaussian elimination, with advantages for some classes of matrices and in the context of pivoting strategies for parallel implementations. The growth factor is an indicator of the numerical stability of an algorithm. In the literature bounds for the growth factor corresponding to Neville elimination with some pivoting strategies have appeared. In this work we determine all the matrices such that the minimal upper bound of the growth factor corresponding to Neville elimination with those pivoting strategies is reached.

Key words: Neville elimination, pivoting strategies, maximal growth factor
MSC 2000: 65F05, 65G05

1 Introduction

The usual direct method to solve a linear system of equations $Ax = b$ is Gaussian elimination (GE). Neville elimination (NE) is an alternative procedure to GE to transform a square matrix A into an upper triangular matrix U, and it has advantages for some classes of matrices and in the context of pivoting strategies for parallel implementations. NE makes zeros in a column of the matrix A by adding to each row a multiple of the previous one. Here we only give a brief description of this procedure (for a detailed and formal introduction we refer to [11]). If $A \in \mathbb{R}^{n\times n}$, NE consists of at most $n-1$ steps:

$$A = A^{(1)} \to \widetilde{A}^{(1)} \to A^{(2)} \to \widetilde{A}^{(2)} \to \cdots \to A^{(n)} = \widetilde{A}^{(n)} = U,$$

where U is an upper triangular matrix.

 ISBN 13: 978-84-613-5510-5

On the one hand, $\widetilde{A}^{(t)}$ can be obtained from the matrix $A^{(t)}$ through an adequate pivoting strategy, so that the rows with a zero entry in column t are the final rows and

$$\widetilde{a}_{it}^{(t)} = 0, \quad i \geq t \quad \Rightarrow \quad \widetilde{a}_{ht}^{(t)} = 0, \quad \forall\, h \geq i.$$

For example, partial pivoting for NE was already introduced in [12]. On the other hand, $A^{(t+1)}$ is obtained from $\widetilde{A}^{(t)}$ making zeros in the column t below the main diagonal by adding an adequate multiple of the ith row to the $(i+1)$th for $i = n-1, n-2, \ldots, t$. If A is nonsingular, the matrix $A^{(t)}$ has zeros below its main diagonal in the first $t-1$ columns. It has been proved that this process is very useful with totally positive matrices, sign-regular matrices and other related types of matrices (see [10] and [11]).

A real matrix is called totally positive (TP) if all its minors are nonnegative. TP matrices arise in a natural way in many areas of Mathematics, Statistics, Economics, etc. (see [7]). In particular, their application to Approximation Theory and Computer Aided Geometric Design (CAGD) is of great interest. For example, coefficient matrices of interpolation or least square problems with a lot of representations in CAGD (the Bernstein basis, the B-spline basis, etc.) are TP. Some recent applications of such kind of matrices to CAGD can be found in [16] and [17]. For applications of TP matrices to other fields see [10]. In [9], [11] and [13] it has been proved that NE is a very useful alternative to GE when working with TP matrices.

In addition, there are some studies that prove the high performance computing of NE for any nonsingular matrix (see [6]). In [5] the backward error of NE has also been analyzed. In [1] we give a sufficient condition that ensures the convergence of iterative refinement using NE for a system $A\,x = b$ with A any nonsingular matrix in $\mathbb{R}^{n\times n}$, and then we apply it to the case where A is TP. Other applications and a study of the stability have been presented in [2].

The growth factor is an indicator of the numerical stability of an algorithm. The growth factor for different pivoting strategies has been studied in [8], [14], [15], [18] and [4] for both Gaussian and Neville elimination. In addition, in [3] the authors have presented some examples where NE outperforms GE, showing the relation of this fact with the growth factor.

2 Matrices with maximal growth factor for Neville elimination

In the backward error analysis of GE with partial pivoting or complete pivoting on a matrix A performed by Wilkinson (see for example page 108 of [19]) it was shown the influence of the growth factor defined by

$$\rho_n(A) := \frac{\max_{i,j,k} |a_{ij}^{(k)}|}{\max_{i,j} |a_{ij}|}$$

where $a_{ij}^{(k)}$ occurring during the elimination process.

 ISBN 13: 978-84-613-5510-5

The growth factor corresponding to GE with partial pivoting for a matrix A is bounded above by 2^{n-1}. In [15] N. J. Higham and D. J. Higham determined all the matrices for which that bound is reached.

The backward error analysis of NE was performed in [5]. Again the growth factor plays an important role in the numerical stability of NE. The growth factor corresponding to NE with a row pivoting strategy such that the magnitude of the multipliers are less than or equal to one for a matrix A is bounded above by 2^{n-1}. For clarity we will denote this growth factor by $\rho_n^r(A)$. In this work we determine all the matrices for which that bound is reached:

Theorem 1 *All real $n \times n$ matrices A for which $\rho_n^r(A) = 2^{n-1}$ are of the form*

$$A = D\,B\,\left[\begin{array}{c|c} T & \theta\, d \\ \hline 0 & \end{array}\right]$$

where $D = diag(\pm 1)$, $B = (b_{ij})_{1\leq i,j\leq n}$ is the lower triangular matrix given by

$$b_{ij} = \begin{cases} 0, & \text{if } i < j, \\ \binom{i-1}{j-1}, & \text{if } i \geq j, \end{cases}$$

T is a nonsingular upper triangular matrix of order $n-1$, d is the vector given by

$$(1, -2, 4, \ldots, (-2)^{n-1})^t,$$

and θ is a scalar such that $\theta = \max_{1\leq i,j\leq n} |a_{ij}|$.

Acknowledgments

This work has been partially supported by the Spanish Research Grant MTM2009-07315, by Gobierno de Aragón and under MEC and FEDER Grant TIN2007-61273.

References

[1] P. Alonso, J. Delgado, R. Gallego and J. M. Peña, *Iterative Refinement for Neville Elimination*, Int. J. Comput. Math. **86(2)** (2009) 341–353.

[2] P. Alonso, J. Delgado, R. Gallego and J. M. Peña, *Neville elimination: an efficient algorithm with application to Chemistry*, to appear in J. Math. Chem.

[3] P. Alonso, J. Delgado, R. Gallego and J. M. Peña, *A collection of examples where Neville elimination outperforms Gaussian elimination*, to appear in Appl. Math. Comput.

[4] P. Alonso, J. Delgado, R. Gallego and J. M. Peña, *Grotwh Factors of Pivoting Strategies Associated to Neville Elimination*, to appear in J. Comp. Appl. Math.

 ISBN 13: 978-84-613-5510-5

[5] P. Alonso, M. Gasca and J. M. Peña, *Backward Error Analysis of Neville Elimination*, Appl. Numer. Math. **23** (1997) 193–204.

[6] P. Alonso, R. Cortina, I. Díaz and J. Ranilla, *Neville Elimination: a Study of the Efficiency Using Checkerboard Partitioning*, Linear Algebra Appl. **393** (2004) 3–14.

[7] T. Ando, *Totally Positive Matrices*, Linear Algebra Appl. **90** (1987) 165–219.

[8] V. Cortés and J. M. Peña, *Growth factor and expected growth factor of some pivoting strategies*, J. Comp. Appl. Math. **202** (2007) 292–303.

[9] J. Demmel and P. Koev, *The Accurate and Efficient Solution of a Totally Positive Generalized Vandermonde Linear System*, SIAM J. Matrix Anal. Appl. **27** (2005) 142–152.

[10] M. Gasca and and C. A. Micchelli, eds., *Total Positivity and its Applications*, Kluwer Academic Publishers, Boston, 1996.

[11] M. Gasca and J. M. Peña, *Total positivity and Neville elimination*, Linear Algebra Appl. **165** (1992) 25–44.

[12] M. Gasca and J. M. Peña, *Scaled pivoting in Gauss and Neville elimination for totally positive systems*, Appl. Numer. Math. **13** (1993) 345–356.

[13] M. Gassó and J. R. Torregrosa, *A Totally Positive Factorization of Rectangular Matrices by the Neville elimination*, SIAM J. Matrix Anal. Appl. **25** (2004) 986–994.

[14] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.

[15] N. J. Higham and D. J. Higham, *Large growth Factors in Gaussian Elimination with Pivoting*, SIAM J. Matrix Anal. Appl. **10** (1989), 155–164.

[16] H. Lin, H. Bao and G. Wang, *Totally positive bases and progressive iteration approximation*, Comput. Math. Appl. **50** (2005) 575–586.

[17] J. M. Peña, *Shape preserving representations in Computer Aided-Geometric Design*, Nova Science Publishers, Inc., New York, 1999.

[18] L. N. Trefethen and R. S. Schreiber, *Average case stability of Gaussian elimination*, SIAM J. Matrix Anal. Appl. **11** (1990) 335–360.

[19] J. H. Wilkinson, *Rounding Errors in Algebraic Processes*, Notes on Applied Science No. 32, Her Majesty's Stationery Office, London, 1963.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Execution time, efficiency and scalability of a block parallel algorithm

Pedro Alonso[1], Raquel Cortina[2], José Ranilla[2] and Antonio M. Vidal[3]

[1] *Department of Mathematics, Universidad de Oviedo, Spain*

[2] *Department of Computer Science, Universidad de Oviedo, Spain*

[3] *Department of Computer Systems and Computation, Universidad Politécnica de Valencia, Spain*

emails: palonso@uniovi.es, raquel@uniovi.es, ranilla@uniovi.es, avidal@dsic.upv.es

Abstract

This paper analyses the performance of several versions of a block parallel algorithm in order to apply Neville elimination in a distributed memory parallel computer. Neville elimination is an alternative procedure to Gauss elimination to transform a square matrix A into an upper triangular one. This analysis must take into account the algorithm behaviour as far as execution time, efficiency/speedup and scalability are concerned. Special attention has been paid to the study of the scalability of the algorithms trying to establish the relationship existing between the size of the block and the performance obtained in this metric. It is important to emphasize the high efficiency achieved for the studied cases. Moreover, the experimental results confirm the theoretical approximation obtaining a tool of analysis of high predicting ability.

Key words: Neville elimination, block parallel algorithms, execution time, efficiency, scalability

1 Introduction

This paper analyses the performance of several versions of a block parallel algorithm in order to apply Neville elimination to a matrix in a parallel computer using a message passing paradigm and identifying the values of the parameters that are necessary to obtain an optimal performance.

With regard to Neville elimination, it is an alternative procedure to that of Gauss to transform a square matrix A into an upper triangular one. Neville elimination makes zeros on an A column adding a multiple of the previous row to each row (for a detailed

 ISBN 13: 978-84-613-5510-5

and formal introduction, we refer to [7]). This strategy has been proved to be specially useful when using certain types of matrices as is the case of those which are totally positive or sign-regular matrices (see [1] and [6]).

A real matrix is called totally positive if all its minors are non-negative. It is possible to come across this sort of matrices in many different branches of science such as Mathematics, Statistics, Economy (see [6]), or Computer Aided Geometric Design (see [12] and [11]). According to [9], [5], [8] and [7], Neville elimination is considered to be an interesting alternative to Gauss elimination for certain types of research. Furthermore, there are other works (see [2], [3] and [4]) that show the advantages of the foresaid procedure in the field of High Performance Computing.

In the developing of the parallel algorithms to solve Numeric Linear Algebra problems, the block-organization seems to be the most efficient one in order to get the highest performance in current machines when dealing both with the good usage of the memory hierarchy of shared memory machines and with the harnessing of the explicit parallelism of those distributed memory ones. Thanks to this organization both efficient and scalable algorithms are usually obtained. Well-known subroutines such as Lapack and ScaLapack use the block organization as their main algorithm strategy design.

In our work, we propose an organization of the block Neville elimination algorithm for computers with the message passing model and we carry out a general analysis based on upper bounds for the three metrics: execution time, efficiency/speedup and scalability. We will concentrate on the most common block distributions and on two different representative machines of the message passing model: a network of workstations and a multicomputer. Special attention has been paid to the study of the scalability of the algorithms trying to establish the relationship existing between the size of the block and the performance obtained in this metric. It should be noted that previous works (see [2], [3] and [4]) have only addressed the scalability for some particular cases. However, in this paper we analyze the general cases obtaining general conclusions.

In the rest of this paper we briefly review some of the aspects that will be considered: the performance model (Section 2), the block parallel algorithm (Section 3) and a reduced set of experiments obtained (Section 4).

2 Performance model

In multicomputers, the physical environment used to share this information is the interconnection network, while the logical paradigm is, in general, so-called *Message Passing*. In message passing the information moves from its origin to its destination establishing communications in which the speakers cooperate actively. In our experiments we have used this kind of paradigm.

The evaluation of a parallel algorithm requires a minimum study of the characteristics of the systems and a theoretical model that could predict its behaviour. Several models have been already proposed and they keep becoming more complex and precise as well as having a more complex application. Together with several authors, this paper

 ISBN 13: 978-84-613-5510-5

has adopted a model based on the compromise between the precision of the predictions and their simplicity of usage (see [10]).

3 Block parallel algorithm

In order to handle a matrix in parallel, we must divide it in such a way that the partitions can be assigned to the different processors. The distribution of the matrix data affects the performance of the parallel system as explained in the following sections. Therefore, determining the best distribution for each algorithm becomes a relevant issue. This section studies a type of generic partition where the matrix $A = (a_{ij})_{1 \leq i,j \leq n}$ is divided into $n_0 \times n_1$ submatrices $A = (A_{ij})_{1 \leq i \leq n_0, 1 \leq j \leq n_1}$ of $m_0 \times m_1$ dimension:

$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1,n_1} \\ A_{21} & A_{22} & \cdots & A_{2,n_1} \\ \vdots & \vdots & \vdots & \vdots \\ A_{n_0,1} & A_{n_0,2} & \cdots & A_{n_0,n_1} \end{pmatrix},$$

where $n_0 = n/m_0$ and $n_1 = n/m_1$. We can assume that n is divisible by m_0 and m_1.

Let us consider a rectangular mesh of $p_0 \times p_1$ processors where the processor in row s and column t is denoted as P_{st}, with $1 \leq s \leq p_0$ and $1 \leq t \leq p_1$. The submatrices will be distributed in a cyclical way among the processors so that each processor will contain different non-adjacent blocks.

The number of submatrices assigned to each processor is the same: $h_0 \times h_1$ where h_0 and h_1 are obtained in the following way:

$$h_0 = \frac{n}{m_0\ p_0} \text{ and } h_1 = \frac{n}{m_1\ p_1}.$$

Let us study the parallel algorithm performance in a jth stage. In this iteration the variable x_j must be removed. Hence, it is necessary to nullify the elements a_{nj}, $a_{n-1,j}$, ..., $a_{j+1,j}$. So that, the following steps must be accomplished:

1. Each active processor P_{st} will send the a_{ik} elements, with $i, k \geq j$, of the last row of each of its submatrices to the immediately inferior processor; that is: $P_{s+1,t}$. The processors of the last row of the processors mesh will send the foresaid elements to the corresponding processors of row 1.

2. Those processors containing a_{ij} elements, with $i > j$, of matrix A will be the ones in charge of calculating the multipliers of the j stage. Let q be the column of the mesh of processors where these processors are placed then the operative processors P_{sq}, with $1 \leq s \leq p_0$, will calculate these multipliers.

3. The forementioned processors will communicate the calculated multipliers to the active processors placed in the same row of the mesh of processors. Therefore, the broadcast will take place from each active processor P_{sq}, with $1 \leq s \leq p_0$, to the active processors P_{sr} with $1 \leq r \leq p_1$.

 ISBN 13: 978-84-613-5510-5

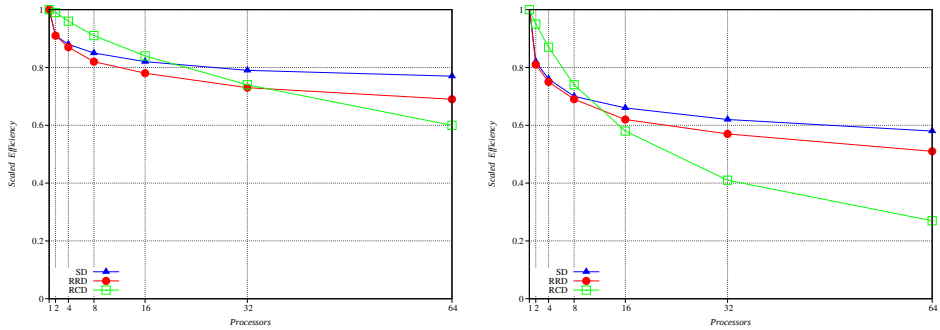


Figure 1: IBM SP2 (left) and network of PCs (right) scaled efficiency

4. Each active processor will update all the elements of matrix A which correspond to the rows and columns with index larger than j.

Taking into account the mentioned steps, we analyze the execution time of the foresaid parallel algorithm in detail. Next we will discuss three particular cases of data partition. Firstly, we will deal with a bidimensional partition in which the coefficient matrix of the system is divided into square submatrices (SD). As far as the classic unidimensional partitions are concerned, we will study those in which the data matrix is divided into complete consecutive rows (RRD) or columns (RCD). In the three cases, the submatrices are cyclically distributed among the processors.

4 Experimental results

Once finished the theoretical studies, we will compare the theoretical model with the empirical values as far as the efficiency and the scalability are concerned.

The experimental results showed in this section have been obtained from two variants of the distributed memory model: an IBM SP2, subcategory MPP, and a network of PCs (a cluster subcategory). The programming paradigm used for the communications is message passing through the implementation MPICH 1.2.7 of the standard MPI 1.2.

As far as the ability of our first distribution (SD) to keep a constant efficiency is concerned, figure 1 shows that when increasing n and W (the number of basic operations required by the fastest-known sequential algorithm to solve the foresaid problem in one processor) equally the efficiency is slightly weakened in the IBM SP2 while this weakening is stronger in the network of PCs. For instance, in the IBM SP2 the efficiency ranges from 0.91 for $(n, p) = (1210, 2)$ to 0.77 for $n = 3841$ and $p = 64$. Thus, there has been a 0.14 loss of efficiency. The efficiency loss is 0.24 in the network of PCs as it drops from 0.81 to 0.57.

Figure 1 shows the evolution of the scaled efficiency as the number of processors and W increase equally in the second distribution (RRD). It is obvious that efficiency does not remain constant being its degradation lower in the IBM SP2 than in the

 ISBN 13: 978-84-613-5510-5

network of PCs. Nevertheless, as many other authors have already pointed out, it may be considered to be almost scalable provided that its scaled efficiency stays higher than 0 ($EE(W,p) > 0$) (see [13]).

Figure 1 shows the scaled efficiency that corresponds to the RCD. We can also observe that the efficiency is reduced to 0.60 and to 0.27 in the IBM SP2 and the network of PCs respectively. In both environments the drop of the efficiency is slightly sharper than in the other distributions.

5 Conclusions

The capability of the algorithms has been analysed by using three different metrics: execution time, efficiency/speedup and scalability.

As far as execution time is concerned, the SD and RRD often obtain good results. Nevertheless, in dealing with this metric, the best distribution is the column-block oriented one since its communication time does not depend on block size.

In RCD the size of the block can be lowered in order to reduce the calculation time without increasing the communication time. However, for other distributions, we must get an optimal device in the size of the block that will not damage the communication time.

If the focus of the analysis were the efficiency, conclusions would be similar to the forementioned in relation to execution time.

It is also important to point out that the best distribution is the one based on the SD, as far as scalability is concerned. In this case, the advantage of the RCD disappears since the size of the block does not influence the scalability.

Acknowledgements

This work has been partially supported by the Spanish Research Grant TIN2007-61273.

References

[1] T. Ando, Totally Positive Matrices. *Linear Algebra and its Applications* 90:165–219 (1987).

[2] P. Alonso, R. Cortina, I. Díaz and J. Ranilla, Analyzing Scalability of Neville Elimination. *Journal of Mathematical Chemistry* 40(1):49–61 (2006).

[3] P. Alonso, R. Cortina, I. Díaz and J. Ranilla, Scalability of Neville elimination using checkerboard partitioning, *International Journal of Computer Mathematics* 85(3-4):309–317 (2008).

[4] P. Alonso, R. Cortina, I. Díaz and J. Ranilla, Blocking Neville elimination algorithm for exploiting cache memories. *Applied Mathematics and Computation* 209:2–9 (2009).

 ISBN 13: 978-84-613-5510-5

[5] J. Demmel and P. Koev, The Accurate and Efficient Solution of a Totally Positive Generalized Vandermonde Linear System. *SIAM Journal on Matrix Analysis and Applications* 27:142–152 (2005).

[6] M. Gasca and C.A. Michelli, *Total Positivity and its Applications*. Kluwer Acad. Publ., Dordrecht, 1996, p.518.

[7] M. Gasca and J.M. Peña, Total Positivity and Neville Elimination. *Linear Algebra and its Applications* 165:25–44 (1992).

[8] M. Gassó M and J.R. Torregrosa, A Totally Positive Factorization of Rectangular Matrices by the Neville elimination. *SIAM Journal on Matrix Analysis and Applications* 25:986–994 (2004).

[9] L. Gemignani, Neville Elimination for Rank-Structured Matrices. *Linear Algebra and its Applications* 428(4):978–991 (2008).

[10] A. Grama, A. Gupta, G. Karypis and V. Kumar, *Introduction to Parallel Computing*. Pearson Education Limited, 2003, p. 636.

[11] H. Lin, H. Bao and G. Wang, Totally positive bases and progressive iteration approximation. *Computers and Mathematics with Applications* 50:575–586 (2005).

[12] J.M. Peña, *Shape preserving representations in Computer Aided–Geometric Design*. Nova Science Publishers, 1999, p. 233.

[13] M. Prieto, R.S. Montero, I.M. Llorente and F. Tirado, A Parallel Multigrid Solver for Viscous Flows on Anisotropic Structured Grids. *Parallel Computing* 29:907–923 (2003).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A numerical method to simulate periodic travelling-wave solutions of some nonlinear dispersive wave equations

Jorge Álvarez[1] and Ángel Durán[1]

[1] *Department of Applied Mathematics, University of Valladolid, Spain*

emails: joralv@eis.uva.es, angel@mac.uva.es

Abstract

The paper proposes a numerical method to simulate periodic travelling-wave solutions of some nonlinear dispersive wave equations. The construction of the method is based on an efficient computation of the elements that characterize these solutions: the initial profile and the velocity of the wave.

Key words: periodic travelling waves, Petviashvilli methods, conserved quantities, conservative methods
MSC 2000: 65M20, 65M99, 35Q53, 76B25

1 Introduction

The purpose of the paper is to introduce a numerical method to simulate travelling-wave solutions of the periodic problem for nonlinear dispersive wave equations of the general form

$$u_t + f(u)_x - Mu_x = 0, \quad x \in (-L, L), \quad t > 0. \tag{1}$$

where $u = u(x,t)$ is a $2L$-periodic, real-valued function of the two real independent variables x, t; f is a smooth, real-valued function of u, representing a nonlinear term and M is a linear, nonnegative, formally self-adjoint operator, characterized as a Fourier multiplier operator by its symbol

$$\widehat{Mv}(\xi) = \alpha(\xi)\widehat{v}(\xi),$$

where $\widehat{\ }$ denotes Fourier transform. Equations of the form (1) appear in many models concerning the propagation of small-amplitude, nonlinear, dispersive long waves, see e. g. [1, 2] and references therein as a modest representation of the literature on (1). Important cases are included, such as the generalized KdV equation ($f(s) = s^p/p,\ p \geq 2,\ M = -\partial_{xx}$), the generalized Benjamin-Ono equation ($f(s) = s^p/p,\ p \geq$

 ISBN 13: 978-84-613-5510-5

2, $M = -\mathcal{H}\partial_x$, where $\mathcal{H}$ stands for the Hilbert transform) or the Benjamin equation ($f(s) = s^2$, $M = -\gamma_1\mathcal{H}\partial_x - \gamma_2\partial_{xx}$, for some parameters γ_1, γ_2).

Periodic travelling-wave solutions of (1) are periodic functions of the form $\phi(x-ct)$, for $c > 0$, representing the velocity of the wave, and they play a relevant role in the models [12]. In general, explicit expressions of these solutions cannot be obtained by analytical techniques and a numerical treatment is necessary. The numerical method we describe here is focused on the elements that characterize these solutions. First we need to implement an efficient computation of the profile ϕ, combining a suitable spatial discretization with an iterative procedure. On the other hand, a correct simulation of the velocity determines the selection of the time integrator.

The paper is structured as follows: in Section 2 we make some hypotheses on (1) and remind some analytical properties of the equations under study that are relevant for our work. The numerical method is treated in Section 3. It includes a description of the spatial discretization, the iterative technique to approximate the initial profiles and the time integration. Some numerical illustrations are shown in Section 4.

2 Preliminaries

Several hypotheses on the nonlinear term f and the symbol α are assumed.

(H1) f is a polynomial of the form $f(z) = a_p z^p + \cdots + a_2 z^2$ with $a_p > 0$, $a_j \geq 0$, $j = 2, \ldots, p-1$, for some $p \geq 2$.

(H2) $\alpha : \mathbb{R} \to \mathbb{R}$ is continuous, even, nonnegative with $\alpha(0) = 0$.

(H3) α is monotone increasing on $[0, \infty)$ and there exists $m > (p-1)/p$ such that $\liminf_{\xi\to\infty} \alpha(\xi)/|\xi|^m > 0$.

We will consider (1) defined in the space $X = H^s_{per}$ of periodic, H^s functions with period $2L$, for some $s \geq 1/2$ and with the usual norm

$$||u||_s = \left(\int_{-\infty}^{\infty} (1 + |\xi|^2)^s |\hat{u}(\xi)|^2 d\xi \right)^{1/2}.$$

Hypotheses (H1)-(H3) are assumed to guarantee the existence of solutions of (1) [4, 7] in X and they include the important cases cited above. For initial data in H^s, the following quantities

$$I(u) = \int_{-L}^{L} u(x,t)dx, \tag{2}$$

$$V(u) = \frac{1}{2}\int_{-L}^{L} u^2(x,t)dx, \tag{3}$$

$$H(u) = \int_{-L}^{L} \left(\frac{1}{2}(u(x,t)Mu(x,t)) - F(u(x,t)) \right) dx, \tag{4}$$

 ISBN 13: 978-84-613-5510-5

where $F' = f,\ F(0) = 0$, are invariants by the solutions of (1). The quantity (4) is the Hamiltonian of the problem, that can be written as

$$u_t = \mathcal{J}\delta H(u), \quad u \in H^s,$$

where δ denotes variational derivative and $\mathcal{J} = \partial_x$.

A second relevant property we mention here is the existence of periodic travelling-wave solutions. They are of the form $u(x,t) = \phi(x - ct)$ where $c > 0$ represents the wave velocity and the profile $\phi = \phi_c$ is $2L$-periodic and satisfies the equation

$$\mathcal{J}(\delta H(\phi_c) + c\delta V(\phi_c)) = 0,$$

that can be written as

$$\delta H(\phi_c) + c\delta V(\phi_c) = A\delta I(\phi_c), \tag{5}$$

that is

$$M\phi_c - f(\phi_c) + c\phi_c = A, \tag{6}$$

where A is an integration constant. We also note that the one-parameter group of translations in space is a symmetry group of (6). This defines an orbit of solutions $\{\phi_c(x - x_0) : x_0 \in \mathbb{R}\}$ whose elements remain in the same level set $\{\varphi / V(\varphi) = V(\phi_c)\}$. The parameter x_0 would play the role of the phase of the wave.

For further purposes (see Section 3) we will make the following assumptions on (6) [14, 3]:

(H4) $f(\phi_c(x)) \geq 0, \quad x \in \mathbb{R}$.

(H5) The linearized operator of (6) at ϕ_c,

$$L_c = c + M - f'(\phi_c),$$

has a unique, negative, simple eigenvalue, the zero eigenvalue is simple and the rest of its spectrum is bounded away from zero.

3 Description of the numerical method

We shall describe our proposal to simulate periodic travelling-wave solutions of (1), whose analytical form is, in general, unknown. As mentioned above, the method is focused on a suitable approximation to the elements that determine these solutions: the profile ϕ and the velocity c.

 ISBN 13: 978-84-613-5510-5

3.1 Spatial discretization

The nonlocal term in (1) makes the spectral-type methods a good selection for the spatial discretization. Here we approximate the solutions of the $2L$-periodic problem of (1) by a Fourier pseudospectral discretization. First we make the description for the 2π-periodic problem and then we adapt the formulation for any interval of periodicity $(-L, L)$ [5, 6, 13]. For an even number N of nodes $x_j = -\pi + jh,\ h = 2\pi/N,\ j \in \mathbb{Z}$, we consider the space S_h of periodic functions $Z = \{Z_j\}_{j\in\mathbb{Z}}$ defined on the grid, with $Z_{j+N} = Z_j$. For each $Z \in S_h$, the discrete Fourier coefficients

$$\widehat{Z}_p = \frac{1}{N}\sum_{0\le j\le N}{}'' Z_j e^{-ipjh}, \quad -\frac{N}{2} \le p \le \frac{N}{2}, \tag{7}$$

provide the information of Z in the Fourier space. In (7), the double prime in the sum denotes that the first and last terms are divided by two. The reconstruction of Z from the Fourier coefficients is carried out by evaluating, at the grid points, the trigonometric interpolation polynomial

$$Z_h(x) = \sum_{-N/2\le p\le N/2}{}'' \widehat{Z}_p e^{ipx}, \tag{8}$$

in such a way that $Z_j = Z_h(x_j)$.

On the other hand, the pseudospectral differentiation operator on Z is obtained by differentiating (8) with respect to x and evaluating at the x_j:

$$(DZ)_j = \sum_{-N/2\le p\le N/2}{}'' \widehat{Z}_p(ip)e^{ipjh}, \quad j \in \mathbb{Z}.$$

In terms of the discrete Fourier coefficients, we have

$$\widehat{(DZ)}_p = (ip)\widehat{Z}_p, \quad -N/2 \le p \le N/2,$$

which means that, in the Fourier space, the operator D diagonalizes and differentiation is represented by the product with the diagonal matrix with elements $ip, -N/2 \le p \le N/2$.

With the Fourier pseudospectral method based on the x_j, the semidiscrete approximation to the 2π-periodic problem of (1) is a map $U : [0,\infty) \to S_h$ satisfying

$$\begin{aligned} &\frac{dU_j}{dt}(t) + (D(f(U)))_j(t) + M(DU)_j(t) = 0, \\ &U_j(0) = u(x_j, 0), \quad 0 \le j \le N-1, \end{aligned} \tag{9}$$

where

- $U(t) = (U_0(t), \ldots, U_{N-1}(t))$ and $U_j(t)$ is an approximation to $u(x_j, t),\ j = 0, \ldots, N-1$.

 ISBN 13: 978-84-613-5510-5

- The expression $f(U(t))$ denotes $(f(U_0(t)),\ldots,f(U_{N-1}(t)))$.

Observe that if $\widehat{U}_p(t)$ is the p-th discrete Fourier component of $U(t)$, the system (9) can be described in a more suitable form

$$\begin{aligned}&\frac{d}{dt}\widehat{U}_p(t) = (ip)(\alpha(p)\widehat{U}_p(t) + \widehat{f(U)}_p(t))\\ &\widehat{U}_p(0) = \widehat{u}_p(0),\end{aligned} \tag{10}$$

where $\widehat{u}_p(0)$ denotes the p-th discrete Fourier component of $(u(x_0,0),\ldots,u(x_{N-1},0))$. System (9) is then implemented in the form (10). This system is stiff, which will influence the choice of the time integrator. On the other hand, the computation of the nonlinear term can be made by minimizing the generation of aliasing errors [5]. It is also well known the properties of convergence of the pseudospectral method, depending on the smoothness of the solution [13]. Finally, the connection with the $2L$-periodic problem of (10) requires to transform the spatial variable in the form $X = \pi(x+L)/L$ and to write (1) with the corresponding scaling. In particular, the pseudospectral differentiation operator must be multiplied by the factor π/L.

3.2 Generation of the initial profile

The combination of the pseudospectral spatial discretization with the adaptation of the Petviashvili's method to compute travelling-wave solutions establishes a technique to generate the initial profile.

Petviashvili's method [15] was originally implemented to compute solitary waves of the initial value problem for the KPI equation, and its application to stationary and solitary-wave solutions of other problems has also been proposed (see e. g. [11] and references therein). The method can be adapted to the periodic case as follows. Denoting by $\widehat{\phi}(k)$ the k-th Fourier coefficient of ϕ, equation (6) generates a system for the Fourier coefficients

$$(c+\alpha(k))\widehat{\phi}(k) - \widehat{f(\phi)}(k) = A\mathbb{I}(k), \quad k\in\mathbb{Z},$$

where $\mathbb{I}(k)$ denotes the k-th Fourier coefficient of the function $u=1$. Then the Fourier coefficients of ϕ satisfy

$$\widehat{\phi}(k) = \frac{\widehat{f(\phi)}(k) + A\mathbb{I}(k)}{(c+\alpha(k))}, \quad k\in\mathbb{Z}. \tag{11}$$

Note that if we multiply (6) by ϕ and integrate in $(-L,L)$ then

$$K = K(\phi) = \frac{\int_{-L}^{L}\phi((c+M)\phi)dx}{\int_{-L}^{L}\phi(A+f(\phi))dx} = 1. \tag{12}$$

For the numerical approximation to the solution of (11), the classical fixed point iteration usually diverges. The adaptation of the Petviashvili's method would introduce a modified iterative scheme with a stabilizing factor

$$\widehat{\phi}(k)^{[\nu+1]} = K(\phi^{[\nu]})^{\gamma}\frac{\widehat{f(\phi^{[\nu]})}(k) + A\mathbb{I}(k)}{(c+\alpha(k))}, \quad k\in\mathbb{Z}, \quad \nu = 0,1,\ldots, \tag{13}$$

 ISBN 13: 978-84-613-5510-5

where $\widehat{\phi}(k)^{[\nu]}$ stands for the ν-th iteration and γ is a free parameter chosen to make (13) be convergent. In the case of the initial value problems and solitary wave solutions, local convergence is obtained under the assumptions (H4)-(H5) and for $\gamma \in (1,(p+1)/(p-1))$ [14]. Furthermore, the fastest rate of convergence occurs for $\gamma^* = p/(p-1)$.

Having in mind the spatial discretization described in the previous subsection, the discrete version of the iterative procedure (13) can be written in terms of the discrete Fourier coefficients of the pseudospectral approximation to the profile ϕ:

$$\widehat{Z}_p^{[\nu+1]} = \widetilde{K}(Z^{[\nu]})^\gamma \frac{\widehat{f(Z^{[\nu]})}_p + A\mathbb{I}(p)}{(c+\alpha(p))}, \quad -N/2 \leq p \leq N/2, \quad \nu = 0,1,\ldots \tag{14}$$

The stabilizing factor $\widetilde{K}(Z)$ is obtained as follows. From the Parseval identity, (12) can be written in terms of the Fourier coefficients as

$$K(\phi) = \frac{\sum_{k=-\infty}^{\infty}(c+\alpha(k))|\widehat{\phi}(k)|^2}{\sum_{k=-\infty}^{\infty}(\widehat{f(\phi)}(k) + A\mathbb{I}(k))\overline{\widehat{\phi}(k)}}.$$

Then, for $Z \in S_h$, we define

$$\widetilde{K}(Z) = \frac{\sum_{p=-N/2}^{N/2}(c+\alpha(p))|\widehat{Z}_p|^2}{\sum_{p=-N/2}^{N/2}(\widehat{f(Z)}_p + A\mathbb{I}(p))\overline{\widehat{Z}_p}}. \tag{15}$$

3.3 Time integration

The choice of the time integrator is determined by the search of a good simulation of the velocity parameter of the periodic travelling wave. Classical discretizations of (1) always include some properties of preservation of discrete versions of the invariants of the problem in the features of the numerical method. Recently, some results ([3, 8] and references therein) show that a better simulation of the parameters of travelling wave solutions is related to the preservation, through the numerical integration, of some invariants of the problem. Explicitly, the analysis of the time propagation of the error shows that this is affected by secular components, associated to the parameters of the wave. These secular terms behave better in those methods that preserve discretized versions of the invariants of the problem, providing a more suitable, in a qualitative sense, simulation of the travelling wave. In [8] this was studied for one of the equations included in (1), the KdV equation, and we conjecture that similar conclusions for the cases covered by (1), under the hypotheses (H1)-(H5), also hold.

Accordingly to this, it seems that preservation of (2), (3) and (4) should be a desirable property for a time integrator in this context. By considering the pseudospectral spatial discretization, we introduce the discrete versions of the invariants

$$\widetilde{I}(Z) \quad = \quad h\sum_{j=0}^{N-1} Z_j, \tag{16}$$

 ISBN 13: 978-84-613-5510-5

$$\widetilde{V}(Z) = \frac{h}{2}\sum_{j=0}^{N-1} Z_j^2, \tag{17}$$

$$\widetilde{H}(Z) = h\sum_{j=0}^{N-1} \frac{1}{2}(Z_j(MZ)_j) - F(Z)_j, \tag{18}$$

for $Z \in S_h$.

It is also necessary to pay attention to other difficulties of the discretization. Then in this case our strategy will be to consider the problem of stiffness of (10) as a first selection criterion of the time integrator to be used, and then studying the preservation of the invariants. To this end, we first choose the simply diagonally implicit Runge-Kutta (SDIRK) method of order three and tableau

$$\begin{array}{c|cc} \frac{3+\sqrt{3}}{6} & \frac{3+\sqrt{3}}{6} & 0 \\ \frac{3-\sqrt{3}}{6} & \frac{-\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \tag{19}$$

The method has a good computational behaviour in the implicit systems for the intermediate stages to be solved at each step. However, (19) does not preserve discrete versions of the invariants (3) and (4). The preservation of (17), (18) will be forced by using the projection technique (see [10, 9]). We can make a brief description of the method in the case of the preservation of $\widetilde{V}$ (the case of $\widetilde{H}$ would be similar). Assume that V_0 is the value of (3) at the initial profile, and let U_n be the numerical approximation to the solution $u(t)$ of (1) at some time discrete value t_n. Then the following approximation U_{n+1} is carried out in two steps:

(i) Compute $\tilde{U}_{n+1}$ by using (19).

(ii) Project the value $\tilde{U}_{n+1}$ onto the manifold

$$\mathcal{M}_0 = \{Z \in S_h / \tilde{V}(Z) = V_0\}$$

The second step is done by solving a constrained minimization problem, see [10, 9] for details.

Note that the first quantity (16) is not included since it is linear and therefore it is preserved by almost all methods used in practice [10]. On the other hand, condition (5) establishes a relation between the variational derivatives of the invariants (2), (3) and (4) at the initial profile. When simulating periodic travelling-wave solutions, this has two consequences. The first one is that we cannot implement a projection to preserve the three quantities at the same time. The second one is that a better performance is obtained when the method preserves two of the quantities, but there is no priority in the selection of the invariants to be conserved (see e. g. [8] for the details). When simulating perturbations of these periodic travelling waves or other periodic solutions of (1), then (5) does not hold and the situation may be different.

 ISBN 13: 978-84-613-5510-5

4 Numerical experiments

In order to illustrate the numerical performance of the method previously described, in this section we will consider the periodic Benjamin-Ono equation as the model problem. This equation is the case of (1) corresponding to $f(u) = u^2/2$ and $M = -\mathcal{H}\partial_x$ where $\mathcal{H}$ is the periodic Hilbert transform

$$\mathcal{H}u(x) = PV\frac{1}{2L}\int_{-L}^{L} cot\left(\frac{\pi}{2L}y\right)u(x-y)dy,$$

which has the symbol $\alpha(k) = |k|$. Since periodic travelling-wave solutions of this problem are known, this will serve us to illustrate the behaviour of the method. We will consider the solution (see e. g. [16])

$$u_L(x,t) = \frac{2c\delta^2}{1-\sqrt{1-\delta^2}\cos(c\delta(x-ct-x_0))}, \quad x \in (-L,L), \quad t > 0, \tag{20}$$

with $c > 0,\ \delta = \pi/(cL),\ x_0 \in \mathbb{R}$. The corresponding profile $\phi_L(x) = u(x,0)$ satisfies (6) with $A = 0$. In the next sections, we will take $L = 16$, $c = 1$ and $x_0 = 0$. The numerical experiments, according to the main goals of the method (explained above), are focused on the computation of the profile and the simulation of the parameters.

4.1 Computation of the profile

As far as the first question is concerned, Table 1 shows, for two different starting iterations, the error in Euclidean norm between the exact profile at the points x_j and the corresponding numerical approximation given by the iteration (14), controlled by a maximum number of iterations and a tolerance for the relative error between two consecutive iterations. The starting profiles are small perturbations of the exact one, in the form $Z_j = \phi_L(x_j) + \epsilon e^{-x_j^2}$, with $\epsilon = 1E-03$ for the second column (ERROR1) and $\epsilon = 1E-01$ for the third column (ERROR2). The experiments are performed with $\gamma = 2$, which is optimal for the iteration in the case of the initial value problem [14]. The results show the convergence of the iteration, although the third column reveals its local character, since for an starting value which is not so close to the exact profile, the convergence is slower. On the other hand, Figure 1 (left) shows the behaviour of

Iteration	ERROR1	ERROR2
5	9.8374E-07	9.5179E-03
6	7.3680E-07	7.1284E-03
7	5.5415E-07	5.3611E-03
8	4.1783E-07	4.0423E-03

Table 1: Errors of the iterative method (14). Starting iteration $Z_j = \phi_L(x_j) + \epsilon e^{-x_j^2}$ with $\epsilon = 1E-03$ (ERROR1) and $\epsilon = 1E-01$ (ERROR2).

 ISBN 13: 978-84-613-5510-5

the error in the stabilizing factor (15) as a function of the number of iterations, for the values $\gamma = 1.1, 2, 2.9$. In Figure 1 (right), we take $\gamma = 0.8, 3.1$. Both are obtained with the initial data used in the second column of Table 1. We have already mentioned that, in order to assure the local convergence, for the corresponding iteration in the initial value problem, γ must be in the range $(1, (p+1)/(p-1))$ ($p = 2$ in our case) [14]. The numerical computations performed here suggest that this also happens in the case of the periodic problem. For values of γ out of this interval, the stabilizing factor do not converge and the the iteration procedure diverges. This is observed in Figure 1 (right).

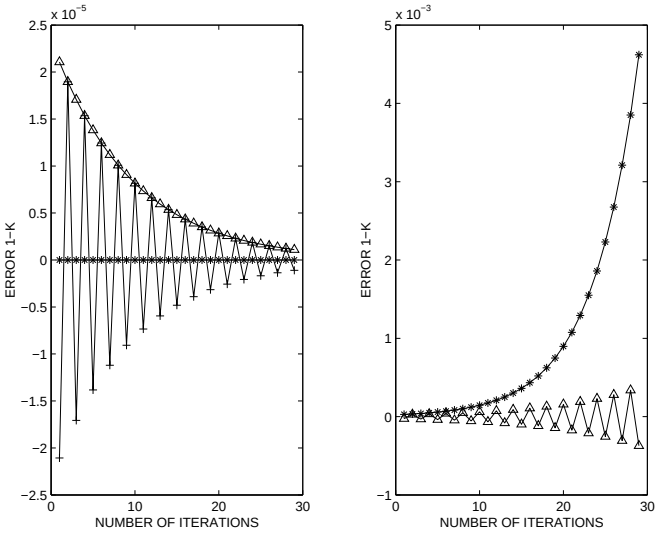


Figure 1: Error in the computation of (15) against number of iterations. Left: $\gamma = 2$ ($*$), $\gamma = 2.9$ ($+$), $\gamma = 1.1$ ($\triangle$). Right: $\gamma = 0.8$ ($*$), $\gamma = 3.1$ ($\triangle$).

4.2 Numerical simulation of the parameters

In order to illustrate the numerical evolution of the velocity, we have simulated the periodic problem with two time integrators: the method (19), denoted by SD, as an example of a nonconservative method, and the scheme (19) combined with a projection to preserve the quantities (16) and (17), denoted by CSD. A Hamiltonian preserving method gives similar results and will not be shown here (see the remarks in the previous section). We have to observe that, although SD preserves (16), the modification to force the conservation of (17) gives a not I-preserving method, and then it is necessary to use a projection involving both quantities.

We first measure the differences in a long time simulation between the two methods, comparing the corresponding approximation with the exact solution (20). Figure 2 shows, in logarithmic scale, the evolution of the error, for different values of the time step, up to a final time $t = 10^3$. Solid lines correspond to SD and broken lines to CSD.

 ISBN 13: 978-84-613-5510-5

The slope of the lines show that, for the SD scheme, errors behave as t^2, being the growth only linear in the case of CSD. This reveals a better performance of the latter for long time simulations.

The most harmful components of the error, considered as a function of time, seem to be related to the parameters. Figure 3 shows, also in log-log scale, the evolution of the error between the velocity of the numerical approximation and the exact one for the same experiments as those of Figure 2. The computation of the numerical velocity has been made in a standard way ([8] and references therein). Left figure corresponds to SD and right figure to CSD. Note that, while the simulation of SD provides a computation of the velocity that grows linearly with time with respect to the exact one, the CSD method gives a constant in time approximation to the exact c, with no secular perturbations. Note also that, due to the relation given in (20), the simulation of c will affect the amplitude of the numerical wave. The experiments performed in this section

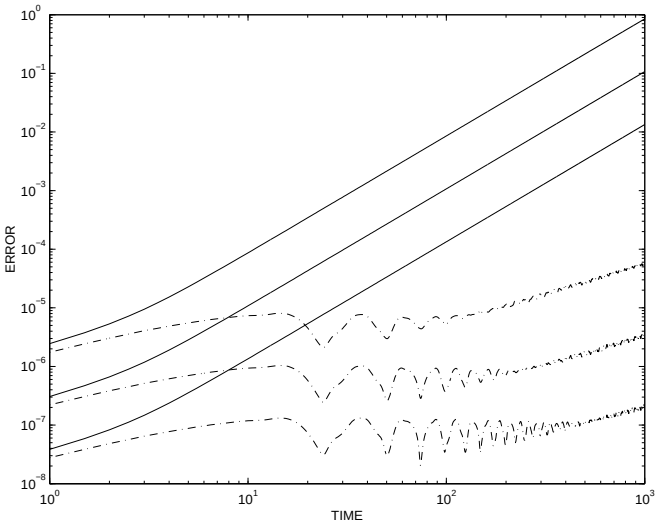


Figure 2: Error vs time in log-log scale. Solid lines: SD. Dashed lines: CSD. The time steps are $\Delta t = 1/80, 1/160, 1/320$.

for the Benjamin-Ono equation, and the theory explained in [8] for the KdV equation, suggest the following behaviour of the parameters of the numerical approximations to periodic travelling-wave solutions of the general problem (1) under the conditions (H1)-(H5). The numerical solution would contain a travelling wave profile $U(x, t_n, c_n, x_{0,n})$ with the main source of the error. The parameters c_n and $x_{0,n}$ are perturbations of c and x_0 respectively, that evolve with time and depend on the integrator used. In the general case, the dominant terms of $x_{0,n}$ contain quadratic in time perturbations of the original phase, while the leading behaviour of c_n, when comparing with c, is linear in time. This would explain the performance of the SD method shown in Figures 2 and 3. This behaviour is improved when the method preserves discrete versions of the quantity (2) and of one of the quantities (3) or (4). In this case, the leading term of

 ISBN 13: 978-84-613-5510-5

the perturbation of c is constant in time and the numerical solution is affected by a change of phase which grows linearly in time. This provides, in a qualitative sense, a better simulation for long times. We have also made the same experiments as above

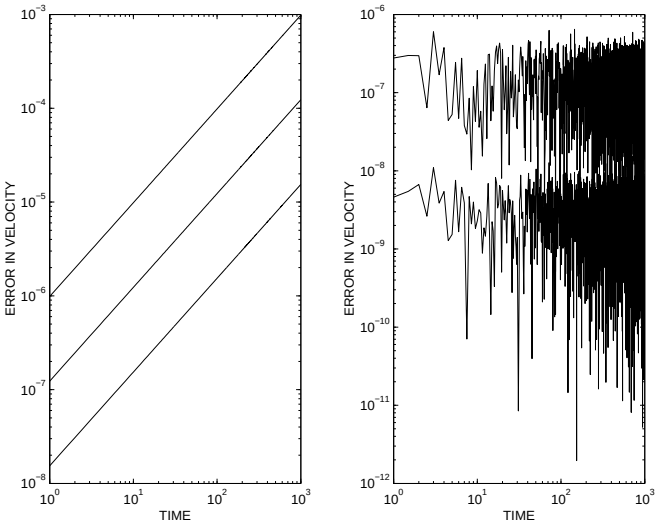


Figure 3: Error in speed vs time in log-log scale. Left: SD with time steps $\Delta t = 1/80, 1/160, 1/320$. Right: CSD with time steps $\Delta t = 1/80, 1/320$.

but with an initial profile obtained after a convergent process using (14) for $\gamma = 2$ and a perturbation of the exact profile as starting iteration. The numerical results in the simulation where similar to those shown here, that were obtained with the exact profile as initial data. This suggests to consider this combination of techniques as an efficient way to approximate periodic travelling-wave solutions of (1) with unknown analytical expression or to simulate perturbations of these waves.

Acknowledgements

This work was supported by: Ministerio de Ciencia e Innovación, under projects MTM2008-00700/MTM, MTM2008-01396-E and MTM2009-06507-E; Junta de Castilla y León, under projects VA001A08 and VA060A09 and Consolider Ingenio Mathematica, under project SAIRT-C4-0189.

References

[1] L. Abdelouhab, J. L. Bona, M. Felland and J. C. Saut, *Nonlocal models for nonlinear, dispersive waves*, Physica D **40** (1989) 360–392.

[2] J. P. Albert, J. L. Bona and J. C. Saut, *Model equations for stratified fluids*, Proc. R. Soc. London A **453** (1997) 1233–1260.

 ISBN 13: 978-84-613-5510-5

[3] J. Álvarez and A. Durán, *Numerical simulation of solitary waves in nonlinear dispersive equations*, submitted.

[4] T. B. Benjamin, J. L. Bona and D. K. Bose, *Solitary-wave solutions of nonlinear problems*, Phil. Trans. R. Soc. London A **331** (1990) 195–244.

[5] J. P. Boyd, *Chebyshev and Fourier Spectral Methods*, 2nd ed. Dover Publications, New York, 2000.

[6] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang, *Spectral Methods in Fluid Dynamics*. Springer-Verlag, New York-Heidelberg-Berlin, 1988.

[7] H. Chen, *Existence of periodic travelling-wave solutions of nonlinear, dispersive wave equations*, Nonlinearity **17** (2004) 2041–2056.

[8] A. Durán, *Time behaviour of the error when simulating finite-band periodic waves. The case of the KdV equation*, J. Comput. Phys. **227** (2008) 2130–2153.

[9] E. Hairer, C. Lubich and G. Wanner, *Geometric Numerical Integration, Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer-Verlag, New York-Heidelberg-Berlin, 2002.

[10] E. Hairer, S. P. Norsett and G. Wanner, *Solving Ordinary Differential equations I. Nonstiff Problems*, 2nd ed. Springer Series in Computational Mathematics 8, Springer-Verlag, New York-Heidelberg-Berlin, 1993.

[11] T. I. Lakoba and J. Yang, *A generalized Petviashvili method for scalar and vector Hamiltonian equations with arbitrary form of nonlinearity*, J. Comput. Phys. **226** (2007) 1668–1692.

[12] P. D. Lax, *Periodic solutions of the KdV equation*, Comm. Pure Appl. Math. **28** (1975) 141–188.

[13] J. Pasciak, *Spectral methods for a nonlinear initial value problem involving pseudo-differential operators*, SIAM J. Numer. Anal. **19** (1982) 142–154.

[14] D. E. Pelinovsky and Y. A. Stepanyants, *Convergence of Petviashvili's iteration method for numerical approximation of stationary solutions of nonlinear wave equations*, SIAM J. Numer. Anal. **42** (2004) 1110–1127.

[15] V. I. Petviashvili, *Equation of an extraordinary soliton*, *Soviet J. Plasma Phys.* **2** (1976) 257–258.

[16] V. Thomée and A. S. Vasudeva Murthy, *A numerical method for the Benjamin-Ono equation*, BIT **38** (1998) 597–611.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A generalization of s-step variants of Gradient methods

J.A. Alvarez-Dios[1], J.C. Cabaleiro[2] and G. Casal[1]

[1] *Departamento de Matemática Aplicada, Universidade de Santiago de Compostela*

[2] *Departamento de Electrónica e Computación, Universidade de Santiago de Compostela*

emails: joseantonio.alvarez.dios@usc.es, jc.cabaleiro@usc.es, xerardo.casal@usc.es

Abstract

The s-step methods were proposed by Chronopoulos to gain efficiency in parallel programming of iterative methods for linear systems. They converge for all symmetric, nonsymmetric definite and some nonsymmetric indefinite coefficient matrices. In this paper we introduce a s-step variant of a general orthogonalization algorithm, we prove convergence and obtain error estimates. From this we derive the well known s-step methods as particular cases, and some new others to our knowledge. This provides a unified framework to derive and study s-step methods. The new methods obtained are convergent for every nonsingular matrix.

Key words: Iterative methods; s-step; large linear systems; Krylov subespace; parallel computation
MSC 2000: AMS codes (65F10, 65Y05, 68W10)

1 Introduction

In iterative method solvers for large linear systems most required computations are vector-vector and matrix-vector operations. In the language of the basic linear algebra subprograms (BLAS) [10], level 1 BLAS operations. On the other hand, BLAS 2 and BLAS 3 operations, based on blocks of submatrices, are much more efficient than BLAS 1 operations on parallel computers with optimized BLAS kernels.

In order to improve the BLAS 2-3/BLAS 1 ratio, an alternative approach using BLAS 3 operations in some iterative methods for linear systems was the s-step methods proposed by Chronopoulos [5, 6]. The efficiency of these methods on parallel computers is corroborated in [4].

The aim is to generalize these s-step variant to other Conjugated Gradient type methods in order to obtain iterative algorithms for the resolution of large linear systems, also valid even in the case of nonsymmetric and/or positive nondefinite matrices, with

 ISBN 13: 978-84-613-5510-5

better performance in parallel programming. For such purpose we present a s-step variant of the general orthogonalization algorithm that can be seen, for example, in [9] and we obtain different s-step variants of this method by previously fixing two parameter matrices.

2 Background

It is assumed throughout this paper that A is in general a real nonsingular matrix of order n, $b \in \mathbb{R}^n$ a column vector and $\mathcal{M}_{n\times s}(\mathbb{R})$ the set of real matrices of order $n \times s$. Denote the symmetric and the antisymmetric part of any matrix A by A^S and A^{aS} respectively. If $v_1, \ldots, v_s$ are vectors, $\pounds\{v_1, \ldots, v_s\}$ will stand for the vector subspace they span. In an analogous way, if $A_1, \ldots, A_s$ are real matrices, $\pounds\{A_1, \ldots, A_s\}$ will stand for the vector subspace spanned by all columns of all matrices. The aim of the iterative methods, object of this paper, is the numerical resolution of the linear system

$$Ax = b \tag{1}$$

whose exact solution will be denoted by c.

We shall now recall some elementary key definitions. For each $v \in \mathbb{R}^n$, $v \neq 0$ and $s \in \mathbb{N}$, $s < n$, we call the vector subspace $\pounds\{v, Av, A^2v, \ldots, A^{s-1}v\}$ a Krylov subspace of order s, and we denote it by $\mathcal{K}_s(A, v)$.

If $\dim(\mathcal{K}_s(A, v)) < s$, and therefore the dimension of $\mathcal{K}_s(A, v)$ were not maximum, the inverse of A would be a polynomial in A of degree $\deg(v) - 1$ at most and we could easily construct the exact solution of the system. We often refer to this circumstance as *lucky breakdown*, which is highly unlikely in practice.

The s-step variant of the *Conjugated Gradient* algorithm (s-CG) was introduced by Chronopoulos and Gear in [5]. Subsequently, in [6], Chronopoulos generalizes this method for some types of positive not necessarily definite and not necessarily symmetric matrices. More specifically, article [6] deals with the s-step variants of the *Generalized Conjugated Residual method* (s-GCR), of the *Minimal Residual* (s-MR) and of the *Orthomin(m)* (s-Orthomin(m)), and particularly for the case s-Orthomin(1) known by *s-Conjugate Residual method* (s-CR).

The convergence of these methods in at most $[n/s]$ iterates is proved in [5, 6] for symmetric positive definite matrices in s-CG, and for symmetric, nonsymmetric definite indefinite matrices with definite symmetric part in s-Orthomin(m) and s-GCR. Therefore these methods are not convergent for every nonsingular matrix.

The s-step variants of the Double Orthogonal Series can be seen in [1], which converge for every nonsingular matrix. Basing ourselves on them, we shall try to construct valid methods for a general nonsingular matrix.

 ISBN 13: 978-84-613-5510-5

3 s-step variant of the General Orthogonalization Algorithm

If matrix A is neither necessarily symmetric nor positive definite, there is a more general algorithm than the Conjugate Gradient method, namely the *General Orthogonalization Algorithm* (GOA). In what follows we describe this method in a summarized way (see [9], for example).

Let $Ax = b$ be the linear system of order n with nonsingular matrix A. Let H, K be square matrices of order n with positive definite symmetric part. We set $N = A^t H^S A$ and $M = L^t N L$, where LL^t is the Cholesky factorization of the symmetric part of K, and then $K^S = LL^t$. For all $r \in \mathbb{R}^n$ let us define $E(r) =< r, Hr >$. From the equality $E(r) =< H^t r, r >=< r, Hr >$ we get $E(r) =< r, \frac{1}{2}(H + H^t)r >=< r, H^S r >$. Then $E(r)$ must be a convex function. Next we write the General Orthogonalization Algorithm (GOA), which is presented in [9]:

Algorithm 3.1 *(GOA).*
Let $x_0 \in \mathbb{R}^n$,
$r_0 = b - Ax_0 = A(x - x_0)$
$g_0 = A^t H^S r_0 = A^t H^S A(x - x_0) = N(x - x_0)$
$p_0 = Kg_0$
For $i = 0, 1, \ldots$ *until convergence* ***Do****:*

$$\alpha_i = \frac{< g_i, p_i >}{< p_i, Np_i >} \tag{2}$$

$$x_{i+1} = x_i + \alpha_i p_i \tag{3}$$

$$g_{i+1} = g_i - \alpha_i N p_i = A^t H^S r_{i+1} \tag{4}$$

$$\beta_{i+1}^l = -\frac{< Kg_{i+1}, Np_l >}{< p_l, Np_l >}, \quad l = 0, \ldots, i \tag{5}$$

$$p_{i+1} = Kg_{i+1} + \sum_{l=0}^{i} \beta_{i+1}^l p_l \tag{6}$$

EndFor

We denote as vectors g_i the general residues, and as vectors p_i the general descent directions. The GOA converges in at most n iterations is proved in [9]. Moreover, if we denote by $E_i = E(r_i) =< r_i, H^S r_i >$, then the following error estimate is also proved in [9]:

$$E_i \leq E_0 \left(1 - \frac{\lambda_{\min}(L^t (K^{-1})^S L)}{\text{cond}(M)}\right)^i \tag{7}$$

where $\text{cond}(M)$ is the condition number of M and $\lambda_{\min}(L^t (K^{-1})^S L)$ the minimum eigenvalue of the matrix $(L^t (K^{-1})^S L)$. Since matrix K is symmetric:

$$E_i \leq E_0 \left(\frac{\text{cond}(M) - 1}{\text{cond}(M) + 1}\right)^{2i}. \tag{8}$$

 ISBN 13: 978-84-613-5510-5

Observe that, if K is a symmetric matrix, then **iv)** is valid for every $i \neq j$ and taking the value of Np_l from (4) to (5), we get $\beta_{i+1}^l = 0$ for all $0 \leq l < i$ and the sum in (6) reduces to the last term. In this case, storage of the preceding directions p_l, $l = 0, \ldots, i-1$, is not necessary to compute p_{i+1}. On the contrary, if K is a nonsymmetric matrix, and more than a few iterations are needed, then the storage requirements become prohibitive. To circumvent this, the general Orthomin(m) method computes p_{i+1} by N-orthogonalizing to the m preceding directions only, m being a parameter previously chosen. Giving matrix H and K particular values in the General Orthogonalization Algorithm, we obtain some known methods [9].

Fixed $n, s \in \mathbb{N}$ and $M \in \mathcal{M}_{n\times s}(\mathbb{R})$, for $v \in \mathbb{R}^n$ we define $\Delta_M(v)$ by the matrix with column vectors $v, Mv, M^2v, \ldots, M^{s-1}v$. Then $\mathcal{K}_s(KN, Kg_0)$ is the vector subspace generated by the column vectors of the matrix $\Delta_{KN}(Kg_0)$. We define the s-step variant of the general orthogonalization algorithm (s-GOA):

Algorithm 3.2 *(s-GOA).*
Let $x_0 \in \mathbb{R}^n$
$r_0 = b - Ax_0$
$g_0 = A^t H^S r_0$
$P_0 = \Delta_{KN}(Kg_0) = Q_0$
For $i = 0, 1, 2, \ldots$ *until convergence* ***Do***

$$W_i = (P_i)^t N P_i \tag{9}$$
$$z_i = (P_i)^t g_i \tag{10}$$
$$y_i = (W_i)^{-1} z_i \tag{11}$$
$$x_{i+1} = x_i + P_i y_i \tag{12}$$
$$g_{i+1} = g_i - NP_i y_i = A^t H^S r_{i+1} \tag{13}$$
$$Q_{i+1} = \Delta_{KN}(Kg_{i+1}) \tag{14}$$

For $j = 0, \ldots, i$ ***Do:***

$$B_{i+1}^j = -W_j^{-1}(P_j)^t N Q_{i+1} \tag{15}$$

EndFor

$$P_{i+1} = Q_{i+1} + \sum_{j=0}^{i} P_j B_{i+1}^j \tag{16}$$

EndFor

By induction on i we can obtain in (13) the following equation for the residual:

$$r_{i+1} = r_i - AP_i y_i. \tag{17}$$

Comparing GOA (algorithm 3.1) with s-GOA (algorithm 3.2) it is easy to verify that BLAS 1 and BLAS 2 become BLAS 2 and BLAS 3 operations, respectively.

We establish the following lemma relating direction and general residual vectors of s-GOA:

 ISBN 13: 978-84-613-5510-5

Lemma 3.1 *It holds that:*

(a) $(P_i)^t N P_j = 0$ *for all* $i \neq j$.

(b) $(P_j)^t g_i = 0$ *for all* $i > j$.

(c) $(P_i)^t g_i = (Q_i)^t g_i$.

(d) $(Q_j)^t g_i = 0$ *for all* $i > j$.

(e) $(P_i)^t N Q_j = 0$ *for all* $i > j$.

(f) $(P_i)^t N P_i = (P_i)^t N Q_i$.

(g) $(P_i)^t g_j = (P_i)^t g_0$ *for all* $i \geq j$.

Proof.–
Part *(a)* is true since we have that $(P_j)^t N P_i = (P_j)^t N Q_i + (P_j)^t N \sum_{k=0}^{i-1} P_k B_i^k$ and $(P_j)^t N P_j B_i^j = -(P_j)^t N Q_i$. Therefore from equations (9) and (15) in Algorithm 3.2, we can easily prove by induction on $i > j$ that $(P_j)^t N P_i = 0$. To prove *(b)*, for a fixed $j \in \mathbb{N}$ we obtain by induction on i that g_i is orthogonal to P_j for all $i > j$:
If $i = j+1$, since $g_{j+1} = g_j - N P_j y_j$ conclude that g_{j+1} is orthogonal to P_j by definition of y_j, W_j and z_j.
Now suppose that g_i is orthogonal to $\mathcal{P}_j$, with $i > j+1$. Then, the orthogonality between g_{i+1} and P_j is a consequence of the induction hypothesis and N-orthogonality between P_i and P_j. The equality *(c)* follows from definition of P_i and *(b)*. Since $Q_j = P_j - \sum_{k}^{j-1} P_k B_j^k$ then *(b)* implies *(d)*.Part *(e)* is also from $Q_j = P_j - \sum_{k}^{j-1} P_k B_j^k$ and *(a)*. The equality *(f)* follows from definition of P_i and *(a)*. The identity $g_j = g_{j-1} - N P_{j-1} y_{j-1}$ and induction give *(g)*. □

We denote by $\{p_0, p_1, \ldots, p_{(i+1)s-1}\}$ the direction vectors computed in GOA and by $p_i^1, \ldots, p_i^s$ the direction vectors of s-GOA in each iteration, and then $P_i = (p_i^1 \,|\ldots|\, p_i^s)$. Now, we can establish the following lemma relating Krylov and direction subspaces generated in both algorithms:

Lemma 3.2 *Let* $i, s \in \mathbb{N}$ *be such that* $s(i+1) \leq n$. *Suppose that* $g_i \neq 0$. *If* $\dim \mathcal{K}_{s(i+1)}(KN, Kg_0) = s(i+1)$ *then:*

$$\pounds\{P_0, \ldots, P_i\} = \bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) = \mathcal{K}_{s(i+1)}(KN, Kg_0) = \pounds\{p_0, p_1, \ldots, p_{(i+1)s-1}\}, \tag{18}$$

where $\bigoplus$ *denotes the direct sum of vectorial subspaces.*
Moreover r_{i+1} *minimizes* $E(r) =< r_i, H^S r_i >$ *over* $x_0 + \pounds\{P_0, \ldots, P_i\}$.

 ISBN 13: 978-84-613-5510-5

Proof.–

It is obvious that $\bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) = \pounds\{Q_0, \ldots, Q_i\}$. Then, the equality

$$\pounds\{P_0, \ldots, P_i\} = \bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) \tag{19}$$

is proved by induction since $P_0 = Q_0$ and the definition of P_i, from which we can also obtain that $Q_i = P_i - \sum_{j=0}^{i-1} P_j B_i^j$.

The equality

$$\bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) = \mathcal{K}_{s(i+1)}(KN, Kg_0) \tag{20}$$

is trivial for $i = 0$. The inclusion

$$\bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) \subset \mathcal{K}_{s(i+1)}(KN, Kg_0) \tag{21}$$

is proved by induction since:

$$(KN)^k Kg_i = (KN)^k (Kg_{i-1} - KNP_{i-1}y_{i-1}). \tag{22}$$

for $k \in \{0, \ldots, s-1\}$ and, because of equality (19) and the induction hypothesis

$$(KN)^{k+1} p_{i-1}^1, \ldots, (KN)^{k+1} p_{i-1}^s \in \mathcal{K}_{s(i+1)}(KN, Kg_0) \tag{23}$$

so $(k+1)+(s\cdot i-1) = k+s\cdot i \leq s(i+1)-1$, and then $(KN)^k Kg_i \in \mathcal{K}_{s(i+1)}(KN, Kg_0)$. The other inclusion

$$\mathcal{K}_{s(i+1)}(KN, Kg_0) \subset \bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j) \tag{24}$$

is also proved by induction. Suppose that the inclusion verifies for $i-1$, $i > 1$ fixed. Since

$$g_i = g_0 + \sum_{j=0}^{is-1} \lambda_j N(KN)^j Kg_0, \tag{25}$$

which stems from (19), (21) and by induction in $g_i = g_{i-1} - NP_{i-1}y_{i-1}$, we have, for $k \in \{0, \ldots, s-1\}$

$$(KN)^k Kg_i = (KN)^k Kg_0 + \sum_{j=0}^{is-1} \lambda_j (KN)^{j+k+1} Kg_0. \tag{26}$$

Now, if we prove that $\lambda_{s\cdot i-1} \neq 0$ then $\mathcal{K}_{s(i+1)}(KN, Kg_0) \subset \bigoplus_{j=0}^{i} \mathcal{K}_s(KN, Kg_j)$. But $\lambda_{s\cdot i-1} \neq 0$ because if $\lambda_{s\cdot i-1} = 0$ in (25), then $Kg_i \in \mathcal{K}_{s\cdot i}(KN, Kg_0)$. From the induction

 ISBN 13: 978-84-613-5510-5

hypothesis and (19), $Kg_i \in < P_0, \ldots, P_{i-1} >$, which implies, by part *(b)* of Lemma 3.1, that $< g_i, Kg_i >= 0$. This is a contradiction if $g_i \neq 0$ because the symmetric part of K is positive definite.
The last equality $\mathcal{K}_{s(i+1)}(KN, Kg_0) = \mathcal{L}\{p_0, p_1, \ldots, p_{(i+1)s-1}\}$ is result **vi)** of GOA's properties previously cited.
Finally, let r_{i+1} be the residual which corresponds to iterate x_{i+1}. From definition of r_{i+1} and by induction we have

$$r_{i+1} = r_0 - \sum_{j=0}^{i} (AP_j y_j). \tag{27}$$

Since $(AP_j)^t H^s r_0 = (P_j)^t g_0 = (P_j)^t g_j = z_j$ for all $j = 0, \ldots, i$, using (27) and part *(a)* of Lemma 3.1 we have that

$$E(r_{i+1}) =< r_0, H^S r_0 > -2 \sum_{j=0}^{i} y_j^t z_j + \sum_{j=0}^{i} y_j^t W_j y_j. \tag{28}$$

Since $E(r)$ is convex, r_{i+1} is the minimal of $E(r)$ over $x_0 + \mathcal{L}\{P_0, \ldots, P_i\}$ if the coefficient vectors y_j, with $j = 0, \ldots, i$, are the solutions of the linear systems $W_j y_j = z_j$, but this is true by the definition of y_j in s-GOA. □
We have to observe that W_i is positive definite and consequently nonsingular. From Lemmas 3.1 and 3.2 we obtain the following convergence theorem:

Theorem 3.1 *If all previous hypothesis hold, the s-step general orthogonalization algorithm converges in at most* $[n/s]$ *iterations.*

Proof.–
Let $i \in \mathbb{N}$. Since Lemmas 3.1 and 3.2, if $g_i \neq 0$ then g_i is orthogonal to $\mathcal{K}_{s \cdot i}(KN, Kg_0)$. But $\dim \mathcal{K}_{s \cdot i}(KN, Kg_0) = s \cdot i$, and then, if $s \cdot i \geq n$ it is necessarily $g_i = 0$. This implies that $r_i = 0$, because $g_i = A^t H^S r_i$ and $A^t H^S$ is nonsingular. □

Let $\tilde{r}_i$ and r_i be the residual vectors in the ith iteration of the GOA and s-GOA, respectively. Since $E(r)$ is a convex function and from Lemma 3.2, if x_0 is the same for GOA and s-GOA then $\tilde{r}_{s \cdot i} = r_i$ in exact arithmetic. Thus we can establish the error estimate:

Theorem 3.2 *Under the hypothesis of Lemma 3.1, if* r_i *is the residual vector in the ith iteration of the s-GOA and* $E_i = E(r_i)$*, it verifies:*

$$E_i \leq E_0 \left(1 - \frac{\lambda_{\min}(L^t (K^{-1})^S L)}{cond(M)}\right)^{s \cdot i} \tag{29}$$

Moreover, if the matrix K *is symmetric, we have:*

$$E_i \leq E_0 \left(\frac{cond(M) - 1}{cond(M) + 1}\right)^{2s \cdot i}. \tag{30}$$

 ISBN 13: 978-84-613-5510-5

Proof.– The proof is obvious from the error estimate (7) and (8) in GOA and since $\tilde{r}_{s\cdot i} = r_i$. □

As seen in GOA, we need all the previous matrices P_j, $j = 0, \ldots, i$ in s-GOA, for the computation of B_{i+1}^j. If more than a few iterations are needed, then the storage requirements become prohibitive. Thanks to the following lemma, when matrix K is symmetric it will only be necessary to store the last of the series of all previous matrices P_j for the computation of matrices B_{i+1}^j.

Lemma 3.3 *If matrix K is symmetric then, for $j = 0, \ldots, i-1$*

$$(P_j)^t N Q_{i+1} = 0. \tag{31}$$

Proof.– Let $j \in \{0, \ldots, i-1\}$ fixed. Then $(P_j)^t N Q_{i+1}$ is a square matrix of order s whose kl element is $< (KN)^{k-1} K g_{i+1}, N p_j^l >$, with $k, l \in \{1, \ldots, s\}$. If K is symmetric then

$$< (KN)^{k-1} K g_{i+1}, N p_j^l > = < (g_{i+1}, (KN)^k p_j^l > . \tag{32}$$

From Lemma 3.2 we get $p_j^l \in \mathcal{K}_{s(j+1)}(KN, Kg_0)$ for $l = 1, \ldots, s$. So, if $j \leq i-1$ and $k \in \{1, \ldots, s\}$ then $(KN)^k p_j^l \in \mathcal{K}_{s(i+1)}(KN, Kg_0)$ because $k + (j+1)s - 1 \leq s + i \cdot s - 1 = s(i+1) - 1$. But again, from *(b)* of Lemma 3.1 and Lemma 3.2, g_{i+1} is orthogonal to $P_0, \ldots, P_i$ whose columns span $\mathcal{K}_{s(i+1)}(KN, Kg_0)$, then g_{i+1} is orthogonal to $\mathcal{K}_{s(i+1)}(KN, Kg_0)$ and thus we conclude that if $0 \leq j \leq i-1$ then the right side of (32) is zero. □

In this way, if K is symmetric, equations (15) and (16) of s-GOA becomes:

$$B_{i+1} = -W_i^{-1} (P_i)^t N Q_{i+1} \tag{33}$$

and

$$P_{i+1} = Q_{i+1} + P_i B_{i+1}. \tag{34}$$

4 Particular cases of the s-step General Orthogonalization Algorithm

From particular choices of matrices H and K, first we shall proceed to obtain the known s-step methods.

Suppose that matrix A is symmetric positive definite. Let $H = A^{-1}$ and K any symmetric positive definite matrix. Then $N = A$ and the s-GOA becomes into the s-step variant of the Preconditioned Conjugate Gradient Algorithm proposed in [4]. In the particular case of $K = I$ we have the s-step variant of the Conjugate Gradient Method [5].

Now choose $H = I$ and $K = A^{-1}$. Then $N = A^2$ and we obtain the s-step variant of the Generalized Conjugate Residual Algorithm proposed in [6]. If matrix A is symmetric positive definite, then K is symmetric and this method is the s-step variant of the Conjugate Residual Algorithm [5].

 ISBN 13: 978-84-613-5510-5

Since matrix K is not symmetric in general we can consider the Orthomin(m) method for this algorithm [6]. The Orthomin(0) is the s-step variant of the Minimal Residual Algorithm proposed in [6], and the Orthomin(1) is the s-step variant of the well known Axelsson's Minimal Residual [2].

If A is a nonsingular matrix, $H = I$ and $K = I$, then $N = A^tA$ and the resulting algorithm is the s-step variant of the Normal Equation which appears in [3].

4.1 s-Minimal Error Algorithm

Let A be a nonsingular matrix, $H = (AA^t)^{-1}$ and $K = A^tA$. Then K is symmetric and $N = I$. In this case we have that $g_i = A^{-1}r_i$ and $z_i = P_i^t g_i$. Then vector z_i depends on A^{-1} whose calculation would render the algorithm useless in practice. To avoid computing g_i we introduce the following matrices:

$$R_i = \Delta_{AA^t}(r_i),\ Q_0 = R_0 \text{ and } Q_i = R_i + Q_{i-1}B_{i-1} \text{ for } i > 0$$

It is obvious that $P_i = A^tQ_i$ and, since $P_i^tP_{i-1} = 0$, we deduce that $Q_i^t(AA^t)Q_{i-1} = 0$. Thus we propose in this paper the s-step variant of the Minimal Error Algorithm.

Algorithm 4.1 (s-Minimal Error).
Let $x_0 \in \mathbb{R}^n$
$r_0 = b - Ax_0$
$Q_0 = \Delta_{AA^t}(r_0)$
For $i = 0, 1, 2, \ldots$ *until convergence* ***Do***
$P_i = A^tQ_i$
$W_i = P_i^tP_i$
$z_i = Q_i^tr_i$
$y_i = W_i^{-1}z_i$
$x_{i+1} = x_i + P_iy_i$
$r_{i+1} = r_i - AP_iy_i$
$R_{i+1} = \Delta_{AA^t}(r_{i+1})$
$B_{i+1} = -W_i^{-1}(AP_i)^tR_{i+1}$
$Q_{i+1} = R_{i+1} + Q_iB_{i+1}$
EndFor

4.2 s-Biconjugate Gradient

If A is a nonsingular matrix, the Biconjugate Gradient method, [8], generates two CG-like sequences of vectors, one based on a system with the original coefficient matrix A, and another one with A^t. In this subsection we propose a s-step variant of the Biconjugate Gradient method. First, we define the following matrices:

$$\mathbf{A} = \begin{pmatrix} 0 & A \\ A^t & 0 \end{pmatrix},\ X = \begin{pmatrix} x^* \\ x \end{pmatrix},\ B = \begin{pmatrix} b \\ b^* \end{pmatrix}. \tag{35}$$

Let

$$\mathbf{H} = (\mathbf{A}^{-1})^t = \begin{pmatrix} 0 & (A^t)^{-1} \\ A^{-1} & 0 \end{pmatrix} \text{ and } \mathbf{K} = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}, \tag{36}$$

 ISBN 13: 978-84-613-5510-5

then $\mathbf{N} = \mathbf{A}^t\mathbf{H}\mathbf{A} = \mathbf{A}$ is a symmetric matrix. Superscript $*$ denotes the array part which is associated to the sequence based on A^t. The s-step variant of the Biconjugate Gradient method can be derived from the s-GOA method. Next we will write the s-Biconjugate Gradient method in terms of n-dimensional vectors. For this purpose we denote by

$$\mathbf{P}_i = \begin{pmatrix} P_i^* \\ P_i \end{pmatrix}, \ \mathbf{Q}_i = \begin{pmatrix} Q_i^* \\ Q_i \end{pmatrix} \text{ and } \ R_i = \begin{pmatrix} r_i \\ r_i^* \end{pmatrix} \tag{37}$$

and enunciate the following lemma:

Lemma 4.1 *In the s-Biconjugate Gradient method, and for all $k \in \{0, 1, 2, \ldots\}$, it holds that:*

(a) $(Q_i^*)^t A^k r_i = Q_i^t \left(A^t\right)^k r_i^*$

(b) $(Q_i^*)^t A^k Q_i = Q_i^t \left(A^t\right)^k Q_i^*$.

Proof.–
It is obvious from the fact that, for all $k \in \{0, 1, 2, \ldots\}$,

$$r_i^t \left(A^t\right)^k r_i^* = (r_i^*)^t A^k r_i. \tag{38}$$

□
The following lemma can be enunciated as a consequence of Lemmas 3.1 and 4.1:

Lemma 4.2 *In the s-Biconjugate Gradient method, and for all $k \in \{0, 1, 2, \ldots\}$, it holds that:*

(a) $(P_i^*)^t A^k P_i = P_i^t \left(A^t\right)^k P_i^*$

(b) $(P_i^*)^t A^k r_{i+1} = P_i^t \left(A^t\right)^k r_{i+1}^*$

(c) $(P_i^*)^t r_{i+1} = P_i^t r_{i+1}^* = 0$

(d) $(P_i^*)^t r_i = P_i^t r_i^* = (Q_i^*)^t r_i = Q_i^t r_i^*$.

Proof.–

We will prove statements *(a)* and *(b)* by induction on i. For $i = 0$ statements *(a)* and *(b)* are true from Lemma 4.1 and since $R_1 = R_0 - \mathbf{A}\mathbf{P}_0 y_0$. Suppose that *(a)* and *(b)* are true for $i - 1$ with $i \geq 1$, then $(P_i^*)^t A^k P_i =$

$$= (Q_i^*)^t A^k Q_i + (Q_i^*)^t A^k P_{i-1} B_{i-1} + B_{i-1}^t (P_{i-1}^*)^t A^k Q_i + B_{i-1}^t (P_{i-1}^*)^t A^k P_{i-1} B_{i-1} \tag{39}$$

and

$$P_i^t \left(A^t\right)^k P_i^* =$$

$$= Q_i^t (A^t)^k Q_i^* + Q_i^t (A^t)^k P_{i-1}^* B_{i-1} + B_{i-1}^t P_{i-1}^t (A^t)^k Q_i^* + B_{i-1}^t P_{i-1}^t (A^t)^k P_{i-1}^* B_{i-1} \tag{40}$$

 ISBN 13: 978-84-613-5510-5

The first summands of the second right hand side of (39) and (40) are equal as a consequence of Lemma 4.1. So are the other corresponding summands by the induction hypothesis on *(a)* and *(b)*, which proves *(a)*. On the other hand, $(P_i^*)^t A^k r_{i+1} = (P_i^*)^t A^k r_i - (P_i^*)^t A^{k+1} P_i y_i$ and $P_i^t \left(A^t\right)^k r_{i+1}^* = P_i^t \left(A^t\right)^k r_i^* - P_i^t \left(A^t\right)^{k+1} (P_i)^* y_i$. Then equality *(b)* is derived from *(a)*, Lemma 4.1 and induction hypothesis since $\mathbf{P}_i = \mathbf{Q}_i + \mathbf{P}_{i-1} B_i$.

Section *(c)* follows from statement *(b)* of Lemma 3.1, and statement *(d)* from previous *(c)*. □

Now, by the previous Lemma 4.2, we can write $W_i = (P_i^*)^t A P_i + P_i^t A^t P_i^* = 2(P_i^*)^t A P_i$ and then $W_i y_i = 2(P_i^*)^t r_i$.

As consequence of section *(b)* of Lemma 4.2, we have that $(P_i^*)^t A Q_{i+1} = P_i^t A^t Q_{i+1}^*$, and then $B_{i+1} = -(W_i)^{-1} \left(P_i^t A^t Q_{i+1}^* + (P_i^*)^t A Q_{i+1}\right) = -2(W_i)^{-1} \left(P_i^t A^t Q_{i+1}^*\right)$.

Finally, we can write the s-Biconjugate Gradient method in the following way:

Algorithm 4.2 (s-Biconjugate Gradient).
Let $x_0, x_0^* \in \mathbb{R}^n$
$r_0 = b - A x_0$
$r_0^* = b^* - A^t x_0^*$
$P_0 = \Delta_A(r_0)$
$P_0^* = \Delta_{A^t}(r_0^*)$
For $i = 0, 1, 2, \ldots$ *until convergence* ***Do***
$W_i = (P_i^*)^t A P_i$
$y_i = W_i^{-1} (P_i^*)^t r_i$
$x_{i+1} = x_i + P_i y_i$
$r_{i+1} = r_i - A P_i y_i$
$r_{i+1}^* = r_i^* - A^t P_i^* y_i$
$Q_{i+1} = \Delta_A(r_{i+1})$
$Q_{i+1}^* = \Delta_{A^t}(r_{i+1}^*)$
$B_{i+1} = -(W_i)^{-1} (P_i^*)^t A Q_{i+1}$
$P_{i+1} = Q_{i+1} + P_i B_{i+1}$
$P_{i+1}^* = Q_{i+1}^* + P_i^* B_{i+1}$
EndFor

Remark: Since matrices $\mathbf{N}$ and $\mathbf{K}$ are not positive definite in general, Theorem 3.1 cannot be used to assure the convergence of the s-Biconjugate Gradient. In practice, we expect convergence to occur in similar conditions to the usual Biconjugate Gradient method.

5 Conclusions and future work

In this work, a s-step variant of the general orthogonalization algorithm which generalizes conjugate gradient methods has been presented. The s-step variants of known iterative methods are derived as particular cases (some of which converging for every nonsingular matrix) and two are unpublished to our knowledge. It has been verified

 ISBN 13: 978-84-613-5510-5

that the convergence of these methods is supported in their s-step variants, by proving some prerequisite lemmas and convergence and error estimate theorems.

The performance gains of parallel implementations of the s-steps methods have been shown in some of the cited references, alongside the numerical results presented. The implementation on parallel computers and an exhaustive numerical analysis of those and other methods is at present under study by the authors.

Acknowledgements

This work was partially supported by Xunta de Galicia (Project PGIDIT06RMA23501PR-2) and Spain Government (Project TIN2004-07797-C02-01).

References

[1] J.A. Alvarez-Dios, J.C. Cabaleiro and G. Casal, *A s-step variant of the double orthogonal series algorithm*, Numerical Mathematics and Advanced Applications. ENUMATH 2005, Springer-Verlag. (2006) 937–944.

[2] O. Axelsson, *Conjugate gradient type methods for unsymmetric and inconsistent systems of equations*, Linear Algebra and its Applications. **29** (1980) 1–16.

[3] A.T. Chronopoulos, *A class of parallel iterative methods implemented on multiprocessors*, Ph.D. diss., University of Illinois at Urbana-Champaing, 1987.

[4] A.T. Chronopoulos and C.W. Gear, *On the efficient implementation of preconditioned s-step conjugate gradient methods on multiprocessors with memory hierarchy*, Parallel Computing. **11** (1989) 37–53.

[5] A.T. Chronopoulos and C.W. Gear, *s-step iterative methods for symmetric linear systems*, Journal of Computational and Applied Mathematics. **25** (1989) 153–168.

[6] A.T. Chronopoulos, *s-step iterative methods for (non)symmetric (in)definite linear systems*, Siam J. Numer. Anal. **28(6)** (1991) 1776–1789.

[7] A. T. Chronopoulos and A. Kucherov, *Block S-step Krylov Iterative Methods*, Numerical Linear Algebra with Applications. **17 (1)** (2010) 3–15.

[8] R. Fletcher, *Conjugate gradient methods for indefinite systems*, Lecture Notes in Mathematics, Springer-Verlag. **506** (1976) 73–89.

[9] P. Joly, *Présentation de synthèse des méthodes de gradient conjugué*, RAIRO. Modélisation Mathématique et Analyse Numérique. **20(4)** (1986) 639–665.

[10] C. Lawson, R. Hanson, D. Kincaid, F. Krogh, *Basic linear algebra subprograms for fortran usage*, ACM Trans. on Math. Soft. **5** (1979) 308–325.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

On an asymptotic sampling formula of Shannon type

Almudena Antuña[1], Juan L.G. Guirao[2] and Miguel A. López[1]

[1] *Departamento de Matemáticas, Universidad de Castilla-La Mancha, E.U. Politécnica de Cuenca, 16071-Cuenca (Castilla-La Mancha), Spain*

[2] *Departamaneto de Matemática Aplicada y Estadística, Universidad Politécnica de Cartagena, Hospital de Marina, 30203-Cartagena (Región de Murcia), Spain*

emails: almudena.antuna@uclm.es, juan.garcia@upct.es, mangel.lopez@uclm.es

Abstract

The aim of the present contribution is to state an asymptotic property $\mathcal{P}$ of type Shannon's sampling theorem, based on normalized cardinal sines, and keeping constant the sampling frequency of a not necessarily band–limited signal. It generalizes in the limit the results stated by Marvasti et al. [7] and Agud et al. [1]. We show that $\mathcal{P}$ is fulfilled for any constant signal working for every given sampling frequency. Moreover, we conjecture that Gaussian maps of the form $e^{-\lambda t^2}$, $\lambda \in \mathbb{R}^+$, hold $\mathcal{P}$. We support this conjecture by proving the equality given by $\mathcal{P}$ for the three first coefficients of the power series representation of $e^{-\lambda t^2}$.

Key words: Band–limited signal, Shannon's sampling theorem, Signal theory
MSC 2000: 41A60, 41A46; Secondary 41A30, 41A45, 41A58, 42C10.

1 Introduction and statement of the main results

A central result of the signal theory in engineering is the well–known Shannon–Whittaker–Kotel'nikov's theorem (see for instance [9] or [11]) working for band–limited maps of $L^2(\mathbb{R})$ (i.e., for Paley–Wiener signals), and based on the normalized cardinal sinus map $\mathrm{sinc}(t)$ defined by

$$\mathrm{sinc}(t) = \begin{cases} 1 & \text{if } t = 0, \\ \frac{\sin(\pi t)}{\pi t} & \text{if } t \neq 0. \end{cases}$$

Another philosopher's stone of the signal processing theory is the Middleton's sampling theorem for band step functions (see [8]). This result was one of the first modifications of the classic Sampling theorem (see [10]) which only works for band–limited maps. After this starting point many different extensions and generalizations of this theorem appeared in the literature trying to obtain approximations of non band–limited signals (see for instance [2] or [4]). Good surveys on these extensions are [3] or [11].

 ISBN 13: 978-84-613-5510-5

In this paper we follow the spirit of the previous results in the sense of trying to obtain approximations of non band–limited signals by using band–limited ones by increasing the band size. But our approach is completely different to the previous ones in the sense that we keep constant the sampling frequency generalizing in the limit the results of Marvasti et al. [7] and Agud et al. [1] .

In this setting, we state the following asymptotic property of type sampling Shannon's theorem where the convergence is considered in the Cauchy's principal value for the series and pointwise for the limit.

Property 1 *Let $f : \mathbb{R} \to \mathbb{R}$ be a map and $\tau \in \mathbb{R}^+$. We say that f holds the property $\mathcal{P}$ for τ if*

$$f(t) = \lim_{n\to\infty} \left(\sum_{k\in\mathbb{Z}} f^{\frac{1}{n}} \left(\frac{k}{\tau} \right) \operatorname{sinc}(\tau t - k) \right)^n . \tag{1}$$

The statement of the main results is:

Theorem 1 *Every constant signal holds property $\mathcal{P}$ for every given $\tau \in \mathbb{R}^+$.*

Conjecture 1 *The Gaussian maps, i.e. maps of the form $e^{-\lambda t^2}$, $\lambda \in \mathbb{R}^+$ hold property $\mathcal{P}$ for every given $\tau \in \mathbb{R}^+$.*

To support our feeling on the truth of the Conjecture 1 we prove, without loss of generality for $\lambda = 1$, that the Gaussian map e^{-t^2} holds expression (1) for the three first coefficients of the power series representation of e^{-t^2}. Note that since the Gaussian map is analytical, for proving formula (1) is enough to show the equality between the coefficients of the power series representation of the Gaussian map and the coefficients of the series stated in the second member of (1) after proving the analitycity of the second member of (1). The statement of our result is the following:

Theorem 2 *Let e^{-t^2} be a Gaussian map. Then the three first coefficients of the power series representation of e^{-t^2} are equal to the three first ones of the second member of expression* (1).

The paper is divided into three sections. In Section 2 we present the ideas and results that have inspired us to formulate property $\mathcal{P}$ and Conjecture 1. Section 3 is devoted to prove Theorem 1 and in Section 4 is proved Theorem 2.

2 On the property $\mathcal{P}$ and Conjecture 1

We state as a property $\mathcal{P}$ an approximation in the limit, through potentials of band–limited maps of the original signal, based on [1] and [7].

In [1] is proven that given a sequence $\{s_k\}_{k\in\mathbb{Z}} \in l^{2/n}(\mathbb{Z})$, $B > 0$, $\tau \geq 2B$ and n odd, there exist exactly n band–limited signals $\{x_r\}$ with bandwidth equal to B such that

 ISBN 13: 978-84-613-5510-5

$x_r^n\left(\frac{k}{\tau}\right) = s_k$. Moreover, is shown that $x_r = e_r x_0$, where $\{e_r\}_{r=0}^{n-1}$ are the roots of unity of order n and $x_0(t) = \sum_{k\in\mathbb{Z}} s_k^{1/n}\mathrm{sinc}(2Bt-k)$.

From this is directly deduced that if we consider an odd number n and a band–limited signal f with bandwidth $\widetilde{B}$ such that the sequence of coefficients $\left\{f\left(\frac{k}{\tau}\right); k\in\mathbb{Z}\right\}$ with $\tau \geq \frac{2\widetilde{B}}{n}$ holds the properties stated in [1], then the signal admits a recomposition of Shannon type in the form

$$f(t) = \left(\sum_{k\in\mathbb{Z}} f^{\frac{1}{n}}\left(\frac{k}{\tau}\right)\mathrm{sinc}(\tau t-k)\right)^n, \tag{2}$$

where clearly the sampling frequency can be choosen bigger than the Nyquist one.

Our aim is to provide a method for approximating non band–limited signal by band–limited ones and keeping the frequency of the sampling constant. And our idea is to take limits in (2) obtaining an equality of the form

$$f(t) = \lim_{n\to\infty}\left(\sum_{k\in\mathbb{Z}} f^{\frac{1}{n}}\left(\frac{k}{\tau}\right)\mathrm{sinc}(\tau t-k)\right)^n,$$

expressed as a property $\mathcal{P}$.

In Section 3 we prove that property $\mathcal{P}$ is held by any constant map for every $\tau\in\mathbb{R}^+$. Thus, the universe of non–trivial signals which hold the conjecture is nonempty (note that $f(t)\equiv 0$ holds $\mathcal{P}$). Our feeling is that there are a big number of representative signals in engineering processes which satisfy property $\mathcal{P}$.

We state as Conjecture 1 to prove that any signal of Gaussian type holds the statement. Note that the Gaussian map, which is mathematically important in itself, plays an important role in the signal theory because the Gaussian map is the unique function which reachs the minimum of the product of the temporal and frecuential width. This minimum is given by the Uncertainty Principle, see [6]. We believe in the working of Conjecture 1 and we support it through Theorem 2 where we show the equality between the three first coefficients of the power series representation of the Gaussian map and property $\mathcal{P}$. For proving completely the conjecture, by the analyticity of the Gaussian map, is enough to prove that expression $\lim_{n\to\infty}\left(\sum_{k\in\mathbb{Z}} e^{-\frac{k^2}{n\tau^2}}\mathrm{sinc}(\tau t-k)\right)^n$ defines an analytical map and to show that the equality works for the rest of coefficients.

3 Proof of Theorem 1

The following lemma will play a key role in the proof of Theorem 1.

Lemma 3 $\sum_{k\in\mathbb{Z}}\mathrm{sinc}(z-k) = 1$ *for every* $z\in\mathbb{C}$.

 ISBN 13: 978-84-613-5510-5

Proof. First of all we shall show that the result works for every $t \in \mathbb{R}$. Indeed, if $t \in \mathbb{Z}$, the result is straight because of

$$\sum_{k\in\mathbb{Z}} \operatorname{sinc}(t-k) = 1 + \sum_{\substack{k\in\mathbb{Z}\\ k\neq t}} \operatorname{sinc}(t-k) = 1 + 0 = 1.$$

Therefore, from now on we assume that $t \in \mathbb{R} \setminus \mathbb{Z}$. Taking simetric terms in the series we obtain

$$\begin{aligned} \sum_{k\in\mathbb{Z}} \operatorname{sinc}(t-k) &= \frac{\sin(\pi t)}{\pi t} + \sum_{k\in\mathbb{N}} \left(\frac{\sin\left(\pi(t-k)\right)}{\pi(t-k)} + \frac{\sin(\pi(t+k))}{\pi(t+k)} \right) \\ &= \frac{\sin(\pi t)}{\pi t} + \frac{2t\sin(\pi t)}{\pi} \sum_{k\in\mathbb{N}} \frac{(-1)^{k+1}}{k^2-t^2}. \end{aligned} \tag{3}$$

On the other hand, for a given $t \in \mathbb{R} \setminus \mathbb{Z}$ is known that

$$\frac{t\pi}{\sin(t\pi)} = 1 + 2t^2 \sum_{k\in\mathbb{N}} \frac{(-1)^k}{t^2-k^2},$$

and therefore

$$\sum_{k\in\mathbb{N}} \frac{(-1)^{k+1}}{k^2-t^2} = \frac{-1}{2t^2} + \frac{\pi}{2t\sin(\pi t)}. \tag{4}$$

Finally, replacing (4) in expression (3) the proof is over for every real number t.

The prove of the result for complex numbers is a consequence of the use of the Analytic Prologation Principle. For applying it, is enough to prove that the series $\sum_{k\in\mathbb{Z}} \operatorname{sinc}(z-k)$ is an analytic function. Indeed, by (3) the series can be written in the form

$$\sum_{k\in\mathbb{Z}} \operatorname{sinc}(z-k) = \frac{\sin(\pi z)}{\pi z} + \frac{2z\sin(\pi z)}{\pi} \sum_{k\in\mathbb{N}} \frac{(-1)^{k+1}}{k^2-z^2}.$$

Obviously, the first term of the previous sum is an analytic map. For proving the analyticity of the second term of the sum we shall prove that the series $\sum_{k\in\mathbb{N}} \frac{(-1)^{k+1}}{k^2-z^2}$ uniformly converges on every compact set $L \subset \mathbb{C}\backslash\mathbb{N}$. In fact, let $s = \max\{|z| : z \in L\}$ and k_0 be such that $k_0 > 2s$, then for every $k \geq k_0$ is $|z| < \frac{k}{2}$ for every $z \in L$. Therefore,

$$\left| \frac{(-1)^{k+1}}{k^2-z^2} \right| \leq \frac{4}{3k^2},$$

which guarantees the uniformly convergency of the series in L and the proof is over. ∎

Remark 4 *We underline that the fact of the series $\sum_{k\in\mathbb{Z}} sinc(z-k)$ defines an analytic function is a direct consequence of the application of the Uniform Convergence Principle for cardinal Series, see [5, pag. 70] or [11, pag. 22] for a more up-to-date reference. We present a direct approach in the proof of Lemma 3 for completness of the arguments.*

 ISBN 13: 978-84-613-5510-5

Proof of Theorem 1. Let $f(t) = C$ be a constant signal and $\tau \in \mathbb{R}^+$. By Lemma 3 we have

$$\begin{aligned}
&\lim_{n\to\infty}\left(\sum_{k\in\mathbb{Z}} f^{\frac{1}{n}}\left(\frac{k}{\tau}\right)\operatorname{sinc}(t\tau-k)\right)^n \\
&= \lim_{n\to\infty}\left(C^{\frac{1}{n}}\sum_{k\in\mathbb{Z}}\operatorname{sinc}(t\tau-k)\right)^n \\
&= \lim_{n\to\infty} C\left(\sum_{k\in\mathbb{Z}}\operatorname{sinc}(t\tau-k)\right)^n = \lim_{n\to\infty} C = C.
\end{aligned}$$

Thus, is shown that f holds property $\mathcal{P}$ ending the proof. ∎

4 Proof of Theorem 2

In the sequel we denote by J a set of consecutive natural numbers in the form $\{0, 1, 2, ...\}$ which eventually can be $\mathbb{N} \cup \{0\}$. By $\#(J)$ we denote the cardinal of the set J and we assume the arithmetic of the infinity (i.e., $\forall k \in \mathbb{N}, \infty \pm k = \infty$), therefore by m_J we denote $\#(J) - 1$.

Given a sequence $\alpha = \{\alpha_n\}_{n\in I}$ of real numbers, by $d(\alpha)$ we denote the *diameter* of the sequence α, i.e., $d(\alpha) = \sup_{1\le n<\#(J)}\{|\alpha_n - \alpha_{n-1}|\}$. As usual by $[\cdot]$ we denote the integer part.

Lemma 5 *Let $\gamma = \{\gamma_n\}_{n\in J}$ be an increasing bounded sequence of real numbers holding the following conditions:*

i) $a = \gamma_0 < \gamma_1 < \ldots < \gamma_{n-1} < \gamma_n < \ldots < b = \sup_{n\in J}\{\gamma_n\}$,

ii) $\{\gamma_n - \gamma_{n-1}\}_{n\in J\setminus\{0\}}$ *is monotonic.*

Let $f : [a, b] \to \mathbb{R}$ be a continuous map of constant sign on $[a, b]$, eventually f can be equal to zero. Then for every sequence $\beta = \{\beta_n\}_{n\in J\setminus\{0\}}$ such that $\beta_k \in [\gamma_{k-1}, \gamma_k]$ and for every $\varepsilon > 0$ there exists $\delta > 0$ such that if $d(\gamma) < \delta$ then

$$\left|\sum_{k=1}^{M} f(\beta_{2k})(\gamma_{2k} - \gamma_{2k-1}) - \frac{1}{2}\int_a^b f(x)\,dx\right| < \varepsilon \tag{5}$$

and

$$\left|\sum_{k=0}^{L} f(\beta_{2k+1})(\gamma_{2k+1} - \gamma_{2k}) - \frac{1}{2}\int_a^b f(x)dx\right| < \varepsilon, \tag{6}$$

where $M = \left[\frac{m_J}{2}\right]$ and $L = \left[\frac{m_J - 1}{2}\right]$.

 ISBN 13: 978-84-613-5510-5

Proof. For proving (5) we assume, without loss of generality, that $f \geq 0$ and $\{\gamma_n - \gamma_{n-1}\}_{n\in J\setminus\{0\}}$ is a decreasing sequence. We shall use the following notation

$$\begin{aligned}
T_o(\gamma,\beta) &= \sum_{k=0}^{L} f(\beta_{2k+1})(\gamma_{2k+1}-\gamma_{2k}),\\
T_e(\gamma,\beta) &= \sum_{k=1}^{M} f(\beta_{2k})(\gamma_{2k}-\gamma_{2k-1}),\\
S_{le}(\gamma) &= \sum_{k=1}^{M} f(\gamma_{2k})(\gamma_{2k}-\gamma_{2k-1}),\\
S_{re}(\gamma) &= \sum_{k=0}^{L} f(\gamma_{2k})(\gamma_{2k+1}-\gamma_{2k}),\\
S_{ro}(\gamma) &= \sum_{k=1}^{M} f(\gamma_{2k-1})(\gamma_{2k}-\gamma_{2k-1}),\\
S_{lo}(\gamma) &= \sum_{k=0}^{L} f(\gamma_{2k+1})(\gamma_{2k+1}-\gamma_{2k}).
\end{aligned}$$

For a given $\varepsilon > 0$, since $T_o + T_e$ is a Riemann sum of f on $[a,b]$, there exists $\delta_0 > 0$ such that if $d(\gamma) < \delta_0$, then

$$\left| T_o + T_e - \int_a^b f(x)dx \right| < \varepsilon. \tag{7}$$

Taking $\varepsilon_1 = \dfrac{\varepsilon}{3(b-a)} > 0$, since the map f is uniformly continuous on the interval $[a,b]$, then there exists $\delta_1 > 0$ such that if $|\beta_{2k} - \gamma_{2k}| < \delta_1$ then $|f(\beta_{2k}) - f(\gamma_{2k})| < \varepsilon_1$ and consequently if $d(\gamma) < \delta_1$

$$\begin{aligned}
|T_e - S_{le}| &= \left| \sum_{k=1}^{M} (f(\beta_{2k}) - f(\gamma_{2k}))(\gamma_{2k} - \gamma_{2k-1}) \right| \\
&< \varepsilon_1 \sum_{k=1}^{M} |\gamma_{2k} - \gamma_{2k-1}| < \varepsilon_1 (b-a) = \frac{\varepsilon}{3}.
\end{aligned} \tag{8}$$

Proceeding in a similar way

$$|T_o - S_{re}| < \frac{\varepsilon}{3}, \quad |T_e - S_{ro}| < \frac{\varepsilon}{3} \quad \text{and} \quad |T_o - S_{lo}| < \frac{\varepsilon}{3}. \tag{9}$$

Now, it is easily deduced that

$$f(\gamma_{2k})(\gamma_{2k+1} - \gamma_{2k}) \leq f(\gamma_{2k})(\gamma_{2k} - \gamma_{2k-1}),$$

and

$$S_{re} - S_{le} \leq f(\gamma_0)(\gamma_1 - \gamma_0).$$

 ISBN 13: 978-84-613-5510-5

So, taking $\delta_2 = \min\left\{\delta_1, \dfrac{\varepsilon}{3f(\gamma_0)}\right\}$ if $f(\gamma_0) \neq 0$ and $\delta_2 = \delta_1$ in other case, if $d(\gamma) < \delta_2$ then

$$S_{re} - S_{le} < f(\gamma_0)\,\delta_2 < \frac{\varepsilon}{3}.$$

Using the previous inequality, (8) and (9) we have that

$$T_o - T_e = (T_o - S_{re}) + (S_{le} - T_e) + (S_{re} - S_{le}) < \varepsilon. \tag{10}$$

On the other hand, it is clear that

$$S_{ro} - S_{lo} \leq 0 \tag{11}$$

and so, using (9) and (11)

$$T_e - T_o = (T_e - S_{ro}) + (S_{lo} - T_o) + (S_{ro} - S_{lo}) < \frac{2\varepsilon}{3} < \varepsilon.$$

From here and (10), if $d(\gamma) < \delta_2$,

$$|T_e - T_0| < \varepsilon.$$

So, taking $\delta = \min\{\delta_0, \delta_2\}$ and using the previous inequality and (7), if $d(\gamma) < \delta$ then

$$\begin{aligned}\left|T_e(\gamma) - \frac{1}{2}\int_a^b f(x)dx\right| \;&\leq\; \frac{1}{2}\,|T_e(\gamma) - T_o(\gamma)| \\ &\quad + \frac{1}{2}\left|T_e(\gamma) + T_o(\gamma) - \int_a^b f(x)dx\right| < \varepsilon,\end{aligned}$$

which is just (5) as we want to show.

The proof of (6) follows in an analogous way. ∎

Lemma 6 *Let $x \in \mathbb{R}^+$, $k \in \mathbb{N}$ and $l_k(x) = \dfrac{1 - e^{-k^2 x}}{k^2 x}$. Then for every k is $\lim\limits_{x \to 0^+} (l_k(x) - l_{k+1}(x)) = 0$ uniformly in k.*

Proof. Note that for every $x \in \mathbb{R}^+$ and every $k \in \mathbb{N}$, $l_k(x)$ is decreasing in k.

We fixed $x \in (0, 1)$. For a given $\varepsilon > 0$ there exist $C > 0$ holding $l_k(x) < \frac{\varepsilon}{2}$ for any k such that $k^2 x \geq C$ and consequently

$$l_k(x) - l_{k+1}(x) < \varepsilon. \tag{12}$$

On the other hand, using the power series representation of the exponential function

 ISBN 13: 978-84-613-5510-5

and the Newton's binomial,

$$\begin{aligned} l_k(x) - l_{k+1}(x) &= \sum_{p=0}^{\infty} \frac{(-1)^p}{(p+1)!} \left((k^2x)^p - ((k+1)^2x)^p\right) \\ &= \sqrt{x} \sum_{p=0}^{\infty} \frac{(-1)^{p+1}}{(p+1)!} \sum_{q=0}^{2p-1} \binom{2p}{q} \left(k\sqrt{x}\right)^q x^{p-\frac{q+1}{2}} \\ &\leq \sqrt{x} \sum_{p=0}^{\infty} \frac{1}{(p+1)!} \left(1+k\sqrt{x}\right)^{2p} \\ &= \sqrt{x}\, \frac{e^{\left(1+k\sqrt{x}\right)^2} - 1}{\left(1+k\sqrt{x}\right)^2} \end{aligned}$$

and if $k^2x < C$ we have the following inequality

$$l_k(x) - l_{k+1}(x) < \sqrt{x} \left(e^{(1+\sqrt{C})^2} - 1\right).$$

Since $\lim_{x\to 0} \sqrt{x}\left(e^{(1+\sqrt{C})^2} - 1\right) = 0$, using the last inequality and (12) the proof is over. ∎

The following proposition will play a key role in the proof of Theorem 2.

Proposition 7 *Let $x \in \mathbb{R}^+$ and $L(x) = \sum_{k\in\mathbb{N}} (-1)^{k+1} l_k(x)$. Then is held*

$$L(x) \leq \frac{\pi}{2} \qquad \text{and} \qquad \lim_{x\to 0^+} L(x) = \frac{1}{2}.$$

Proof. We consider the functions $\alpha_k(x) = \operatorname{arctg} l_k(x)$ on $[0, \frac{\pi}{2}]$. Let $x \in \mathbb{R}^+$ fixed. We note that $\alpha_k(x)$ is a decreasing sequence on k. It is easily deduced that using the Intermediate Value Theorem

$$L(x) = \sum_{k\in\mathbb{N}} (l_{2k-1}(x) - l_{2k}(x)) = \sum_{k\in\mathbb{N}} (\operatorname{tg}\alpha_{2k-1}(x) - \operatorname{tg}\alpha_{2k}(x)),$$

and therefore

$$L(x) = \sum_{k\in\mathbb{N}} \frac{\alpha_{2k-1}(x) - \alpha_{2k}(x)}{\cos^2 \beta_{2k-1}}, \tag{13}$$

for suitable $\beta_{2k-1} \in (\alpha_{2k}(x), \alpha_{2k-1}(x))$.
Note that $\alpha_k(x) \in (0, \frac{\pi}{4}]$ for all $k \in \mathbb{N}$ and consequently $0 < \beta_k < \frac{\pi}{4}$. Thus,

$$L(x) \leq 2 \sum_{k\in\mathbb{N}} (\alpha_{2k-1}(x) - \alpha_{2k}(x)) \leq 2 \sum_{k\in\mathbb{N}} (\alpha_k(x) - \alpha_{k+1}(x)) = 2\,\alpha_1(x) \leq \frac{\pi}{2}.$$

 ISBN 13: 978-84-613-5510-5

Since $\int_0^{\frac{\pi}{4}} \frac{dt}{\cos^2 t} = 1$, then

$$\begin{aligned} \left| L(x) - \frac{1}{2} \right| &= \left| L(x) - \frac{1}{2} \int_0^{\frac{\pi}{4}} \frac{dt}{\cos^2 t} \right| \\ &\leq \left| L(x) - \frac{1}{2} \int_0^{\alpha_1(x)} \frac{dt}{\cos^2(t)} \right| + \left| \frac{1}{2} \int_{\alpha_1(x)}^{\frac{\pi}{4}} \frac{dt}{\cos^2(t)} \right|. \end{aligned} \tag{14}$$

On the one hand, given $\varepsilon > 0$ clearly there exists $\delta_0 > 0$ such that if $x < \delta_0$ then

$$\int_{\alpha_1(x)}^{\frac{\pi}{4}} \frac{dt}{\cos^2 t} < \varepsilon. \tag{15}$$

On the other hand, using Lemma 5 for $f(x) = \frac{1}{\cos^2(x)}$, $\gamma = \{-\alpha_{r+1}\}_{r=0}^{\infty}$, $\widetilde{\beta} = \left\{\widetilde{\beta}_r\right\}_{r=1}^{\infty}$ such that $\widetilde{\beta}_k \in [\gamma_{k-1}, \gamma_k]$, $\widetilde{\beta}_{2k-1} = -\beta_{2k-1}$, $a = -\alpha_1(x)$ and $b = 0$, there exists $\delta_1 > 0$ such that if $d(\gamma) < \delta_1$ then

$$\left| L(x) - \frac{1}{2} \int_0^{\alpha_1(x)} \frac{dx}{\cos^2(x)} \right| < \frac{\varepsilon}{2}. \tag{16}$$

Since $\arctan(\cdot)$ is a continuous map on $(0, \frac{\pi}{4}]$, for δ_1 by Lemma 6 there exists $\delta_2 > 0$ such that if $x < \delta_2$ then $d(\alpha) < \delta_1$.

Therefore, taking $x < \delta = \min\{\delta_0, \delta_2\}$, and replacing (15) and (16) in (14) we obtain

$$\left| L(x) - \frac{1}{2} \right| < \varepsilon,$$

finishing the proof. ∎

Proof of Theorem 2. The aim of the proof is to show that the limit of the three first nonzero coefficients of the power series representations of

$$\left(\sum_{k \in \mathbb{Z}} e^{\frac{-k^2}{n\tau^2}} \operatorname{sinc}(\tau t - k) \right)^n$$

and e^{-t^2} are equal for every $t \in \mathbb{R}$ and $\tau > 0$ given. Indeed, for every $m \in \mathbb{N} \cup \{0\}$ and $n \in \mathbb{N}$ we fix the following notation

$$B_m^\tau = \frac{(-1)^m (\pi\tau)^{2m}}{(2m+1)!}; \tag{17}$$

$$C_{m,n}^\tau = \begin{cases} \frac{1}{2}, & \text{if } m = 0, \\ \tau^{2m} \sum_{k \in \mathbb{N}} \frac{(-1)^{k+1}}{k^{2m}} e^{\frac{-k^2}{n\tau^2}}, & \text{if } m \geq 1; \end{cases} \tag{18}$$

$$D_{m,n}^\tau = \sum_{p=0}^{m} B_p^\tau C_{m-p,n}^\tau. \tag{19}$$

 ISBN 13: 978-84-613-5510-5

Let

$$g(t,n)=\sum_{k\in\mathbb{Z}} e^{\frac{-k^2}{n\tau^2}}\,\mathrm{sinc}(\tau t-k).$$

Note that by the analitycity is enough to consider pointwise convergence for all $t\in(0,\frac{1}{\tau})$. Now, using expressions (17), (18), (19) and the power series of the sine function, the map $g(t,n)$ can be written in the form

$$\begin{aligned}
g(t,n) &= \mathrm{sinc}(\tau t)+\frac{2\tau t\sin\pi\tau t}{\pi}\sum_{k\in\mathbb{N}}\frac{(-1)^k}{\tau^2t^2-k^2}\,e^{\frac{-k^2}{n\tau^2}}\\
&= \frac{\sin\pi\tau t}{\pi}\left(\frac{1}{\tau t}+2\tau t\sum_{m=0}^{\infty}t^{2m}\left(\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}\tau^{2m}}{k^{2(m+1)}}\,e^{\frac{-k^2}{n\tau^2}}\right)\right)\\
&= \frac{2t}{\pi\tau}\left(\sum_{m=0}^{\infty}\frac{(-1)^m}{(2m+1)!}(\pi\tau t)^{2m+1}\right)\left(\sum_{m=-1}^{\infty}C^{\tau}_{m+1,n}\,t^{2m}\right)\\
&= 2\left(\sum_{m=0}^{\infty}B^{\tau}_m\,t^{2m}\right)\left(\sum_{m=0}^{\infty}C^{\tau}_{m,n}\,t^{2m}\right)=2\sum_{m=0}^{\infty}D^{\tau}_{m,n}\,t^{2m}.
\end{aligned}$$

and therefore

$$(g(t,n))^n=2^n\left(\sum_{m=0}^{\infty}D^{\tau}_{m,n}\,t^{2m}\right)^n=2^n\sum_{m=0}^{\infty}E^{\tau}_{m,n}\,t^{2m}.$$

For $m=0$ it is clear that $E^{\tau}_{0,n}=(D^{\tau}_{0,n})^n=(B^{\tau}_0\,C^{\tau}_{0,n})^n=\frac{1}{2^n}$ and hence

$$\lim_{n\to\infty}2^nE^{\tau}_{0,n}=1.$$

For $m=1$ is

$$\begin{aligned}
E^{\tau}_{1,n} &= n(D^{\tau}_{0,n})^{n-1}\,D^{\tau}_{1,n}=n\,\left(B^{\tau}_0C^{\tau}_{0,n}\right)^{n-1}\,\left(B^{\tau}_0C^{\tau}_{1,n}+B^{\tau}_1C^{\tau}_{0,n}\right)\\
&= \frac{n}{2^{n-1}}\left(C^{\tau}_{1,n}+\frac{B^{\tau}_1}{2}\right)\\
&= \frac{n\tau^2}{2^{n-1}}\left(\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^2}\,e^{\frac{-k^2}{n\tau^2}}-\frac{\pi^2}{12}\right).
\end{aligned}$$

So, using $\displaystyle\sum_{k=1}^{\infty}\frac{(-1)^{k+1}}{k^2}=\frac{\pi^2}{12}$,

$$2^nE^{\tau}_{1,n}=-2\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{\frac{k^2}{n\tau^2}}=-2L\left(\frac{1}{n\tau^2}\right)$$

where $L(\cdot)$ is introduced in Proposition 7 and now by such result we obtain

 ISBN 13: 978-84-613-5510-5

$$\lim_{n\to\infty} 2^n E^\tau_{1,n} = -1. \tag{20}$$

For $m = 2$ it follows that

$$\begin{aligned} E^\tau_{2,n} &= n(D^\tau_{0,n})^{n-1} D^\tau_{2,n} + \frac{n(n-1)}{2}(D^\tau_{0,n})^{n-2}(D^\tau_{1,n})^2 \\ &= n(B^\tau_0 C^\tau_{0,n})^{n-1}\left(B^\tau_2 C^\tau_{0,n} + B^\tau_1 C^\tau_{1,n} + B^\tau_0 C^\tau_{2,n}\right) \\ &\quad + \frac{n(n-1)}{2}(B^\tau_0 C^\tau_{0,n})^{n-2}\left(B^\tau_0 C^\tau_{1,n} + B^\tau_1 C^\tau_{0,n}\right)^2 \\ &= \frac{n}{2^{n-1}}\left(\frac{B^\tau_2}{2} + B^\tau_1 C^\tau_{1,n} + C^\tau_{2,n}\right) + \frac{n(n-1)}{2^{n-1}}\left(C^\tau_{1,n} + \frac{B^\tau_1}{2}\right)^2 \end{aligned}$$

Therefore

$$2^n E^\tau_{2,n} = F^\tau_n + G^\tau_n \tag{21}$$

where

$$\begin{aligned} F^\tau_n &= 2n\left(\frac{B^\tau_2}{2} + B^\tau_1 C^\tau_{1,n} + C^\tau_{2,n}\right), \\ G^\tau_n &= 2n(n-1)\left(C^\tau_{1,n} + \frac{B^\tau_1}{2}\right)^2. \end{aligned}$$

We will take the limit in each part separately. Since

$$G^\tau_n = \frac{n-1}{2n}\left(2^n E^\tau_{1,n}\right)^2,$$

from (20) we obtain

$$\lim_{n\to\infty} G^\tau_n = \frac{1}{2} \tag{22}$$

To determine the limit of F^τ_n, replacing each B^τ_j and $C^\tau_{j,n}$ by (17) and (18), we get

$$\begin{aligned} F^\tau_n &= 2n\tau^4\pi^4\left(\frac{1}{2.5!} - \frac{1}{3!\pi^2}\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^2}e^{\frac{-k^2}{n\tau^2}} + \frac{1}{\pi^4}\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^4}e^{\frac{-k^2}{n\tau^2}}\right) \\ &= 2n\tau^4\pi^4\left(\frac{1}{2.5!} + \frac{1}{3!\pi^2}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{k^2}\right. \\ &\quad \left. + \frac{1}{\pi^4}\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^4}e^{\frac{-k^2}{n\tau^2}} - \frac{1}{3!\pi^2}\sum_{k=1}^{\infty}\frac{(-1)^{k+1}}{k^2}\right) \end{aligned}$$

 ISBN 13: 978-84-613-5510-5

Using again $\sum_{k=1}^{\infty}\frac{(-1)^{k+1}}{k^2}=\frac{\pi^2}{12}$ and applying $\sum_{k=1}^{\infty}\frac{(-1)^{k+1}}{k^4}=\frac{7\pi^4}{720}$, the above expression becomes

$$\begin{aligned}
F_n^{\tau} &= 2n\tau^4\pi^4\left(\frac{-7}{720}+\frac{1}{3!\pi^2}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{k^2}+\frac{1}{\pi^4}\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^4}e^{\frac{-k^2}{n\tau^2}}\right)\\
&= 2n\tau^4\pi^4\left(\frac{1}{3!\pi^2}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{k^2}-\frac{1}{\pi^4}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{k^4}\right)\\
&= \frac{\tau^2\pi^2}{3}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{\frac{k^2}{n\tau^2}}-2\tau^4\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{\frac{k^4}{n}}\\
&= \frac{\tau^2\pi^2}{3}L\left(\frac{1}{n\tau^2}\right)-2\tau^2\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{\frac{k^4}{n\tau^2}}.
\end{aligned}$$

Therefore, since $\lim_{n\to\infty}\sum_{k\in\mathbb{N}}(-1)^{k+1}\frac{1-e^{\frac{-k^2}{n\tau^2}}}{\frac{k^4}{n\tau^2}}=\sum_{k\in\mathbb{N}}\frac{(-1)^{k+1}}{k^2}=\frac{\pi^2}{12}$, using Proposition 7 is

$$\lim_{n\to\infty}F_n^{\tau}=\frac{\tau^2\pi^2}{3}\cdot\frac{1}{2}-2\tau^2\cdot\frac{\pi^2}{12}=0.$$

So, from here and (22), taking limits in (21) we get

$$\lim_{n\to\infty}2^nE_{2,n}^{\tau}=0+\frac{1}{2}=\frac{1}{2}.$$

Note that from the results obtained for $m=0,1,2$ is stated that the limit of the three first nonzero coefficients of the power series representations of

$$\left(\sum_{k\in\mathbb{Z}}e^{\frac{-k^2}{n\tau^2}}\operatorname{sinc}(\tau t-k)\right)^n$$

are equal to $\frac{(-1)^m}{m!}$, coefficients of the power series representation of e^{-t^2}, ending the proof. ∎

Acknowlegments

This work has been partially supported by MCI (Ministerio de Ciencia e Innovación) and FEDER (Fondo Europeo Desarrollo Regional), grant number MTM2008–03679/MTM and Fundación Séneca de la Región de Murcia, grant number 08667/PI/08. The authors want to thank to Professors J. Garay and R.G. Catalán for their value comments formulated in a constructive spirit.

 ISBN 13: 978-84-613-5510-5

References

[1] L. Agud and R.G. Catalán, *New Shannon's sampling recomposition*, Rev. Acad. Ciencias Zaragoza, **56** (2001), 45–48.

[2] P.L. Butzer, S. Ries and R.L. Stens, *Approximation of continuous and discontinuous functions by generalized sampling series,* Jour. Appr. Theo., **50**, (1987), 25-39.

[3] P.L. Butzer and R.L. Stens, *Sampling theory for not necessarily band-limited functions: a historical overview,* SIAM review, **34**, 4, (1992), 40-53.

[4] J.A. Gubner, *A new series for approximating Voight functions,* Jour. Phys. A: Math., **27**, (1994), L745-L749.

[5] J.R. Higgings, *Five short stories about the cardinal series*, Bull. Amer. Math. Soc., **12** (1985), 45–89.

[6] H.J. Landau, H.O Pollak *Prolate spheroidal wave functions,* Fourier analysis and uncertainly, Bell. Sys. Tech. Jour., **40**, 1, (1961), 65–84.

[7] F. Marvasti and A.K. Jain, *Zero crossing bandwidth compression, and restoration of nonlinearly distorted band–limited signals*, J. Optical Soc. Amer., **3** (1986), 651–654.

[8] D. Middleton, *An introduction to statistical communication theory,* McGraw-Hill, New York, 1960.

[9] C.E. Shannon, *Communication in the presence of noise,* Proc. IRE, **137**, (1949), 10–21.

[10] E.T. Whittaker, *On the functions which are represented by the expansions of the interpolation theory,* Proc. Roy. Soc. Edinburgh, **35**, (1915), 181-194.

[11] A.I. Zayed, *Advances in Shannon's Sampling Theory,* Ed. CRC Press, (1993).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Java/C++/Fortran cross-platform performance and consistency issues for large simulation codes

Satya Baboolal[1]

[1] *School of Computer Science, University of KwaZulu-Natal, Durban, South Africa.*

emails: baboolals@ukzn.ac.za

Abstract

The suitability of different programming languages for scientific computing has been the subject of many debates and studies. Java is a popular multi-purpose programming language and it is not surprising that many recent studies have been conducted on its performances in various application areas, in particular, in the scientific computing arena. In this instance, one aspect of Java involves its documented unpredictable behaviour in respect of floating point computations. Since the Java virtual machine's floating point behaviour has been designed to adhere very strictly to IEEE standard for binary floating point systems with the intention of achieving portability and consistency, we find that this restriction can lead to inconsistent behaviours across various platforms. Moreover, such behaviour is not confined to Java alone.

The other aspect which has interested the scientific computing community has been the speed benchmarks for different languages employed in this arena.

We attempt to gain useful insight into these two performance aspects by porting into Java, C/C++ and Fortran77, a code employing a two-dimensional high-resolution finite-difference scheme for simulating nonlinear wave propagation a multi-fluid plasma under electromagnetic fields and comparing the relative performances of these implementations on 32-bit and 64-bit PC platforms.

Key words: Java floating point, IEEE floating point, numerical simulation

1 Introduction

In this report we are concerned, in the first instance, in examining the behaviour of Java as candidate of the IEEE 754 standard (1985) and revised IEEE 854 (1987, 2008) standard [1, 2, 3] for binary floating point representation and computation so we briefly review some salient features of this standard:

Here, floating-point numbers in general are normalized before storage and can be represented in one of the forms [1, 2]:

 ISBN 13: 978-84-613-5510-5

single precision (32 bits):

$$\text{s E F} \equiv \text{s eeee eeee fff ffff ffff ffff ffff ffff}$$

with value $(-1)^s \times 1.F \times 2^{E-127}$

double precision (64 bits):

$$\text{s E F} \equiv \text{s eee eeee eeee ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff}$$

with value $(-1)^s \times 1.F \times 2^{E-1023}$

where the sign bit (bit 31) is $s = 0$ for a positive number and $s = -1$ for a negative number, the binary fraction F=fff...ff occupies bits 0-22, the exponent E=eeeeeeee, which is biased (by 127) to avoid storing negative values occupies bits 23-30 so that $\text{E}_{max} = 11111111_2 = 255$ and $\text{E}_{min} = 0$ in the first case. These extreme E values are reserved for special conditions, so that the allowable range of E is 00000001 ... 11111110 (1... 254) giving an exponent range E-127 = -126... +127. Similar considerations apply to the double precision case. In addition the IEEE standard includes provisions for extended-precision numbers, for handling denormal numbers, i.e. numbers obtained from calculations whose results fall in the range between the smallest non-zero number that can be represented in the floating point system and zero (on the positive side), for infinities and for not-a-numbers (NANs). Moreover, the default behaviour specified by the IEEE is to allow the computations to continue in spite of the occurrence(s) of these special values, by masking such particular exceptions. This may or may not be desirable in every situation.

Most present day processors have floating point units (FPUs) which implement the IEEE standard by default. In particular, on Intel x86 processors [4, 5], floating point behaviour can be controlled by setting its floating point control word register (FPCSR), a special 16-bit register. Then the current control word in the FPCSR, will control the arithmetic accuracy employed in calculating intermediate results in the 80-bit FPU general registers, control how rounding is done when register contents are manipulated and stored in memory, and how denormals are handled, amongst other effects. Corresponding to the FPCSR, is another floating point status-word register, which the CPU sets, depending on the result of the last executed floating point instruction.

The x87 instruction set includes the FLDCW (load control word) and FSTCW (store control word) machine instructions for manipulating the FPCSR. For example, the instruction

```
FLDCW   639
```

will set it to the hexadecimal value 027F, allowing for 53-bit mantissa precision and rounding to the nearest floating point number, and the instruction

```
FLDCW   895
```

will set it to the hexadecimal value 037F, allowing for 64-bit mantissa precision and

 ISBN 13: 978-84-613-5510-5

rounding to the nearest floating point number.

In addition, more convenient functions may be available in some operating systems with particular language bindings.

On more recent Intel processors, additional floating point operations [4, 5] can carried out on separate processing units within the CPU as streaming pipelined instructions (MMX, SSE, SSE2, SSE3, ...) which can be enabled by compiler switches. These machine instructions can be controlled by a separate combined control-status word register, MXCSR. This is a 32-bit register which allows one to set flags to handle denormal processing, rounding and so on. In this paper we shall deal essentially with the former x87 instructions, since they are the more accurate, as the latter employ the default IEEE register precision, a crucial aspect for our codes.

2 Code for simulating nonlinear waves in electromagnetic plasmas

We handle performance issues here by multi-language implementations of a code for simulating electromagnetic shock-like structures in a plasma fluid consisting of singly charged ions an electrons subject to the electromagnetic field. A complete description of the model and algorithm used is given elsewhere[6]. It suffices to mention that it employs a two-dimensional high-resolution Riemann-solver-free central difference scheme on staggered grids to numerically solve model equations cast into the first-order PDE hyperbolic system form [6]:

$$\frac{\partial U}{\partial t}+\frac{\partial F(U)}{\partial x}+\frac{\partial G(U)}{\partial y}=S(U) \tag{1}$$

In the above $U(x,y,z,t)$ is the unknown (m-dimensional) vector, $F(U)$ is the x-flux vector, $G(U)$ is the y-flux vector and $S(U)$ is a source vector function, with x and y the only two spatial coordinates considered (for no variation in the z direction) and t is the time coordinate.

To numerically integrate this system, a uniform rectangular grid with spacings Δx and Δy in the respective X and Y directions is used to obtain [6],

$$\begin{aligned}\bar{U}^{n+1}_{j+\frac{1}{2},k+\frac{1}{2}} &= \frac{1}{4}\left[\bar{U}^{n}_{jk}+\bar{U}^{n}_{j,k+1}+\bar{U}^{n}_{j+1,k}+\bar{U}^{n}_{j+1,k+1}\right]\\ &+\frac{1}{16}\left[U^{n}_{xj,k}-U^{n}_{xj+1,k}+U^{n}_{xj,k+1}-U^{n}_{xj+1,k+1}\right]\\ &+\frac{1}{16}\left[U^{n}_{yj,k}-U^{n}_{yj,k+1}+U^{n}_{yj+1,k}-U^{n}_{yj+1,k+1}\right]\\ &-\frac{\Delta t}{2\Delta x}\left[F^{n+\frac{1}{2}}_{j+1,k}-F^{n+\frac{1}{2}}_{j,k}+F^{n+\frac{1}{2}}_{j+1,k+1}-F^{n+\frac{1}{2}}_{j,k+1}\right]\\ &-\frac{\Delta t}{2\Delta y}\left[G^{n+\frac{1}{2}}_{j,k+1}-G^{n+\frac{1}{2}}_{j,k}+G^{n+\frac{1}{2}}_{j+1,k+1}-G^{n+\frac{1}{2}}_{j+1,k}\right]\\ &+\frac{\Delta t}{4}\left[S^{n+\frac{1}{2}}_{j+1,k+1}+S^{n+\frac{1}{2}}_{j+1,k}+S^{n+\frac{1}{2}}_{j,k+1}+S^{n+\frac{1}{2}}_{j,k}\right].\end{aligned} \tag{2}$$

 ISBN 13: 978-84-613-5510-5

This scheme advances the cell average vectors $\bar{U}^n_{j,k}$ where j, k are the spatial discretization indices and n is the time level index, with time spacing Δt. It is used in conjunction with the derivative array aproximations (U_x and U_y) and suitable boundary conditions. For more details consult [6].

3 Multi-platform implementations

We have coded the complete time-evolutionary algorithm into Fortran 77, C/C++ and Java and linked the DISLIN package [7] to include real modelling-time graphics. Double precision words (64-bits) were used for all the floating-point variables throughout. The codes were run under Microsoft Windows XP 32-bit (Xp32) on various PC configurations (Pentium IV desktop, Notebook with Intel Centrino CPU, Intel Pentium Core 2 Quad). Additional tests on the Core 2 Quad machine were done with the Salford (32-bit) and Gfortran (32- and 64-bit) compilers under the Microsoft windows Xp 64-bit (Xp64) and under SUSE Linux 64-bit (SuSe64) operating systems. By selecting a somewhat modest-sized problem (XY grid size of 201×201 discretization points) our findings for various compiler suites, with their default settings are tabulated in summary form below:-

Compiler/OS	**Stable?**	**Consistent?**
Salford FTN95 (Fortran 77)/Xp32	Yes, over long times	—
Mingw g77/ Xp32	Yes, over long times	Yes, with above
Gfortran 32-bit/ Xp32	Yes, over long times	Yes, with above
Mingw C (GNU c)/Xp32	Yes, over long times	Yes, with above
MS Visual C++ Express 8/Xp32	Emits NANs over long times	No: results meaningless
Sun Java 5/ Xp32	Emits NANs over long times	No: results meaningless
Salford FTN95 (Fortran 77)/Xp64	Emits NANs over long times	No: results meaningless
Gfortran 32-bit/ Xp64	Emits NANs over long times	No: results meaningless
Gfortran 64-bit/ Xp64	Emits NANs over long times	No: results meaningless
Gfortran 64-bit/ SuSe64	Yes, over long times	Yes, with above 1st four

We observe that stable and consistent behaviour is obtained in the first four and last cases in the table with Fig. 1 giving a typical result of the evolution of the electron

 ISBN 13: 978-84-613-5510-5

fluid density as a shock wave. Such computations prevail over several thousands time steps, whilst remaining stable and giving meaningful physical results.

For all other cases the results obtained are unstable and quite meaningless. Fig. 2 depicts such results for codes written in Visual C++ and Sun Java 5.

These codes emit NANs or meaningless results, of no physical significance. Upon investigation of the discrepancies we find that both MS Visual C++ and Sun Java adhere to the IEEE standard strictly: in particular, the most significant feature of departure is that the FPU register precision is 64-bits (53-bit mantissa) whilst all the first four compilers employ 80-bits (64-bit mantissa). Thus intermediate results in the FPU registers can suffer from a significant loss in precision even before being rounded to 64-bits double-precision words for storage in memory. On large simulations codes such as here, such errors can accumulate and swamp the computations over time. Another IEEE feature is that floating point over/under flows are masked to allow computations to continue, regardless of the occurrence of NANs at intermediate steps, which is the situation observed in the last two implementations.

Furthermore, tests were performed on an Intel Core 2 Quad machine by installing 64-bit operating systems and compilers. We find here that, under MS Windows Xp-64 all the compilers including the Gfortran 64-bit compiler fail to achieve consistency and stability. However, under SUSE Linux-64 we can again achieve consistency and stability with the Gfortran 64-bit compiler by setting the appropriate command-line switches, as indicated in the next section.

4 Code fixes for numerical consistency

4.1 Fortran/C++/Java 5 MS Windows Xp-32 implementations

In the case of Salford FTN95,GCC(MinGW C,C++) and Visual C++ 2008 we can set the floating point control word either with inline assembly code or by using WIN32 functions, such as the call to the system function

```
_control87(_PC_64,MCW_PC);
```

or

```
_control87(0x0008001F,0xFFFFFFFF);
```

These details are available in other works [4].

However, for Sun Java we cannot employ these means. We have thus created a C++ DLL which may be called from Java by employing the following process:

1. Create a (Mingw) C++ project e.g cpplibDLL
2. Set the project options to WIN32 DLL The default output file name will be cpplibDLL.dll
3. Include the standard header jni.h (Sun Microsystems') and your cpplibDLL.h

 ISBN 13: 978-84-613-5510-5

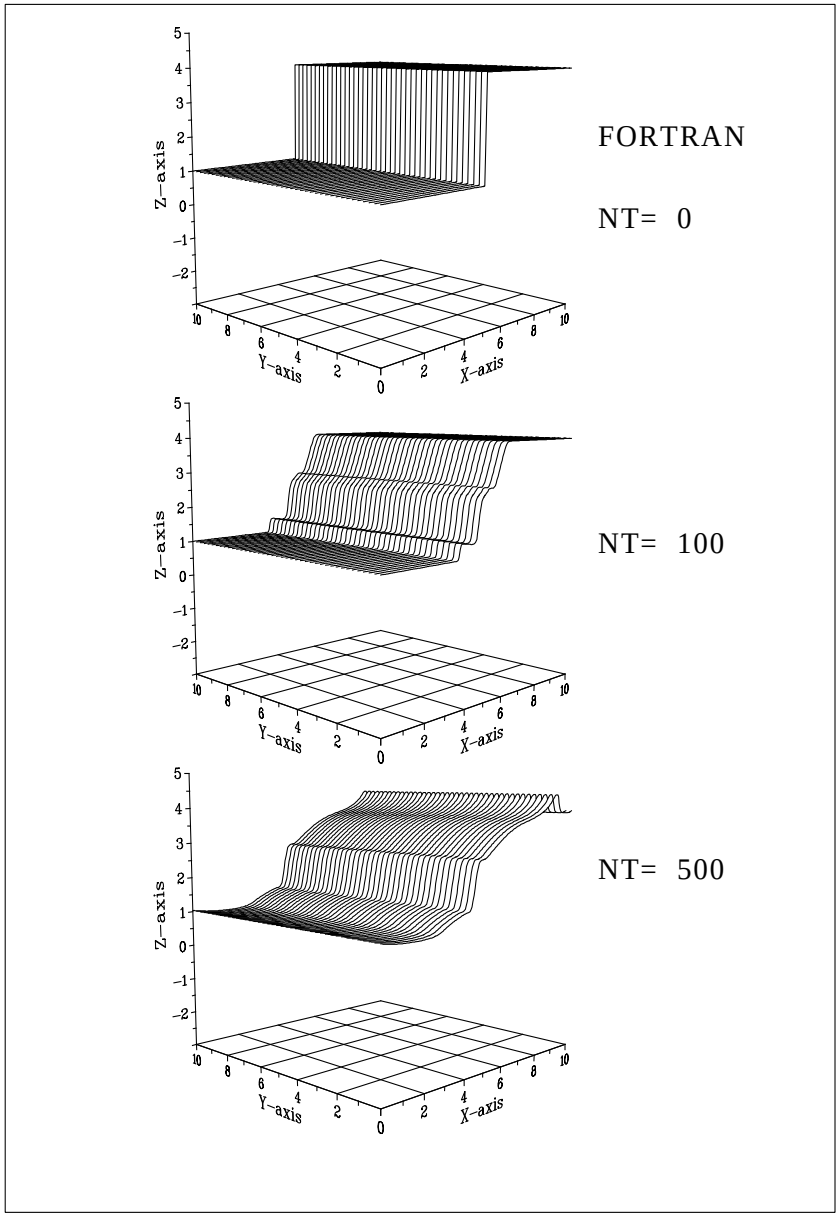


Figure 1: Salford FTN95/GNU C computed electron density shock structures

 ISBN 13: 978-84-613-5510-5

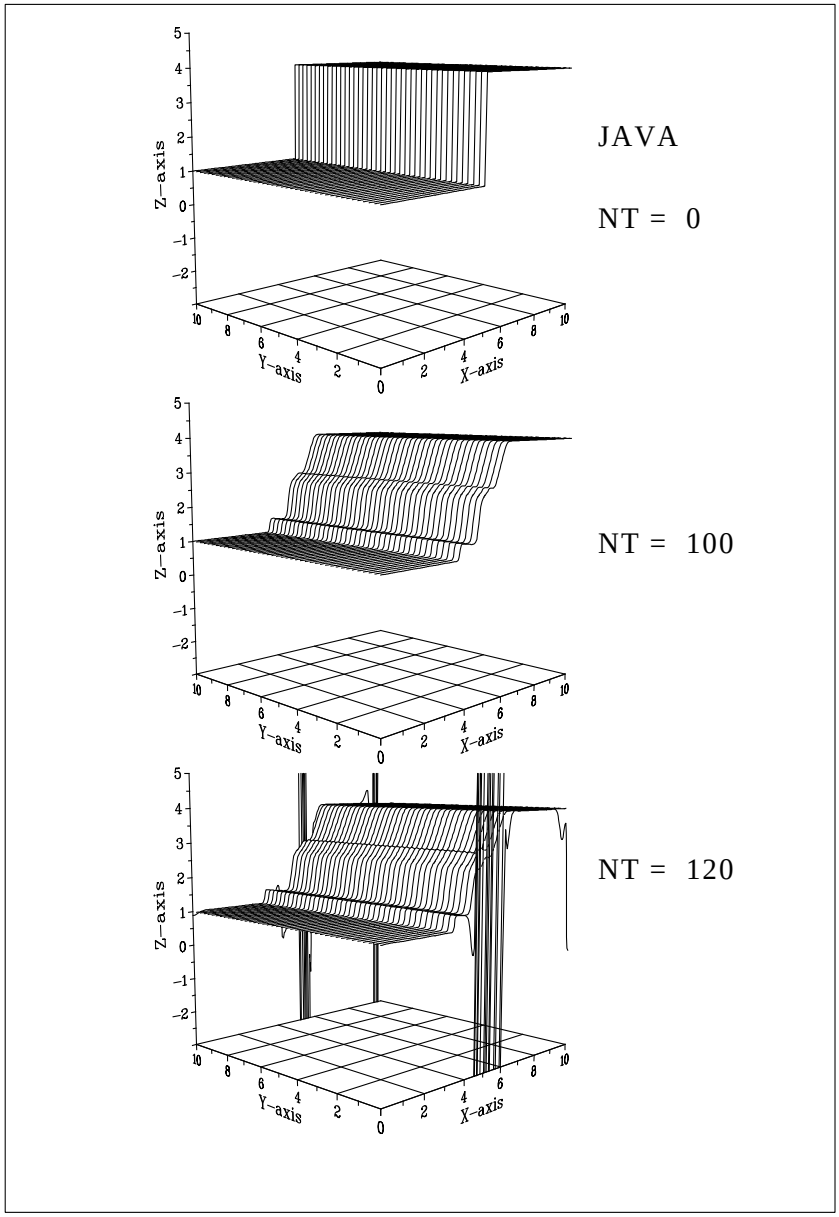


Figure 2: Sun Java 5/VC++ computed electron density shock structures

 ISBN 13: 978-84-613-5510-5

4. Compile/build this into cpplibDLL.dll in your working directory
5. Create the Java invoking program javacppprog.java
6. Compile and run the Java program.

Then the following code implementations may be used:

```
//cpplibDLLcpp file looks like:
//---------------------------
#include "cpplibDLL.h"
#include "jni.h"
#include <stdio.h>
#include <stdlib.h>
#include <float.h>

//To create an export function for
//the export library:
JNIEXPORT void JNICALL
Java_javacppprog_controlWord(
           JNIEnv *env, jobject obj)
{
  printf("Hello from C++ function
          controlWord()!\n");
  printf( "Original: 0x%.8x\n",
           _control87(0,0));
//Set FPU control word:
  _control87(0x0008001F,0xFFFFFFFF);
//system("PAUSE");
  printf( "New: 0x%.8x\n",
          _control87(0,0));
  return;
}

//etc...for other functions
//-------------------------

//cpplibDLL.h header file
//-----------------------
/*
The header file may be generated by:
javah  javacppprog.java
or you can edit the above-
mentioned header such as:
*/

#include "c:\...\jni.h"
#ifndef _Included_cpplibDLL
#define _Included_cpplibDLL

#ifdef __cplusplus
```

 ISBN 13: 978-84-613-5510-5

```
extern "C" {
#endif

JNIEXPORT void JNICALL
Java_javacppprog_controlWord(
              JNIEnv *, jobject);
JNIEXPORT void JNICALL
Java_javacppprog_controlWord2(
               JNIEnv *, jobject);
#ifdef __cplusplus
}
#endif
#endif
//end of header file cpplibDLL.h
//------------------------------

//javacppprog.java
//----------------
import java.io.IOException;
import java.text.NumberFormat;
 .........
public class javacppprog{
 static final int NxPts = 201,
                  NyPts = 201,
                  MEQS = 16,
                     ...,
 ..............
 ..............
 public static native void controlWord();
 ....
 static{System.loadLibrary("cpplibDLL");}
 public static void main(String [] args)
   {
    int NGRIDA, NDH, NT, ........
    ......................
    //Set FPU control word:
      controlWord();
      .....
    //Main Time/While loop
    //---------------------
      while (NT < NSTEPS)
       {
        controlWord();
        t=t+dt;
        .........
       }//end main while loop
       .....
    } //main
} //end of class javacppprog
```

 ISBN 13: 978-84-613-5510-5

Now, setting the FPU control word as in the Fortran/Mingw value we obtain stable and meaningful results in agreement with those Fig. 1. The results from the similarly amended Visual C++ program are the same. We note that in both these cases we found it necessary to repeatedly apply this setting in the time-loop of the code since returns from certain functions can cause the compiler or Windows default setting to be resumed. The computational cost of this process is negligible in comparison to the time-loop traversal time for any realistic simulation.

4.2 Sun Java 6 MS Windows Xp-32 implementations

In the case of Java 6 we find that attempts to set the control word as in the above section fail, with the results as in Fig. 2 again. In fact Java 6 masks this low-level function call. At this stage we can only speculate that this must be a deliberate design feature in order to protect the Java working environment.

4.3 MS Windows Xp-64 implementations

We find, in these cases that even setting the control word as above does not fix the problem. In fact, the Xp-64 operating system overrides the settings, so that the IEEE standard is maintained. For instance, the command line to invoke the Gfortran 64-bit compiler from an install-directory's bin sub-directory for compilation of some **prog.for** and force the generation of code for the the x87 FPU with 80-bit register precision is:

```
..\bin\x86_64-pc-mingw32-gfortran -c
        -mpc80 -mfpmath=387 prog.for
```

But, even this imposition is ignored under Windows Xp-64, since the latter switches off access to the MMX and x87 units. It appears that [5] that Microsoft would migrate to developing code employing floating point computations only for the SSE2 (and successor) architectures, thus dropping the x87 legacy architecture, which might explain our observations.

4.4 SUSE Linux-64 implementations

Here although the default floating point behaviour is IEEE compliant, the use of the x87 FPU can be enforced by the command-line:

```
gfortran -c -mpc80 -mfpmath=387 prog.for
```

5 Speed tests

To gain some insight into the relative performance of Java with respect to speed, we have conducted some benchmark tests on the Java-, C- and Fortran-code versions and compare the CPU (user single code/thread + operating system support code) execution

 ISBN 13: 978-84-613-5510-5

times of the three versions of our 2D code. The results are summarized in the table for a 500 time step run in each case:

Compiler/OS	**CPU time (secs)**
Salford FTN95(Fortran 77)/MS Win Xp32	264.141
Mingw C(GNU C)/MS Win Xp32	155.968
Sun Java6/MS Win Xp32	200.031

Thus it is clear that Java is competitive with C and moreover, performs better than Fortran when the latter is runnning with no code optimizations.

6 Conclusion

By means of multi-language codings of an algorithm for the numerical integration of hyperbolic systems for 3-D electromagnetic plasma fluid equations allowing wave propagation in two dimensions [6] we have conducted floating point consistency tests and speed benchmarks. Our findings indicate that for languages, such as Salford Fortran 95 and GNU Mingw Fortran and C/C++, that employ 80-bit FPU accuracy in the FPU registers for intermediate calculations, overriding the IEEE standard of 64-bit accuracy, the results over long time runs are stable and consistent, in agreement with previous results. When, we employ compilers (Sun Java 5/6 and MS Visual C++ Express 8) that strictly adhere to the IEEE standard in this respect, the results obtained are not consistent with the previous case, and more so, degenerate into meaningless computations which just evolve NANs.

However, for 32-bit operating systems and compilers (MS Visual C++ Express 8, Java 5), when in the latter we set the processor FPU control word to override the IEEE accuracy to 80-bits we obtain stable and consistent results as before. Exceptionally, in the case of Java 6 (and later) attempts to set the control word fail.

Under Windows Xp-64, all compilers (32- and 64-bit) are restricted by the operating system to the IEEE default, and hence the codes fail. Nevertheless, under SUSE Linux-64, we find Gfortran to exhibit stable behaviour when the compiler is invoked to generate x87 code.

As far as speed benchmarks are concerned, our tests on 32-bit compilers indicate the C++ performs best, followed (surprisingly) by Java 6 and then Salford FTN95. Thus Java may be seen to be competitive for large simulation codes, were it not for inconsistencies in floating point behaviour.

References

[1] W. Kahan, *Lecture Notes on the Status of the IEEE Standard 754 for Binary Floating-Point Arithmetic*, Dept. of Elect. Eng. and Computer Science, University of California, Berkeley, (1996) 1–29.

[2] D. Goldberg, *What Every Computer Scientist Should Know About Floating-Point Arithmetic*, ACM Comput. Surv. **23** (1991) 5-48.

 ISBN 13: 978-84-613-5510-5

[3] W. Kahan and J.D. Darcy, *How Java's Floating-Point Hurts Everyone Everywhere*, Dept. of Elect. Eng. and Computer Science, University of California, Berkeley, (1998, 2004), 1-81. Available online: http://www.cs.berkeley.edu/ wkahan/JAVAhurt.pdf
(Originally presented at ACM 1998 Workshop on Java for High-Performance Network Computing, Stanford University, March 1, 1998).

[4] Shawn D. Casey, *x87 and SSE floating point assists in IA-32: Flush-to-zero (FTZ) and Denormals-are-zero (DAZ)*, Intel Corp., (2007, 2008). Available online: http://software.intel.com/en-us/articles/x87-and-sse-floating-point-assists-in-ia-32-flush-to-zero-ftz-and-denormals-are-zero-daz/

[5] Microsoft Corp., *Run-time library reference*, http://msdn.microsoft.com/en-us/library (Dec. 2009).

[6] S. Baboolal, *High-resolution numerical simulation of 2D nonlinear wave structures in electromagnetic fluids with absorbing boundary conditions*, J. Comput. Appl. Math. **234** (2010) 1710-1716.

[7] H. Michels, *Dislin Graphics Package: ver. 9.4*, Max Planck Instutute for Solar System Research, Katlenburg-Lindau, Germany. Available online: http://www.mps.mpg.de/dislin, November, 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Solving Large-Scale Linear Systems on Clusters using Secondary Storage

J. M. Badía[1], M. Castillo[1], J.I. Climente[2], M. Marqués[1], R. Mayo[1], J.L. Movilla[2], J. Planelles[2] and E. S. Quintana-Ortí[1]

[1] *Depto. de Ingeniería y Ciencia de Computadores, Universidad Jaume I, 12.071-Castellón (Spain)*

[2] *Depto. de Química Física y Analítica, Universidad Jaume I, 12.071-Castellón (Spain)*

emails: badia@icc.uji.es, castillo@icc.uji.es, climente@qfa.uji.es, mmarques@icc.uji.es, mayo@icc.uji.es, movilla@qfa.uji.es, planelle@qfa.uji.es, quintana@icc.uji.es

Abstract

We present a package, built upon libraries PLAPACK and POOCLAPACK, that facilitates the use of out-of-core and parallel techniques for the solution of large-scale dense linear systems to non-experienced programmers. The complexity of dealing with secondary memory storage necessary to accommodate huge data structures and the use of parallel distributed-memory message-passing architectures are thus made transparent to users. The techniques described in this work allow the solution of this type of systems on a wide range of computer facilities, from commodity workstations to complex high performance computer systems. Our package is tested on the solution of a real problem arising in condensed matter physics on a cluster of commodity computers. The experimental results for systems with up to 100,000 equations illustrate the benefits of exploiting both process-level and thread-level parallelism as well as demonstrate that the use of secondary storage exerts a moderate impact on performance.

Key words: Linear systems, high performance computing, out-of-core algorithms, LU factorization.

1 Introduction

Large-scale dense linear algebra problems, involving matrices with hundreds of thousands of rows and columns, arise in boundary element methods for integral equations in electromagnetism and acoustics, radial function methods, estimation of Earth's gravitational field, molecular dynamic simulations, and quantum chemistry, among others [1, 2, 3, 7, 9]. When the data structures involved in these problems are too large

 ISBN 13: 978-84-613-5510-5

to fit in memory, the only solution is to rely on disk storage. Although such additional memory can be accessed via virtual memory, careful design of out-of-core (OOC) algorithms is generally required to attain high performance.

In this paper we extend the POOCLAPACK library [6] with the HDSS package, a user-friendly application programming interface (API) to build OOC dense linear algebra objects and to execute its OOC dense linear algebra routines. We believe our high-level object-oriented API can be of wide appeal to a majority of scientists and engineers, who need this class of environments to elaborate complex analyses, modeling, and simulations, and who have little or no experience with parallel programming and OOC techniques. As an additional contribution, our paper evaluates the performance of the API/POOCLAPACK duo on a linear system with up to $\mathcal{O}(100,000)$ equations arising in a condensed matter physics application. Both process-level or thread-level parallelism are exploited for the efficient solution of these systems on a cluster of commodity computers equipped with Intel multi-core technology and connected via a high-speed Infiniband switch.

The rest of paper is structured as follows. Section 2 exposes the key routines of the HDSS interface, illustrating their use by means of a simple example. In Section 3 we evaluate the codes on a problem provided by a group from our university that conducts research on . Finally, concluding remarks are summarized in the Section 4.

2 The HDSS Package

Our HDSS interface is based on PLAPACK [8] and its extension POOCLAPACK for OOC computations. PLAPACK (Parallel Linear Algebra Package) provides a collection of parallel routines for the solution of dense linear algebra operations such as dense linear systems, linear least-squares problems, and eigenvalue computations on message-passing architectures. PLAPACK features an a object-based orientation, abstracting the user from the layout of the data among the memory spaces of the nodes in a distributed-memory platform. POOCLAPACK (Parallel OOC LAPACK) offers additional flexibility to customize both the in-core and OOC algorithms. This in turn allows to code OOC algorithms in such a way that data I/O becomes straight-forward, reducing the porting effort and improving performance.

HDSS builds upon these two libraries, with the primary goal of providing a collection of routines to help non-expert programmers to develop efficient, parallel OOC codes by hiding details on the storage infrastructure and its use. Figure 1 shows the layering of the libraries employed in our work and its use from HDSS.

A secondary objective for HDSS is to assist the programmer of parallel dense linear algebra routines, by easing the extraction of a large fraction of the performance of current clusters. Moreover, our interface cannot only be used on parallel machines, but also efficiently profit from modern multi-core commodity computers. In pursue of this goal, the routines in the interface offer the possibility of exploiting parallelism at two different levels. At the bottom level, thread-level parallelism can be extracted by accessing multi-threaded implementation of BLAS (Basic Linear Algebra Subpro-

 ISBN 13: 978-84-613-5510-5

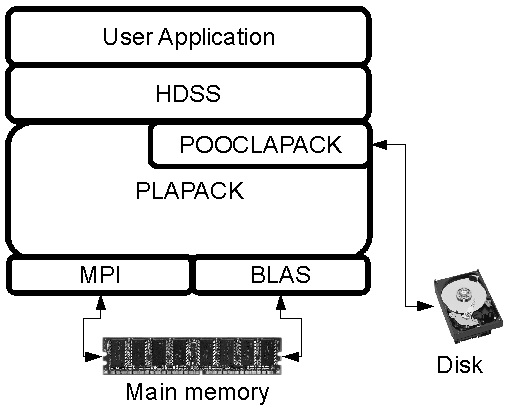


Figure 1: Architecture of parallel OOC dense linear algebra libraries.

grams). At the top level, the number of MPI processes and the data layout can be easily configured from the HDSS routines to optimize the use of task- (or process-) level parallelism.

HDSS exhibits an object-oriented style akin to those of PLAPACK and POOCLAPACK. This paradigm hides programming details which usually entail most of the errors during the development process (basically, the intricate indexing present in most traditional linear algebra codes and features related to the storage layout of the data matrices). HDSS employs objects to represent matrices and vectors, which are defined in HDSS using the `HDSS_Obj` datatype. These objects are implemented internally as a structure containing an OOC matrix consisting of a distributed `PLA_Obj` matrix with an attached file on each MPI process.

The usage of the routines in the HDSS package is illustrated with the excerpt of Fortran-90 code in Figure 2. The code in that figure can be executed sequentially or in parallel invoking the `mpirun` launcher. In the second case, routine `hdss_init_env` initializes the parallel environment and routine `hdss_finalize` terminates it. The code between the two previous routines can be viewed as a parallel region where all processes execute the same code by default. Following the MIMD (multiple instruction, multiple data) programming model, code executed by different processes can be customized using the process' unique identifier returned by routine `hdss_get_myid`. Several environment parameters can be extracted using the appropriate routines of the API. For example, routine `hdss_get_numprocs` returns the number of processes participating in the parallel region.

Once the environment is initialized and the execution of the parallel region has begun, the code in Figure 2 performs three main actions: First, it creates and initializes the objects containing the OOC matrices of the linear system $AX = B$, namely, `A_ooc` for the matrix containing the entries of $A \in \mathbb{R}^{ndimA \times ndimA}$ and `B_ooc` for those of matrix $B \in \mathbb{R}^{ndimA \times ndimB}$.

The matrices involved in the problem are so large that they may not fit into the

 ISBN 13: 978-84-613-5510-5

```
1   C     /* ********************************************************************** */
2   C     /* Initialization misc.: environment, process identifier, etc. */
3
4         call hdss_init_env()
5         call hdss_ooc_set_scratch_dir( '/state/partition1' )
6         call hdss_get_myid( me )
7         call hdss_get_slab( slab )
8         call hdss_get_numprocs( nprocs )
9
10  C     /* ********************************************************************** */
11  C     /* Store ndimA x ndimA matrix in OOC HDSS object A_ooc by slabs of columns */
12
13        allocate( R(ndimA, (slab/nprocs)+1)) )
14        call hdss_ooc_create_matrix( A_ooc, 'A', ndimA, ndimA )
15        do j = 1, ndimA, slab
16           col_begin = j
17           col_end   = min(ndimA, j+slab-1)
18           call hdss_get_mycolumns(col_begin, col_end, me_col_begin, me_col_end )
19
20  C        /* initialize slab of columns in-core buffer R */
21  C        /* ... */
22
23           call hdss_ooc_matrix_set_columns( R, A_ooc, col_begin, col_end )
24        enddo
25
26  C     /* ********************************************************************** */
27  C     /* Repeat the process for ndimA x ndimB matrix in OOC HDSS object B_ooc */
28  C     /* ... */
29
30  C     /* ********************************************************************** */
31  C     /* Solve system and store the solution in B_ooc */
32
33        call hdss_ooc_lu(A_ooc, ipiv_ooc)
34        call hdss_ooc_lu_solve(A_ooc, ipiv_ooc, B_ooc)
35
36  C     /* ********************************************************************** */
37  C     /* Retrieve solution by slabs from OOC B_ooc to in-core buffer R */
38        do j = 1, ndimB, slab
39           col_begin = j
40           col_end   = min(ndimB, j+slab-1)
41           call hdss_ooc_matrix_get_columns( B_ooc, R, col_begin, col_end )
42
43  C        /* Do something with the solution slab in in-core buffer R */
44  C        /* ... */
45        end do
46
47  C     /* ********************************************************************** */
48  C     /* Free memory and terminate environment */
49        deallocate( R )
50        call hdss_ooc_free_matrix( A_ooc )
51        call hdss_ooc_free_matrix( B_ooc )
52        call hdss_finalize()
```

Figure 2: Fragment of sample code that allocates and initializes two OOC matrices, A and B (lines 13–29), solves the linear system $AX = B$ overwriting the contents of B with the solution (lines 34–35), and retrieves the solution by blocks of columns (lines 39–46).

 ISBN 13: 978-84-613-5510-5

aggregated main memory of the cluster nodes. The solution is to rely on the aggregated secondary memory (i.e., the disks) of the system. Thus, the OOC library stores the elements on each process on a different file in the local disk the process is running on. Function `hdss_ooc_set_scratch_dir` indicates the directory where these files are stored for each process.

An additional problem of dealing with large matrices is that, during the initialization, one cannot store all their elements in the main memory of the processors; thus, this stage also needs to proceed by blocks. In our example, we perform this local manipulation by blocks of columns, or *slabs*, that are stored in an in-core buffer `R` on the main memory of each processor; see the loop in line 15. During each iteration of this loop, all processes invoke routine `hdss_ooc_matrix_set_columns` to initialize a slab containing columns `col_begin` to `col_end` of the matrix. The routine distributes the block of columns among the different processor following PLAPACK bidimensional data layout. To assist in this slab-wise (or panel-wise) generation of the matrices, routine `hdss_get_mycolumns` returns the indexes of the initial and final columns of each block of columns, `me_col_begin` and `me_col_end` respectively.

After the initialization is completed, the program solves the linear system. It first uses routine `hdss_ooc_lu` to compute the LU factorization of matrix A, returning the pivoting information in the HDSS object `ipiv_ooc`. It then solves the system invoking routine `hdss_ooc_lu_solve` which overwrites object `B_ooc` with the solution to the system.

Finally, once the system is solved, the solution is retrieved by slabs, using the in-core buffer `R` to store them, with the help of routine `hdss_ooc_matrix_get_columns`.

In summary, most routines in the HDSS API can be considered collective operations in the sense that they must be invoked by all processes to perform some collective action in parallel. For example, all processes perform the same invocation to the routines to create or fill the HDSS objects with data, to retrieve the data from the objects, or to perform the LU factorization or the solution of the linear system.

3 Experimental results

All experiments in this section were performed on a cluster composed of 4 nodes, each equipped with two Intel Quad Core E5520 processors operating at 2.27 GHz, with 36 GB of DDR3 main memory, and a single 250 GB SATA-HD at 7200 rpm. Peak performance for a single node of this system in single precision arithmetic is 145.28 Gflops (128×10^9 floating-point arithmetic operations per second). Thus, the global peak performance of the system is $4 \times 145.28 = 581.12$ Gflops. The interconnect is a QDR Infiniband high-performance switch. Multi-threaded implementations of BLAS and LAPACK are provided by MKL 10.2.

In the following, we report the performance of the OOC codes for the LU decomposition in single-precision arithmetic. (The theoretical cost of the LU decomposition of a dense matrix, in terms of floating-point arithmetic operations, is $2n^3/3$, and thus this is the operation that dominates the cost of the system solution procedure via Gaussian

 ISBN 13: 978-84-613-5510-5

elimination.) The data matrices appear in the modeling of the electrostatic properties of recently synthesized metal-semiconductor nanostructures [5, 4]. For realistic models, the size of the linear systems can be as large as $500,000 \times 500,000$. Single precision arithmetic provides enough accuracy for the application. With these constraints, storage of a $100,000 \times 100,000$ matrix (the coefficient of the linear system) requires more than 40 GB of memory, which does not fit into the main memory of a single node.

To check the correction of the solution, in our tests with small problems we computed the relative residual $\mathcal{R}(X^*) = \|AX^* - B\|_F / \|X^*\|_F$, where X^* stands for the computed solution of the system $AX = B$. In all those tests, we obtained $\mathcal{R}(X^*) \leq 10^{-6}$.

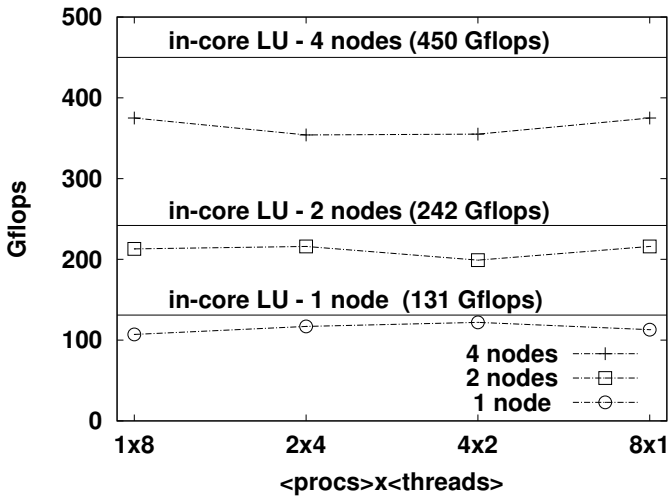
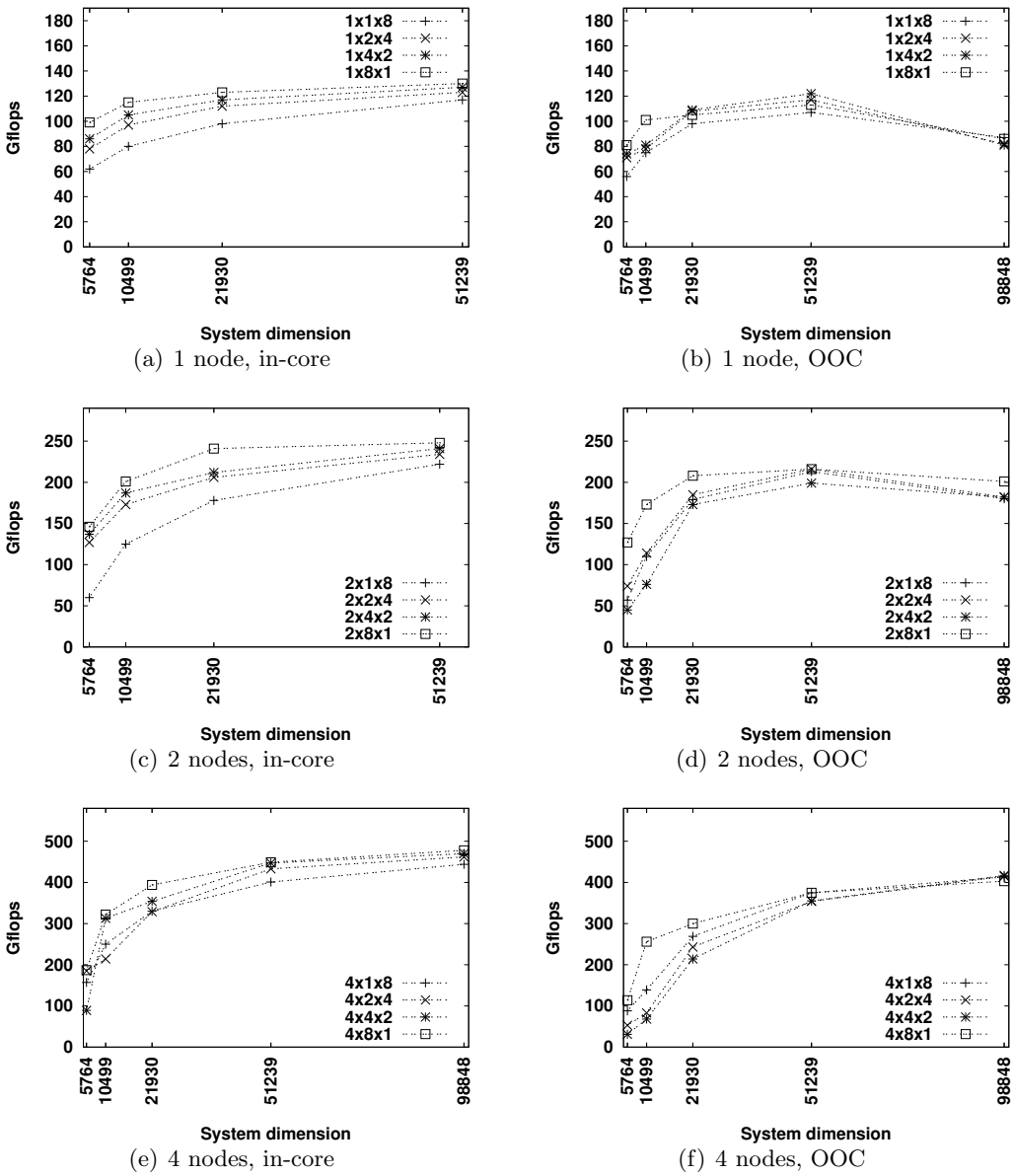


Figure 3: Performance of OOC codes in the solution of a linear system of dimension 51,239.

We first evaluate the in-core and OOC versions in order to asses the impact of the disk storage on performance. In Figure 3 we report the performance of the OOC codes as well as the maximum performance attained with the execution of the in-core codes on 1, 2 and 4 nodes for a system with 51,239 equations. For 1 and 2 nodes, the performance of the OOC codes is close to that of the in-core counterpart. If the number of nodes is increased up to 4, the difference becomes more significant: when four nodes are used to solve a problem of this dimension, the time needed to access the data on disk is comparable to computation time, which yields an important decrease in performance.

The plots in Figure 4 illustrate that, for the in-core codes, the exploitation of process-level parallelism ($1 \times 8 \times 1$) clearly outperforms that of thread-level parallelism ($1 \times 1 \times 8$). The behaviour of the OOC codes is similar.

 ISBN 13: 978-84-613-5510-5



(a) 1 node, in-core (b) 1 node, OOC

(c) 2 nodes, in-core (d) 2 nodes, OOC

(e) 4 nodes, in-core (f) 4 nodes, OOC

Figure 4: Performance on 1, 2 and 4 nodes of the in-core and OOC codes. Four different parallelization alternatives are shown in each plot which illustrate different combination of the nodes/processes/threads numbers (<nodes>x<process>x<threads>). For example, pure process-level parallelism is exploited by the tuples <nodes>x8x1 while pure thread-level parallelism is given by the combinations <nodes>x1x8.

 ISBN 13: 978-84-613-5510-5

4 Conclusions

In this paper we present a package that allows the solution of huge linear systems of equations on a wide variety of platforms, from commodity workstations to large clusters of computers. Our package is based on an OOC extension of PLAPACK library, namely, POOCLAPACK.

We have designed our HDSS API so that the codes in these libraries can be easily used by novel parallel programmers. The routines in the package involve a reduced number of parameters and hide all the aspects regarding the distribution and storage of the data and the parallel handling of the different tasks.

We have conducted an experimental analysis on a cluster equipped multi-core processors and a high-speed interconnect. The results show that the package allows to solve systems with up to hundreds of thousands of equations. We have exploited and combined two different levels of parallelism, process-level (using PLAPACK) and thread-level parallelism (using a multi-threaded implementation of the BLAS kernels). Our results show good parallel performances, that in the case of the in-core version of the routines are close to the peak of the processors: 478 Gflops on 32 cores. The performance of the OOC codes is more modest for relatively small systems, but as the dimension of the problem is increased, the performance gap between the two versions narrows.

Acknowledgements

Continuous support from CICYT project TIN2008-06570-C04-01, MCINN project CTQ2008-03344, UJI-Bancaixa project P1-1A2009-03 and FEDER are gratefully acknowledged.

References

[1] M. Baboulin. *Solving large dense linear least squares problems on parallel distributed computers. Application to the Earth's gravity field computation.* Ph.D. dissertation, INPT, March 2006. TH/PA/06/22.

[2] Po Geng, J. Tinsley Oden, and Robert van de Geijn. Massively parallel computation for acoustical scattering problems using boundary element methods. *Journal of Sound and Vibration*, 191(1):145–165, 1996.

[3] Brian C. Gunter. *Computational methods and processing strategies for estimating Earth's gravity field.* PhD thesis, The University of Texas at Austin, 2004.

[4] T. Mokari, E. Rothenberg, I. Popov, R. Costi, and U. Banin. Selective growth of metal tips onto semiconductor quantum rods and tetrapods. *Science*, 304:1787–1790, 2004.

 ISBN 13: 978-84-613-5510-5

[5] J.L. Movilla, J.I. Climente, and J. Planelles. Dielectric polarization in axially-symmetric nanostructures: A computational approach. *Comput. Phys. Comm.*, 181:92–98, 2010.

[6] Wesley C. Reiley and Robert A. van de Geijn. POOCLAPACK: Parallel Out-of-Core Linear Algebra Package. Technical Report CS-TR-99-33, Department of Computer Sciences, The University of Texas at Austin, Nov. 1999.

[7] N. Schafer, R. Serban, and D. Negrut. Implicit integration in molecular dynamics simulation. In *ASME International Mechanical Engineering Congress & Exposition*, 2008. (IMECE2008-66438).

[8] Robert A. van de Geijn. *Using PLAPACK: Parallel Linear Algebra Package*. The MIT Press, 1997.

[9] Y. Zhang, T. K. Sarkar, R. A. van de Geijn, and M. C. Taylor. Parallel MoM using higher order basis function and PLAPACK in-core and out-of-core solvers for challenging EM simulations. In *IEEE AP-S & USNC/URSI Symposium*, 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Erasure decoding for Gabidulin codes

Régis F. Babindamana[1] **and Cheikh T. Gueye**[1]

[1] *Departement de Mathmatiques Informatique, Facult des Sciences et Techniques, UNIVERSITE CHEIKH ANTA DIOP*

emails: `regisbab@ucad.sn`, `cheikht.gueye@ucad.edu.sn`

Abstract

We present a new approach of the decoding algorithm for Gabidulin Codes. In the same way as efficient erasure decoding for Generalized Reed Solomon codes by using the structure of the inverse of the VanderMonde matrices, we show that, the erasure decoding Gabidulin code can be seen as a computation of three matrice and an affine permutation , instead of computing an inverse from the generator or parity check matrix. This significantly reduces the decoding complexity compared to others algorithms
For r erasures, where $r = n - k$, the erasure algorithm decoding for $Gab_{n,k}(g)$ Gabidulin code compute the r symbols by simple multiplication of three matrices. That requires $r^2+r(k-1)$ Galois field multiplications, $r(r-1)+2rk$ field additions, $r^2 + r(k + 1)$ field negations and $r(k + 1)$ field inversions.

Key words: Gabidulin Codes, Generalized Reed solomon codes, Vandermonde matrix, Cauchy matrix

1 Introduction

The Gabidulin codes are introduced in [2]. These codes are maximum rank distance (MRD). They meet the best possible rank distance $d = n + 1 - k$, where k is the dimension of the code and n its lenght.
The Generalized Reed Solomon Codes are Maximum Distance Separable (MDS) for the Hamming distance. (i.e. $d' = n' - k' + 1$ where n' and k' are respectvely the lenght and the dimension of code).
Given an [n', k'] Generalized Reed Solomon Codes (GRS); his generator matrix $G(\alpha', v')$ can be written : $G(\alpha', v') = V(\alpha')D(v')$ where $V(\alpha')$ is a Vandermonde matrix and $D(v')$ a diagonal matrix.
In [1] we describe the existence of an affine permutation ψ that conserves the Hamming distance and transforms a Gabidulin code into a GRS code.

 ISBN 13: 978-84-613-5510-5

Let be a $Gab_{n,k}(g)$ Gabidulin code, H his parity check matrix, H is also a Gabidulin code of parameters (n, r). Then $\psi(H)$ is a GRS code. So, we can write $\psi(H) = V(\alpha)D(v)$. This transformation allows to use the structure of the inverse of the VanderMonde matrix in order to construct our algorithm.
This paper is organized as follows. First we recall the definitions known on diagonals, vandermonde and Cauchy matrices and we recall results on the GRS codes and the Gabidulin codes which take us in the section 2. In the section 3 we state mains results and we describe our erasure decoding algorithm. In section 4 we discuss the complexity of our decoding algorithm.

2 Preliminary

In this section we present the results known that we will use in order to prove our results.

2.1 Diagonal matrix

Definition 2.1 *Let be K a finite field. Given $v = (v_1, ..., v_n)$, where $(v_1, ..., v_n) \in K$. we define $D(v)$ to be the $n \times n$ diagonal matrix as*

$$D(v) = \begin{pmatrix} v_1 & 0 & . & . & . & 0 \\ 0 & v_2 & . & . & . & 0 \\ . & . & . & & & . \\ . & . & & . & & . \\ . & . & & & . & . \\ 0 & 0 & . & . & . & v_n \end{pmatrix}$$

2.2 VanderMonde Matrix

Definition 2.2 *Given k non-zero and distinct elements $\alpha = (\alpha_1, ..., \alpha_k)$ we define the $k \times k$ VanderMonde matrix as*

$$V(\alpha) = v(\alpha_1, ..., \alpha_k) = \begin{pmatrix} 1 & 1 & . & . & . & 1 \\ \alpha_1 & \alpha_2 & . & . & . & \alpha_k \\ \alpha_1^2 & \alpha_2^2 & . & & & \alpha_k^2 \\ . & . & & . & & . \\ . & . & & & . & . \\ \alpha_1^{k-1} & \alpha_2^{k-1} & . & . & . & \alpha_k^{k-1} \end{pmatrix}$$

2.2.1

Let us put $f_i(z) = \prod_{1 \leq t \leq k, i \neq t}(z - \alpha_t) = \Sigma_{1 \leq r \leq k} a_{ij} z^r$.
The inverse of the VanderMonde matrix is given by

$$(v(\alpha_1, ..., \alpha_k)^{-1})_{ij} = \frac{a_{ij}}{\prod_{1 \leq t \leq k, i \neq t}(\alpha_i - \alpha_t)}$$

 ISBN 13: 978-84-613-5510-5

2.3 Cauchy Matrix:

Definition 2.3 *Let K be a field, $x_i \in K$ for $1 \leq i \leq k$ and $y_j \in K$ for $1 \leq j \leq r$ such that $\{x_1, ..., x_k\}$ are pairwise distinct and $\{y_1, ..., y_r\}$ are pairwise distinct and $x_i + y_j \neq 0$ for $1 \leq i \leq k$ and $1 \leq j \leq r$.*
The matrix

$$\begin{pmatrix} \frac{1}{x_1+y_1} & \frac{1}{x_1+y_2} & . & . & . & \frac{1}{x_1+y_r} \\ \frac{1}{x_2+y_1} & \frac{1}{x_2+y_2} & . & . & . & \frac{1}{x_2+y_r} \\ . & . & . & & & . \\ . & . & & . & & . \\ . & . & & & . & . \\ \frac{1}{x_k+y_1} & \frac{1}{x_k+y_2} & . & . & . & \frac{1}{x_k+y_r} \end{pmatrix}$$

is called a Cauchy matrix over K generated by $\{x_1, ..., x_k\}$ and $\{y_1, ..., y_r\}$.

2.4 Generalized Cauchy Matrix :

A $k \times r$ matrix $\mathbf{A}$ is a generalized Cauchy matrix if $A = D(c)CD(d)$
Where $\mathbf{C}$ is a $k \times r$ Cauchy matrix and $c = (c_1, ..., c_k)$, $c_i \neq 0$ for $1 \leq i \leq k$ and $d = (d_1, ..., d_r)$, $d_j \neq 0$ for $1 \leq j \leq r$.

2.5 Generalized Cauchy Codes

Let $k \in \mathbf{N}$ and $k < n$ for some $n \in \mathbf{N}$.
Let $\mathbf{C}$ be $k \times (n-k)$ Cauchy matrix over a field $\mathbf{K}$. Let $c = (c_1, ..., c_k)$ such that $c_i \in \mathbf{K}$ and $c_i \neq 0, \forall\ 1 \leq i \leq k$ and $d = (d_1, ..., d_{n-k})$ where $d_j \in \mathbf{K}$ and $d_j \neq 0\ \forall\ 1 \leq j \leq n-k$. Let $A = D(c)CD(d)$ ($\mathbf{A}$ is a generalized Cauchy matrix by definition). Then the code generated by the generator matrix $[I_k|A]$ is called the generalized cauchy code.

2.6 Generalized Reed Solomon Codes

Definition 2.4 *Let $GF(q^m)$ be a finite field with q^m elements. Let $n \in N$ with $1 \leq n \leq q^m$ and $\alpha = (\alpha_1, ..., \alpha_n)$ an n-tuple of distinct elements of $GF^(q^m)$ and let $v = (v_1, ..., v_n)$ be an n-tuple of non -zero elements of $GF^(q^m)$. Let $k \in N$ with $1 \leq k \leq n$. Then the Generalized Reed Solomon codes, denoted by : $GRS_{n,k}(\alpha, v)$ is $GRS_{n,k}(\alpha, v) = \{v_1 f(\alpha_1), ..., v_n f(\alpha_n)/f \in GF(q^m)[x], deg(f) \leq k-1\}$.*
We can thus write the generator matrix of Generalized Reed Solomon code as

$$G = \begin{pmatrix} v_1 & v_2 & . & . & . & v_n \\ v_1\alpha_1 & v_2\alpha_2 & & & & v_n\alpha_n \\ . & & & & & \\ . & & & & & \\ . & . & & . & . & . \\ {v_1\alpha_1}^{k-1} & {v_2\alpha_2}^{k-1} & & . & . & {v_n\alpha_n}^{k-1} \end{pmatrix}$$

is noted $GRS_k(\alpha, v)$.

 ISBN 13: 978-84-613-5510-5

The set of $GRS_k(\alpha, v)$ codes is called by Generalized REED SOLOMON codes family.
So,

$$G = \begin{pmatrix} 1 & 1 & . & . & . & 1 \\ \alpha_1 & \alpha_2 & & & & \alpha_n \\ . & & & & & \\ . & & & & & \\ . & . & . & . & . & . \\ {\alpha_1}^{k-1} & {\alpha_2}^{k-1} & . & . & . & {\alpha_n}^{k-1} \end{pmatrix} \times \begin{pmatrix} v_1 & 0 & . & . & . & 0 \\ 0 & v_2 & & & & 0 \\ . & & & & & \\ . & & & & & \\ . & . & . & . & . & . \\ 0 & 0 & . & . & . & v_n \end{pmatrix}$$

i.e. $G(\alpha, v) = V(\alpha)D(v)$, where $V(\alpha)$ is a Vandermonde matrix and $D(v)$ a diagonal matrix.

2.7 Gabidulin Codes

Definition 2.5 *Let $g_1, ..., g_n \in GF(q^m)^n$ be* n *elements, which are linearly independent over* GF(q). *The matrix*

$$\boldsymbol{G} = \begin{pmatrix} g_1^{[0]} & . & . & . & g_n^{[0]} \\ . & . & . & & . \\ . & & . & & . \\ . & & . & & . \\ {g_1}^{[k-1]} & . & . & . & {g_n}^{[k-1]} \end{pmatrix}$$

where $[i] = q^i$ is of rank k, *is a generator matrix of Gabidulin code.*

2.7.1 Properties

- The linear code C with generator matrix G reaches the Singleton bound for the rank metric, that is, let d be the minimum rank distance of C, we have $d - 1 = n - k$.
- The dual of the Gabidulin code is a Gabidulin code.

Remark 2.6 *In the sequel, we will note $Gab_{n,k}(g)$ the Gabidulin code of the lenght n, the dimension k and generated by g*

2.7.2 Definition of affine permutation

1. Let be $Gab_{n,k}(g)$ a generator matrix of Gabidulin code generated by $g = (g_1, ..., g_n)$ ψ_{ij} is defined by

$$\psi_{ij} : GF(q^m) \longrightarrow GF(q^m)$$

$$x \longmapsto a_{ij}x$$

 ISBN 13: 978-84-613-5510-5

where

$$a_{ij} = \frac{\prod_{\lambda_1,\lambda_2,...,\lambda_k \in GF(q)} (g_j - \lambda_i g_i - \sum_{l=1, l\neq i}^{k} \lambda_l g_l)}{g_j - g_i}$$

with $\lambda_i \neq 0$ and $(\lambda_1, \lambda_2, ..., \lambda_k) \neq (0, 0, ..., 0)$.
We are going to extend the action of ψ_{ij} to $GF(q^m)^n$ by the following form :

$$\psi : (GF(q^m))^n \longrightarrow (GF(q^m))^n$$

$$(x_1, x_2, x_3, x_j, ..., x_n) \longmapsto (x_1, x_2, x_3, \psi_{ij}(x_j), ..., \psi_{in}(x_n))$$

2. ϕ is affine permutation define by:

$$\phi : (GF(q^m))^n \to (GF(q^m))^n$$

$$(x_1, ..., x_r, x_{r+1}, ..., x_n) \mapsto (x_1, ..., x_r, \phi_{i,r+1}(x_{r+1}), ..., \phi_{i,n}(x_n))$$

ϕ is an extension of ϕ_{ij} and $\phi_{ij}(x_j) = a_{ij}^{-1} x_j$ for $x_j \in GF(q^m)$

3 Main results

In this section we state our main results

3.1 The Erasure Decoding Algorithm

Proposition 3.1 *if c is a codeword of the Gabidulin code and H its parity matrix, the following conditions are equivalents*

1. $H^t c = 0$
2. $\psi(H)^t \phi(c) = 0$.

Proof 3.2 *Let us consider the $[n, k]$ Gabidulin code generated by $g = (g_1, ..., g_n)$. Its parity matrix $H = Gab_{n,r}(h)$ is such that*

$$H = \begin{pmatrix} h_1 & h_2 & . & . & . & h_n \\ h_1^{[1]} & h_2^{[1]} & . & . & . & h_2^{[1]} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ h_1^{[d-2]} & h_2^{[d-2]} & . & . & . & h_n^{[d-2]} \end{pmatrix}$$

 ISBN 13: 978-84-613-5510-5

with $d = n - k + 1$
Let be $c = (c_1, ..., c_n)$ *is a word.*
If c *is codeword of Gabidulin code, then* $H^t c = 0$*, that is to say*

$$\begin{pmatrix} h_1 & h_2 & . & . & . & h_n \\ h_1^{[1]} & h_2^{[1]} & . & . & . & h_2^{[1]} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ h_1^{[d-2]} & h_2^{[d-2]} & . & . & . & h_n^{[d-2]} \end{pmatrix} \times \begin{pmatrix} c_1 \\ c_2 \\ . \\ . \\ . \\ c_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix}$$

$$\Longrightarrow \begin{cases} h_1 c_1 + h_2 c_2 + ... + h_n c_n = & 0 \\ h_1^{[1]} c_1 + h_2^{[1]} c_2 + ... + h_n^{[1]} c_n = & 0 \\ . & . \\ . & . \\ . & . \\ h_1^{[d-2]} c_1 + h_2^{[d-2]} c_2 + ... + h_n^{[d-2]} c_n = & 0 \end{cases}$$

If $c = (c_1, ..., c_n)$*, then*

$${}^t\phi(c) = \begin{pmatrix} c_1 \\ . \\ . \\ . \\ c_r \\ \phi(c_{r+1}) \\ . \\ . \\ . \\ \phi(c_n) \end{pmatrix}$$

So,

$$\psi(H)^t\phi(c) = \begin{pmatrix} h_1 & h_2 & \dots & h_r & \psi_{1,r+1}(h_{r+1}) & \dots & \psi_{1,n}(h_n) \\ h_1^{[1]} & h_2^{[1]} & \dots & h_r^{[1]} & \psi_{2,r+1}(h_{r+1}^{[1]}) & \dots & \psi_{2,n}(h_n^{[1]}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_1^{[d-2]} & h_2^{[d-2]} & \dots & h_r^{[d-2]} & \psi_{r,r+1}(h_{r+1}^{[d-2]}) & \dots & \psi_{r,n}(h_n^{[d-2]}) \end{pmatrix} \times \begin{pmatrix} c_1 \\ . \\ . \\ . \\ c_r \\ \phi(c_{r+1}) \\ . \\ . \\ . \\ \phi(c_n) \end{pmatrix}$$

where $\phi = (\phi_{ij})$*, with* $r - 1 \leq j \leq n$ *and* i *depending of the line of the* $\psi(H)$ *matrix.*
So, we have $\psi(H)^t\phi(c) =$

 ISBN 13: 978-84-613-5510-5

$$\begin{pmatrix} h_1c_1 + & \dots + & h_rc_r & + \psi_{1,r+1}(h_{r+1})\phi_{1,r+1}(c_{r+1}) & + \dots & + \psi_{1,n}(h_n)\phi_{1,n}(c_n) \\ h_1^{[1]}c_1 + & \dots + & h_r^{[1]}c_r & + \psi_{2,r+1}(h_{r+1}^{[1]})\phi_{2,r+1}(c_{r+1}) & + \dots & + \psi_{2,n}(h_n^{[1]})\phi_{2,n}(c_n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_1^{[d-2]}c_1 + & \dots & + h_r^{[d-2]}c_r & + \psi_{r,r+1}(h_{r+1}^{[d-2]})\phi_{r,r+1}(c_{r+1}) + & \dots + & \psi_{r,n}(h_n^{[d-2]})\phi_{r,n}(c_n) \end{pmatrix}$$

since $\psi_{ij}(X) = a_{ij}X$ *and* $\phi_{ij}(y) = a_{ij}^{-1}$, *then we have* $\psi_{ij}(h_j)\phi_{ij}(c_j) = h_jc_j$, *with* $r+1 \leq j \leq n$ *and* $1 \leq i \leq r$
Therefore $\psi(H)^t\phi(c) = 0$ *is equivalent to* $H^tc = 0$

Remark 3.3 *Let be* $Gab_{n,k}(g)$ *a Gabidulin code. If* c *is not a codeword of Gabidulin code, we have the following equivalent.* $H^tc \neq 0$, *then* $\psi(H)^t\phi(c) \neq 0$.

Theorem 3.4 *Let* $c \in Gab_{n,k}(g)$ *and* $Gab_{n,k}(g)^\perp = H_{n,r}(h)$ *with* $r = n-k$. *Let us suppose that the erasures are at locations* $\{e_1, ..., e_r\} \subset \{1, ..., n\}$. *Let* $\{d_1, ..., d_k\} = \{1, ..., n\} \setminus \{e_1, ..., e_r\}$ *be the non-erasure locations, then.*

$$\begin{pmatrix} c_{e_1} \\ . \\ . \\ . \\ c_{e_r} \end{pmatrix} = -D(y)CD(X) \begin{pmatrix} \phi_{d_1,d_1}(c_{d_1}), \\ . \\ . \\ ., \\ \phi_{d_k,d_k}(c_{d_k}) \end{pmatrix}$$

With $y = \left(\frac{v_1^{-1}}{\xi_1}, \ . \ . \ ., \ \frac{v_r^{-1}}{\xi_r} \right)$, $X = \left(\ v_{d_1}\zeta_1, \ . \ . \ ., \ v_{d_k}\zeta_k \ \right)$, *where* $(v_1, ..., v_n)$ *is come from by calculating* $\psi(H) = V(u)D((v_1, ..., v_n))$

$\xi_i = \Pi_{1 \leq t \leq r}(u_{e_i} - u_{e_t}) \qquad 1 \leq i \leq r \ e_i \neq e_t$

$\zeta_j = \Pi_{1 \leq t \leq r}(u_{d_j} - u_{e_t}) \qquad 1 \leq j \leq k$

C *is a cauchy matrix generated by* $\{-u_{e_1}, ..., -u_{e_r}\}$ *and* $\{u_{d_1}, ..., u_{d_k}\}$

Proof 3.5 *Let be* G *the generator matrix of* $Gab_{n,k}(g)$ *over* $GF(q^m)$ *and* H *its parity matrix.* $H = Gab_{n,r}(h)$ *where* $r = n-k$
Since $\psi(H)$ *is a generator matrix of GRS code i.e.* $\psi(H) = GRS_{n,r}(u, v)$,
we have: $\psi(H) = [l_1, ..., l_n] = V_r(u)D(v)$ *where* V_r *is a* $r \times n$ *VanderMonde matrix and* D *is a* $n \times n$ *diagonal matrix.*
Let us put $\psi(M) = D(\widehat{v})[V(u_{e_1}, ..., u_{e_r}]^{-1}\psi(H)$ *where* $\widehat{v} = (v_1^{-1}, ..., v_r^{-1})$, $\psi(M)$ *is a* $r \times n$ *matrix.*
Let be $c = (c_1, ..., c_r)$ *and* $\phi(c) = (c_1, ..., c_r, \phi(c_{r+1}, ..., \phi(c_n))$
$\psi(M)^t\phi(c) = D(\widehat{v})[V(u_{e_1}, ..., u_{e_r}]^{-1}\psi(H)^t\phi(c)$

 ISBN 13: 978-84-613-5510-5

Let us show that $\psi(M)$ is a parity matrix of the code of generator matrix $\phi(G)$

*if c is a codeword of $Gab_{n,k}(g)$, $\psi(H)^t\phi(c) = 0$ according to Proposition*3.1.
Moreover $D(\widehat{v}) \neq 0$ and $[V(u_{e_1}, ..., u_{e_r}]^{-1} \neq 0$ because they are invertible matrices.
So, $\psi(M)^t\phi(c) = 0 \Longleftrightarrow \psi(H)^t\phi(c) = 0$.

In the same way, if c is not a codeword of Gabidulin code, $\psi(M)^t\phi(c) \neq 0 \Longleftrightarrow \psi(H)^t\phi(c) \neq 0$.
Therefore $\psi(M)$ is a parity matrix of the code of generator matrix $\phi(G)$.

Let us put $\widehat{\xi} = [\frac{1}{\xi_1}, ..., \frac{1}{\xi_r}]$, *with* $\xi_i = \prod_{1\leq t\leq r, e_i\neq e_t} (u_{e_i} - u_{e_t})$

Let us put $f_i(z) = \prod_{1\leq t\leq r, e_i\neq e_t} (z - u_{e_t}) = \sum_{1\leq s\leq r-1} a_{i,s}z^s$

$$[V(u_{e_1}, ..., u_{e_r}]^{-1} = \left(\frac{a_{i,j-1}}{\prod_{1\leq t\leq r, e_i\neq e_t} (u_{e_i} - u_{e_t})} \right)_{ij} = \left(\frac{1}{\xi_i} \times a_{i,j-1} \right)_{ij}$$

$$= \begin{pmatrix} \frac{1}{\xi_1} & 0 & \dots & 0 \\ 0 & \frac{1}{\xi_2} & & \\ \vdots & 0 & \ddots & \\ 0 & \dots & & \frac{1}{\xi_r} \end{pmatrix} \times \begin{pmatrix} a_{1,0} & \dots & a_{1,r-1} \\ \vdots & \ddots & \vdots \\ a_{r,0} & \dots & a_{r,r-1} \end{pmatrix}$$

We have, by replacing $\psi(H)$ by its expression
$\psi(M) = D(\widehat{v})[V(u_{e_1}, ..., u_{e_r})]^{-1}\psi(H)$

$= D(\widehat{v})[V(u_{e_1}, ..., u_{e_r})]^{-1}V_r(u)D(v)$

$$= D(\widehat{v}) \begin{pmatrix} \frac{1}{\xi_1} & 0 & \dots & 0 \\ 0 & \frac{1}{\xi_2} & & \\ \vdots & 0 & \ddots & \\ 0 & \dots & & \frac{1}{\xi_r} \end{pmatrix} \times \begin{pmatrix} a_{1,0} & \dots & a_{1,r-1} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ a_{r,0} & \dots & a_{r,r-1} \end{pmatrix} V_r(u)D(v)$$

$$\psi(M) = \begin{pmatrix} \frac{v_1^{-1}}{\xi_1} & 0 & \dots & 0 \\ 0 & \frac{v_2^{-1}}{\xi_2} & & \\ \vdots & 0 & \ddots & \\ 0 & \dots & & \frac{v_r^{-1}}{\xi_r} \end{pmatrix} \times \begin{pmatrix} a_{1,0} & \dots & a_{1,r-1} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ a_{r,0} & \dots & a_{r,r-1} \end{pmatrix} \times \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ u_1^{r-1} & \dots & a_n^{r-1} \end{pmatrix} D(v)$$

Let us put $W = D(\widehat{v}\widehat{\xi})$

 ISBN 13: 978-84-613-5510-5

$$\psi(M) = W \begin{pmatrix} \sum_{t=0}^{r-1} a_{1,t}u_1^t & \dots & \sum_{t=0}^{r-1} a_{1,t}u_n^t \\ \vdots & \ddots & \vdots \\ \sum_{t=0}^{r-1} a_{r,t}u_1^t & \dots & \sum_{t=0}^{r-1} a_{r,t}u_n^t \end{pmatrix} D(v)$$

Let us put

$$S = \begin{pmatrix} \sum_{t=0}^{r-1} a_{1,t}u_1^t & \dots & \sum_{t=0}^{r-1} a_{1,t}u_n^t \\ \vdots & \ddots & \vdots \\ \sum_{t=0}^{r-1} a_{r,t}u_1^t & \dots & \sum_{t=0}^{r-1} a_{r,t}u_n^t \end{pmatrix}$$

Let us denote s_i with $1 \leq i \leq n$ the columns of S , we have $S = [S_1, ..., S_n]$, where $S_i = [f_1(u_i), f_2(u_i), ..., f_r(u_i)]^\perp$ with f_i defined as in subsection 3.1.
$\psi(M) = WSD(v)$
$\psi(M) = D(\widehat{v}\widehat{\xi})SD(v)$
$\psi(M) = D(\widehat{v})D(\widehat{\xi})SD(v)$
Let be $i \in \{e_1, ..., e_r\}$, let us put $i = e_\lambda$ for $\lambda \in \{1, ..., r\}$. Then

$$S_{e_\lambda} = \begin{pmatrix} f_1(u_{e_\lambda}) \\ \vdots \\ f_r(u_{e_\lambda}) \end{pmatrix} = \begin{pmatrix} \prod_{1 \leq t \leq r, e_1 \neq e_t} (u_{e_\lambda} - u_{e_t}) \\ \vdots \\ \prod_{1 \leq t \leq r, e_r \neq e_t} (u_{e_\lambda} - u_{e_t}) \end{pmatrix}$$

Thus, we have

$$f_\mu(u_{e_\lambda}) = \begin{cases} \prod_{1 \leq t \leq r, e_\mu \neq e_t} (u_{e_\lambda} - u_{e_t}) & if \quad e_\mu = e_\lambda \\ 0 & otherwise \end{cases}$$

For $\mu \in \{1, ..., r\}$. Since we can write $\xi_\mu = \prod_{1 \leq t \leq r, e_\lambda \neq e_t} (u_{e_\lambda} - u_{e_t})$
and so, we can rewrite $f_\mu(u_{e_\lambda}) = \begin{cases} \xi_\mu & if \quad e_\mu = e_\lambda \\ 0 & otherwise \end{cases}$
When $i \notin \{e_1, ..., e_r\}$, then $i = d_\rho$ for some $\rho \in \{1, ..., k\}$

 ISBN 13: 978-84-613-5510-5

Then

$$S_{d_\rho} = \begin{pmatrix} f_1(u_{d_\rho}) \\ \vdots \\ f_r(u_{d_\rho}) \end{pmatrix} = \begin{pmatrix} \prod\limits_{1\leq t\leq r, e_1\neq e_t} (u_{d_\rho} - u_{e_t}) \\ \vdots \\ \prod\limits_{1\leq t\leq r, e_r\neq e_t} (u_{d_\rho} - u_{e_t}) \end{pmatrix} = \zeta_\sigma \begin{pmatrix} \frac{1}{u_{d_\rho}-u_{e_1}} \\ \vdots \\ \frac{1}{u_{d_\rho}-u_{e_r}} \end{pmatrix}$$

where $\zeta_\mu = \prod\limits_{1\leq t\leq r} (u_{d_\rho} - u_{e_t})$
Since we have $\psi(M)^t\phi(c) = 0$

$\iff D(\widehat{v})D(\widehat{\xi})SD(v)^t\phi(c) = 0$

$$D(v)^t\phi(c) = \begin{pmatrix} v_1 & 0 & \dots & 0 \\ 0 & v_2 & 0 & \\ \vdots & \ddots & & 0 \\ 0 & \dots & 0 & v_n \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_r \\ \phi(c_{r+1}) \\ \vdots \\ \phi(c_n) \end{pmatrix}$$

and we can $D(v)^t\phi(c) = [v_1c_1, ..., v_rc_r, v_{r+1}\phi_{r+1,r+1}(c_{r+1}), ..., v_n\phi_{n,n}(c_n)]^\perp$
and so by multiplying by the S matrix at left we have :

$$SD(v)^t\phi(c) = \prod_{i=1}^{n} s_iv_ic_i$$

$$SD(v)^t\phi(c) = \prod_{i=1}^{r} s_{e_i}v_{e_i}c_{e_i} + \prod_{i=1}^{k} s_{d_i}v_{d_i}\phi_{d_i,d_i}(c_{d_i})$$

We have $D(\widehat{v})D(\widehat{\xi})SD(v)^t\phi(c) = WSD(v)^t\phi(c)$

$$WSD(v)^t\phi(c) = \prod_{i=1}^{r} Ws_{e_i}v_{e_i}c_{e_i} + \prod_{i=1}^{k} Ws_{d_i}v_{d_i}\phi_{d_i,d_i}(c_{d_i}) \tag{1}$$

1. *Now, when* $t \in \{e_1, ..., e_r\}$ *and* $t = e_\lambda$

$$Ws_{e_\lambda}v_{e_\lambda}c_{e_\lambda} = W \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \xi_{e_\lambda}v_{e_\lambda}c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

 ISBN 13: 978-84-613-5510-5

$$Ws_{e_\lambda}v_{e_\lambda}c_{e_\lambda} = D([\frac{v_1^{-1}}{\xi_1},...,\frac{v_r^{-1}}{\xi_r}]) \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \xi_{e_\lambda}v_{e_\lambda}c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$Ws_{e_\lambda}v_{e_\lambda}c_{e_\lambda} = \begin{pmatrix} \frac{v_1^{-1}}{\xi_1} & 0 & \dots & 0 \\ 0 & \frac{v_2^{-1}}{} & 0 & \\ \vdots & \ddots & & 0 \\ 0 & \dots & 0 & \frac{v_r^{-1}}{\xi_r} \end{pmatrix} \times \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \xi_{e_\lambda}v_{e_\lambda}c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Where $\xi_{e_\lambda}v_{e_\lambda}$ is at the λ^{th} position in the array. Simplifying further we have

$$Ws_{e_\lambda}v_{e_\lambda}c_{e_\lambda} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{v_{e_\lambda}^{-1}}{\xi_{e_\lambda}}\xi_{e_\lambda}v_{e_\lambda}c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

With c_{e_λ} is at the λ^{th} position in the array. Thus, we have

$$Ws_{e_\lambda}v_{e_\lambda}c_{e_\lambda} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ c_{e_\lambda} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

further we have

$$\prod_{i=1}^{r} Ws_{e_i}v_{e_i}c_{e_i} = \begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} \tag{2}$$

 ISBN 13: 978-84-613-5510-5

2. *When* $t \notin \{e_1, .., e_r\}$ *then* $t \in \{d_1, .., d_k\}$ *and let us put* $t = d_\rho$

$$Ws_{d_\rho} v_{d_\rho} \phi_{d_\rho, d_\rho}(c_{d_\rho}) = W \begin{pmatrix} \frac{1}{u_{d_\rho} - u_{d_1}} \\ \vdots \\ \frac{1}{u_{d_\rho} - u_{d_r}} \end{pmatrix} \zeta_\rho v_{d_\rho} \phi_{d_\rho, d_\rho}(c_{d_\rho})$$

and so,

$$\sum_{i=1}^{k} Ws_{d_i} v_{d_i} \phi_{d_i, d_i}(c_{d_i}) = W \sum_{i=1}^{k} (\begin{bmatrix} \frac{1}{u_{d_t} - u_{d_1}} \\ \vdots \\ \frac{1}{u_{d_t} - u_{d_r}} \end{bmatrix} \zeta_t v_{d_t} \phi_{d_t, d_t}(c_{d_t}))$$

We can simplifying further by writing

$$\sum_{i=1}^{k} (\begin{bmatrix} \frac{1}{u_{d_t} - u_{d_1}} \\ \vdots \\ \frac{1}{u_{d_t} - u_{d_r}} \end{bmatrix} \zeta_t v_{d_t} \phi_{d_t, d_t}(c_{d_t})) = \begin{pmatrix} \frac{1}{u_{d_1} - u_{e_1}} & \cdots & \frac{1}{u_{d_k} - u_{e_1}} \\ \vdots & \ddots & \vdots \\ \frac{1}{u_{d_1} - u_{e_r}} & \cdots & \frac{1}{u_{d_1} - u_{e_r}} \end{pmatrix} \times \begin{pmatrix} \zeta_1 v_{d_1} \phi_{d_1, d_1}(c_{d_1}) \\ \vdots \\ \zeta_k v_{d_k} \phi_{d_k, d_k}(c_{d_k}) \end{pmatrix}$$

where

$$C = \begin{pmatrix} \frac{1}{u_{d_1} - u_{e_1}} & \cdots & \frac{1}{u_{d_k} - u_{e_1}} \\ \vdots & \ddots & \vdots \\ \frac{1}{u_{d_1} - u_{e_r}} & \cdots & \frac{1}{u_{d_1} - u_{e_r}} \end{pmatrix}$$

is a Cauchy matrix generated by $\{-u_{e_1}, ..., -u_{e_r}\}$ *and* $\{u_{d_1}, ..., u_{d_k}\}$
Thus ,

$$\sum_{i=1}^{k} Ws_{d_i} v_{d_i} \phi_{d_i, d_i}(c_{d_i}) = WC \begin{pmatrix} \zeta_1 v_{d_1} \phi_{d_1, d_1}(c_{d_1}) \\ \vdots \\ \zeta_k v_{d_k} \phi_{d_k, d_k}(c_{d_k}) \end{pmatrix} \tag{3}$$

Considering the relations (2) *and* (3)*, the relation* (1 =) *become, since* $\psi(M)^t \phi(c) = 0$

$$\prod_{i=1}^{r} Ws_{e_i} v_{e_i} c_{e_i} + \prod_{i=1}^{k} Ws_{d_i} v_{d_i} \phi_{d_i, d_i}(c_{d_i}) = 0$$

Thus

$$\begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} + WC \begin{pmatrix} \zeta_1 v_{d_1} \phi_{d_1, d_1}(c_{d_1}) \\ \vdots \\ \zeta_k v_{d_k} \phi_{d_k, d_k}(c_{d_k}) \end{pmatrix} = 0$$

$$\iff \begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} = -WC \begin{pmatrix} \zeta_1 v_{d_1} \phi_{d_1, d_1}(c_{d_1}) \\ \vdots \\ \zeta_k v_{d_k} \phi_{d_k, d_k}(c_{d_k}) \end{pmatrix}$$

$$\implies \begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} = -D([\frac{v_1^{-1}}{\xi_1}, ..., \frac{v_r^{-1}}{\xi_r}])C \begin{pmatrix} \zeta_1 v_{d_1} \phi_{d_1, d_1}(c_{d_1}) \\ \vdots \\ \zeta_k v_{d_k} \phi_{d_k, d_k}(c_{d_k}) \end{pmatrix}$$

 ISBN 13: 978-84-613-5510-5

So, we have

$$\begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} = -D([\frac{v_1^{-1}}{\xi_1}, ..., \frac{v_r^{-1}}{\xi_r}]) \times C \times D([v_{d_1}\zeta_1, ..., v_{d_k}\zeta_k]) \times \begin{pmatrix} \phi_{d_1,d_1}(c_{d_1}) \\ \vdots \\ \phi_{d_k,d_k}(c_{d_k}) \end{pmatrix}$$

3.1.1 Algorithm

1. compute ξ_i, ζ_j, the cauchy matrix C and ϕ
2. compute $X' = \left(\ v_{d_1}\zeta_1\phi_{d_1,d_1}(c_{d_1}), \ . \ . \ ., \ v_{d_k}\zeta_k\phi_{d_k,d_k}(c_{d_k}) \ \right)$
3. compute $\widehat{X} = CX'$
4. compute $-D\left(\ \frac{v_1^{-1}}{\xi_1}, \ . \ . \ ., \ \frac{v_r^{-1}}{\xi_r} \ \right) \widehat{X}$

gives the values at the erasure locations $\{e_1, ..., e_r\}$.

4 Analysis of Decoding complexity

In this section we discuss the complexity of the algorithm. Since the codes are defined in a field GF. It has two operations:addition and multiplication. Thus, we do an analysis of the complexity of the decoding algorithm by counting the number of additions, negations (additive inverse of an element),multiplications and inversions (multiplicative inverse of an element) required for the decoding algorithm.

Proposition 4.1 *The erasure decoding algorithm for Gabidulin codes describe above, require $r^2 + r(k+1)$ negations, $r(r-1) + 2rk$ additions, $r^2 + r(k-1)$ multiplications and $r(k+1)$ inversions.*

Proof 4.2

1. $\xi_i = \Pi_{1 \leq t \leq r}(\alpha_{e_i} - \alpha_{e_t}) \qquad 1 \leq i \leq r \ e_i \neq e_t$ *is calculated in $r-1$ negations, $r-1$ additions and $r-2$ multiplications.*
2. $\zeta_j = \Pi_{1 \leq t \leq r}(\alpha_{d_j} - \alpha_{e_t}) \qquad 1 \leq j \leq k$. *to calculate ζ_j we need k negations, k additions and $k-1$ multiplications*
3. $-D\left(\ \frac{v_1^{-1}}{\xi_1}, \ . \ . \ ., \ \frac{v_r^{-1}}{\xi_r} \ \right)$ *requires r^2 negations, $r(r-1)$ additions, $r(r-1)$ multiplications and r inversions*
4. $\left(\ v_{d_1}\zeta_1\phi_{d_1,d_1}(c_{d_1}), \ . \ . \ ., \ v_{d_k}\zeta_k\phi_{d_k,d_k}(c_{d_k}) \ \right)^{\perp}$ *requires rk negations, rk additions and rk multiplications*
5. *C is a $r \times k$ cauchy matrix, we need r negations, rk additions and rk inversions*

Thus in total, we need : $r^2+r(k+1)$ negations, $r(r-1)+2rk$ additions, $r^2+r(k-1)$ multiplications and $r(k+1)$ inversions.

 ISBN 13: 978-84-613-5510-5

.

Remark 4.3 *To decode each codeword, we first calculed* $x_1 = \left[\ v_{d_1}\zeta_1\phi_{d_1,d_1}(c_{d_1}), \ . \ . \ ., \ v_{d_k}\zeta_k\phi_{d_k,d_k}(c_{d_k}) \ \right]^{\perp}$ *which takes* k *multiplications as* $v_j\zeta_j$ *and* $\phi_{d_j,d_j}(c_{d_j})$ *have already been calculated.*
Next, we calculate $x_2 = Cx_1$ *which takes* $r(k-1)$ *additions and* rk *multiplications.*
Finally, we calculate

$$\begin{pmatrix} c_{e_1} \\ \vdots \\ c_{e_r} \end{pmatrix} = D(-[\frac{v_1^{-1}}{\xi_1}, ..., \frac{v_r^{-1}}{\xi_r}])x_2$$

, which takes r *multiplications.*
Thus, in total we have $r(k-1)$ *additions and* $rk+n$ *multiplications.*

If

$$\psi(M) = -D([\frac{v_1^{-1}}{\xi_1}, ..., \frac{v_r^{-1}}{\xi_r}])CD(v_{d_1}\zeta_1\phi_{d_1,d_1}(c_{d_1}), ..., v_{d_k}\zeta_k\phi_{d_k,d_k}(c_{d_k}))$$

is compute first, then decoding takes rk *multiplications and* $r(k-1)$ *additions while setting up the structures for decoding requires further* $2rk$ *multiplications. Thus, totaling* $:r^2 + r(k-1) + 2rk = r^2 + r(3k-1)$ *multiplications.*

5 Conclusion

In this paper we described an erasure decoding algorithm for Gabidulin codes by utilizing the structure of the inverse of the VanderMonde matrix. We have shown that this new algorithm compute the erasure locations fixed by a single multiplication of three matrices; two of which are diagonal matrices and the other is a cauchy matrix. This reduces significantly the decoding complexity compared to matrix inverse based decoding algorithm.

 ISBN 13: 978-84-613-5510-5

References

[1] R. F. Babindamana and C. T. Gueye, *Gabidulin codes that are generalized Reed Solomon codes*, International Journal of Algebra **4**(3) (2010) 119–142.

[2] E. Gabidulin, *Theory of code with maximum rank distance*, problemy Peredachi Informatsii **21**(1) (1985) 1–12.

[3] P. Gaborit, *Shorter keys for code based cryptography*, International Workshop on Coding and Cryptography September, 2004.

[4] R. Lidl and H. Niederreiter, *Introduction to finite fields and their applications*, Cambridge University Press 1997.

[5] P. Loidreau, *Metrique rang et cryptographie*, mmoire d'Habilitation a diriger des Recherches Universit Pierre et Marie Curie, Paris 6 25 janvier 2007.

[6] P. Loidreau *Etude et Optimisation des Cryptosystmes Cl Publique fonds sur la Theorie des Codes Correcteurs d Erreurs*, 4 mai 2001 , thse prsente l' Ecole Nationale Superieure de Techniques Avances (ENSTA), Universit Paris 6.

[7] R. M. Roth and G. Seroussi, *On generator matrices of MDS codes*, IEEE Trans. Inform. Theory **31**(6) (1985) 826–830.

[8] M. Shretha and L. Xu, *Efficient Erasure Decoding for Generalized Reed Solomon Codes*, preprint.

[9] L. Xu and Bruck, *X-Code: MDS array codes with optimal encoding*, IEEE Transactions on Information Theory **45**(1) (1999) 272–276.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Mapping MMOFPS over Heterogeneous Distributed Systems

Ignasi Barri[1], Miquel Orobitg[1], Concepció Roig[1] and Francesc Giné[1]

[1] *Computer Science Department, University of Lleida*

emails: {ignasibarri, orobitg, roig, sisco}@diei.udl.cat

Abstract

In this paper we explore the problem of mapping game services of MMOFPS (First Person Shooter games) games in a hybrid architecture, called *OnDeGas* (On Demand Game Service), that combines the functionalities of a centralized server infrastructure with a distributed P2P topology. We propose and analyze two mapping strategies, for *OnDeGas*, that differ in the way that they tackle the heterogeneity, in the number of cores, of nodes in the P2P area. We show through simulation that both mapping mechanisms are able to provide a distributed platform that scales on demand. However, they differ in performance, as it can be seen that taking into account heterogeneity provides a better use of resources and faster mapping decisions at the expense of having more communication overhead and a broader variability of latency values.

Key words: MMOFPS, Mapping, Heterogeneity, Distributed System.

1 Introduction

Massively Multiplayer Online Games (MMOG) are the most popular genre in the computer game world [6]. They can be divided into three categories: *MMORPG* (Massively Multiplayer Online Role Playing Games), *MMORTS* (Massively Multiplayer Online Real Time Strategy) and *MMOFPS* (Massively Multiplayer Online First Person Shooter). The execution requirements vary with the way of playing in each of them [11]. On the one hand, MMORPG and MMORTS can have thousands of players in a single party, so bandwidth is an important feature for supporting them [5]. On the other hand, in MMOFPS, players are divided into many isolated game sessions each with a handful of players who are continuously interacting. Thus, response latency is the key factor in this case.

According to that, the strategies to optimize the execution of MMOGs, in current distributed platforms, are different depending on the category they belong to. In this paper, we focus on the optimization of MMOFPS games. Traditionally, client-server

 ISBN 13: 978-84-613-5510-5

systems have been the platforms to provide service to massively networked games. However, when the number of players increases, this approach reaches its limits due to problems of scalability.

The research community has proposed some alternatives to overcome client-server limits with decentralized structures, where each machine contributes to, and benefits from, a large service oriented network. Some works have focused on the game itself to serve MMORPG. These solutions are based on splitting the game world into different subspaces and distributing them into the decentralized nodes [8] [7], or to group players according to the load on the map [9]. Unfortunately, these proposals do not fit into the specific features of the MMOFPS games, mainly due to their low latency requirements and the length of game sessions. In the specific case of MMOFPS, Bharambe et al [3] proposes a solution for a pure P2P system. The increase in latency time, inherent to this kind of architecture, is solved by proposing new rules in many features of current MMOFPS games, such as the size of the AOI (Area of Interest) of players in order to decrease the number of messages that players transfer to each other. These improvements need to be included in the internal code of the game, which implies important implementation efforts. Unlike Bharambe works, our efforts are focused to propose a scalable architecture oriented to MMOFPS, which assures the players' latency below a specific satisfaction threshold without changing the internal code of games.

According to our aims, we propose a new system named *OnDeGaS* (On Demand Game Service), devoted to mapping the MMOFPS game sessions without affecting the game's internal code. *OnDeGaS* is a hybrid system, which is made up of a central server and a set of temporary servers, distributed throughout a P2P topology. The central server executes several game services as long as it is not overloaded; in the case of an overload situation in the central server, due to a peak number of players, new game sessions are mapped into the P2P topology. An initial version of *OnDeGaS* was reported in [1], focusing on the assignment of game sessions of a MMOFPS in the distributed area, without considering physical differences in player' machines. However, heterogeneity is an inherent property of these nodes and so, it is a key issue to improve the efficiency of the system. For this reason, in this work, two mapping policies, *Non_Heterogeneity_aware* and *Heterogeneity_aware*, are proposed and analyzed. Whenever the central server is overloaded, *Non_Heterogeneity_aware* mechanism selects a new distributed game server, among waiting players, taking exclusively latency criteria into account. The *Heterogeneity_aware* policy applies a variation of previous lookup mechanism taking advantage of all available computational resources throughout the system. Thus, it maps the new game sessions into the available cores of the game servers already created.

The effectiveness of our approaches has been evaluated by means of simulation. Our results show that the *OnDeGaS* scales on demand. Moreover, the *Heterogeneity_aware* mechanism provides a better use of resources and faster mapping decisions, at expense of having more communication overhead and higher latency variability. However, *Non_Heterogeneity_aware* policy, has more stable latency values, slower mapping decisions and avoids the communication overhead.

The remainder of this paper is organized as follows. Section 2 describes the *OnDe-*

 ISBN 13: 978-84-613-5510-5

GaS system composed of a set of algorithms and methodologies. Section 3 evaluates the performance of the proposed mapping policies of the system in terms of scalability and QoS. Section 4 outlines the main conclusions and future work.

2 *OnDeGaS* System Description

In this section, the *OnDeGaS* system is described globally, discussing the components, their operation and implementation details.

2.1 System Model

Figure 1 shows the *OnDeGaS* system that is composed of two main areas: one central area performing central services and a distributed area with several zones that grow in a P2P like fashion.

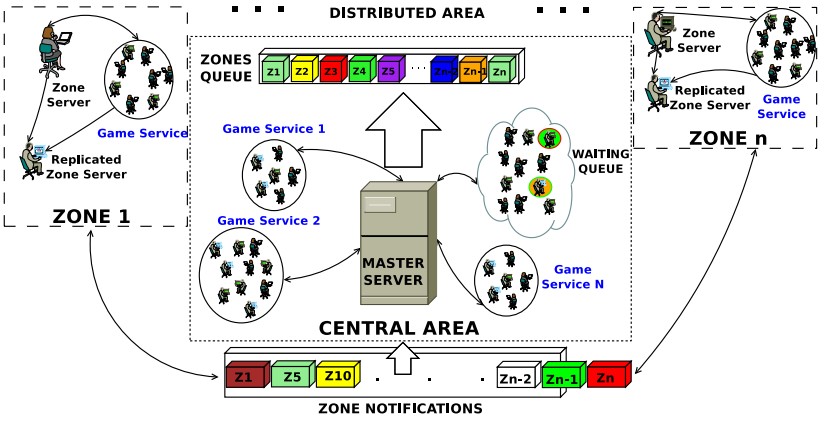


Figure 1: *OnDeGaS* system global vision.

The central area is devoted to performing the global control of the system and also to supplying players with services. The components of the central area are the following:

- *Master Server (MS).* It is the system's main server and acts as the bootstrap point. All the players requesting to enter the system will attempt to connect to it.

- *Waiting Queue (WQ).* This is a logical space in MS used to insert those players who cannot be served due to overload situations. It is a transitory state for players, who will be distributed in a short term.

- *Zones Queue (ZQ).* This is a logical space in MS used to keep the information about the created zones updated. This information is used for distributing players to the already created zones.

 ISBN 13: 978-84-613-5510-5

The distributed area is composed of players' machines that are logically grouped in zones, which are locally circumscribed places running physically out of the MS conforming a distributed platform. A Zone number i, Z_i, in the system, has the following components:

- *Zone Server (ZS).* This is the current server of the Zone.
- *Replicated Zone Server (RZS).* This is the current replicated server of the Zone. It has the role of implementing fault tolerance policies.

Regarding games that are to be executed on this platform, we distinguish the following elements:

- *Player (P_i).* A Player number i, P_i, is a client who connects to the system in order to play a MMOFPS.
- *Game Service (GS).* It is an instance of a game, where a set of players is connected to play. Each GS will be hosted in the MS or in a single core of a ZS. At any moment, each GS can be in two different states: *active* when players are interacting in the GS, or *over*, when the GS has ended due to player disconnections or caused by the rules of the GSs. Normally, in MMOFPS, the *number of players* per GS is in the order of tens, while the *length of the GS* is in the order of a few minutes.
- *Zones Notifications (Z^N).* These are the set of N Zones that have sent a message to the MS to notify that their respective GSs are over. In this case, the MS will decide if the zone's players can be reaccepted.

2.2 System Operation

The operation of the *OnDeGaS* platform is a hybrid between the classical centralized client-server model, performed in the central area, and the distributed P2P model, performed in Zones. The main idea of system operation consists of executing a set of GSs in the central area until it reaches the limit of its capabilities. When no more players can be accepted by the MS, new zones are dynamically created to avoid large waiting times for players, and to provide scalability to the system.

The system operation is controlled by the continuous execution of Algorithm 1, which has two input flows: new player connections (P_i) and the set of zones (Z^N) that have ended their GS and want to enter the MS.

At each iteration of Algorithm 1, the MS checks its state (*MS.state*). If it is overloaded, new player connections will be en-queued to the WQ. After en-queuing, the MS checks if the number of players in the WQ is greater than or equal to a predefined value α, or whether the uptime of the WQ is greater than or equal to a predefined value β, too. If any of these two conditions is true, the MS will execute the mapping function (*MS.mapping(WQ)*) to distribute players. This mapping policy can be undertaken using one of the two following mechanisms that will be discussed in next subsection:

 ISBN 13: 978-84-613-5510-5

```
Input: ∀P_i connecting to MS
Input: Z^N = {Z_i, Z_{i+1}, ..., Z_{n-i}, Z_n} notifying to MS
while True do
    switch MS.state() do
        case MS.state() == OVERLOAD
            if ∃ P_i then MS.enqueue(WQ, P_i);
            if (WQ.size() ≥ α) or (WQ.uptime() ≥ β) then MS.mapping(WQ);
        end
        case MS.state() == NOT OVERLOADED
            if WQ.uptime() ≥ β then
                forall P_i in WQ do
                    MS.accept(P_i);
                end
            if Z^N ≠ ∅ then MS.reaccept(P_i);
            if ∃ P_i then
                if MS.state() == NOT OVERLOADED then
                    MS.accept(P_i);
                else MS.enqueue(WQ, P_i);
        end
    end
end
```

Algorithm 1: OnDeGaS main Algorithm.

(a) *Non_Heterogeneity_aware*, that distributes one GS per zone independently of the physical characteristics of nodes and, (b) *Heterogeneity_aware* that distributes GSs to zones according to the number of cores of player's nodes.

In the case that the MS state is not overloaded, Algorithm 1 evaluates the three following conditional statements:

- The first condition evaluates if the uptime of the WQ is greater than or equal to β; if so, players located in the WQ will be accepted to play in the MS (*MS.accept()*). This acceptance flow acts like a *FIFO*, the first player en-queued in the WQ is the first to be reconnected to the MS if it has enough space. This way, the *OnDeGaS* system rewards the players with more patience.

- The second conditional statement gives priority to enter to the MS, players of those zones that sent an *over* message to notify that they have finished the GS, and they want to start another GS in the MS. This happens whenever a round of the game has finished and players are waiting for the next round. In this case, if the set of notifying zones, Z^N, is not empty, the MS executes the *Reaccept* function. For each Zone, Z_i, in Z^N, the MS re-accepts it, if it has enough free space for all players of Z_i. In this case, Z_i is deleted from Z^N. Note that the algorithm tries to prioritize that all players of the re-accepted Zone are connected to the same GS in the MS to avoid fragmentation of the zones' players. If there is not enough space in the MS, a deny message is sent to Z_i notifying that it can

 ISBN 13: 978-84-613-5510-5

start a new GS. Note that distributed players are playing continuously in the MS or in the zones, and the time transitions are of the order of seconds, which is an acceptable delay for the players.

- The last conditional statement evaluates the existence of new players trying to connect to the system. These new players will be accepted to the MS if it is not overloaded, or they will be en-queued to the WQ (*MS.enqueue()*) in other case.

2.2.1 Mapping function

The mapping function *MS.mapping(WQ)* is in charge to distribute the players located in the WQ to a Zone. Depending on the mapping policy, players located in the WQ will be assigned to a new Zone (*Non_Heterogeneity_aware*) or to a to an already created Zone in the distributed area (*Heterogeneity_aware*).

Note that the two mapping policies proposed in this paper are based on the latency requirements of such kind of games, and the difference between both is focused on taking into account heterogeneity of nodes according to the number of cores.

Non_Heterogeneity_aware. The mapping function *MS.mapping(WQ)* will create a new Zone as it is shown in Algorithm 2. This function executes the *lowest latency* function to find the best ZS, and the best RZS (the implementation details for the *lowestLatency* function are discussed in subsection 2.3). Then, all players in the WQ are linked to the new ZS (*ZS.accept()*). In addition, these players are also linked to RZS (*RZS.accept()*) with the aim that RZS keeps the same information as ZS updated. Finally, a Zone Z_i comprises the ZS, RZS and the players that both of them manage. Thus, a fault tolerance mechanism is maintained by the system (see section 2.3).

```
Input: WQ
MS.mapping(WQ):
begin
    ZS, RZS = MS.lowestLatency(WQ);
    foreach P_i ∈ WQ do
        ZS.accept(P_i);
        RZS.accept(P_i);
    end
    Z_i = {ZS ∪ RZS};
end
```

Algorithm 2: *Non_Heterogeneity_aware* mapping function.

Heterogeneity_aware. The mapping function *MS.mapping(WQ)* (see Algorithm 3) will firstly try to distribute the players located in the WQ to an already created Zone contained in ZQ. If the previous function fails, then the MS will create a new Zone.

This mapping function begins with a loop that will add, those zones of ZQ whose ZS has at least one free core (*ZS.freeCores()*) to a local variable called *AvailableZones*. Each free processor can host a GS composed of all players located in the WQ. If there is

 ISBN 13: 978-84-613-5510-5

```
Input: ZQ,WQ
MS.mapping(WQ):
begin
    AvailableZones =∅;
    forall ((ZS and RZS) ∈ Z_i)) ∈ ZQ do
        if (ZS.freeCores() and RZS.freeCores()) then
            AvailableZones =AvailableZones +{Z_i};
    end
    if (AvailableZones != ∅) then
        Z_i=MS.lowestLatency(AvailableZones);
        Z_i.Accept(WQ);
    else
        ZS=MS.lowestLatency(WQ);
        RZS=MS.lowestLatency(WQ-{ZS});
        if (ZS.freeCores() > RZS.freeCores()) then swap(ZS,RZS);
        ZS.accept(WQ);
        RZS.accept(WQ);
        Z_i = {ZS ∪ RZS};
        ZQ = ZQ + Z_i;
    end
end
```

Algorithm 3: *Heterogeneity_aware* Mapping function.

a zone available, the function will select the one which has the lowest latency between the respective ZS and the MS (*MS.lowestLatency()*). Then the ZS and RZS of the selected Zone will accept all players located in the WQ. If no Zone is found, then a new Zone is created; first of all, it is executed the *lowest Latency* function to find the best ZS and the best RZS, in latency and computational resource terms. Moreover, the function ensures that RZS is able to serve at least the same number of GSs as the ZS to avoid problems when the fault tolerance mechanisms acts (*if* statement with function *swap*). Then, all players in the WQ are linked to the new ZS (*ZS.accept()*) and to the RZS (*RZS.accept()*) with the aim of RZS keeping the same information as the updated ZS. Thus, a fault tolerance mechanism is maintained by the system (see section 2.3). Then, a Zone Z_i comprises the ZS, the RZS and the set of players previously located in the WQ. Finally, the created zone is added to the ZQ variable, in order to reuse them in future situations of overload in MS.

2.3 Implementation Issues

The following issues need to be considered for the proper performance of the system:

Lowest Latency Functionality. *Lowest_Latency* function presented in Algorithms 2 and 3 is based on a loop that checks the latency of all the ZS located in the ZQ with respect to the MS. Then, it selects the closest Zone to the MS to assign the players located in the WQ. It guarantees that the ZS will be very similar to the MS (in

 ISBN 13: 978-84-613-5510-5

latency terms) for the most players located in the WQ.

System Overload State. The state of system overload is determined by the number of concurrent players playing in the MS. According to many authors, this is the most important factor for determining the overload condition in MMOFPS [2, 11, 5], as it has been proved experimentally that the number of concurrent players is directly related to the CPU and network usage.

Free Cores Functionality. In Algorithm 3, the *freeCores* function is used. This function returns the number of free cores of the ZS or RZS (depending on which node executes the function). The studies carried out by Ye and Cheng in [11] show that with an idle processor, is possible to easily provide an MMOFPS with QoS service. Likewise, our system assumes that the player's computational resources are totally dedicated to the MMOFPS and therefore, it is feasible to take advantage of all these computational resources of a player. Thus, the maximum number of GSs that a Zone is able to execute is equal to the number of cores of the ZS, being that, a ZS reserves a core to run its own GS when the ZS is involved in a player role, apart from the ZS role.

Fault Tolerance. In the *OnDeGaS* system, the fault tolerance mechanism is introduced by the use of the RZS. The role of the RZS is to replace the ZS in case of failure. For this reason, players in the distributed area play against the ZS and its RZS, and the ZS sends the game state to both, players and the RZS. In the previous work reported by the authors [1], there is a detailed explanation and performance analysis of the fault tolerance mechanism, when it is applied to a homogeneous distributed platform with a simple processor in each node.

3 Experimental Results

In this section, an experimentation process is conducted to demonstrate the feasibility and good performance of the proposed *OnDeGaS* system as well as to compare the influence in performance of the two presented mapping mechanisms for the distributed area. The idea is to show that it configures a dynamically scalable-on-demand platform and also that *OnDeGaS* provides a good game experience system with both mapping policies, albeit with differences in performance in terms of latency, number of created zones, waiting time for players located in the WQ and communication costs.

The experimentation was performed through simulation using SimPy [10]. SimPy is a discrete-event simulation language based on standard Python. SimPy tools have been used with python classes to implement the nodes of the platform, which can develop four distinct roles: player, ZS, RZS and MS. SimPy procedures, allow random behavior of the simulation to be created to represent the real behavior of a gamer.

Each simulation consists of $100,000$ player connections to the MS. The connections are sequential with constant inter-arrival time ($\approx$ 1 second) to submit the MS to a constant stress situation or constant peak load, in order to verify that the distributed area is dynamically adapted to the on-demand queries of players. When the MS reaches its limit, $2,000$ concurrent players, (since the computational resources of a typical single machine server can support $2,000$ to $6,000$ concurrent clients [8]), no more players will

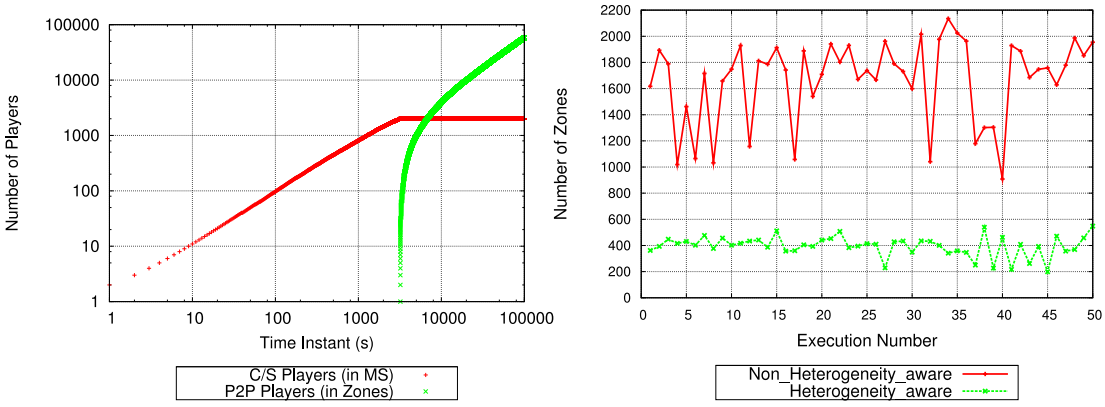
 ISBN 13: 978-84-613-5510-5

be accepted, and new ones will be distributed to zones. Another important issue is the calculus of the players' latency against MS. This is determined by a triangulated heuristic, delimiting the 2-Dimensional Euclidean Space to ($x = [-110, +110], y = [-110, 110]$). This methodology is based on the relative coordinates explained in [4]. Furthermore, each player has a lifetime determined by a Weibull distribution scaled from 0 seconds up to 24 hours. For the parameters α and β used in Algorithm 1, we considered the values of 32 players and 120 seconds respectively, it having been demonstrated in [1], that they are appropriate values to ensure a good performance of the whole system. The GS of a Zone is over (able to try a reconnection to the MS) when 900 seconds have passed [3] since the Zone creation on average; following an exponential distribution.

The platform of the distributed area was considered with one core per node in the non heterogeneous case, while an even distribution of 2, 4 and 8 cores in each node has been considered when the *Heterogeneity_aware* mapping policy was applied.

According to the assumptions and functionalities established in the simulation process, we show in next Subsection, the performance provided by the *OnDeGaS* system according to the mapping mechanisms. The cases of the study are: ability to scale the distributed area determined by the number of created zones and the QoS of the system, measured by the zone's average latency and the waiting queue time.

3.1 Performance Evaluation



(a) Players in MS and Zones.

(b) Created Zones.

Figure 2: Study of the scalability performance of the mapping policies.

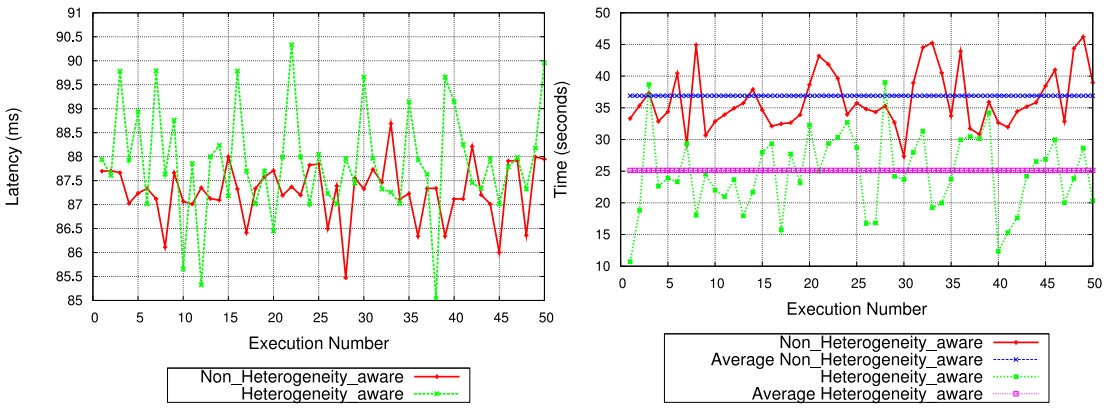
The scalability of the *OnDeGaS* system indicates its ability to manage more zones on demand while the latency values of the whole system are maintained under an acceptable threshold.

Figure 2a outlines, in logarithmic scale, the number of players served in the central and the distributed area. The line which starts at time 0 represents the concurrent MS players over time. When the MS reaches to its limit (2,000 concurrent players),

 ISBN 13: 978-84-613-5510-5

i.e. it is overloaded, no more players will be accepted, and new ones will be distributed to zones (top line). As can be observed, Figure 2a shows the huge difference between the number of concurrent players centralized in the MS and those playing distributed in zones. Note that this extra number of players would be rejected by the server of a traditional client-server architecture, thus reflecting the benefits of our proposal.

We studied the evolution in the number of created zones, for 50 different simulations, using both mapping policies. Figure 2b shows the average number of created zones. The top solid line corresponds to *Non_Heterogeneity_aware* mapping alternative with 1 core per node. The bottom dashed line represents the case of *Heterogeneity_aware* mapping policy where players can can have 2, 4 or 8 cores with the same proportion of each. As can be observed, the distribution performed by *Heterogeneity_aware* is able to exploit the additional cores of the heterogeneous system, as it creates a lower number of zones with more GSs in all cases. On average, the number of created zones falls from 1690 with the *Non_Heterogeneity_aware* policy to 400 with the other mapping policy.



(a) Latency in zones.

(b) Players time elapsed in the Waiting Queue.

Figure 3: QoS evaluation of mapping policies.

Figure 3 shows the impact of the two mapping mechanisms in the QoS of the system. Figure 3a shows the average latency in zones, in ms, with the same system assumptions that were evaluated in Figure 2b. As can be observed, latency values have similar average in both cases. However, they flow in a broader interval in the *Heterogeneity_aware* case. This is due to the fact that in this alternative fewer zones are created and so, the set of potential ZS to distribute players located in the WQ was fewer. However, it is worth remarking that in all the cases, for both mapping policies, latency values are below the maximum acceptable threshold for MMOFPS (180 ms).

Figure 3b shows the waiting time for players located in the WQ before they are distributed to zones. The experiment reveals a significant impact depending on the mapping policy. Whenever a new Zone is created, a new ZS and RZS must be searched for. This process takes some seconds, which can be considered constant (30 seconds on average). This situation happens continuously with *Non_Heterogeneity_aware* mapping

 ISBN 13: 978-84-613-5510-5

policy as indicated by the average of 36.9 ms. Nevertheless, this happens less often with the other mapping mechanism, as more players are mapped to zones already created, giving an average of 25.12 ms in this case.

Another aspect to consider is the influence in the communication overhead associated to the process of creating new zones of both mapping policies. There is no need to check the already created zones for *Non_Heterogeneity_aware*, then the communication cost between the MS and the rest of zones in that case is none. However, for *Heterogeneity_aware*, before creating a new Zone, Algorithm 3 checks in ZQ if there is any ZS and RZS with a free core to host a new GS in order to avoid a new Zone creation. Then, in the worst case it has to be managed a communication between the MS and the rest of zones existing in the distributed area before creating a new one.

To conclude, the proposed mapping mechanisms for MMOFPS game sessions are able both to provide a distributed platform that scales on demand that keeps latency values under an acceptable threshold. Regarding to differences, we have shown that the *Heterogeneity_aware* mechanism exploits better the resource capabilities of nodes and maps players quite faster, while the system is penalized by an increase in communications between the central and the distributed area and a latency variability.

4 Conclusion and Future Work

In this paper, we presented an hybrid system, called *OnDeGaS* (On Demand Game Service), that fits the scalability and latency requirements of MMOFPS networked games. The proposed new system is made up of a Master Server, carrying out centralized functionalities, and several zones that make up a distributed P2P network. Whenever the central server is overloaded, new Zones are created according to two different policies: *Non_Heterogeneity_aware* and *Heterogeneity_aware* policy. *Non_Heterogeneity_aware* mechanism selects a new distributed game server, among waiting players, taking exclusively latency criteria into account. On the other hand, the *Heterogeneity_aware* policy maps, whenever it is possible, the new game sessions into the available cores of the game servers already created.

By means of simulation, it has been demonstrated that the system is able to scale according to the demand. It has also been shown that this scalability does not damage the average latency, as it is always possible to achieve in distributed area an average latency below the maximum threshold allowed in MMOFPSs. Moreover, the player's waiting time is reduced if *Heterogeneity_aware* mapping policy is applied at the expense of increasing the communication costs between the central and the distributed area.

Future work is oriented towards modeling MMORPGs requirements and extending our hybrid architecture to them. Another important key would be to merge the current simulator with a network simulator, to study the QoS of the game, and test the network problems derived from an MMOG. Finally, the implementation of a prototype of the simulated architecture would be an important step for the deployment of this proposal.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work was supported by the MEyC-Spain under contract TIN2008-05913 and the CUR of DIUE of GENCAT and the European Social Fund.

References

[1] I. Barri, F. Giné, and C. Roig. A Scalable Hybrid P2P System for MMOFPS. In *Parallel, Distributed, and Network-Based Processing, Euromicro Conference on*, 2010.

[2] D. Bauer, S. Rooney, and P. Scotton. Network infrastructure for massively distributed games. In *NetGames '02: Proceedings of the 1st workshop on Network and system support for games*, 2002.

[3] A. Bharambe, J. Douceur, J. R. Lorch, T. Moscibroda, J. Pang, S. Seshan, and X. Zhuang. Donnybrook: Enabling large-scale, high-speed, peer-to-peer games. In *SIGCOMM '08: Proceedings of the 2008 conference on Applications, technologies, architectures, and protocols for computer communications*, 2008.

[4] T. S. Eugene and H. Zhang. Predicting internet network distance with coordinates-based approaches. In *INFOCOM*, 2001.

[5] G. Huang, M. Ye, and L. Cheng. Modeling system performance in mmorpg. In *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*, 2004.

[6] Research in China. China online games market report 2007-2008, 2008. http://www.researchinchina.com/Htmls/Report/2008/1944.html.

[7] J. Keller and G. Simon. Solipsis: A massively multi-participant virtual world. In *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA*, 2003.

[8] B. Knutsson, Honghui Lu, Wei Xu, and B. Hopkins. Peer-to-peer support for massively multiplayer games. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 2004.

[9] Huey-Ing Liu and Yun-Ting Lo. Dacap - a distributed anti-cheating peer to peer architecture for massive multiplayer on-line role playing game. In *CCGRID '08: Proceedings of the 2008 Eighth IEEE International Symposium on Cluster Computing and the Grid*, 2008.

[10] IBM Developers Works. Charming python: Simpy simplifies complex models (simulate discrete simultaneous events for fun and profit). 2002.

[11] M. Ye and L. Cheng. System-performance modeling for massively multiplayer online role-playing games. *IBM Syst. J.*, 2006.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A quasi-linear algorithm for calculating the infimal convolution of convex quadratic functions

L. Bayón[1], J.M. Grau[1], M.M. Ruiz[1] and P.M. Suárez[1]

[1] *Department of Mathematics, University of Oviedo,Spain*

emails: bayon@uniovi.es, grau@uniovi.es, mruiz@uniovi.es, pedrosr@uniovi.es

Abstract

In this paper we present an algorithm of quasi-linear complexity for exactly calculating the infimal convolution of convex quadratic functions. The algorithm exactly and simultaneously solves a separable uniparametric family of quadratic programming problems resulting from varying the equality constraint.

Key words: Algorithm Complexity, Infimal Convolution, Quadratic Programming
MSC 2000: 90C20, 90C25, 90C60

1 Introduction

The infimal convolution operator is well known within the context of convex analysis. For a survey of the properties of this operation, see [1].

Definition 1. *Let $F, G : \mathbb{R} \longrightarrow \bar{\mathbb{R}} := \mathbb{R} \cup \{+\infty, -\infty\}$ be two functions. We denote as the Infimal Convolution of F and G the operation defined as follows:*

$$(F \bigodot G)(x) := \inf_{y \in \mathbb{R}} \{F(x) + G(y - x)\}$$

Furthermore, if $A = \{1, ..., N\}$, we have that

$$(\bigodot_{i \in A} F_i)(\xi) = \inf_{\sum_{i \in A} x_i = \xi} \sum_{i \in A} F_i(x_i)$$

When the functions are considered to be constrained a certain domain, $Dom(F_i) = [m_i, M_i]$, the above definition continues to be valid by redefining $F_i(x) = +\infty$ if $x \notin Dom(F_i)$. In this case, the equivalent definition may be expressed as follows:

$$\Psi^A(\xi) := (\bigodot_{i \in A} F_i)(\xi) = \min_{\substack{\sum_{i \in A} x_i = \xi \\ m_i \leq x_i \leq M_i}} \sum_{i \in A} F_i(x_i)$$

 ISBN 13: 978-84-613-5510-5

This operator has a microeconomic interpretation that is quite precise: if Ψ^A is the infimal convolution of several production cost functions, $\Psi^A(\xi)$ represents the joint cost for a production level ξ when the latter is shared out among the different units in the most efficient way possible.

In this paper we present an algorithm that leads to the determination of the analytic optimal solution of a particular quadratic programming (QP) problem: Let $\{F_i\}_{i\in A}$ be a family of strictly convex quadratic functions:

$$F_i(x_i) = \alpha_i + \beta_i x_i + \gamma_i x_i^2$$

We denote by $\left\{\Pr^A(\xi)\right\}_{\xi\in\mathbb{R}}$ the family of separable convex QP Problems:

$$\begin{aligned} &\text{minimize:} && \sum_{i\in A} F_i(x_i) \\ &\text{subject to:} && \sum_{i\in A} x_i = \xi; \quad m_i \le x_i \le M_i, \forall i \in A \end{aligned}$$

QP problems have long been a subject of interest in the scientific community. Thousands of papers [2] have been published that deal with applying QP algorithms to diverse problems. Within this extremely wide-ranging field of research, some authors have sought the analytic solution for certain particular cases of QP problems with additional simplifications. For example, [3] presents an algorithm of linear complexity for the case of a single equality constraint (fixed ξ), only including constraints of the type $x_i \ge 0$. The present paper generalizes prior studies, presenting an algorithm of quasi-linear complexity, $O(N\ \log(N))$, for the family of problems $\left\{\Pr^A(\xi)\right\}_{\xi\in\mathbb{R}}$. This supposes a substantial improvement to a previous paper by the authors [4] in which an algorithm was presented that, as we shall show in this paper, is one of quadratic computational complexity, $O(N^2)$.

2 Algorithm

In this section, we first present the necessary definitions to build our algorithm.

Definition 2. *Let us consider in the set $A\times\{m,M\}$ the binary relation $\preccurlyeq$ defined as follows:*

$$\begin{aligned} (i,m) \preccurlyeq (j,m) &\Longleftrightarrow F_i'(m_i) < F_j'(m_j) \text{ or } (F_i'(m_i) = F_j'(m_j) \text{ and } i \le j) \\ (i,m) \preccurlyeq (j,M) &\Longleftrightarrow F_i'(m_i) < F_j'(M_j) \text{ or } (F_i'(m_i) = F_j'(M_j) \text{ and } i \le j) \\ (i,M) \preccurlyeq (j,m) &\Longleftrightarrow F_i'(M_i) < F_j'(m_j) \text{ or } (F_i'(m_i) = F_j'(M_j) \text{ and } i \le j) \\ (i,M) \preccurlyeq (j,M) &\Longleftrightarrow F_i'(M_i) < F_j'(M_j) \text{ or } (F_i'(m_i) = F_j'(M_j) \text{ and } i \le j) \end{aligned}$$

Definition 3. *We denote by g the isomorphism*

$$g(n) := (g_1(n), g_2(n)), \quad g : (\{1,2,\cdots,2N\},\le) \longrightarrow (A\times\{m,M\},\preccurlyeq)$$

which at each natural number $n \in \{1,2,\cdots,2N\}$ corresponds to the n-th element of $A\times\{m,M\}$ following the order established by $\preccurlyeq$.

 ISBN 13: 978-84-613-5510-5

We now present the optimization algorithm that leads to the determination of the optimal solution. The algorithm generates all the feasible states of activity/inactivity of the constraints on the solution of the problem. We build a sequence $(\Omega_n, \Theta_n, \Xi_n)$ starting with the triad $(A, \varnothing, \varnothing)$, which represents the fact that all the constraints on minimum are active and ending with the triad $(\varnothing, \varnothing, A)$, which represents the fact that all the constraints on maximum are active. Each step of the process consists in decreasing the number of active constraints on minimum by one unit or increasing the number of active constraints on maximum by one unit, following the order established by the relation $\preccurlyeq$. Let us consider the following recurrent sequence, $X_n := (\Omega_n, \Theta_n, \Xi_n)$, $n = 0, \ldots, 2N$:

$\Omega_0 = A$	$\Theta_0 = \varnothing$	$\Xi_0 = \varnothing$
If $g_2(n) = M : \Omega_n = \Omega_{n-1}$	$\Theta_n = \Theta_{n-1} - \{g_1(n)\}$	$\Xi_n = \Xi_{n-1} \cup \{g_1(n)\}$
If $g_2(n) = m : \Omega_n = \Omega_{n-1} - \{g_1(n)\}$	$\Theta_n = \Theta_{n-1} \cup \{g_1(n)\}$	$\Xi_n = \Xi_{n-1}$

We prove the following proposition.

Proposition 1. *The function Ψ^A (infimal convolution) is piecewise quadratic, continuous and, if $\Theta_n \neq \varnothing$, $\forall n, 0 < n < 2N$, then it also belongs to class C^1. Specifically, if $\phi_n \leq \xi \leq \phi_{n+1}$, with*

$$\phi_{n+1} = \phi_n + \frac{1}{2}\left[s_{n+1} - s_n\right]\frac{1}{\widehat{\gamma}_n}; \quad s_n = \begin{cases} F'_{g_1(n)}(m_{g_1(n)}) & \text{if} \quad g_2(n) = m \\ F'_{g_1(n)}(M_{g_1(n)}) & \text{if} \quad g_2(n) = M \end{cases}$$

we have

$$\Psi^A(\xi) = \widehat{\alpha}_n + \widehat{\beta}_n(\xi - \mu_n) + \widehat{\gamma}_n(\xi - \mu_n)^2$$

where

$$\mu_n = \begin{cases} \mu_{n-1} - m_{g_1(n)} & \text{if} \quad g_2(n) = m \\ \mu_{n-1} + M_{g_1(n)} & \text{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\alpha}_n = \begin{cases} \widehat{\alpha}_{n-1} + \alpha_{g_1(n)} - \frac{\left(\widehat{\beta}_{n-1} + \beta_{g_1(n)}\right)^2}{4\left(\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}\right)} - F_{g_1(n)}\left(m_{g_1(n)}\right) & \text{if} \quad g_2(n) = m \\ \widehat{\alpha}_{n-1} - \alpha_{g_1(n)} - \frac{\left(\widehat{\beta}_{n-1} - \beta_{g_1(n)}\right)^2}{4\left(\widehat{\gamma}_{n-1} - \gamma_{g_1(n)}\right)} - F_{g_1(n)}\left(M_{g_1(n)}\right) & \text{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\beta}_n = \begin{cases} \frac{1}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}}\left[\widehat{\beta}_{n-1} \cdot \gamma_{g_1(n)} + \beta_{g_1(n)} \cdot \widehat{\gamma}_n\right] & \text{if} \quad g_2(n) = m \\ \frac{1}{\widehat{\gamma}_{n-1} - \gamma_{g_1(n)}}\left[-\widehat{\beta}_{n-1} \cdot \gamma_{g_1(n)} + \beta_{g_1(n)} \cdot \widehat{\gamma}_n\right] & \text{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\gamma}_n = \begin{cases} \frac{\widehat{\gamma}_{n-1} \cdot \gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}} & \text{if} \quad g_2(n) = m \\ -\frac{\widehat{\gamma}_{n-1} \cdot \gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} - \gamma_{g_1(n)}} & \text{if} \quad g_2(n) = M \end{cases}$$

 ISBN 13: 978-84-613-5510-5

3 Computational Complexity of the Algorithm

In this section we analyze the complexity of the previous algorithm and compare it to the one presented in [4]. Given the family of strictly convex quadratic functions $F_i(x_i) = \alpha_i + \beta_i x_i + \gamma_i x_i^2$ with $i = 1.....N$ and $Dom(F_i) = [m_i, M_i]$, each one of these shall be represented by the list $\{m_i, M_i, \alpha_i, \beta_i, \gamma_i\}$. The union of all these functions constitutes the input for the algorithm:

$$\{\{m_1, M_1, \alpha_1, \beta_1, \gamma_1\}, \{m_2, M_2, \alpha_2, \beta_2, \gamma_2\}, \cdots, \{m_N, M_N, \alpha_N, \beta_N, \gamma_N\}\}$$

The output shall represents the infimal convolution, which we symbolize as:

$$\{\{\phi_1, \phi_2, \widehat{\alpha}_1, \widehat{\beta}_1, \widehat{\gamma}_1\}, \cdots, \{\phi_n, \phi_{n+1}, \widehat{\alpha}_n, \widehat{\beta}_n, \widehat{\gamma}_n\}, \cdots, \{\phi_{2N-1}, \phi_{2N}, \widehat{\alpha}_{2N}, \widehat{\beta}_{2N}, \widehat{\gamma}_{2N}\}\}$$

The algorithm presents the following phases:

A) Construction of the set $A \times \{m, M\}$.
B) Ordering of the set $A \times \{m, M\}$ following the ordering relation $\preccurlyeq$.
C) Construction of the recurrent sequence $X_n := (\Omega_n, \Theta_n, \Xi_n)$, $n = 0, \ldots, 2N$.
D) Construction of the sequence $s_n,\ n = 0, \ldots, 2N$.
E) Construction of the sequences $\widehat{\alpha}_n, \widehat{\beta}_n, \widehat{\gamma}_n,\ n = 1, \ldots, 2N-1$.
F) Construction of the sequences $\phi_n,\ n = 1, \ldots, 2N$.

We prove that:

Proposition 2. *The complexity of the aforementioned algorithm is quasi-linear:* $O(N \log(N))$, *and the complexity of the algorithm* [4] *is quadratic:* $O(N^2)$.

References

[1] T. Stromberg, *The operation of infimal convolution*, Diss. Math. **352** (1996).

[2] N. I. M. Gould, PH. L. Toint, *A Quadratic Programming Bibliography*, http://www.optimization-online.org/DB_HTML/2001/02/285.html, (2001).

[3] S. Cosares, D. S. Hochbaum, *Strongly polynomial algorithms for the quadratic transportation problem with a fixed number of sources*, Math. Oper. Res. **19**(**1**) (1994), 94-111.

[4] L. Bayon, J. M. Grau, M. M. Ruiz and P. M. Suarez, *An analytic solution for some separable convex quadratic programming problems with equality and inequality constraints*, Journal of Mathematical Inequalities, Preprint (2010).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Efficient Simulation of Scroll Wave Turbulence in a Three-Dimensional Reaction-Diffusion System

Youssef Belhamadia[1], André Fortin[2] and Yves Bourgault[3]

[1] *Campus Saint-Jean and Mathematical Department, University of Alberta, Edmonton Canada*

[2] *Mathematical Department , Laval University, Quebec, Canada*

[3] *Mathematical Department , University of Ottawa, Ottawa, Canada*

emails: youssef.belhamadia@ualberta.ca, afortin@giref.ulaval.ca, ybourg@uottawa.ca

Abstract

In this work, an efficient numerical method based on an adaptive finite element technique is presented for simulating three-dimensional scroll waves turbulence in cardiac tissue. The proposed numerical method enhances the accuracy of the prediction of the electrical wave fronts. Illustrations of the performance of the proposed method are presented using three-dimensional re-entrant waves.

Key words: Monodomain model, finite element method, anisotropic mesh adaptation, FitzHugh-Nagumo model.

1 Introduction

Scroll wave turbulence in cardiac tissue is known as fibrillation and implies cardiac failure. From mathematical point of view, fibrillations can be represented by the monodomain model with an appropriate ionic model. The monodomain model is a reaction-diffusion system and consists of a nonlinear partial differential equation for the transmembrane potential coupled with an ordinary differential equation for the recovery variable. This model is computationally very expensive and is known to require extremely fine meshes (Bourgault et al. [9]).

To overcome these difficulties, many methods have been developed in the literature. This includes parallel computing techniques with a fixed spatial mesh (Colli Franzone and Pavarino [10]), fully and semi implicit time-stepping descritizations (Bourgault and coauthors [9] and[13]) and operator-splitting methods (Lines et al. [16, 17]).

 ISBN 13: 978-84-613-5510-5

In our work [5, 4, 7, 6], two- and three-dimensional mesh adaptation methods have been introduced to capture transmembrane potential fronts using the monodomain and bidomain model. The technique consists in locating finer mesh cells near the front position while a coarser mesh is used away from the front. In this work, a three dimensional adaptive algorithm is presented for accurately computing time-evolving scroll wave. Although the scroll wave is in a noncoherent (turbulent) state, the method proposed reduces the computational grid size, and concentrates the elements near the depolarization and repolarization front positions, which leads to an efficient solutions.

This paper is organized as follows. Next section is devoted to the monodomain model and adaptive mesh technique, and section 3 presents three-dimensional numerical results showing the accuracy of the proposed method.

2 Mathematical Models and Adaptive Mesh Technique

The monodomain model used in this work takes the following form:

$$\begin{cases} \dfrac{\partial U}{\partial t} = & \nabla \cdot (D \nabla U) + I_{\text{ion}}(U,V) + I_s, \\ \dfrac{\partial V}{\partial t} = & G(U,V). \end{cases} \tag{1}$$

Where U describes the transmembrane potential and V describes the recovery variable. I_s is the current due to an external stimulus, and the nonlinear terms $I_{\text{ion}}(U,V)$ and $G(U,V)$ depend on the ionic model. In this work, a modified version of a piecewise linearized FitzHugh-Nagumo model is used [2]:

$$I_{\text{ion}} = kU(1-U)\left(U - \frac{V+b}{a}\right) \quad \text{and} \quad G(U,V) = g(U) - V.$$

where

$$g(U) = \begin{cases} 0 & \text{if } U < \frac{1}{3} \\ 1 - 6.75U(U-1)^2, & \text{if } \frac{1}{3} \leq U \leq 1 \\ 1 & \text{si } U > 1. \end{cases}$$

A finite element method is used to solve the nonlinear system of equation (1). The variational formulation of this system is straightforward and obtained by multiplying this system by test functions (ϕ, ψ) and integrating by parts the second order terms.

In all our numerical simulations, a quadratic (P_2) finite elements and a fully implicit backward second order scheme are employed for the spatial and time discretizations, respectively. For more details about a comparison between different time-stepping

 ISBN 13: 978-84-613-5510-5

schemes and spatial dicretizations, the reader is refereed to Belhamadia [5].

An adaptive time-dependent meshing algorithm for accurately simulating moving three-dimensional transmembrane potential is now presented. The adaptive strategy employed in this work uses an error estimators based on a definition of edge lengths using a solution dependent metric (see Habashi and coauthors [14, 1, 11] and Hecht and Mohammadi [15],and Belhamadia et al. [8]).

The overall adaptive strategy is the following:

1. Start from the solutions $U^{(n-1)}$, $U^{(n)}$, $V^{(n-1)}$ and $V^{(n)}$ and a mesh $\mathcal{M}^{(n)}$ at time $t^{(n)}$;
2. Solve the system (1) on mesh $\mathcal{M}^{(n)}$ to obtain a first approximation of the solutions (denoted $\tilde{U}^{(n+1)}$ and $\tilde{V}^{(n+1)}$) at time $t^{(n+1)}$;
3. Adapt the mesh starting from the mesh $\mathcal{M}^{(n)}$ and the solution dependent metric calculated with the solution time-variations
$$\frac{\tilde{U}^{(n+1)} + U^{(n)} + U^{(n-1)}}{3} \quad \text{and} \quad \frac{\tilde{V}^{(n+1)} + V^{(n)} + V^{(n-1)}}{3}$$
to obtain a new mesh $\mathcal{M}^{(n+1)}$;
4. Reinterpolate $U^{(n-1)}$, $U^{(n)}$, $V^{(n-1)}$ and $V^{(n)}$ on mesh $\mathcal{M}^{(n+1)}$;
5. Solve the system (1) on mesh $\mathcal{M}^{(n+1)}$ for U^{n+1} and V^{n+1}.
6. Next time step: go to step 2.

The step 3 depends on time discretizations scheme. In this work, a fully implicit backward second order scheme for time-stepping is employed. Thus, the mesh is required to represent the solutions at time $t^{(n-1)}$, $t^{(n)}$, and $t^{(n+1)}$.

3 Numerical Results

In this section, the performance of the adaptive method is presented using a three dimensional scroll wave turbulence. The computational domain is the cube $[0, 60] \times [0, 60] \times [0, 60]$. Homogeneous Neumann conditions are imposed on all sides, and the initial condition is a scroll wave as presented in figure 1. This was obtained with the technique described in Ezscroll software by Barkley et al. [3, 12] and using the following parameters:

 ISBN 13: 978-84-613-5510-5

$a = 0.84$	$b = 0.08$
$\epsilon = 0.07$	$D = 1$
$\Delta t = 0.1$	

The scroll wave began to break up after some transient rotations. This wave travels across the whole computational domain, calling for grids that are uniformly fine. Indeed, coarse grids lead to wrong propagation speed and wave trajectories. The adaptive technique presented in this work reduces the total number of element since the mesh is refined only in the vicinity of the front position while keeping sufficient resolution in other regions.

Figure 2 presents the adapted meshes and the transmembrane potential at different times. As can be seen, the adapted mesh evolves with time and, at each time step, elongated elements are obtained at the appropriate position to capture the depolarization and repolarization fronts, in spite of the fact that these fronts correspond to sharp gradients of the transmembrane potential.

4 Conclusions

An efficient numerical method for the scroll wave turbulence in a three-dimensional reaction-diffusion system was presented. The accuracy of the numerical solutions was obtained by using an anisotropic time-dependent adaptive method. It will be interesting to compare the adaptive method with regular meshes and also to see how the method performs with a realistic heart geometry in case of scroll wave turbulence.

Acknowledgements

The author wish to acknowledge the financial support of NSERC.

References

[1] D. Ait Ali Yahia, G. Baruzzi, W. G. Habashi, M. Fortin, J. Dompierre, and M.-G. Vallet. Anisotropic Mesh Adaptation: Towards User-Independent, Mesh-Independent and Solver-Independent CFD. Part II: Structured Grids. *Int. J. Numer. Meth. Fluids*, 39:657–673, 2002.

[2] M. Bar and Eiswirth M. Turbulence Due to Spiral Breakup is a Continuous Excitable Medium. *Physical Review E*, 48(3):1635–1638, 1993.

[3] D. Barkley. A Model for Fast Computer Simulation of Waves in Excitable Media. *Physica D*, 49:61–70, 1991.

 ISBN 13: 978-84-613-5510-5

[4] Y. Belhamadia. An Efficient Computational Method for Simulation of Electrophysiological Waves. *Conf Proc IEEE Eng Med Biol Soc.*, pages 5922–5925, 2008.

[5] Y. Belhamadia. A Time-Dependent Adaptive Remeshing for Electrical Waves of the Heart. *IEEE Transactions on Biomedical Engineering*, 55(2, Part-1):443–452, 2008.

[6] Y. Belhamadia, Y. Bourgault, and A. Fortin. Mathematical Simulation of the Electrical Activity of the Heart. *32nd Conference proceeding of the Canadian Medical and Biological Engineering Society*, 2009.

[7] Y. Belhamadia, A. Fortin, and Y. Bourgault. An Accurate Numerical Method for Monodomain Equations Using a Realistic Heart Geometry. *Mathematical Biosciences*, 220(2):89–101, 2009.

[8] Y. Belhamadia, A. Fortin, and É. Chamberland. Three-Dimensional Anisotropic Mesh Adaptation for Phase Change Problems. *Journal of Computational Physics*, 201(2):753–770, 2004.

[9] Y. Bourgault, M. Ethier, and V.G. LeBlanc. Simulation of Electrophysiological Waves With an Unstructured Finite Element Method. *Mathematical Modelling and Numerical Analysis*, 37(4):649–662, 2003.

[10] P. Colli Franzone and L. F. Pavarino. A Parallel Solver for Reaction-Diffusion Systems in Computational Electrocardiology. *Math. Models and Methods in Applied Sciences*, 14(6):883–911, 2004.

[11] J. Dompierre, M.-G. Vallet, Y. Bourgault, M.. Fortin, and W. G. Habashi. Anisotropic Mesh Adaptation: Towards User-Independent, Mesh-Independent and Solver-Independent CFD. Part III: Unstructured Meshes. *Int. J. Numer. Meth. Fluids*, 39:675–702, 2002.

[12] M. Dowle, R. M. Mantel, and D. Barkley. Fast Simulations of Waves in Three-Dimensional Excitable Media. *Int. J. Bif. Chaos*, 7(11), 1997.

[13] M. Ethier and Y. Bourgault. Semi-implicit time-discretization schemes for the bidomain model. *SIAM Journal of Numerical Analysis*, 46(5):2443–2468, 2008.

[14] W. G. Habashi, J. Dompierre, Y. Bourgault, D. Ait Ali Yahia, M. Fortin, and M.-G. Vallet. Anisotropic Mesh Adaptation: Towards User-Independent, Mesh-Independent and Solver-Independent CFD. Part I: General Principles. *Int. J. Numer. Meth. Fluids*, 32:725–744, 2000.

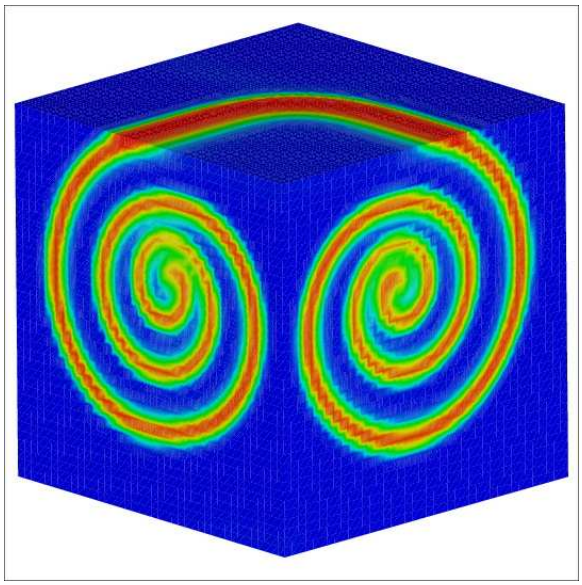
[15] F. Hecht and B. Mohammadi. Mesh Adaptation by Metric Control for Multi-Scale Phenomena and Turbulence. *AIAA*, 97–0859, 1997.

[16] G.T. Lines, M.L. Buist, P. Grottum, A.J. Pullan, J. Sundnes, and A. Tveito. Mathematical models and numerical methods for the forward problem in cardiac electrophysiology. *Comput. Visual. Sc.*, 5:215–239, 2003.

 ISBN 13: 978-84-613-5510-5

[17] G.T. Lines, P. Grottum, and A. Tveito. Modeling the electrical activity of the heart: A bidomain model of the ventricles embedded in a torso. *Comput. Visual. Sc.*, 5:195–213, 2003.

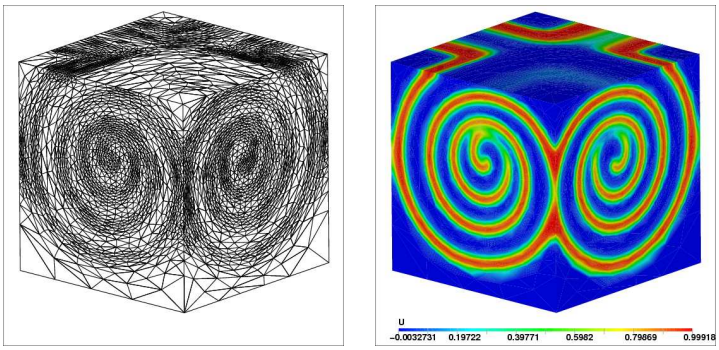
 ISBN 13: 978-84-613-5510-5



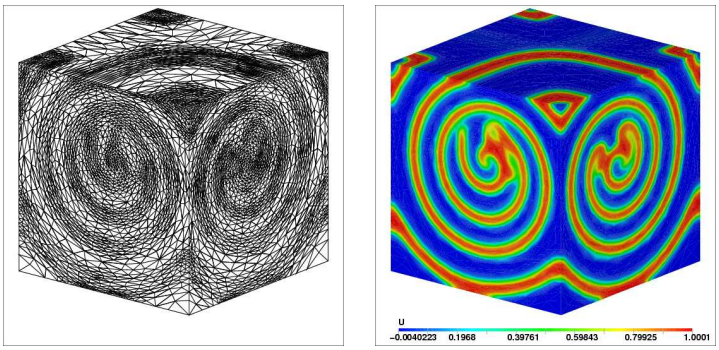
b) Initial condition

Figure 1: Initial Scroll Wave

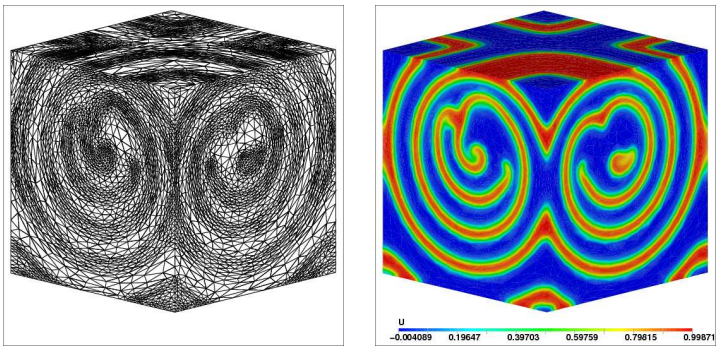
 ISBN 13: 978-84-613-5510-5



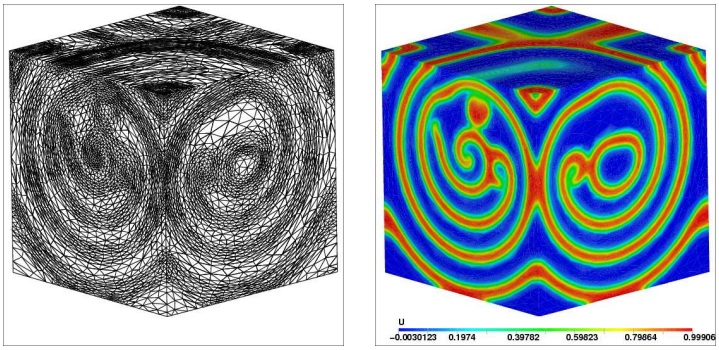
a) $t = 3$ b) $t = 3$



c) $t = 6$ d) $t = 6$



a) $t = 9$ b) $t = 9$



c) $t = 12$ d) $t = 12$

Figure 2: Time evolution of adapted meshes and transmembrane potential

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Multichannel acoustic signal processing on GPU

Jose A. Belloch[1], Antonio M. Vidal[2], Francisco J. Martínez-Zaldívar[3] and Alberto Gonzalez[3]

[1] *Audio and Communications Signal Processing Group. Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universidad Politécnica de Valencia (Spain)*

[2] *Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia (Spain)*

[3] *Departamento de Comunicaciones, Universidad Politécnica de Valencia (Spain)*

emails: jobelrod@iteam.upv.es, avidal@dsic.upv.es, fjmartin@dcom.upv.es, agonzal@dcom.upv.es

Abstract

Massive convolution is the basic operation in multichannel acoustic signal processing. Dealing with multichannel signals takes a big computational cost requiring the use of multiple resources from the CPU. Graphical Processor Units (GPU), a high parallel commodity programmable co-processors, can carry out a multichannel convolution faster. However, the fact of transferring data from/to the CPU to/from the GPU prevents to carry out a real-time application. In this paper, an algorithm with a pipeline structure is developed, what allows to perform a massive real-time convolution.

Key words: Massive convolution, Multichannel audio processing, FFT, GPU, CUDA

1 Introduction

Multichannel acoustic signal processing has experienced a great development in recent years, due to an increase in the number of sound sources used in playback applications available to users, and the growing need to incorporate new effects and to improve the experience of hearing [1].

Several effects, as the synthesis of 3D sound, are achieved through multichannel signal processing, with an efficient implementation of the massive convolution. It consists of carrying out different convolutions of different channels in a parallel way. All these operations require high computing capacity.

 ISBN 13: 978-84-613-5510-5

GPU offer us the possibility of parallelizing these operations, letting us not only to obtain the result of the processing in much less time, but also to free up CPU resources.

The paper is organized as follows. Section 2 describes the convolution and the problem of its implementation on GPU. In Section 3, an efficient GPU implementation of massive convolution is presented. Section 4 is reserved for the results of different tests on GPU. Finally Sections 5 is devoted to the conclusions, and the paper closes with some references.

2 Convolution on GPU

2.1 Convolution Algorithm in GPU

The convolution describes the behavior of a linear, time-invariant discrete-time system with input signal x and output signal y [2]:

$$y[n] = \sum_{j=0}^{N-1} x[j]h[n-j], \tag{1}$$

Signal x will the input to the system, in our case, samples from audio signal. The known signal h is the response of the system to a unit-pulse input. The output signal y contains the samples of the desired acoustic effects. N, M and $L = N + M - 1$ will be the lenghts of x, h and y respectively.

Convolution theorem [2] states that if x and h are padded with zeros to the length L, then the Discrete Fourier Transform of y is the point-wise product of the Discrete Fourier Transforms of x and h. In other words, convolution in one domain (e.g., time domain) equals point-wise multiplication in the other domain (e.g., frequency domain). This way of computing the convolution is advantageous because the number of operations is smaller than implementing the convolution in the time domain.

There exist different libraries that implement efficient FFT algorithms. They allow to obtain the Discrete Fourier Transform of a signal, in a CPU (like MKL [7] or IPP [8]) or in a GPU (like CUFFT [5] from NVIDIA whose performances have been analized in [9]).

The use of a GPU may offer two benefits: less execution time due to a high level of parallelization of the computations and the freeing up resources of the CPU.

Let us consider x and input audio signal, h an acoustic filter and y the desired output audio signal of our system. The execution of the convolution using a GPU can be enumerated in the next steps: first, the lenghts of x and h must be checked; then both signals must be transferred from the CPU to the GPU; next, the FFT (from CUFFT) is applied to each signal obtaining X and H; the frequency domain output Y is obtained multiplying point-wise X and H; the time domain output y is obtained applying the IFFT to Y; Finally, y is transferred from the GPU to the CPU. Figure 1 shows this process.

We can observe that:

1. Long time of the algorithm is spent in transfers between the CPU and the GPU.

 ISBN 13: 978-84-613-5510-5

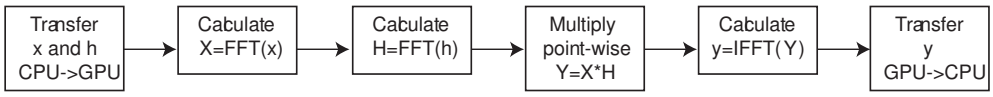


Figure 1: Steps in order to calculate convolution of signals x and h on GPU.

2. Signals must be sent to the GPU before beginning the operations, and the whole output signal must be received at the CPU to be reproduced.

In spite of the parallelism in operations that offered by the GPU, the transfer time penalty prevents us to carry out a real time application in a GPU. More even, if the signal x is compound by several channels, then a multiple convolutions would be required. On the other hand, if a CPU is used to make a massive convolution, all our resources would be used and no more applications could be run at the same time.

2.2 Convolution of Large Signals

In a real-time environment, the length of signal x can not be known a priori. There exist techniques that allow us to cut the signal in chunks, and from the convolution of each chunk we can obtain the convolution of the whole signal. One of these techniques is called overlap-save [3] and it consists of:

1. Chunks of L samples are taken, where L will be either the next power of two, bigger than M (length of h) or 512.

2. In the first chunk, the first $M-1$ samples will be padded with zeros.

3. From the second and following chunks, the first $M-1$ samples will be duplicated from the last $M-1$ samples of the previous chunk.

4. Following the steps of the previous subsection, $y_0[n]$, $y_1[n]$, $y_2[n]$, ..., are obtained as the result of the convolution of $x_0[n]$, $x_1[n]$, $x_2[n]$, ..., with h respectively.

5. From each chunk result, the first $M-1$ samples will not be valid values so they will be eliminated.

3 Pipelined Algorithm of convolution on GPU

Recently, the new CUDA toolkit 3.0 [4] lets use CUFFT [5] with the property *concurrent copy and execution*. Therefore, the latency of transferring data from the CPU to the GPU and vice versa can be overlapped by computations. That fits perfectly with the steps described in the previous section.

In fact, in order to maximize the overlapping of the computations in the GPU and the communications between CPU and GPU, a matrix can be configured with each chunk obtained with the signal samples.

 ISBN 13: 978-84-613-5510-5

In this matrix, the first $M-1$ values of one row will coincide with the last $M-1$ values of the previous one, except the first configured matrix at the start of the algorithm whose first $M-1$ values from the first row will be zeros. This matrix will have the following shape with R rows and L columns:

The last $M-1$ samples from the last row of the matrix will be kept in an internal buffer in order to occupy the first $M-1$ positions of the next matrix to be filled.

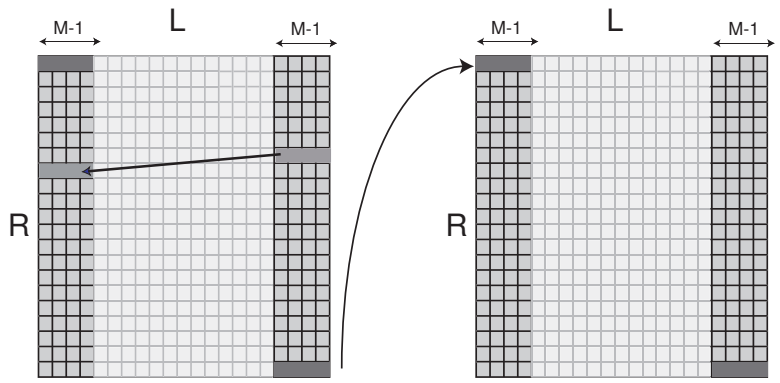


Figure 2: Samples are sent to GPU in a chunks-matrix- configuration. $matrix_{i-1}$ (left side) shares a $M-1$ samples with, previously sent to $matrix_i$ (right side).

The *concurrent copy and execution* property lets sending the $matrix_i$ using the asynchronous transfer while carrying out the other tasks in parallel:

1. Beginning to configure the next matrix $matrix_{i+1}$ with new samples.
2. Execution of the convolution algorithm at the GPU with the matrix which was previously sent $matrix_{i-1}$. So, R chunk-convolutions (the matrix sent to the GPU has R rows) will be executed in a parallel way.
3. Chunk-convolutions results from $matrix_{i-2}$ will be sent back from GPU to the CPU

The unit-impulse response h will have been sent to the GPU before sending the first matrix. h will be kept in the GPU memory and reused over and over with all the convolutions.

All the previous tasks can be viewed in a pipeline configuration as shown in Figure 3.

3.1 Extrapolation to a multichannel signal: Massive Convolution

Dealing with a multichannel-signal will be totally scalable due to an equal distribution of the resources. So, the matrix that contains the chunks will be divided in the number of channels of the signal.

In the same way if more than one effect is going to be applied, each of the impulse responses would be sent and kept in the GPU.

 ISBN 13: 978-84-613-5510-5

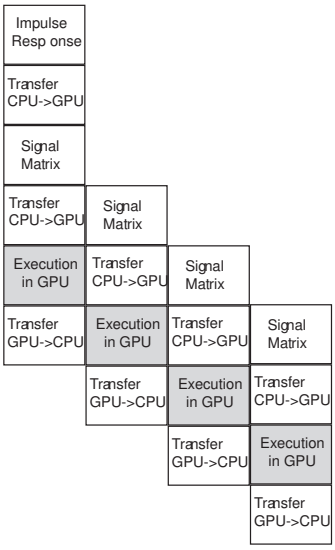


Figure 3: Pipeline Configuration.

The parallel architecture of the GPU give us freedom to configure several possibilities such as: Apply the same effect to all the channel signals. Apply one specific effect to one channel and other effect to the rest. Even, apply one effect to a determinate number of channels.

So, all the combinations are possible, and therefore, the possibilities of mixing several acoustic effects, as well.

4 Results

Many are the tests that are being carried out in order to know the achievement of the massive convolution. One of the most significative resolves around the comparison between the convolution algorithm in GPU, shown at Figure 1 (implemented, for example, in [6]), and the pipeline algorithm. In this case, as it can be shown in table 2, the second achieves the convolution of the signal in half of the time than the first one.

Test has been carried out on a signal x and a impulse-response h compound by 176400 samples and 220 coefficients respectively. Results are shown in Table 1. A time comparison with the basic convolution algorithm is shown in Table 2.

As it can be appreciated, performance are improved if an algorithm in convolution with a pipeline configuration is used.

5 Conclusions

With this article, it has been revealed that GPU can be used for carrying out a massive convolution of multichannel-acoustic signals in real-time. It has been possible thanks to the pipeline configuration that is now available with the new CUDA Toolkit 3.0. It

 ISBN 13: 978-84-613-5510-5

must be pointed that one of the advantages of using a GPU, is the fact of freeing up CPU resources, letting us to run more applications in the CPU.

R//L	**512**	**1024**	**2048**	**4096**	**8192**
32	625.92	833.28	802.83	836.83	870.70
64	730.80	745.64	809.62	890.21	876.21
128	761.43	819.39	865.55	906.63	981.27
256	870.89	913.35	1094.1	995.02	1111.4
512	937.24	951.26	949.20	994.04	1206.3
1024	1005.2	110.51	1080.4	1278.7	1622.8
2048	1089.1	1274.8	1436.7	1603.1	1969.1

Table 1: Time in miliseconds of the pipelined algorithm varying number of rows (R) and columns (C) of the matrix.

Type of Algorithm	**Time**
Convolution Algorithm in GPU	1330ms
Configuration Pipeline (Best Performance)	802.83ms

Table 2: Comparison between basic and pipelined algorithm.

Acknowledgements

This work was financially supported by the Spanish Ministerio de Ciencia e Innovacion(Projects TIN2008-06570-C04-02 and TEC2009-13741), Universidad Politecnica de Valencia through "Programa de Apoyo a la Investigacion y Desarrollo (PAID-05-09)", Regional Government Generalitat Valenciana through grant PROMETEO/2009/013 and NVIDIA through CUDA Community program.

References

[1] E. Torick, *Highlights in the history of multichannel sound*, J. Audio. Eng. Soc. **46** (1998) 27–31.

[2] S. S. Soliman, D. S.Mandyam and M. D. Srinath, *Trade paperback, Prentice Hall*, ISBN:0135184738, 1997.

[3] Overlap-save method: "http://en.wikipedia.org/wiki/Overlap-save-method"

[4] CUDA Toolkit 3.0: "http://developer.nvidia.com/object/cuda-3-0-downloads.html"

 ISBN 13: 978-84-613-5510-5

[5] CUFFT library: "http://developer.download.nvidia.com/compute/cuda/3-0/toolkit/docs/CUFFT-Library-3.0.pdf"

[6] GPU Computing SDK code samples: "http://developer.nvidia.com/object/cuda-3-0-downloads.html"

[7] MKL library: "http://software.intel.com/en-us/intel-mkl/"

[8] MKL library: "http://software.intel.com/en-us/intel-ipp/"

[9] P. Alonso, J. A. Belloch, A. Gonzalez, E. S. Quintana-Orti, A. Remon and A. M. Vidal, *Evaluacion de bibliotecas de altas prestaciones para el calculo de la FFT en procesadores multinúcleo y GPUs*, II Workshop en Aplicaciones de Nuevas Arquitecturas de Consumo y Altas Prestaciones, Freeman, Mostoles (Madrid), 2009.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Ecoepidemic models with identifiable infectives I: disease in the prey

Sabrina Belvisi[1], Nicola Tomatis[1] and Ezio Venturino[1]

[1] *Dipartimento di Matematica "Giuseppe Peano", Università di Torino, via Carlo Alberto 10, 10123 Torino*

emails: sabrinabelvisi@libero.it, nico.tomatis@hotmail.it, ezio.venturino@unito.it

Abstract

In this paper we consider an ecoepidemic model with disease in the prey, in which the disease-carriers are identifiable by other individuals of their own population. Therefore the contact with infectious can be avoided and thereby the disease incidence decreased. We model the situation and investigate the long term behavior of the system, showing that bifurcations leading to sustained limit cycles may occur.

Key words: Ecoepidemics, identifiable disease-carriers, Capasso-Serio epidemic model
MSC 2000: AMS codes (92D25, 92D30, 92D40)

1 Introduction

Population theory is a branch of mathematical biology dealing with the study of interacting populations. From the early classical models on single populations, predator-prey models have been developed in the second decade of the past century. From the latter, many investigations followed, extending also to other types of interactions, like competition and commensalism.

Epidemiology investigates the spreading of infectious diseases in populations, with the goal of fighting and possibly eradicating them. The role of mathematical modeling of epidemics in this context appears to be fundamental, as the infectious individuals, i.e. those that are able to propagate by contact the infection, are not usually recognizable. However, the human behavior is in general different. In fact, when an epidemic spreads, people tend to take measures in order not to be infected. This has been remarked and used as the basis for proposing a model, now well known in the literature, [2].

Ecoepidemiology is a rather recent subject of investigation, merging the epidemiological features with those of interacting systems of populations. The issues tackled

 ISBN 13: 978-84-613-5510-5

are important, since diseases are present in the real world, and therefore influence environments in which clearly more than just one species is present. For a summary of some of the earlier results in this relatively new field of study, see Chapter 7 of [5].

In this paper, we address the problem of how to modify the classical ecoepidemic models when the disease carriers are identifiable by the other individuals in the population, and therefore avoided in order not to catch the disease.

2 Model formulation

We consider here a predator-prey model in which prey can catch an infectious disease, which spreads by contact among an infectious and a susceptible. The disease is unrecoverable, i.e. once contracted, the infected individual carries it for its lifespan. We denote by S the healthy prey, by I the infected prey and by P the predators. The differential equations describing this ecosystem are:

$$\begin{aligned} \frac{dS}{dt} &= rS\left(1-\frac{S}{K}\right)-\gamma\frac{IS}{A+I^2}-aSP, \\ \frac{dI}{dt} &= \gamma\frac{IS}{A+I^2}-\mu I-bIP, \\ \frac{dP}{dt} &= -mP+eaSP+ebIP. \end{aligned} \tag{1}$$

The model is characterized by the interactions among the three populations in the environment under consideration. To capture their meaning, we focus at first on the interpretation of the system's parameters.

The meaning of the parameters of the model are as follows: K is the environment carrying capacity for the prey; r represents the healthy prey's reproduction rate; a is the predation rate on healthy prey, while b is the one upon sick prey; m is the predators' mortality, μ the sick prey's mortality; γ denotes the disease incidence rate; e is the conversion factor, a pure number, i.e. dimensionless, beetween zero to one; A has a role similar to the half saturation constant in the Holling type II model.

We now describe each equation in the model (1). The first term of the first equation represents the logistic growth of the prey. The second one in the first equation represents the disease incidence term, i.e. it counts all the healthy prey which become sick upon contact with an infected one and leave their class. The new characteristic of the ecoepidemic model being introduced here is exactly this term. We are indeed making the assumption that for the particular disease in consideration the disease carriers are in fact recognizable. In fact, the functional response of sound prey is to try to avoid contact with an infectious individual when the disease is widespread i.e. there are many infectious around. Notice in fact that as $I \to \infty$, for the functional response we have

$$\lim_{I\to\infty}\frac{I}{A+I^2}=0.$$

The third term represents instead the reduction of the prey number due to predation. As for the second equation the first term is once again the disease incidence, this time

 ISBN 13: 978-84-613-5510-5

accounting for the susceptibles that become new infected. The second one represents the natural plus disease-related prey mortality, and the last one describes losses due to predation. Note that the predation rate is here different from the one related to be sound prey. In the third equation the first term is the predators' mortality; then there are two growth terms, accounting for gains due to predation of sound and of infected prey respectively resulting in new individuals. Remark also that in this model the infected prey do not reproduce, nor do they contribute to intraspecific competition nor there is vertical transmission of the disease, i.e. newborns are always born sound.

3 Equilibria

We are now interested in finding the equilibrium points $E_k \equiv (S_k, I_k, P_k)$, i.e. the points to which the system tends as time flows. Easily, the origin $E_0 = (0, 0, 0)$ is an equilibrium point; the other boundary ones are

$$E_1 = (K, 0, 0)\,, \quad E_2 = \left(\frac{m}{ea}, 0, \frac{r}{a}\left(1 - \frac{m}{eaK}\right)\right), \quad E_3 = \left(\frac{\mu}{\gamma}(A + T_3^2), T_3, 0\right),$$

and then there is the coexistence equilibrium

$$E_4 = \left(\frac{m - ebT_4}{ea}, T_4, \frac{\gamma m + \gamma ebT_4 + \mu Aea + \mu T_4^2 ea}{bea(A + T_4^2)}\right).$$

In the above expressions, T_3 denotes the solution of the following quartic equation

$$r\mu x^4 + (2r\mu A - rK\gamma)x^2 + \gamma^2 Kx - rKA\gamma + r\mu A^2 = 0$$

and T_4 represents instead a root of the following cubic equation

$$reb^2x^3 + (rKeab + a^2K\mu e - rmb)x^2 + rAeb^2x - aK\gamma m + a^2K\mu Ae + rKAeab - rAmb = 0.$$

The feasibility condition for E_2 is as follows

$$m < eaK, \tag{2}$$

and one for the interior equilibrium E_4 places an upper bound on the location of the positive root, namely

$$\frac{m}{eb} \geq T_4. \tag{3}$$

A condition ensuring a positive root for the quartic is

$$\mu A < K\gamma. \tag{4}$$

In fact, we do not have the closed form expression for the point E_3, but it is possible to determine some necessary conditions for existence. From the equations of the system (1), setting $P = 0$ we get the equations:

$$S_1 = \frac{K}{r}\left(r - \frac{\gamma I}{A + I^2}\right) \quad S_2 = \frac{\mu}{\gamma}\left(A + I^2\right)$$

 ISBN 13: 978-84-613-5510-5

From the graphs of the two functions it is possible to obtain the existence condition (4) as indicated above.

A condition instead giving a positive root for the cubic is given by

$$a^2K\mu Ae + rKAeab < rAmb + aK\gamma m.$$

Here again, we do not have the closed form expression for the point E_4, but it is possible to determine some necessary conditions for existence as well. From the equations of the system (1), we find

$$S_3 = K\frac{(\mu a + rb)(A + I^2) + \gamma bI}{aK\gamma + rb(A + I^2)}, \quad S_4 = -\frac{b}{e}I + \frac{m}{ea}$$

and by graphing these functions, we get the existence and feasibility condition:

$$S_5 < \frac{m}{ea}, \quad S_5 = \frac{KA(\mu a + rb)}{aK\gamma + rbA}.$$

4 Stability analysis of equilibria.

The general form of the Jacobian matrix of the system (1) at the generic point (S, I, P) is the following

$$\begin{pmatrix} r - \frac{2rS}{K} - \frac{\gamma I}{A+I^2} - aP & \frac{\gamma S(I^2-A)}{(A+I^2)^2} & -aS \\ \frac{\gamma I}{A+I^2} & \frac{\gamma S(A-I^2)}{(A+I^2)^2} - \mu - bP & -bI \\ eaP & ebP & -m + eaS + ebI \end{pmatrix}$$

We begin by considering the origin. One of the eigenvalues of this Jacobian matrix, r, is positive, the other ones are $-\mu$ and $-m$. Hence E_0 is an unstable equilibrium.
We then consider E_1: the stability conditions of this equilibrium are

$$m > eaK, \quad A\mu > \gamma K. \tag{5}$$

The characteristic equation for E_2 factors, to give explicitly an eigenvalue

$$\frac{\gamma S_2}{A} - \mu - bP_2,$$

and then the quadratic equations

$$\lambda^2 Kea + rm\lambda + rmeaK - rm^2 = 0. \tag{6}$$

For such equation, the Routh Hurwitz conditions ensure two negative roots if and only if this condition is satisfied

$$eaK > m \tag{7}$$

Similarly from the explicit first eigenvalue we obtain

$$\mu + \frac{br}{a} > \frac{\gamma m}{aeA} + \frac{brm}{a^2eK}. \tag{8}$$

We do not have the closed form expression for the points E_3 and E_4, so that it is not possible to perform the stability analysis theoretically, to this end we will have to use numerical simulations.

 ISBN 13: 978-84-613-5510-5

5 Biological significance of the results

The instability of equilibrium E_0 means that even the presence of a small number of predators or prey makes the system evolve, toward population values that will never all be zero at the same time. The equilibrium E_1 corresponds to the situation in which predators and sick prey vanish, i.e. only sound prey survive in the system at the environment's carrying capacity K. It is attained if and only if conditions (5) hold, i.e. the mortality of predators must be higher than a quantity that depends on the product of the predator's hunting rate on sound prey a by the conversion factor and the carrying capacity, e and K. A similar condition holds for the infected. If their combined, natural plus disease-related mortality μ exceeds a quantity that depends on the disease incidence coefficient γ, scaled via the constant A, as well as once again on the the sustem's prey carrying capacity, K. Combining these result we can claim that the sound-prey-only equilibrium is attained if the reduced mortality rates of infected prey and predators do not exceed the carrying capacity of the environment, where "reduced" means the ratio of each mortality respectively by the assimilation term due to hunting and the disease incidence, namely

$$K > \max\left\{\frac{m}{ea}, A\frac{\mu}{\gamma}\right\}.$$

The disease-free equilibrium point E_2 is feasible if (2) holds, i.e. the opposite of that one of stability of previous point, and (8). The requests seem reasonable: the mortality of predators should be limited, while that of diseased prey must exceed a threshold value. Note that E_1 is stable if and only if E_2 is infeasible.

Equilibrium E_3 describes the situation where healthy and diseased prey survive, while the predators become extinct. The simulations indicate that there is a set of parameters for which the equilibrium is reached.

The interior equilibrium E_4 represents the situation when predators thrive together with healthy and diseased. Again the simulations show that this equilibrium is attained for a particular set of parameters.

6 Simulations

To verify theoretical results we have done some graphical simulations with Matlab.

We start from point E_1. Assigning the following values to the parameters,

$$r = 0.9, K = 2, \gamma = 0.1, A = 1, a = 0.1, b = 0.2, \mu = 0.9, m = 0.4, e = 0.6$$

we get the result shown in Figure 1, showing that the equilibrium is indeed attained, as theoretically predicted. Note indeed that the stability condition (5) is satisfied for this parameter choice.

For the point E_2, we consider the parameter values

$$r = 0.9, K = 2, \gamma = 0.1, A = 1, a = 0.9, b = 0.3, \mu = 0.9, m = 0.3, e = 0.8$$

 ISBN 13: 978-84-613-5510-5

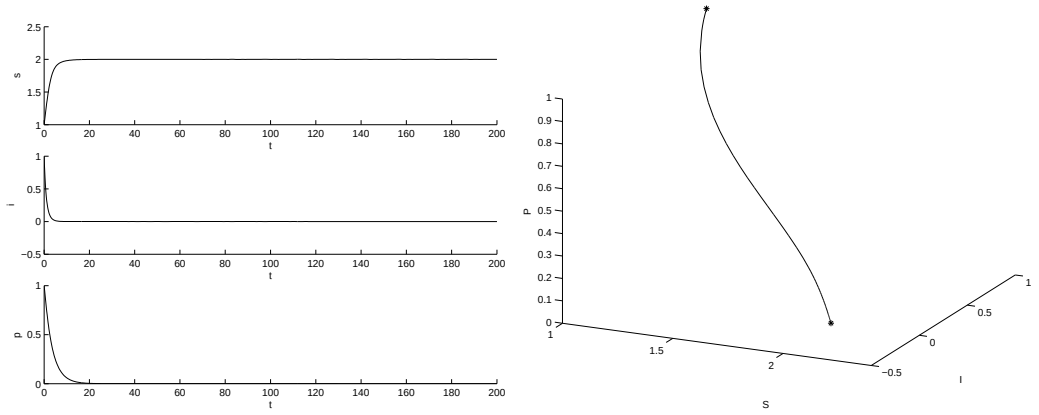


Figure 1: Equilibrium E_1 is reached. On the left the solutions as function of time, on the right the trajectory in the phase space.

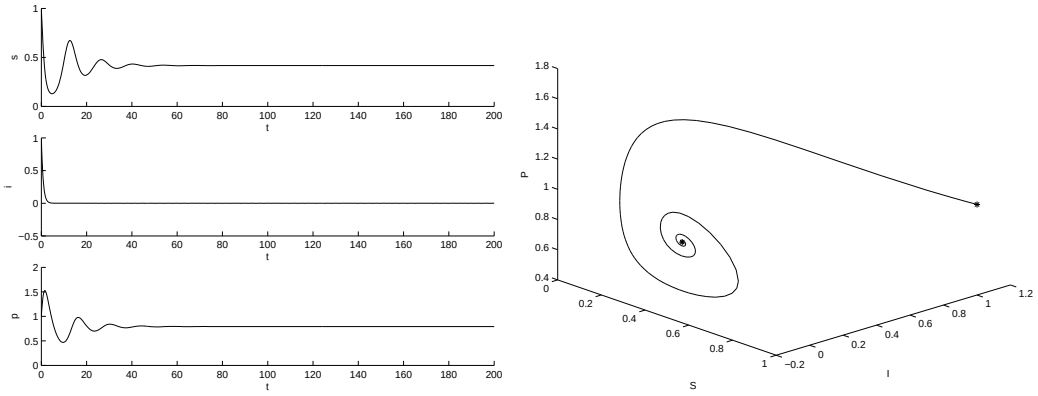


Figure 2: Stability of the disease-free equilibrium E_2. On the left the solutions as function of time, on the right the trajectory in the phase space.

obtaining the behavior shown in Figure 2. Once more, the disease-free equilibrium is reached by the system's trajectories, verifying the theoretical results. Indeed both conditions (7) and (8) are satisfied, for the above parameter choice.

We do not have the analytical coordinates of the remaining equilibria E_3 and E_4. To investigate their behavior, simulations are the only resource. For the former, the following parameter values

$$r = 0.3, K = 20, \gamma = 1, A = 0.8, a = 0.1, b = 0.1, \mu = 0.6, m = 0.1, e = 0.9$$

provide a stable behavior, as depicted in Figure 3.

Around the point E_4, the values

$$r = 10, K = 30, \gamma = 1, A = 0.01, a = 0.1, b = 0.1, \mu = 0.5, m = 0.1, e = 0.9$$

 ISBN 13: 978-84-613-5510-5

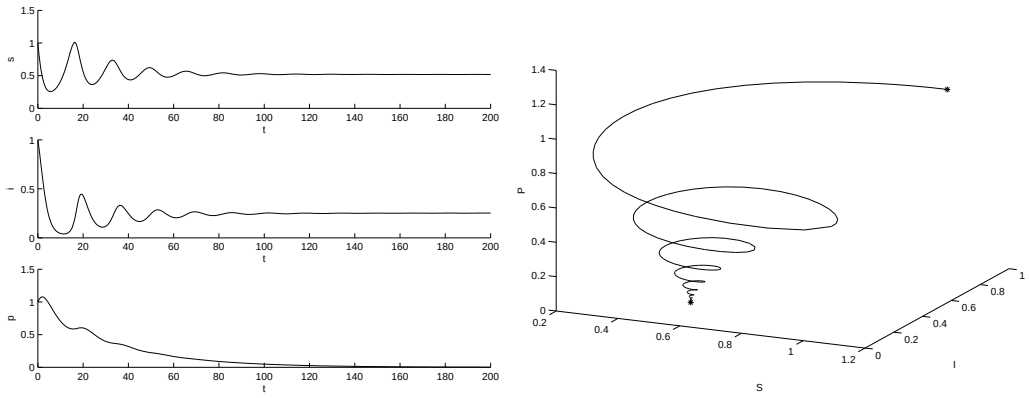


Figure 3: Stability of the predator-free equilibrium E_3. On the left the solutions as function of time, on the right the trajectory in the phase space.

provide the empirical verification that the interior equilibrium in such situation is stable and therefore can be attained, as shown in Figure 4.

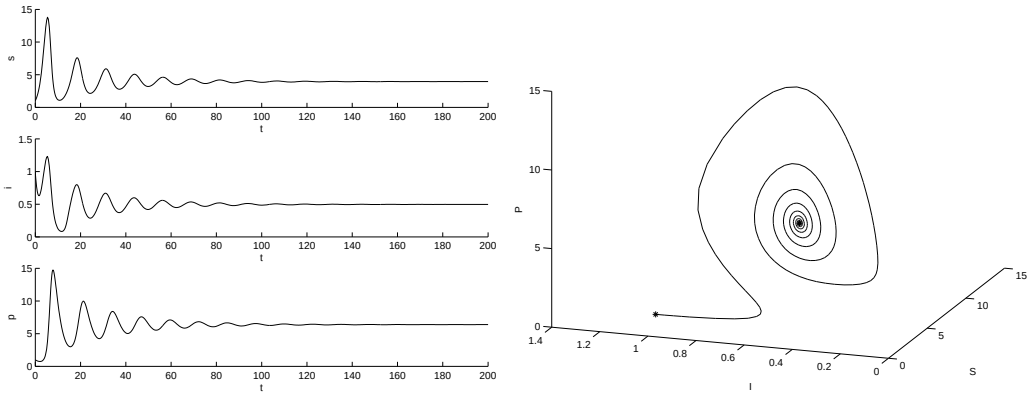


Figure 4: Stability of equilibrium E_4 indicating that coexistence of the ecosystem at a constant level is possible. On the left the solutions as function of time, on the right the trajectory in the phase space.

At E_4 a situation that induces oscillatory behavior arises. In fact if we change the parameter values and consider

$$r = 0.9, K = 30, \gamma = 0.11, A = 0.1, a = 0.1, b = 0.1, \mu = 0.5. m = 0.4, e = 0.9$$

we have the behavior shown in Figure 5.

We have also investigated some of the limit cycles arising. Taking as reference values for the parameters the following values,

$$r = 10, \quad K = 30, \quad \gamma = 1, \quad A = .01, \quad a = .1, \quad b = .1, \quad \mu = .5, \quad m = .05, \quad e = .9,$$

 ISBN 13: 978-84-613-5510-5

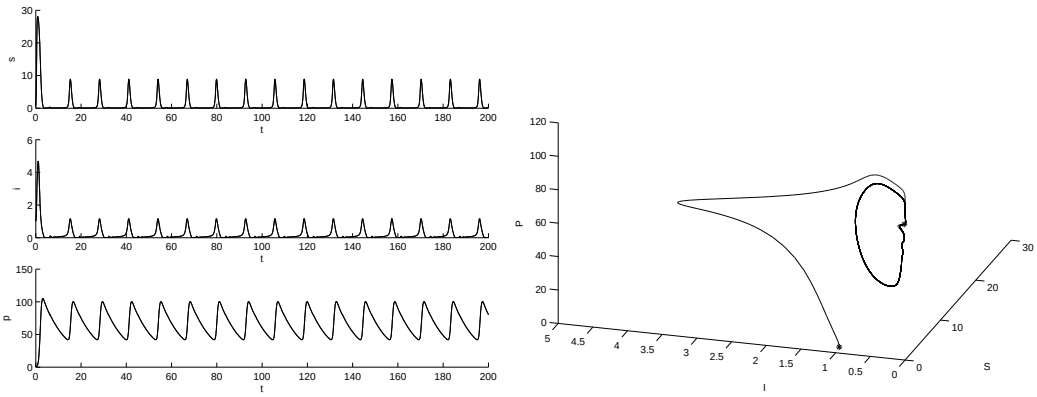


Figure 5: Limit cycles arising around E_4. Thus coexistence can be attained also via stable oscillations. On the left the solutions as function of time, on the right the trajectory in the phase space.

and letting one of them vary at each time, we have constructed some bifurcation diagrams. We report below some of the results.

In Figure 6 we provide a bifurcation diagram as a function of the parameter m, in Figure 7 the bifurcation diagram as a function of the parameter a, in Figure 8 the one relative to the parameter b, in Figure 9 the one relative to γ, finally in Figure 10 the one of μ. Here, stars denote the largest value of the limit cycle and circles the smallest one.

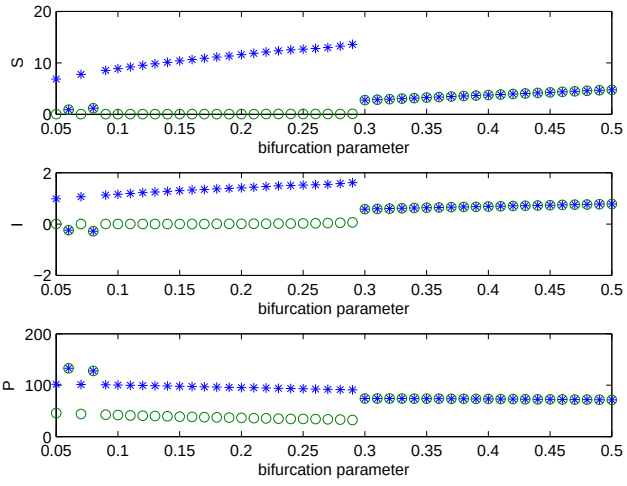


Figure 6: Bifurcation diagram as a function of the parameter m.

 ISBN 13: 978-84-613-5510-5

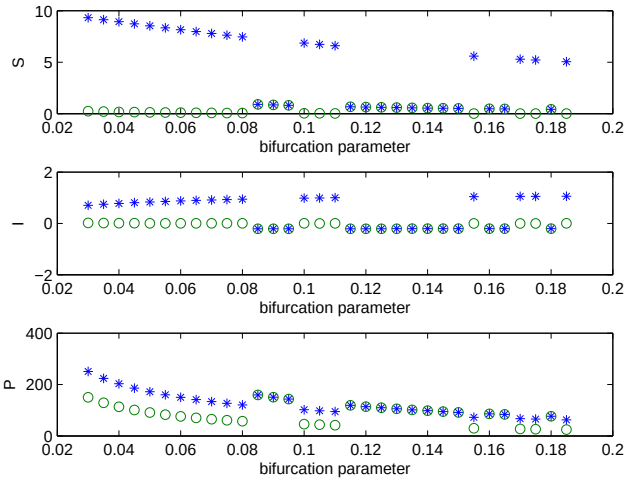


Figure 7: Bifurcation diagram as a function of the parameter a.

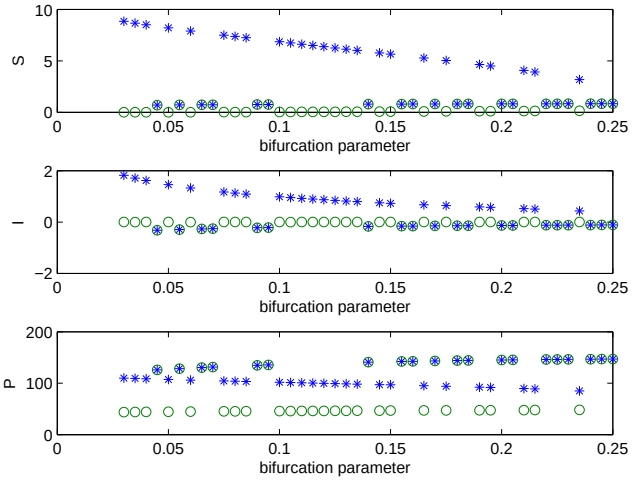


Figure 8: Bifurcation diagram as a function of the parameter b.

7 Conclusions

In this investigation we have introduced a response against infected individuals in predator-prey ecoepidemic models, in which the disease spreads among prey. We have examined the equilibria and studied the limit cycles that originate when the latter bifurcate. These persistent oscillations are similar to the ones found in the classical model for these situations, [2], in contrast instead to what is reported in [4]. With respect to other similar models using Holling type I interaction terms, [6], the oscillations here

 ISBN 13: 978-84-613-5510-5

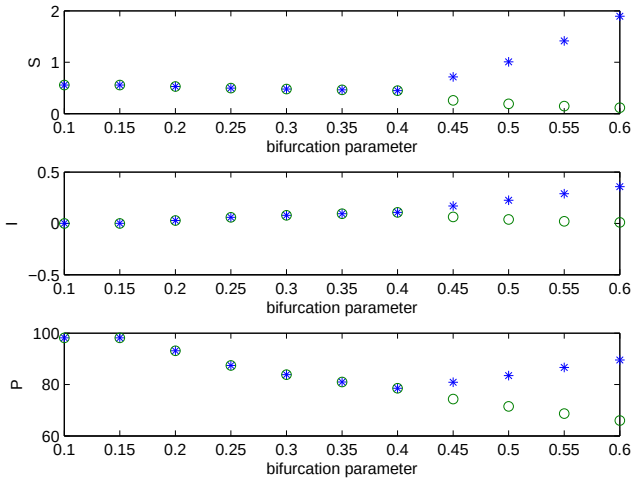


Figure 9: Bifurcation diagram as a function of the parameter γ.

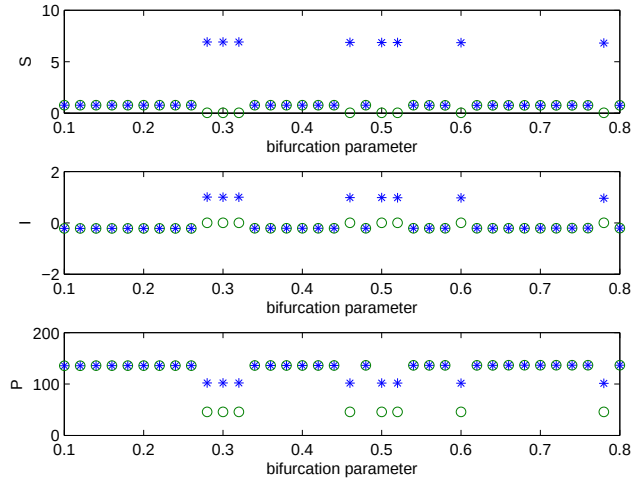


Figure 10: Bifurcation diagram as a function of the parameter μ.

found constitute a novelty. But a similar model using Holling type II interactions shows also limit cycles, [3].

References

[1] O. Arino and A. El abdllaoui, J. Mikram, J. Chattopadhyay, *Infection in prey population may act as biological control in ratio-dependent predator-prey*

 ISBN 13: 978-84-613-5510-5

models, Nonlinearity **17** (2004) 1101–1116.

[2] V. Capasso, G. Serio, *A generalization of the Kermack-McKendrick deterministic epidemic model*, Math. Biosc. **42** (1978) 43–61.

[3] J. Chattopadhyay and O. Arino, *A predator-prey model with disease in the prey*, Nonlinear Analysis **36** (1999) 747–766.

[4] J. Chattopadhyay, R. R. Sarkar, G. Goshal, *Removal of infected prey prevent limit cycles oscillations in an infected prey-predator system – a mathematical study*, Ecological Modelling **156** (2002) 113–121.

[5] H. Malchow, S. Petrovskii, E. Venturino, *Spatiotemporal patterns in Ecology and Epidemiology*, CRC, Boca Raton, 2008.

[6] E. Venturino, *Epidemics in predator-prey models: disease among the prey*, in O. Arino, D. Axelrod, M. Kimmel, M. Langlais: Mathematical Population Dynamics: Analysis of Heterogeneity, Vol. one: Theory of Epidemics, Wuertz Publishing Ltd, Winnipeg, Canada, p. 381-393, 1995.

[7] E. Venturino, *The effects of diseases on competing species*, Math. Biosc., **174** (2001) 111–131.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A new numerical method for Volterra integral equations of the second kind

M. I. Berenguer[1], D. Gámez[1], A. I. Garralda-Guillem[1], M. Ruiz Galán[1] and M. C. Serrano Pérez[1]

[1] *Department of Applied Mathematics, University of Granada (Spain)*

emails: maribel@ugr.es, domingo@ugr.es, agarral@ugr.es, mruizg@ugr.es, cserrano@ugr.es

Abstract

In this work we present a numerical method to approximate the solution of the Volterra integral equation of the second kind. The properties of Schauder bases and fixed point theorem are the fundamental tools used for this purpose.

Key words: Volterra integral equation, fixed point, Schauder bases, numerical methods.

MSC 2000: 45D05, 65R20, 47H10.

1 Introduction

Modeling many problems of science, engineering, physics and other disciplines leads to linear and nonlinear Volterra integral equations of the second kind:

$$x(t) = y_0(t) + \int_{\alpha}^{t} K(t, s, x(s))ds, \qquad t \in [\alpha, \alpha + \beta], \tag{1}$$

where $y_0 : [\alpha, \alpha+\beta] \to \mathbb{R}$ and the kernel $K : [\alpha, \alpha+\beta]^2 \times \mathbb{R} \to \mathbb{R}$ are assumed to be known continuous functions and the unknown function to be determined is $x : [\alpha, \alpha + \beta] \to \mathbb{R}$.

These are usually difficult to solve analytically and in many cases the solution must be approximated transforming the integral equation into a linear or nonlinear system that can be solved by direct or iterative methods (see for example [1] and [4]). The purpose of this paper is to develop an effective method for approximating the solution of (1). This is the previous work of an forthcoming paper and generalizes to the nonlinear case the results for solving the linear case appearing in [2]. Among the main advantages of our numerical method as opposed to the classical ones, we can point out that it is not necessary to solve algebraic equation systems and the integrals involved are immediate and we do not require any quadrature method to calculate them.

 ISBN 13: 978-84-613-5510-5

2 Analytical tools to be used

Let $C([\alpha, \alpha+\beta])$ be the Banach space of all continuous and real-valued functions on $[\alpha, \alpha+\beta]$, endowed with its usual sup-norm. Observe that (1) is equivalent to the problem of finding fixed points of the operator $T : C([\alpha, \alpha+\beta]) \longrightarrow C([\alpha, \alpha+\beta])$ defined by

$$(Tx)(t) := y_0(t) + \int_\alpha^t K(t, s, x(s))\, ds, \quad t \in [\alpha, \alpha+\beta] \text{ and } \ x \in C([\alpha, \alpha+\beta]). \qquad (2)$$

To establish the existence of fixed points of (2), we will use the version of the **Banach fixed-point theorem** (see [7]) which we enunciate below: *Let $(X, \|\cdot\|)$ be a Banach space, let $F : X \longrightarrow X$ and let $\{\mu_n\}_{n\geq 1}$ be a sequence of nonnegative real numbers such that the series $\sum_{n\geq 1} \mu_n$ is convergent and for all $x, y \in X$ and for all $n \geq 1$, $\|F^n x - F^n y\| \leq \mu_n \|x - y\|$. Then F has unique fixed point $u \in X$. Moreover, if $\overline{x}$ is an element in X, then we have that for all $n \geq 1$, $\|F^n \overline{x} - u\| \leq (\sum_{i=n}^{\infty} \mu_i)\|F\overline{x} - \overline{x}\|$. In particular, $u = \lim_n F^n(\overline{x})$.*

Schauder bases will be another important tool in development work. They have been previously used successfully in the numerical study of integral and differential equations (see [2], [3], [5] and [8]). If $\{x_n\}_{n\geq 1}$ is a Schauder basis of a Banach space X, we denote the sequence of (continuous and linear) finite dimensional *projections* by $\{P_n\}$ and the associated sequence of (continuous and linear) *coordinate functionals* by $\{x_n^*\}_{n\geq 1}$ in X^*. Reader is referred to [2], [6] and [9], where can see the construction of the usual Schauder bases $\{b_n\}_{n\geq 1}$ in $C([\alpha, \alpha+\beta])$ and $\{B_n\}_{n\geq 1}$ of $C([\alpha, \alpha+\beta]^2)$.

3 Development of the numerical method and example

Before presenting the proposed method, we establish the following two preliminary results:

Proposition 1. *Assume that in (1) the kernel K satisfies a Lipschitz condition in its third variable:*

$$|K(t, s, x) - K(t, s, y)| \leq M|x - y| \quad \text{for all } t, s \in [\alpha, \alpha+\beta] \ \text{ and } x, y \in \mathbb{R}$$

for some constant $M > 0$. Then the integral equation (1) has a unique solution $x \in C[\alpha, \alpha+\beta]$. In addition, for each $\overline{x} \in C([\alpha, \alpha+\beta])$, the sequence $\{T^n\overline{x}\}_{n\geq 1}$ in $C([\alpha, \alpha+\beta])$ converges uniformly to the unique solution x and for all $n \geq 1$,

$$\|T^n\overline{x} - x\| \leq \frac{(M\beta)^n}{n!} e^{M\beta} \|T\overline{x} - \overline{x}\|.$$

Proposition 2. *Let $T : C([\alpha, \alpha+\beta]) \longrightarrow C([\alpha, \alpha+\beta])$ be the continuous integral operator defined in (2). Let $x \in C[\alpha, \alpha+\beta]$, and let us consider the function $\Phi \in C([\alpha, \alpha+\beta]^2)$, defined by $\Phi(t, s) = K(t, s, x(s))$. Let $\{\lambda_n\}_{n\geq 1}$ be the sequences of*

 ISBN 13: 978-84-613-5510-5

scalars satisfying $\Phi = \sum_{n\geq 1} \lambda_n B_n$. *Then for all* $t \in [\alpha, \alpha+\beta]$ *we have that*

$$(Tx)(t) = y_0(t) + \sum_{n\geq 1} \lambda_n \int_\alpha^t B_n(t,s)\, ds \tag{3}$$

where $\lambda_1 = \Phi(t_1,t_1)$ *and for* $n \geq 2$, $\lambda_n = \Phi(t_i,t_j) - \sum_{k=1}^{n-1} B_k^*(\Phi) B_k(t_i,t_j)$ *with* $\sigma(n) = (i,j)$, *where* $\sigma = (\sigma_1,\sigma_2) : \mathbb{N} \longrightarrow \mathbb{N}\times\mathbb{N}$ *is the bijective mapping defined by*

$$\sigma(n) := \begin{cases} (\sqrt{n},\sqrt{n}), & \text{if} \quad [\sqrt{n}] = \sqrt{n} \\ (n-[\sqrt{n}]^2, [\sqrt{n}]+1), & \text{if} \quad 0 < n-[\sqrt{n}]^2 \leq [\sqrt{n}] \\ ([\sqrt{n}]+1, n-[\sqrt{n}]^2-[\sqrt{n}]), & \text{if} \quad [\sqrt{n}] < n-[\sqrt{n}]^2 \end{cases} .$$

and $[a]$ *denote the integer part of* $a \in \mathbb{R}$.

In view of Propositions 1 and 2, (3) gives the unique solution $x(t)$ of (1). The problem is that generally this expression can not be calculated explicitly. The idea of the proposed method is to truncate to calculate approximately a sequence of iterations and projections that converge to the solution. More specifically, let $\overline{x} : [\alpha, \alpha+\beta] \longrightarrow \mathbb{R}$ be a continuous function, and $n_1, n_2, n_3, \ldots, \in \mathbb{N}$. Consider the continuous functions

$$z_0(t) := \overline{x}(t), \qquad t \in [\alpha, \alpha+\beta] \tag{4}$$

and for $r \in \mathbb{N}$, we define

$$L_{r-1}(t,s) := K(t,s,z_{r-1}(s)) \qquad (t,s \in [\alpha,\alpha+\beta]). \tag{5}$$

$$z_r(t) := y_0(t) + \int_\alpha^t Q_{n_r^2}(L_{r-1}(t,s))\, ds \qquad (t \in [\alpha,\alpha+\beta]). \tag{6}$$

In order to obtain the convergence of the sequence $\{z_r\}_{r\geq 1}$ to the unique solution of (1) we introduce the following notation: If $\{t_n\}_{n\geq 1}$ is the dense subset of distinct points in $[\alpha,\alpha+\beta]$ we considered to define the Schauder basis, let T_n be the set $\{t_j, 1 \leq j \leq n\}$ ordered in an increasing way for $n \geq 2$. Let ΔT_n denote the maximum distance between two consecutive points of T_n.

Theorem 3. *With the previous notation, let* $\overline{x} \in C([\alpha,\alpha+\beta]$, $y_0 \in C^1([\alpha,\alpha+\beta]$ *and* $K \in C^1([\alpha,\alpha+\beta]^2 \times \mathbb{R})$ *with* K, $\dfrac{\partial K}{\partial t}$, $\dfrac{\partial K}{\partial s}$, $\dfrac{\partial K}{\partial x}$, *satisfying the Lipschitz global condition of the third variable. Then:*

a) Then $\left\{\dfrac{\partial L_{r-1}}{\partial t}\right\}_{r\geq 1}$, $\left\{\dfrac{\partial L_{r-1}}{\partial s}\right\}_{r\geq 1}$ *are uniformly bounded.*

b) There is $\rho > 0$ *such that for all* $r \geq 1$ *and* $n_r \geq 2$

$$\|L_{r-1} - Q_{n_r^2}(L_{r-1})\| \leq \rho \Delta T_{n_r}.$$

The main result that establishes that the sequence defined in (4), (5) and (6) approximates the solution of (1) as well as giving an upper bond of the error committed is given below:

 ISBN 13: 978-84-613-5510-5

Theorem 4. *Let $K \in C([\alpha, \alpha+\beta]^2 \times \mathbb{R})$ such that K satisfies a global Lipschitz condition in the third variable and let $\overline{x} \in C([\alpha, \alpha+\beta]$. Let $m \in \mathbb{N}$, and assume that certain positive numbers $\varepsilon_1, \ldots, \varepsilon_m$ satisfy*

$$\|Tz_{r-1} - z_r\| < \varepsilon_r, \quad r = 1, \ldots, m$$

and let x be the exact solution of the integral equation (1). Then

$$\|x - z_m\| \leq \frac{(M\beta)^m}{m!} e^{M\beta} \|T\overline{x} - \overline{x}\| + \sum_{r=1}^{m} \varepsilon_r \frac{(M\beta)^{m-r}}{(m-r)!},$$

where M is the Lipschitz constant of K.

Under the hypothesis of Theorem 3, there is $\rho > 0$ such that for $r \geq 1$ and $n_r \geq 2$,

$$\|Tz_{r-1} - z_r\| \leq \beta \|L_{r-1} - Q_{n_r^2}(L_{r-1})\| \leq \beta\rho\Delta T_{n_r}.$$

Hence, given certain $\varepsilon_1, \ldots, \varepsilon_m > 0$, we can find m positive integers $n_1, \ldots, n_m$ such that $\|Tz_{r-1} - z_r\| < \varepsilon_r$, and by Theorem 4 we can state the convergence of $\{z_r\}_{r\geq 1}$ and an estimation of the error.

Example 5. Consider the equation

$$\begin{cases} x(t) = \dfrac{1}{3} t\cos(t^3) + t^3 - \dfrac{t}{3} + \displaystyle\int_0^t t\ s^2\ \sin(x(s))ds \qquad (t \in [0,1]) \\ y(0) = 0 \end{cases},$$

whose exact solution is $x(t) = t^3$.

To construct the Schauder basis in $C([0,1]^2)$, we considered the particular choice $t_1 = 0$, $t_2 = 1$ and for $n \in \mathbb{N} \cup \{0\}$, $t_{i+1} = \frac{2k+1}{2^{n+1}}$ if $i = 2^n + k + 1$ where $0 \leq k < 2^n$ are integers. To define the sequence $\{z_r\}_{r\geq 1}$, we take $z_0(t) = 1$ and $n_r = j$ (for all $r \geq 1$). In the following table we exhibit, for $j = 9$ and 17, the absolute errors committed in eight representative points (t_i) of $[0,1]$ when we approximate the exact solution x by the iteration z_2.

t_i	0.125	0.250	0.375	0.5	0.625	0.750	0.875	1
$j = 9$ $\lvert z_2(t_i) - x(t_i)\rvert$	1.6E−7	6.0E−6	4.7E−5	2.9E−4	6.2E−4	1.7E−3	3.1E−3	2.1E−3
$j = 17$ $\lvert z_2(t_i) - x(t_i)\rvert$	4.7E−8	1.5E−6	1.2E−5	9.4E−5	1.6E−4	4.9E−4	8.7E−4	1.1E−4

Acknowledgements

Research partially supported by M.E.C. (Spain) and FEDER project no. MTM2006-12533, and by Junta de Andalucía Grant FQM359.

 ISBN 13: 978-84-613-5510-5

References

[1] C.T.H. Baker, *A perspective on the numerical treatment of Volterra quations.* J. Comput. Appl. Math. 125 (2000), pp. 217–249.

[2] M.I. Berenguer, D. Gámez, A.I. Garralda-Guillem, M. Ruiz Galán and M.C. Serrano Pérez, *Analytical Techniques for a Numerical Solution of the Linear Volterra Integral Equation of the Second Kind*, Abstr. Appl. Anal. Volume 2009, (2009) Article ID 149367, 12 pages, doi: 10.1155/2009/149367.

[3] M. I. Berenguer, M. A. Fortes, A. I. Garralda Guillem and M. Ruiz Galán, *Linear Volterra integrodifferential equation and Schauder bases*, Applied Mathematics and Computation, vol. 159, no. 2, pp. 495-507, 2004.

[4] H. Brunner and P.J. van der Houwen, *The Numerical Solution of Volterra Equations*, North-Holland, Amsterdam, 1986.

[5] D. Gámez, A. I. Garralda Guillem and M. Ruiz Galán, *High-order nonlinear initial-value problems countably determined*, Journal of Computational and Applied Mathematics, vol. 228, no. 1, pp. 77-82, 2009.

[6] B. Gelbaum and J. Gil de Lamadrid, *Bases on tensor products of Banach spaces*, Pacific J. Math. 11, (1961), pp. 1281–1286.

[7] G. J. O. Jameson, *Topology and Normed Spaces*, Chapman and Hall, London, 1974.

[8] A. Palomares and M. Ruiz Galán, *Isomorphisms, Schauder bases in Banach spaces and numerical solution of integral and differential equations*, Numer. Funct. Anal. Optim. 26 (2005), pp. 129–137.

[9] Z. Semadeni, *Product Schauder bases and approximation with nodes in spaces of continuous functions*, Bull. Acad. Polon. Sci. 11 (1963), pp. 387–391.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Is there anything left to say on enzyme kinetic constants and quasi-steady state approximation?

Alberto Maria Bersani[1] **and Guido Dell'Acqua**[1]

[1] *Dipartimento di Metodi e Modelli Matematici (Me.Mo.Mat.), "Sapienza" University, Via A. Scarpa 16, 00161 - Rome, Italy*

emails: bersani@dmmm.uniroma1.it, dellacqua@dmmm.uniroma1.it

Abstract

In this paper we re-examine the commonly accepted meaning of the two kinetic constants characterizing any enzymatic reaction, according to Michaelis-Menten kinetics. Expanding in terms of exponentials the solutions of the ODEs governing the reaction, we determine a new constant, which corrects some misinterpretations of current biochemical literature.

Key words: Michaelis-Menten kinetics, quasi-steady state approximations, asymptotic expansions
MSC 2000: 41A58, 41A60, 92C45

1 Introduction

The question addressed in the title of this paper is not merely a rethoric one. Our answer, of course, is definitely *yes*: we do think that there is still a lot of room in this field. Formulated more than one century ago, the Michaelis-Menten-Briggs-Haldane approximation, or standard quasi-steady state approximation (sQSSA) [24, 7, 33], still represents a milestone in the mathematical modeling of enzymatic reactions. Nevertheless, the hypothesis of quasi-steady state is crucial for the interpretation of the reaction and must be handled with much care. It is based on the assumption that the complex can be considered "substantially" constant, but this statement has led to many misinterpretations of the model. In fact, as Heineken et al. showed in [15], the correct mathematical interpretation of the quasi-steady state assumption is that when we expand asymptotically the solutions of the ODEs governing the process with respect to an appropriate parameter, the sQSSA is the zero order approximation of the solution. As already observed by Briggs and Haldane by a chemical point of view, when the parameter of the expansion is sufficiently small this approximation is valid. Heineken et al. used the parameter given by the ratio of the initial concentrations of enzyme E and substrate S, obtaining the well-known chemical requirement.

 ISBN 13: 978-84-613-5510-5

In 1987 Fraser [13] pointed out that, geometrically speaking, the steady state assumption for chemical reactions is an approximation in the phase space to the slow manifold, i. e., the singular trajectory which strongly attracts all fast transient flow. He also described an iterative scheme to approximate this singular trajectory without any restrictions on the rate constants of the system. The same arguments were applied to the Michaelis-Menten mechanism in 2006 by Calder and Siegel [8]. In 1988 Segel [32] and in 1989 Segel and Slemrod [33] obtained the Michaelis-Menten approximation expanding the solutions in terms of a new parameter, including the Michaelis constant and showing that the sQSSA is valid in a wider range of parameters than the one supposed before. However it is well known that while *in vitro* the condition on the concentrations can be easily fulfilled, *in vivo* it is not always respected [34, 35, 36, 1], in particular when the reaction is not isolated but is part of complex reaction networks. This means that, though very useful, this approximation cannot always be applied.

Michaelis-Menten kinetics has recently become one of the most important tools in the field of Systems Biology and in particular of mathematical modeling of intracellular enzyme reactions, but in most literature any *apriori* analysis of the applicability of sQSSA is absent, even in very complex reaction networks. This fact has led to several problems concerning the study of particular phenomena, like oscillations [12, 28], bistability [10], ultrasensitivity [29] or Reverse Engineering [27]. Following [20], recent papers [6, 38, 39, 26, 10, 29, 40, 23, 11, 28, 2] have introduced and explored a new approximation, called total quasi-steady state approximation (tQSSA), which has been shown to be always roughly valid in the case of an isolated reaction. Nevertheless, since it is in any case an approximation, also the tQSSA can dramatically fail, as shown in [28], in more complex mechanisms, involving more than one reaction, but it is doubtless that it is valid in a much wider range of parameter than the sQSSA [10, 26, 27, 28, 31].

One of the main problems of the mathematical treatment of the sQSSA is the misinterpretation of the hypothesis that the complex time concentration has zero derivative. Many papers and even monographies tend to indicate, probably for the sake of simplicity, the "substantial" equilibrium as a real equilibrium [21, 42, 30, 14], which is obviously not true; in this case any simplification can be definitely misleading. As observed in [15], p. 97, this use of the equations seems scandalous to any mathematicians and can bring to results which are absolutely inconsistent and false. In this work we want to re-examine some mathematical aspects of Michaelis-Menten reaction and of the sQSSA, trying to clarify some aspects of the enzyme reactions; in particular we discuss the biochemical and mathematical meaning of the tQSSA, comparing it with the sQSSA, then we analyse the consequences of the misuse of the sQSSA, reconsidering the meaning of the two kinetic constants V_{max} and K_M; finally we introduce an expansion in terms of exponentials, which is valid for every choice of the parameters and enzyme initial concentrations; this expansion is the most appropriate to approximate the asymptotic behavior of the solution for large values of t, in absence of product degradation; moreover we use it to solve a serious incoherence present in literature, related to the biochemical interpretation of the constant K_M.

 ISBN 13: 978-84-613-5510-5

2 Notations, definitions and main known results

The model of biochemical reactions was set forth by Henri in 1901 [16, 17, 18] and Michaelis and Menten in 1913 [24] and further developed by Briggs and Haldane in 1925 [7]. This formulation considers a reaction where a substrate S binds an enzyme E reversibly to form a complex C. The complex can then decay irreversibly to a product P and the enzyme, which is then free to bind another molecule of the substrate. This process is summarized in the scheme

$$E + S \underset{d}{\overset{a}{\rightleftarrows}} C \xrightarrow{k} E + P, \tag{1}$$

where a, d and k are kinetic parameters (supposed constant) associated with the reaction rates: a is the second order rate constant of enzyme-substrate association; d is the rate constant of dissociation of the complex; k is the catalysis rate constant. Following the mass action principle, which states that the concentration rates are proportional to the reactant concentrations, the formulation leads to an ODE for each complex and substrate involved. We refer to this as the *full system*. From now on we will indicate with the same symbols the names of the enzymes and their concentrations. The ODEs describing (1) are

$$\begin{aligned} \tfrac{dS}{dt} &= -a(E_T - C)S + d\,C, \\ \tfrac{dC}{dt} &= a(E_T - C)S - (d+k)C, \end{aligned} \tag{2}$$

with initial conditions

$$S(0) = S_T, \quad C(0) = 0, \tag{3}$$

and conservation laws

$$E + C = E_T, \quad S + C + P = S_T\ . \tag{4}$$

Here E_T is the total enzyme concentration assumed to be free at time $t = 0$. Also the total substrate concentration, S_T, is free at $t = 0$. This is called the Michaelis-Menten (MM) kinetics [24, 3]. Let us observe that (2) – (4) asimptotically admits only the trivial solution given by $C = S = 0$, $P = S_T$ and $E = E_T$. This means that all the substrate eventually becomes product due to the irreversibility, while the enzyme eventually is free and the complex concentration tends to zero. Assuming that the complex concentration is approximately constant after a short transient phase leads to the usual Michaelis-Menten (MM) approximation, or *standard quasi-steady state approximation* (sQSSA): we have an ODE for the substrate while the complex is assumed to be in a quasi-steady state (i. e., $\frac{dC}{dt} \approx 0$):

$$C \cong \frac{E_T \cdot S}{K_M + S} \qquad , \qquad \frac{dS}{dt} \cong -kC \cong -\frac{V_{max} S}{K_M + S}, \quad S(0) = S_T, \tag{5}$$

where

$$V_{max} = k\,E_T, \quad K_M = \frac{d+k}{a}\ . \tag{6}$$

 ISBN 13: 978-84-613-5510-5

and K_M is the *Michaelis constant*. Applying a quasi-steady approximation reduces not only the dimensionality of the system, passing from two equations (full system) to one (MM approximation or sQSSA). It reduces also its stiffness and thus speeds up numerical simulations greatly, especially for large networks as found *in vivo*. It allows also a theoretical investigation of the system which cannot be obtained with the numerical integration of the full system. Moreover, the kinetic constants in (1) are usually not known, whereas finding the kinetic parameters for the MM approximation is a standard *in vitro* procedure in biochemistry. See e.g. [3] for a general introduction to this approach. We stress here that this is an approximation to the full system, and that it is valid only under suitable hypotheses, e. g., when the enzyme concentration is much lower than either the substrate concentration or the Michaelis constant K_M, i. e., (see, for example, [33])

$$\varepsilon_{MM} := \frac{E_T}{S_T + K_M} \ll 1 \tag{7}$$

This condition is usually fulfilled for *in vitro* experiments, but often breaks down *in vivo* [36, 35, 34, 1]. We refer to [31] for a nice, general review of the kinetics and approximations of (1). It is useful to quote also the recent papers [12, 41, 25, 10, 28] which discuss the applicability of the sQSSA. In order to solve this problem, in 1955 Laidler [20], discussing the mathematical theory of the transient phase, found expressions for the behavior of P in the quasi-steady state and found several sufficient conditions for the applicability of the approximations. These conditions were much more general than $\frac{E_T}{S_T} \ll 1$. The importance of Laidler's results can be understood comparing his approach to a recent one, based on the *total quasi-steady state approximation* (tQSSA). It was introduced by Borghans et al. [6] and refined by Tzafriri [38] for isolated reactions. It arises introducing the *total substrate*

$$\overline{S} = S + C, \tag{8}$$

and assuming that the complex is in a quasi-steady state as for the sQSSA. Reaction (1) then gives the tQSSA [6, 20]:

$$\frac{d\overline{S}}{dt} \cong -k\, C_-(\overline{S}), \quad \overline{S}(0) = S_T, \tag{9}$$

where

$$C_-(\overline{S}) = \frac{(E_T + K_M + \overline{S}) - \sqrt{(E_T + K_M + \overline{S})^2 - 4E_T\overline{S}}}{2}. \tag{10}$$

Numerical integration of (9) gives the time behavior of $\overline{S}$ and then (8) and (10) give the corresponding C and S. Tzafriri [38] showed that the tQSSA (9) is valid whenever

$$\varepsilon_{tQSSA} := \frac{K}{2S_T}\left(\frac{E_T + K_M + S_T}{\sqrt{(E_T + K_M + S_T)^2 - 4E_T S_T}} - 1\right) \ll 1, \tag{11}$$

 ISBN 13: 978-84-613-5510-5

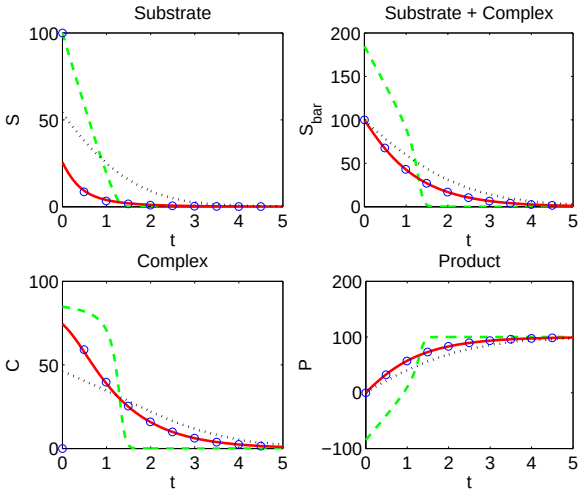


Figure 1: Dynamics of the model (1) for a=1, k=1, d=4, $S_T = 100$, $E_T = 89$. Plots show (top-left, bottom-right) S, $\bar{S}$, C, P. Circles: numerical solution of the full system; dashed line: sQSSA; solid: tQSSA; dotted: first order approximation of tQSSA. Notice that the sQSSA and the tQSSA, representing only the outer approximations, do not (and are not expected to) satisfy the initial condition for C. This is why the initial boundary layer is missed.

(where $K = \dfrac{k}{a}$), and that this is at least roughly valid for any sets of parameters, in the sense that $\varepsilon_{tQSSA} \leq \frac{K}{4K_M} \leq \frac{1}{4}$. This means that, for *any* combination of parameters and initial conditions, (9) gives a decent approximation to the full system (2). The parameter K is known as the Van Slyke-Cullen constant. The *dissociation constant* $K_D = \dfrac{d}{a}$ [3] is related to the previous kinetic constants by the simple formula $K_D = K_M - K$. Let us remark that, in recent literature, the sQSSA is applied to complex enzyme reaction networks, like, e.g., the MAPK cascade, without any *a priori* analysis on its applicability, setting to zero not only the derivatives of the complex concentrations, but also, surprisingly, the complex concentrations themselves (see, e.g., [22, 19, 9]). This produces serious inconsistencies with experimental observations and has resulted in the discovery of the so-called "substrate sequestration" hypothesis [4, 5], which states that the enzyme can sequester a significant amount of substrate by binding to it, making this sequestered fraction of the substrate no longer accessible to other kinases. The importance of the choice of $\bar{S}$ as one of the system variables lies in the fact that substrate sequestration is naturally included in the total substrate. Indeed, the latter takes into account both the free and the "sequestered" substrate.

 ISBN 13: 978-84-613-5510-5

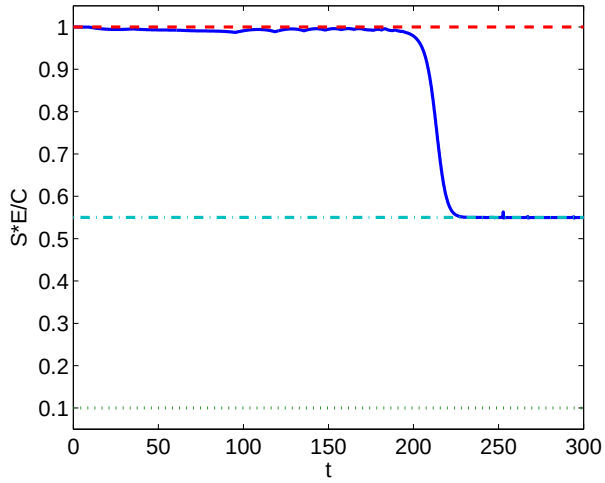


Figure 2: Plot of $\frac{ES}{C}$ for a=1, k=0.9, d=0.1, $S_T = 100$, $E_T = 0.55$. Solid line: numerical solution of the full system; dashed: K_M; dashed-dotted: K_W; dotted: K_D. Parameters and initial conditions were chosen to give $K_W = \dfrac{K_M + K_D}{2}$.

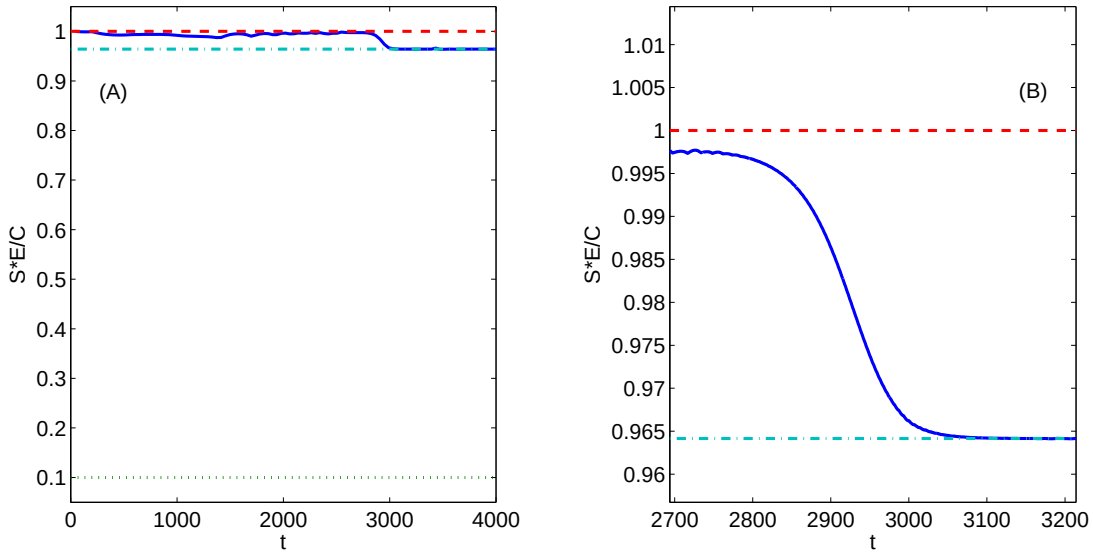


Figure 3: (A) Plot and (B) zoom of $\frac{ES}{C}$ for a=1, k=0.9, d=0.1, $S_T = 100$, $E_T = 0.04$. Solid line: numerical solution of the full system; dashed: K_M; dashed-dotted: K_W; dotted: K_D.

 ISBN 13: 978-84-613-5510-5

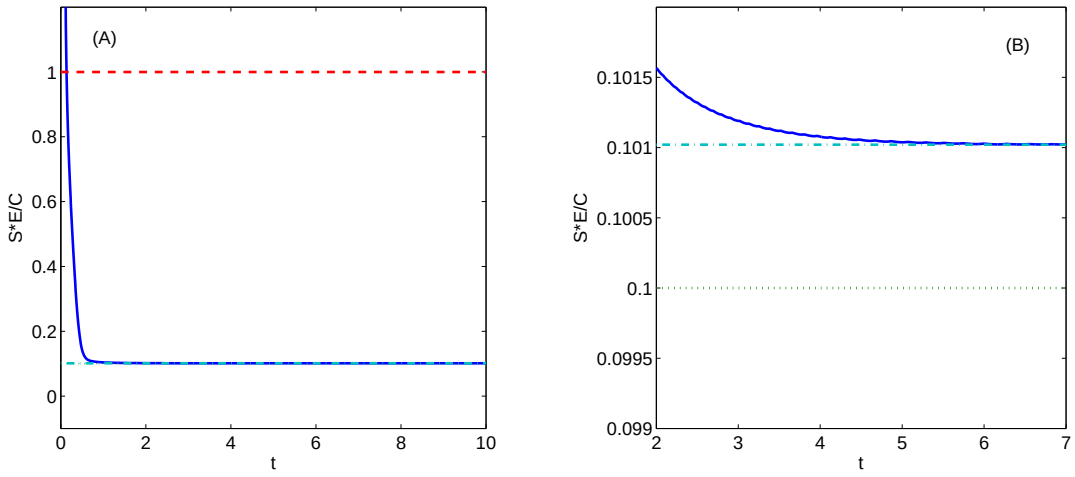


Figure 4: (A) Plot and (B) zoom of $\frac{ES}{C}$ for a=1, k=0.9, d=0.1, $S_T = 100$, $E_T = 89$. Solid line: numerical solution of the full system; dashed: K_M; dashed-dotted: K_W; dotted: K_D.

3 Use and misuse of the Quasi-Steady State Approximation (sQSSA)

The roles of V_{max}, the maximal reaction velocity, and K_M, the Michaelis constant, become essential when characterizing biochemical reactions *in vitro* as well as *in vivo*. Moreover, the description of cooperative reactions, inhibition and many other biochemical processes have up to now exploited the fundamental ideas of the MM scheme, i. e., the sQSSA and the parameters V_{max} and K_M (see, e.g., [3]). However, these approximations cannot be expected to be valid *in vivo*. Figure 1 shows that, for particular values of the parameters and the initial conditions, the sQSSA cannot be adequate to approximate the solutions of the full system at the beginning of the process, failing widely also to approximate the time in which the system reaches its equilibrium. This is due to the fact that (7) is not always fulfilled. The dependence of the product velocity

$$v := \frac{dP}{dt} = kC \tag{12}$$

on the concentration of S is based on the *a priori* (and not always true) assumption that the sQSSA is valid. In this case

$$v = kC \cong \frac{V_{max} \cdot S}{K_M + S} \ . \tag{13}$$

Consequently V_{max} is usually intended as the limit of the "initial velocity" for the S concentration tending to infinity and K_M as the value of S such that

$$v(S = K_M) = \frac{V_{max}}{2} \ . \tag{14}$$

 ISBN 13: 978-84-613-5510-5

Since the tQSSA is much more appropriate than the sQSSA, we can use formula (10) and very simple algebra to define in a more appropriate way K_M (if $S_T > K_M$):
i) when the value of the total substrate is equal to $\overline{S} = K_M + \frac{E_T}{2}$, then the rate of P is equal to $\frac{V_{max}}{2}$:

$$v\left(\overline{S} = K_M + \frac{E_T}{2}\right) = \frac{V_{max}}{2} \tag{15}$$

This result can also be found in [37]. Let us remark, by the way, that if we used the Tzafriri approximating formula , we would obtain the following definition:
ii) when the value of the total substrate is equal to $\overline{S} = K_M + E_T$, then the velocity of P is equal to $\frac{V_{max}}{2}$:

$$v\left(\overline{S} = K_M + E_T\right) = \frac{V_{max}}{2} \tag{16}$$

Then the estimate given by (16) becomes largely incorrect for high values of E_T.

4 The equilibrium constant revisited

Though the sQSSA is based on the approximation $\frac{dC}{dt} \cong 0$, several biochemistry textbooks (see for example [21, 42, 30, 14]), in order to simplify the mathematics, consider the approximation as a true equality, leading to a misinterpretation of the QSSA. As a consequence, the Michaelis constant is determined by equating to zero the right hand side of the second equation of (2) [42, 30, 14], obtaining

$$K_M = \frac{E \cdot S}{C} = \frac{(E_T - C) \cdot S}{C} \ . \tag{17}$$

Actually, as shown in Figure (1), the derivative of C is equal to zero only at time $t = t_{max}$, when C reaches its maximum value. Consequently we cannot declare that the right hand side in (17) remains constant. On the other hand, we could interpret K_M as the equilibrium value for $\frac{E \cdot S}{C}$, reached for large t (supposing that no degradation, product inhibition or back reaction phenomena are involved), in the same way as the dissociation constant K_D is interpreted in the original Michaelis-Menten reaction, where $k = 0$ [30]. Actually, while this last reaction, which is completely reversible, reaches a steady-state where both S and C are different from zero, in reaction (1), as remarked above, S and C tend to zero and consequently we cannot use (17), which gives an undefined ratio, for $t \to \infty$. Thus the equality $K_M = \frac{E \cdot S}{C}$ is valid for every reaction only at $t = t_{max}$. We can however try to solve the indetermination of the ratio for $t \to \infty$ in the following way. From Figure (1) we can observe that, after the transient phase, all the reactants seem to follow asymptotically an exponential behavior, with negative exponent. If we suppose that the asymptotic decay of C is proportional to $e^{-\alpha t}$, for some α, formula (12) implies that also $S_T - P$ will be asymptotically proportional to $e^{-\alpha t}$. By means of the conservation laws (4) we can conclude that also S and $E_T - E$

 ISBN 13: 978-84-613-5510-5

will follow the same asymptotic behavior as C. Thus let us expand S and C in powers of $\mathrm{e}^{-\alpha t}$: we have

$$S(t) = S_0 + S_1\,\mathrm{e}^{-\alpha t} + S_2\,\mathrm{e}^{-2\alpha t} + o(\mathrm{e}^{-2\alpha t}) \tag{18}$$

$$C(t) = C_0 + C_1\,\mathrm{e}^{-\alpha t} + C_2\,\mathrm{e}^{-2\alpha t} + o(\mathrm{e}^{-2\alpha t}) \tag{19}$$

After some computations, we get then

$$S_{as}(t) \cong S_1\,\mathrm{e}^{-\alpha t} \tag{20}$$

$$C_{as}(t) \cong \frac{\alpha}{k-\alpha}\,S_1\,\mathrm{e}^{-\alpha t} \tag{21}$$

where

$$\alpha = \frac{a}{2}(K_M + E_T)\left[1 - \sqrt{1 - \frac{4kE_T}{a(K_M+E_T)^2}}\,\right] \tag{22}$$

There is still an unknown parameter, S_1, which could be estimated from experimental data via a least-squares procedure.

We are now in position to state the main results of this section.

Theorem 4.1. *For $t \to \infty$*

$$\frac{E\,S}{C}(t) \cong \frac{E_{as}\,S_{as}}{C_{as}}(t) \to \left(\frac{k-\alpha}{\alpha}\right) E_T =: K_W \tag{23}$$

The constant K_W, here introduced for the first time, gives the exact asymptotic value of the ratio $\dfrac{E\,S}{C}$ and, in contrast with biochemical literature [21, 42, 30, 14], in general is different from K_M. This result is clearly illustrated in Figures (2) - (4), where we have plotted the time course of the ratio $\dfrac{E\,S}{C}$, where the values E, S, C are obtained by the numerical integration of system (2) - (4). Finally, let us state some important properties of K_W.

Theorem 4.2. *For any admissible choice of the kinetic parameters and the initial conditions, the following inequalities hold:*

$$K_D \le K_W \le K_M\ . \tag{24}$$

Varying appropriately the parameter values, we can obtain for K_W every value between K_D and K_M. In particular,

Theorem 4.3. *For any admissible choice of the kinetic parameters and for any $\bar{K} \in (K_D\,, K_M)$, there exists $\bar{E}_T$ such that $\frac{ES}{C} \to \bar{K}$ when $t \to \infty$.*

 ISBN 13: 978-84-613-5510-5

References

[1] K. R. Albe, M. H. Butler and B. E. Wright, *Cellular Concentration of Enzymes and Their Substrates*, J. Theor. Biol., 143 (1990), pp. 163–195.

[2] D. Barik, M. R. Paul, W. T. Baumann, Y. Cao and J. J. Tyson, *Stochastic Simulation of Enzyme-Catalized Reactions with Disparate Time Scales*, Biophys. J., 95 (2008) pp. 3563–3574.

[3] H. Bisswanger, *Enzyme Kinetics. Principles and Methods*, Wiley-VCH, Weinheim, (2002).

[4] N. Blütghen, *Sequestration Shapes the Response of Signal Transduction Cascades*, IUMBM Life, 58 (2006), pp. 659–663.

[5] N. Blütghen, F. J. Bruggermann, S. Legewie, H. Herzel, H. V. Westerhoff and B. N. Kholodenko, *Effects of sequestration on signal transduction cascades*, FEBS J., 273 (2006), pp. 895–906.

[6] J. Borghans, R. de Boer and L. Segel, *Extending the quasi-steady state approximation by changing variables*, Bull. Math. Biol., 58 (1996), pp. 43–63.

[7] G. E. Briggs and J. B. S. Haldane, *A note on the kinetics of enzyme action*, Biochem. J., 19 (1925), pp. 338–339.

[8] M. S. Calder and D. Siegel, *Properties of the Michaelis/Menten mechanism in phase space*, J. Math. Anal. Appl., 339 (2008), pp. 1044–1064.

[9] V. Chickarmane, B. N. Kholodenko and H. M. Sauro, *Oscillatory dynamics arising from competitive inhibition and multisite phosphorylation*, J. Theor. Biol., 244 (2006), pp. 68–76.

[10] A. Ciliberto, F. Capuani and J. J. Tyson, *Modeling networks of coupled enzymatic reactions using the total quasi-steady state approximation*, PLoS Comput. Biol., 3 (2007), pp. 463–472.

[11] J. W. Dingee and A. B. Anton, *A New Perturbation Solution to the Michaelis-Menten Problem*, AIChE J., 54 (2008), pp. 1344–1357.

[12] E. H. Flach and S. Schnell, *Use and abuse of the quasi-steady-state approximation*, IEE Proc.-Syst. Biol., 153 (2006), pp. 187–191.

[13] S. J. Fraser, *The steady state and equilibrium approximations: A geometrical picture*, J. Chem. Phys., 88 (1988), pp. 4732–4738.

[14] G. G. Hammes, *Thermodynamics and kinetics for the biological sciences*, Wiley-Interscience, (2000).

 ISBN 13: 978-84-613-5510-5

[15] F. G. Heineken, H. M. Tsushiya and R. Aris, *On the Mathematical Status of the Pseudo-steady State Hypothesis of Biochemical Kinetics*, Math. Biosc., 1 (1967), pp. 95–113.

[16] V. Henri, *Recherches sur la loi de l'action de la sucrase*, C. R. Hebd. Acad. Sci., 133 (1901), pp. 891–899.

[17] V. Henri, *Über das Gesetz der Wirkung des Invertins*, Z. Phys. Chem., 39 (1901), pp. 194–216.

[18] V. Henri, *Théorie générale de l'action de quelques diastases*, C. R. Hebd. Acad. Sci., 135 (1902), pp. 916–919.

[19] B. N. Kholodenko, *Negative feedback and ultrasensitivity can bring about oscillations in the mitogen–activated protein kinase cascade*, Eur. J. Biochem., 267 (2000), pp. 1583–1588.

[20] K. J. Laidler, *Theory of the transient phase in kinetics, with special reference to enzyme systems*, Can. J. Chem. 33, (1955), pp. 1614–1624.

[21] A. L. Lehninger, *Lehninger Principles of Biochemistry*, W.H. Freeman & Company, 2008.

[22] N. I. Markevich, J.B. Hoek and B. N. Kholodenko, *Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades*, J. Cell. Biol., 164 (2004), pp. 353–359.

[23] S. MacNamara, A. M. Bersani, K. Burrage and R. B. Sidje, *Stochastic chemical kinetics and the total quasi-steady-state assumption: application to the stochastic simulation algorithm and chemical master equation*, J. Chem. Phys., 129 (2008), pp, 095105-1/095105-13.

[24] L. Michaelis and M. L. Menten, *Die kinetik der invertinwirkung*, Biochem. Z., 49 (1913), pp. 333–369.

[25] L. Noethen and S. Walcher, *Quasi-steady state in Michaelis-Menten system*, Nonlinear Anal., 8 (2007), pp. 1512–1535.

[26] M. G. Pedersen, A. M. Bersani and E. Bersani, *The Total Quasi Steady-State Approximation for Fully Competitive Enzyme Reactions*, Bull. Math. Biol., 69 (2005), pp. 433–457.

[27] M. G. Pedersen, A. M. Bersani, E. Bersani and G. Cortese, *The Total Quasi-Steady State Approximation for Complex Enzyme Reactions*, Matematics and Computers in Simulation (MATCOM), 79 (2008), pp. 1010–1019.

[28] M. G. Pedersen, A. M. Bersani and E. Bersani, *Quasi Steady-State Approximations in Intracellular Signal Transduction – a Word of Caution*, J. Math. Chem., 43 (2008), pp. 1318–1344.

 ISBN 13: 978-84-613-5510-5

[29] M. G. Pedersen and A. M. Bersani, *The Total Quasi-Steady State Approximation Simplifies Theoretical Analysis at Non-Negligible Enzyme Concentrations: Pseudo First-Order Kinetics and the Loss of Zero-Order Ultrasensitivity*, J. Math. Biol., 60 (2010), pp. 267–283.

[30] N. C. Price and L. Stevens, *Fundamentals of Enzymology*, Oxford Univ. Press, (1989).

[31] S. Schnell and P. K. Maini, *A century of enzyme kinetics. Reliability of the K_M and v_{max} estimates*, Comments Theor. Biol., 8 (2003), pp. 169–187.

[32] L. A. Segel, *On the validity of the steady-state assumption of enzyme kinetics.* Bull. Math. Biol., 50 (1988), pp. 579–593.

[33] L. A. Segel and M. Slemrod, *The quasi steady-state assumption: a case study in pertubation*, SIAM Rev., 31 (1989), pp. 446–477.

[34] A. Sols and R. Marco, *Concentrations of metabolites and binding sites, Implications in metabolic regulation*, Curr. Top. Cell. Regul., 2 eds. B. Horecker and E. Stadtman (1970), pp. 227–273.

[35] P. A. Srere, *Enzyme Concentrations in Tissues*, Science, 158 (1967), pp. 936–937.

[36] O. H. Straus and A. Goldstein, *Zone Behavior of Enzymes*, J. Gen. Physiol., 26 (1943), pp. 559–585.

[37] P. Toti, A. Petri, V. Pelaia, A. M. Osman, M. Paoloni and C. Bauer, *A linearization method for low catalytic activity enzyme kinetic analysis*, Biophys. Chem., 114 (2005), pp. 245–251.

[38] A. R. Tzafriri, *Michaelis-Menten kinetics at high enzyme concentrations*, Bull. Math. Biol., 65 (2003), pp. 1111–1129.

[39] A. R. Tzafriri and E. R. Edelman, *The total quasi-steady-state approximation is valid for reversible enzyme kinetics*, J. Theor. Biol., 226 (2004), pp. 303–313.

[40] A. R. Tzafriri and E. R. Edelman, *Quasi-steady-state kinetics at enzyme and substrate concentrations in excess of the Michaelis-Menten constant*, J. Theor. Biol., 245 (2007), pp. 737–748.

[41] N. G. Walter, *Michaelis-Menten is dead, long live Michaelis-Menten!*, Nat. Chem. Biol., 2 (2006), pp. 66–67.

[42] E. N. Yeremin, *The foundations of chemical kinetics*, MIR Pub., Moscow (1979).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Integrating Manufacturing Execution and Business Management systems with soft computing

Alba Berzosa[1], Javier Sedano[1], José R. Villar[2], Emilio S. Corchado[3] and Enrique de la Cal[2]

[1] *Department of Artificial Intelligence and Applied Electronics, Castilla y León Technological Institute*

[2] *Department of Computer Science, University of Oviedo*

[3] *Department of Computer Science and Automatica, University of Salamanca*

emails: alba.berzosa@itcl.es, javier.sedano@itcl.es, villarjose@uniovi.es, escorchado@usal.es, delacal@uniovi.es

Abstract

A Manufacturing Execution System (MES) is a highly complex, large, multi-task application that is used to manage production in companies and factories. It monitors and tracks every aspect of all factory-based manufacturing processes. One of the challenges of a MES is to find ways of integrating it with other information techology (IT) systems; i.e., business process management (BPM) systems, so that compatible information may be shared between both systems. This work studies the integration of a local company MES into a BMP to assist with budgeting, in which a data set is gathered from the MES and a soft computing model helps the expert with cost-level estimation. Various modelling methods are used, such as fuzzy rule based ones, in order to determine whether white box or black box models are suitable for the task. The results of the study show how information may be integrated between manufacturing and business management software.

Key words: Manufacturing Execution Systems, Fuzzy Rule Based Systems, Applied Soft Computing

1 Introduction

Over recent years, the presence of IT applications in industry has increased considerably. IT has been applied to different tasks such as assisting with production or on-line process management and manufacturing, which includes what are nowadays known as Enterprise Requirements Planning (ERP) and Manufacturing Resources Planning (MRP) [11, 19]. Manufacturing Execution Systems (MES) are information systems that

 ISBN 13: 978-84-613-5510-5

are used to manage the way in which manufacturing resources -equipment, employees and inventories- are planned [2, 18].

The objective of a MES depends on whether it is implemented in the context of a production control system or for manufacturing monitoring and supervision. In the former case, the objective is to provide the company with a research laboratory for products and processes, while in the latter, the MES is considered a computer-aided system that assists with decision-making processes related to manufacturing.

However, designing and deploying a user-friendly MES, which has to fulfil the above-mentioned objectives, represents a significant challenge, owing in great part to the complexity of the different production systems, plants and products in use. In this study, several soft-computing techniques are applied, in order to assist with budgeting for a plastic products factory. The main objective of this study, however, is to develop a computer-based assistant to detect faults and loss of competitiveness in the production system. The problem is defined in the following section, while in section 3 the chosen models are described and the results are discussed. Finally, the conclusions and future lines of work are outlined.

2 The case of a plastic products factory

In this study, the system will be applied to a plastic products factory in Spain. It manufactures different products, such as tubes, sheets, bags, polypropylene sheets, garbage bags and others. Its production process is divided into a storage area, an extrusion area, and a printing and clothing area.

The schema of the local plastic bags factory is depicted in Figure 1, where the production system is totally supervised and monitored. Each machine includes its own control system based on Programmable Logic Controllers (PLC). There are up to 75 machines, each producing a range of different products. There are also several Human Machine Interfaces (HMIs) all connected to an ethernet network; a Data Acquisition System (DAQ) which collects various process signals, among which pressures and temperatures. The operators can control and operate the machines that are programmed to manufacture the product. Finally, the monitoring and supervising computers are connected to this network to request information from the PLCs and DAQs. This is known as the Manufacturing Control System (MCS). The company has recently started to store all available data in a data-base management system to broaden the capacity of its staff to plan production processes in the factory, as the amount of available data was rather small.

This is the scenario into which the MES has to be integrated. Production dynamics characteristics should firstly be determined. For this purpose, manufacturing conditions in the current operational stage have to be defined, in the form of data that may be gathered from the MCS network. Once the manufacturing dynamics data have been gathered, then a model of the present production operation may be obtained [4]. In other words, the relevant variables for measurement and storage need to be determined.

 ISBN 13: 978-84-613-5510-5

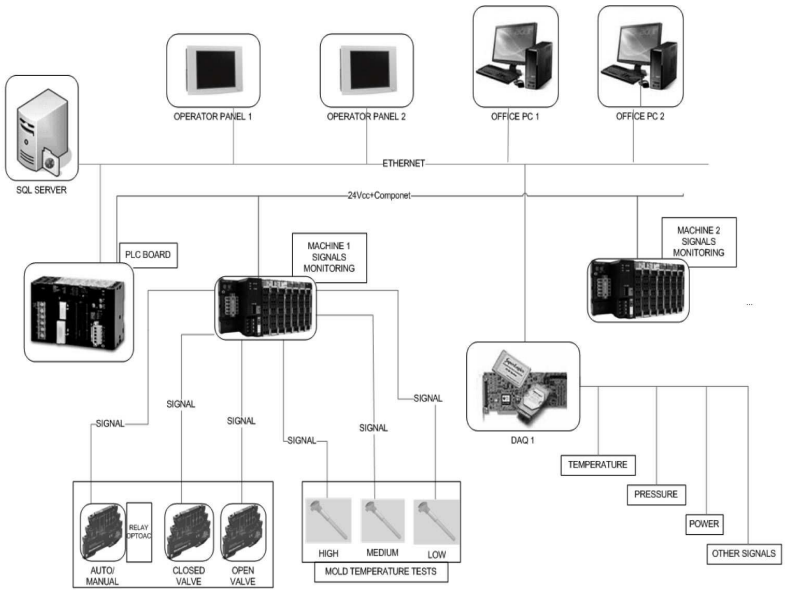


Figure 1: Schematic diagram of the MES installed in the plastic products factory. The PLCs controlling each machine and the DAQs and HMIs connected through the field network constitute the MCS.

2.1 The expected objectives

The final objective of this study is to develop a computer-based assistant to detect faults and loss of competitiveness in the production system. Consequently, the available data from the MCS should be examined in order to design the final data base; rather than storing all the signals, it was only intended to store those signals that were sufficiently informative of the process evolution in the MCS. As this represents a virtually costless task, the factory representative and the research group agreed to present a prototype for a simpler task; the factory would invest in such a system according to the obtained results.

The simpler task involved assisting the staff in budgeting a manufactured product. The working method was as follows: a client requests a product, following which a staff member assigns the job to a certain machine chain and a cost is estimated. This is not automated yet, so before assigning a machine chain, the employee must analyse several plots and reports. So, the challenge was to develop a model to automatically assist the staff in establishing the cost level for a tuple <product, client, machine>. They collected a data set of 1471 examples, including the available historical records of 22 input variables such as client identification, product identification, the machine, the operator, units produced and length of operation, among others. The output of the data set was a variable indicating whether the cost was high, medium or low.

 ISBN 13: 978-84-613-5510-5

3 Generating the models for computer-aided decision making

Several tasks were carried out once the data set was defined. Firstly, the data set had to be analysed and pre-processed, in order to determine whether there were any dependent variables. It was also analysed to decide whether it was necessary to normalise and partition the data. KEEL software was used [1] in all the experimental and modelling stages.

3.1 Soft Computing tools and algorithms used

KEEL stands for Knowledge Extraction based on Evolutionary Learning. KEEL software is a research and educational tool for modelling data mining problems which implements more than one hundred algorithms, including classification, regression, clustering, etc. Moreover, it includes data pre-processing and post-processing algorithms, statistical tests and reporting facilities. Finally, it has a module for data set analysing and formatting, which was used for the first task in this experiment.

As the model would be used as a IT support tool, it was considered desirable to obtain a white box model, such as Fuzzy Rule Based Systems or Decision Trees. Several different techniques provided the ability to manage the type of available data. Different techniques compared the results and the viability of the models. The statistical methods included Quadratic Discriminant Analysis (QDA) [12], the Multinomial Logistic regression model with a ridge estimator (LOG) [3], the Kernel Classifier (KC with 0.01 and 0.05 sigma values) [12], and the K-nearest neighbour (KNN with 1 and 3 K values) [7]. The fuzzy rule-based methods included the Fuzzy Adaboost rule learning method (ADA) [10], the Fuzzy GA-P algorithm (FGAP) [15] and the Ishibuchi Hybrid Fuzzy GBML (HFG) [9]. Finally, the decision tree and decision tree rule-based methods were the well-known C4.5 [13] and C4.5 rule-based methods. (C45R) [14].

In the QDA algorithm, the cost of classifying an example X with class k is calculated through Eq. 1, where π_k is the unconditional prior class k probability estimated from the weighted sample, and μ_k and Σ_k are the population mean vector and covariance matrix for the k class. Hence, an example X is assigned with the minimum cost class as stated in Eq. 2.

$$d_k(X) = (X - \mu_k)^T \Sigma_k^{-1} (X - \mu_k) + \ln |\Sigma_k| - 2 \ln \pi_k \tag{1}$$

$$d_{\hat{k}}(X) = min_{1 \leq k \leq K} d_k(X) \tag{2}$$

The LOG algorithm is based on the standard logistic regression. The probability that the class k correctly classifies the example $X = \{X_1, ..., X_p\}$ is calculated following Eq. 3,where the parameter $\beta = \{\beta_1, \ldots, \beta_p\}$ is estimated, i.e., with the maximum likelihood estimation obtained by maximising Eq. 4. It is classified in the class with the higher probability, as in the example.

 ISBN 13: 978-84-613-5510-5

$$p(k|X) = \frac{\exp(\sum_{j=1}^{p} \beta_j X_j)}{1 + \exp(\sum_{j=1}^{K} \beta_j X_j)} \tag{3}$$

$$l(\beta) = \sum_{k} [k \log p(k|X) + \neg k \log\{1 - p(k|X)\}] \tag{4}$$

The Kernel method is a classifier that uses the Bayes rule using a "non-parametric estimation of the density functions through a Gaussian kernel function" as stated in [8]. In the KEEL software, covariance matrix tuning is carried out by means of an ad-hoc method. On the other hand, the K-nearest neighbour method classifies the example X with the majority class in the K examples of the data set with a shorter distance to X. Note that the use of the KNN implies that a metric is defined in the space to measure the distance between examples.

The Fuzzy Adaboost method is based on boosting N weak fuzzy classifiers (that is, N unreliable fuzzy classifiers are weighted according to their reliability) so that the whole outperforms each of the individual classifiers. Moreover, each example in the training data set is also weighted and tuned in relation to the evolution of the whole classifier.

The GAP is a Fuzzy Rule-Based Classifier learned using the Genetic Programming principles but using the Simulated Annealing algorithm to mutate and to evolve both the structure of the classifier and the parameters. At each iteration, the whole Fuzzy Rule set will evolve.

The Ishibushi Hybrid Fuzzy Genetic Based Machine Learning method represents a Pittsburgh style genetic learning process which is hybridised with the Michigan style evolution schema: after generating the $(N_{pop} - 1)$ new Fuzzy Rule sets, a Michigan style evolutionary scheme is applied to each of the rules for all the individuals. Recall that each individual is a complete Fuzzy Rule set.

The Ishibushi Hybrid Fuzzy Genetic Based Machine Learning method represents a Pittsburgh-style genetic learning process which is hybridised with the Michigan style evolution schema: after generating the $(N_{pop} - 1)$ new Fuzzy Rule sets, a Michigan style evolutionary scheme is applied to each of the rules for all the individuals. Recall that each individual is a complete Fuzzy Rule set.

Finally, the C4.5 algorithm is a well-known decision-tree method based on information entropy and information gain. A node in the decision tree is supposed to discriminate between examples of a certain class based on a feature value. At each node, the feature that produces the higher normalised information gain is then chosen. In the case of C4.5R, the decision tree is presented as rules, where each node in the path from the root to a leaf is considered an antecedent of the rule. These rules are then filtered to eliminate redundant or equivalent rules.

3.2 The experimentation and results

After analysing the original data set it was found that most of the examples corresponded to the tuning of the plant, which could therefore be discarded. In addition,

 ISBN 13: 978-84-613-5510-5

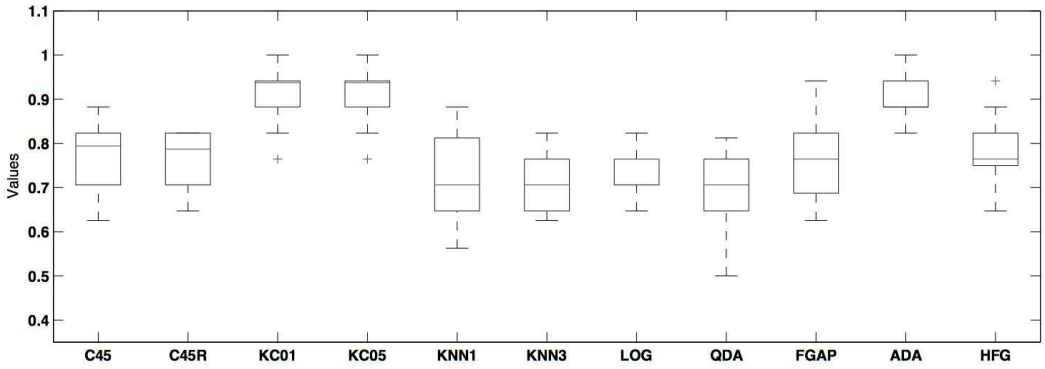


Figure 2: Boxplot of the classifiers results for the {Medium, High} experiments.

there was also a large quantity of totally erroneous samples, which were also discarded. Finally, the data set included 168 examples corresponding to 9 machines.

Several relationships were found, such as the one between the number of faulty units and the weight of discarded material. In the end, the data set included information on the product, the machine, client identification and the number of units to produce. The output variable was the class of the cost level, which could be Low, Medium or High.

The second task consisted of the modelling step, in which the modelling algorithm had to be chosen and the statistical tests carried out. The 9 methods described in the previous Sub-Section were used to obtain a classifier.

Two series of experiments were designed. The first experiment generated two classifiers. On the one hand, one discriminated between Low and ¬Low classes, on the other hand, the second classifier, which was run when a ¬Low example was found, discriminated between Medium and High classes. As a result of the first experiment, two different data sets were generated: one contained the examples characterised with class Low or ¬Low, and another one contained only the ¬Low examples characterised by the corresponding class Medium or High. The second experiment made use of all 150 examples in the data set to generate a 3-class classifier. Finally, in both cases, since the number of examples was so small, the 10-fold cross-validation schema was selected and performed in a KEEL environment.

The results from the first experiment are presented in Table 1, Figure 2 and Figure 3. As it can be seen, the kernel methods and Fuzzy AdaBoost, although not interpretable, were found to be the best models. On the other hand, in view of the results and considering the standard deviation of the FGAP and the HFG algorithms, it could be said that these two methods may improve their performance by means of a better definition of their parameters (population and sub-population sizes, number of islands, etc.) and a larger number of generations. It is worth remarking on the ease with which the problem of discriminating between Medium and High may be solved, provided no Low class classifications are involved.

 ISBN 13: 978-84-613-5510-5

	{Low, ¬Low}			{Medium, High}		
	GCE	SGCE	CC	GCE	SGCE	CC
C4.5	0.2276	0.0748	0.7724	0.1018	0.1220	0.8982
C4.5R	0.2324	0.0620	0.7676	0.1018	0.1230	0.8982
KC01	0.0949	0.0651	0.9051	0.0949	0.0651	0.9051
KC05	0.1143	0.0879	0.8857	0.1018	0.0758	0.8982
KNN1	0.2860	0.1002	0.7140	0.2464	0.1746	0.7536
KNN3	0.2857	0.0695	0.7143	0.3107	0.2295	0.6893
LOG	0.2504	0.0530	0.7496	0.0750	0.0829	0.9250
QDA	0.3040	0.0858	0.6960	0.0911	0.0820	0.9089
FGAP	0.2335	0.0973	0.7665	0.0893	0.0810	0.9107
ADA	0.0945	0.0598	0.9055	0.0500	0.0829	0.9500
HFG	0.2206	0.0800	0.7794	0.0750	0.0829	0.9250

Table 1: Mean results of the classifiers for the {Low, ¬Low} {Medium, High} experiments. GCE, SGCE and CC stand for Global Classification Error, standard deviation of the GCE and the percentage of correctly classified examples.

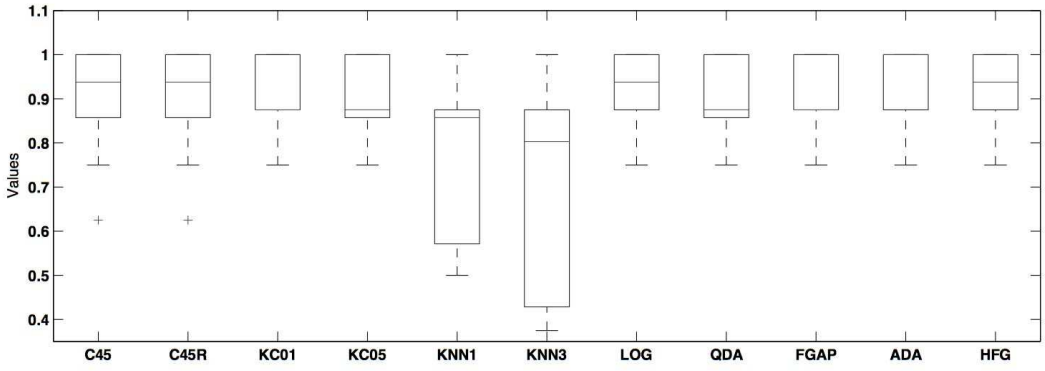


Figure 3: Boxplot of the classifiers results for the {Medium, High} experiments.

 ISBN 13: 978-84-613-5510-5

	GCE	SGCE	CC
C4.5	0.2974	0.0441	0.7026
C4.5R	0.3103	0.0967	0.6897
KC01	0.1077	0.0648	0.8922
KC05	0.1077	0.0531	0.8923
KNN1	0.3445	0.0796	0.6555
KNN3	0.3684	0.1120	0.6316
LOG	0.2434	0.0840	0.7566
QDA	0.3338	0.0857	0.6662
FGAP	0.4118	0.0975	0.0975
ADA	0.1783	0.0785	0.8217
HFG	0.3857	0.0799	0.6143

Table 2: Mean results for the {Low, Medium, High} classifier experiment. GCE, SGCE and CC stand for Global Classification Error, standard deviation of the GCE and the percentage of correctly classified examples.

The results of the first experiment did not prepare us for the results of the second experiment. A much poorer performance of the methods was observed, despite method C4.5, which is unable to manage a three-class problem. Only the kernel methods keep track of the problem. The reason for these results is related to the kind of features involved in the modelling; several of them being integer valued features with an unknown upper limit. As an example, the number of units to be produced is quite dependent on the machine, as each machine has a maximum production rate. But this data was not given for the experimentation, so it was not possible to normalize those variables which, in turn, make the classifier worse.

A main conclusion may be drawn from this experimentation: the data set should be more informative and representative of the problem, if better models are to be generated. The company should rely on an in-depth analysis of available data and measurements, but it is also necessary for it to study the relationships between the variables under study, i.e. using Cooperative Maximum Likelihood Hebbian Learning (CMLHL) [6] as shown in [17, 16]. The results illustrate the way in which the research team may help the company to design their MES.

4 Conclusions and future work

A MES development to improve its capacity and link up with other business management applications has been tested in this work. A computer assisted-budgeting problem has been solved through the application of different computing techniques. Nevertheless, it was shown that the data gathered from a MCS must be carefully chosen and the amount of data should be representative and informative of the real process. A clear list of the objectives to be accomplished by the MES should be prepared prior to the collection and analysis of relevant data.

 ISBN 13: 978-84-613-5510-5

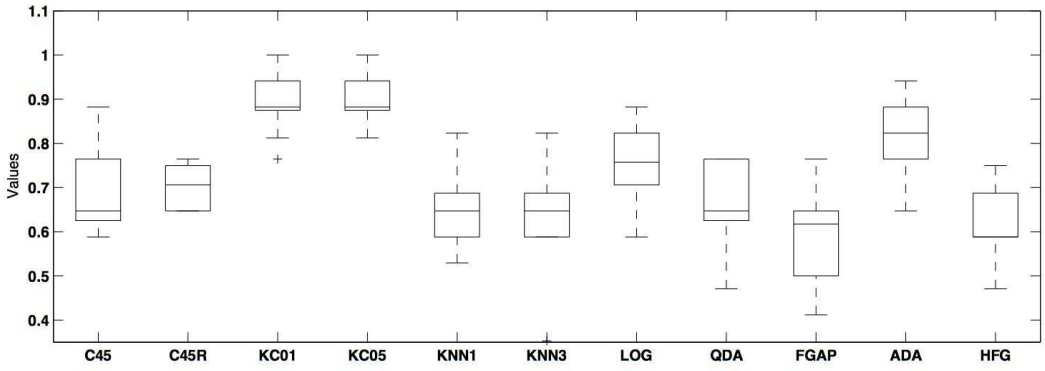


Figure 4: Boxplot of the classifiers results for the {Low,Medium, High} experiments.

Future work will include modelling the relationships between operators, machines, products and the overall performance of the plant, so that resource planning may be introduced. More knowledge and data should be gathered from the plant, such as machine operating limits. Finally, a complete analysis of the data through the use of well-known techniques (such as CMLHL) would contribute to reliable MES design and engineering.

Acknowledgements

This research work has been funded by the Spanish Ministry of Science and Innovation, under grant TIN2008-06681-C06-04 and the Spanish Ministry of Science and Innovation through project PID 560300-2009-11

References

[1] J. Alcalá-Fdez, L. Sánchez, S. García, M.J. del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, J.C. Fernández, F. Herrera, *KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems*, Soft Computing **13:3** (2009) 307–318.

[2] Apriso Corporation, *Manufacturing Execution Systems Strategy Update: Trends and Tips for 2010*, http://www.bitpipe.com/detail/RES/1268690867_382.html.

[3] S. le Cessie and J.C. van Houwelingen, *Ridge Estimators in Logistic Regression*, Applied Statistics **41:1** (1992) 191-201.

[4] R. S. Chen and Y. S. Tsai and C. C. Chang, *Design and implementation of an intelligent manufacturing execution system for semiconductor manufacturing*

 ISBN 13: 978-84-613-5510-5

industry, Proceedings of the 2006 IEEE International Symposium on Industrial Electronics (2006) 2948–2953.

[5] B. K. Choi and B. H. Kim, *MES (manufacturing execution system) architecture for FMS compatible to ERP (enterprise planning system)*, International Journal of Computer Integrated Manufacturing **15:3** (2002) 274–284.

[6] E. Corchado and C.Fyfe, *Connectionist techniques for the identification and suppression of interfering underlying factors*, International Journal of Pattern Recognition and Artificial Intelligence **17:8** (2003) 1447-1466.

[7] T.M. Cover and P.E. Hart, *Nearest Neighbor Pattern Classification*, IEEE Transactions on Information Theory **13** (1967) 21-27.

[8] Salvador García and Francisco Herrera, *An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons*, Journal of Machine Learning Research **9** (2008) 2677-2694.

[9] H. Ishibuchi and T. Yamamoto and T. Nakashima, *Hybridization of Fuzzy GBML Approaches for Pattern Classification Problems*, IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics **35:2** (2005) 359-365.

[10] M.J. del Jesus and F. Hoffmann and L. Junco and L. Sánchez, *Induction of Fuzzy-Rule-Based Classifiers With Evolutionary Boosting Algorithms*, IEEE Transactions on Fuzzy Systems **12:3** (2004) 296-308.

[11] M. McClellan, *Introduction to Manufactoring Execution Systems*, Proceedings of MES Conference and Exposition, USA 2001.

[12] G.J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*, John Wiley and Sons, 2004.

[13] J.R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kauffman, 1993.

[14] J.R. Quinlan, *MDL and Categorical Theories (Continued)*. Machine Learning: Proceedings of the Twelfth International Conference. Lake Tahoe California (United States of America, 1995) 464-470.

[15] L. Sánchez and I. Couso and J.A. Corrales, *Combining GP Operators With SA Search To Evolve Fuzzy Rule Based Classifiers*, Information Sciences **136:1-4** (2001) 175-192.

[16] J. Sedano and J. R. Villar and E. S. Corchado and L. Curiel and P. M. Bravo, *Modelling a Pneumatic Drill Process by a two-steps AI Model*, International Journal of Computer Mathematics **86:10-11** (2009) 1769-1777.

[17] J. Sedano and L. Curiel and E. Corchado and E. de la Cal and J. R. Villar,*A soft computing method for detecting lifetime building thermal insulation failures*,Integrated Computer-Aided Engineering **10:2** (2010) 103-115.

 ISBN 13: 978-84-613-5510-5

[18] B. S. de Ugarte and A. Artiba and R. Pellerin, *Manufacturing execution system - a literature review*, Production planning and control **20:6** (2009) 525–539.

[19] L. Van Dyk, *Manufacturing execution systems, MEng dissertation*, University of Pretoria, Pretoria, http://upetd.up.ac.za/thesis/available/etd-11092006-125332/, (1999).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Fairness Scheduling for Multi-Cluster Systems Based on Linear Programming

Héctor Blanco[1], Alberto Montañola[1], Fernando Guirado[1] and Josep Lluís Lérida[1]

[1] *Department of Computer Science and Industrial Engineering, University of Lleida*

emails: hectorblanco@diei.udl.cat, alberto@diei.udl.cat, f.guirado@diei.udl.cat, jlerida@diei.udl.cat

Abstract

Clusters of computers that act in a collaborative manner to execute parallel jobs are known as multi-clusters. In a multi-cluster environment, it is possible to treat computational problems that require more resources than those available in only one cluster. However, the degree of complexity of the scheduling process is greatly increased to take advantage of multi-cluster capabilities, and the scheduler must take into account the co-allocation process that distributes the tasks of parallel jobs across cluster boundaries.

In this work, a scheduling strategy is presented based on a linear programming model, which brings together the parallel jobs in the system queue that fit into the system and allocates them simultaneously, instead of assigning them individually as is usual in the literature. The proposed scheduling technique is shown to reduce the execution times of the parallel jobs by about 8% on average, and the waiting times by about a 35% compared with other scheduling techniques in the literature. This reduction in response time provides greater resource utilization and improved overall system performance.

Key words: Multi-cluster systems, co-allocation, job scheduling, mixed integer programming

1 Introduction

Nowadays the use of clusters of computers is becoming common in all kinds of research laboratories or institutions. Computation problems that would require the use of more computational resources than just one of these clusters can offer can be resolved by the use of multiple clusters in a collaborative manner. These environments are known as multi-clusters and are distinguished from grids by their use of dedicated interconnection networks among clusters with a known topology and predictable performance characteristics [1].

 ISBN 13: 978-84-613-5510-5

A critical aspect of exploiting the resources in a multi-cluster environment is the challenge of scheduling [2]. Multi-cluster schedulers can take advantage on distributed resources among different clusters to allocate those jobs that cannot be assigned to one single cluster or to take profit of the underutilized resources. This allocation strategy, known as co-allocation, can maximize the job throughput by reducing the queue waiting times and then the job response times [3]. However, mapping jobs across the cluster boundaries can result in rather poor overall performance when co-allocated jobs contend for inter-cluster network bandwidth. Additionally, the heterogeneity of resources increases the degree of complexity to the scheduling problem [4][5].

The scheduling process in multi-cluster environments can be solved by two approaches [2]; (a) using a multi-scheduler mechanism, where each cluster has its own job queue and scheduler, and all the clusters are coordinated by a global meta-scheduler. In this situation, the job queue of each cluster can be directly accessed by the user in order to allocate those jobs that fit into the cluster, and those jobs that do not fit are delegated to the meta-scheduler to be co-allocated. Or (b), when there is only one global scheduler with a single system queue, where all jobs to be executed in the multi-cluster are waiting to be assigned. This second option is the considered in the present work.

The scheduling strategies by applying co-allocation in multi-cluster environments have awakened great interest in recent years. The work in [2], analyzes the performance of four different scheduling strategies to deal with co-allocation, based on job queues, local to each cluster or global for all of them. This work concludes that using a global scheduler, as in our case, simply allowing co-allocation without any restriction is not desirable and requires more complex strategies. Thus, in the literature, the authors have dealt with the co-allocation process in the multi-cluster environments by developing different approaches. Some of these, applying load-balancing techniques minimize the execution time of the jobs from the system queue [6][7]. The work presented in [8] applies a linear programming based approach for modeling and solving the allocation of jobs by attempting to avoid inter-cluster links saturation. Another point of view is presented in [4], which characterizes the bandwidth requirements of the parallel jobs that are co-allocated in order to minimize the inter-cluster links usage and obtain the lowest execution time. This work was extended in [3], where the computation needs are also taken into account to reduce the execution time of the parallel jobs, while preventing the saturation of the interconnection links.

Nevertheless, those previous works follow a FCFS scheme to allocate the jobs from the system queue. This means that they consider all the available resources to allocate the current job, but as they do not take into account the other jobs in the system queue, the current allocation could affect the next assignments negatively. The main problem, and also the big challenge, is the capacity of the techniques to extent the scheduling process to more than one job at the same time.

In the present work, we extend the Mixed-Integer Programming model (MIP) presented in [3] by adding the power to allocate multiple jobs simultaneously in a heterogenous multi-cluster environment, in order to obtain the best overall performance for a set of parallel jobs. Additionally, we propose a scheduling strategy, named PAS

 ISBN 13: 978-84-613-5510-5

for ***P**ackage **A**llocation **S**trategy*, which selects those jobs from the system queue that can be concurrently executed with the available resources. Once the package of jobs is selected, the MIP model proposed is responsible for finding the best possible allocation for the set of jobs. With our approach, the best resources for each parallel job are obtained considering the other applications that can be executed concurrently in the multi-cluster environment, and thus, the scheduling process will be able to reduce the global response times while making better use of the resources and also preventing the saturation of the inter-cluster links.

The rest of the paper is organized as follows. In Section 2, we present the scheduling strategy proposal based on minimizing the global response time for multiple allocated jobs by using a Mixed-Integer Programming model that takes into account the heterogeneity and non-dedicated nature of multi-cluster resources. Section 3 presents the experimentation and the results obtained from comparison with other scheduling strategies presented in the literature. Finally, the conclusions and future work are presented in Section 4.

2 Multiple Job Scheduling Strategy

In [3] we presented a new execution time model for parallel jobs. The goodness of this model was that it defines the execution time by considering both processing resource availability and communication resource utilization. This model was applied in a Mixed-Integer Programming model (MIP) in order to find the allocation of the current job that minimizes its execution time, while avoiding the negative effects of sharing the communication links and processing resources. The drawback of our previous proposal was that jobs in the system queue were selected and scheduled individually in a FCFS manner, allocating the best resources for each job without considering the effects on the execution time of the remaining jobs in the system queue.

One of the main contributions of the present work is the improvement of the Mixed-Integer Programming model presented in order to achieve the best possible allocation of multiple jobs, i.e. one that provides the minimum overall execution time.

In subsection 2.1, the execution time model of parallel jobs in heterogeneous multi-cluster environments is defined and in subsection 2.2, we define the multiple job allocation problem as a MIP (Mixed-Integer Programming) model where the best solution is the one that minimizes the global execution time of a set of jobs. Finally in subsection 2.3 the proposed multiple job scheduling strategy is presented.

2.1 Multi-cluster and Parallel Job Models

We define a multi-cluster as a collection of arbitrary sized clusters with heterogeneous resources. Each cluster has its own internal switch. Clusters are connected to each other by single dedicated links by means of a central switch.

Formally, a multi-cluster can be defined as a system comprised by α heterogeneous clusters $\{C_1..C_\alpha\}$ interconnected by means of dedicated links $\{\mathcal{L}_1..\mathcal{L}_\alpha\}$, where each cluster C_i $(1 \leq i \leq \alpha)$ is also made up of β_i nodes $C_i = \{N_i^1..N_i^{\beta_i}\}$, see Figure 1.

 ISBN 13: 978-84-613-5510-5

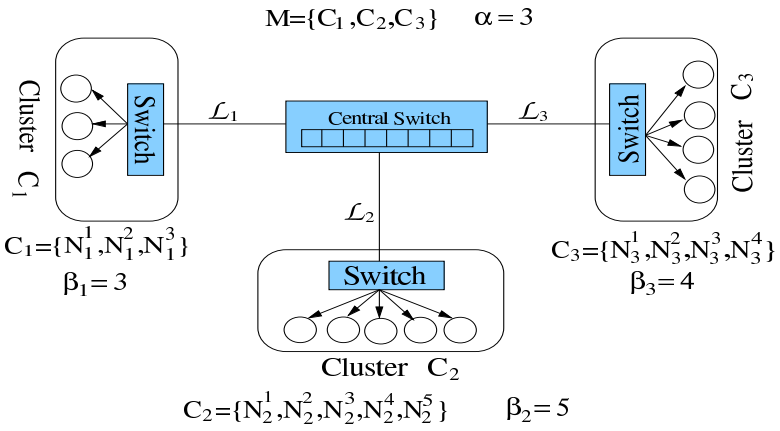


Figure 1: Diagram of a multi-cluster topology

We assume that parallel jobs are not supposed to be malleable, the processing and communicating requirements of every job task are very similar, and the job tasks follow an all-to-all communication pattern.

Taking this assumptions into account, the execution time (T_e) of a parallel job in a heterogeneous and non-dedicated environment can be defined as its execution time in a dedicated environment ($\overline{T_e}$) delayed by a slowdown factor (SD) produced by the heterogeneity and non-dedicated nature of the slowest allocated resources, and expressed by equation 1.

$$T_e = \overline{T_e} \cdot SD \tag{1}$$

However, the slowdown of a parallel application depends on the capacity and availability of resources of both computation and communication, and thus, we can express SD based on processing SP and communication SC slowdowns by equation 2.

$$SD = \sigma \cdot SP + (1 - \sigma) \cdot SC \tag{2}$$

where σ denotes the weighting factor that measures the relevance of the processing time with respect to the communication time of the corresponding job.

2.1.1 Processing Characterization

We assume that parallel job tasks are generally similar in size and they are executing separately, and thus, the job execution time is bounded by the slowest allocated resource. Taking this into account, the job processing slowdown (SP) is obtained from the allocated resource with maximum processing slowdown, expressed by equation 3.

$$SP^j = max\{SP_r | r \in \mathcal{P}^j\} \tag{3}$$

 ISBN 13: 978-84-613-5510-5

where $\mathcal{P}^j$ denotes the set of processing nodes allocated to job j. In a heterogeneous and non-dedicated environments the computing resources capabilities can be quite different. To measure these differences we use the Effective Power metric (Γ_r) defined in [3], which relates the computing power of each resource with its availability. Thus, $\Gamma_r = 1$ when resource r has full capacity to run tasks at full speed, and otherwise $\Gamma_r < 1$. Assuming this, the processing slowdown of such resource SP_r is inversely proportional to its Effective Power weight, $SP_r = (\Gamma_r)^{-1}$.

2.1.2 Communication Characterization

The parallel job co-allocation consumes a certain amount of bandwidth across inter-cluster network links (BW_k^j). These are shown by equation 4.

$$BW_k^j = \left(t_k^j \cdot PNBW^j\right) \cdot \left(\frac{n_T^j - t_k^j}{n_T^j - 1}\right), \quad \forall\, k \in 1..\alpha \tag{4}$$

where n_T^j is the total number of tasks of the job j, t_k^j denotes the total number of tasks allocated to cluster C_k and $PNBW^j$ is the average per-node bandwidth requirement by job j from the jobs. The first term in the equation is the total bandwidth required by all the nodes associated with job j in cluster C_k. The second term represents the communication percentage of job j in other cluster nodes (not in C_k) that will use the inter-cluster link k.

The degree of saturation of inter-cluster links relates the available bandwidth of each link (ABW_k) with the bandwidth requirements of the allocated parallel applications, which is calculated by equation 5.

$$BW_k^{sat} = \frac{ABW_k}{\sum_{j,k} BW_k^j} \quad \forall\, k \in 1..\alpha \tag{5}$$

When the required bandwidth is lower than the available, the link is not saturated and the communications will not suffer delays. Otherwise, the network link is saturated drastically reducing the performance of all the jobs sharing the link.

Thus, the job communicating slowdown (SC) is obtained from the slowest, most saturated, communication link used by the job, calculated as the inverse of the saturation bandwidth by equation 6.

$$SC^j = max\{(BW_k^{sat})^{-1} | k \in 1..\alpha\} \tag{6}$$

2.2 Mixed-Integer Programming Model

In [3] we presented a Mixed-Integer Programming model (MIP) in order to find the best allocation for a parallel job, i.e. one that minimizes its execution time. However, the allocation of the best available resources to an application without considering the requirements of the other applications in the system queue impairs the overall system performance.

 ISBN 13: 978-84-613-5510-5

Input arguments:

1. PCK: Queue of jobs to be matched.
2. τ^j: number of tasks making up job $j \in PCK$.
3. $PNBW^j$: per-task bandwidth requirement for each job $j \in PCK$.
4. σ^j: weighting factor that relates the processing and communication time.
5. $\mathcal{L}$: set of inter-cluster links.
6. μ: set of multi-cluster resources.
7. Γ_r: Effective Power weight of node $r \in \mathcal{P}$
8. ABW_l: maximum communication capacity for each inter-cluster l link $\in \mathcal{L}$

Output parameters:

9. $X_{(j,r)}$, $j \in PCK$ and $r \in \mathcal{P}$: $X_{(j,r)}$=1 when j is matched to resource r, and 0 otherwise.
10. $\mathcal{P}_j$: Set of allocated resources to job j, $\mathcal{P}_j = \{r \in \mu \mid X_{(j,r)} = 1 \text{ and } j \in PCK\}$
11. SP^j: processing slowdown. $SP^j = max\{SP_{(j,r)} \mid j \in PCK \text{ and } r \in \mathcal{P}_j\}$.
12. SC^j: communication slowdown. $SC^j = max\{SC_{(j,l)} \mid j \in PCK \text{ and } l \in \mathcal{L}\}$.

Objective Function:

13. $min\{\sum_{1,j} \overline{T_e^j} \cdot SD^j\}$

Constraints:

14. Gang matching.
15. Non inter-cluster link saturation.

Figure 2: MIP model definition for multiple job allocation

The new proposal is described in Figure 2. The objective function and constraints correspond to linear expressions that concern the simultaneous allocation of several parallel jobs on a multi-cluster. In this case, the solution is presented as binary values, 1 or 0, indicating the allocation, or not, of each processing node to each parallel job. The objective function corresponds to the lowest global execution time for a set of jobs, giving us the best possible allocation, which may or may not be co-allocated between different clusters.

In order to find the best allocation, information about job requirements and multi-cluster status are required (lines 1-8). The information about each job j corresponds to the number of tasks (τ^j), the required per-node bandwidth ($PNBW^j$), and the weighting factor (σ^j), which measures the relevance of the processing and communication time in the total job execution time. For multi-cluster resources, its status is specified by the Effective Power weight of each resource (Γ_r) and the availability of the communications links (ABW_l).

The set of output variables (lines 9-12) consists of an array of binary decision variables $X_{(j,r)}$, with values of 1 or 0 when a task of job j is allocated in node r, or not, respectively. The SP^j and SC^j variables, obtained by equations (3) and (6) respectively, represent the processing and communicating slowdowns obtained for each allocated job, and provide the job Slowdown SD^j using equation 2.

 ISBN 13: 978-84-613-5510-5

2.2.1 Objective function

When there are many possible solutions, the objective function defines the quality of the solution. Our main aim was the allocation of multiple jobs in heterogeneous and non-dedicated resources over a multi-cluster system, obtaining the lowest execution time for the job set.

In order to deal with multiple jobs obtaining a fair allocation for all of them, we attempted to minimizing the global execution time of the entire job set PCK. This is done by summing the individual obtained execution times for all the allocated jobs, as shown in equation 7.

$$min \left\{ \sum_{1,j} \overline{T_e^j} \cdot SD^j \right\} \tag{7}$$

where the execution time for each allocated job is expressed based on the execution time measured in a dedicated environment $\overline{T_e^j}$ lengthened by the job slowdown SD^j obtained by equation 2.

2.2.2 Constraints

The constraints (lines 14-15) define a feasible matching scheme. In this model, two main constraints that must be satisfied are defined, the *gang matching* and *non-saturation* of the inter-cluster links.

The gang matching constraint ensures that all the tasks in each job are assigned, according to equation 8.

$$\sum_{j \in PCK, r \in \mathcal{P}} X_{(j,r)} = \tau^j \tag{8}$$

where τ^j is the number of tasks for the job j. This ensures that the sum of resources allocated for each job j corresponds to its number of tasks.

The non-saturation constraint ensures that the bandwidth consumed on inter-cluster links once the set of jobs is allocated, does not exceed the total capacity of these links, thus preventing saturation and delay of the parallel jobs. This constraint is formalized by equation 9.

$$SC^j \leq 1, \forall j \in PCK \tag{9}$$

where SC^j is the communication slowdown calculated by eq. (6), for each allocated job j.

2.3 Package Allocation Scheduling Strategy

A common feature of most on-line scheduling strategies in cluster, multi-cluster and grid environments, is the individual allocation of resources to applications. First, the strategy selects the next job to be executed by order of arrival or according to a priority criteria. When there are insufficient resources to run the selected job, the scheduler can

 ISBN 13: 978-84-613-5510-5

wait for the release of enough resources in order to follow a *First Come First Served* (FCFS) schema, or select a new job from the system queue that can be executed with the existing resources by applying such a schema as *Fit Processors First Served* (FPFS), *backfilling*, etc. Once a job is selected, it is individually allocated to the most appropriate resources according to the chosen allocation strategy.

However, allocating the best available resources to a job without considering the requirements of the rest of the jobs in the system queue can impair the performance of future allocations and therefore the overall system performance. The proposed strategy, named *Package Allocation Scheduling* (PAS), is able to package those jobs that can be executed in the available resources and allocates them using the MIP model proposed in the previous subsection. To do this, the PAS strategy implements a job selection function ($\mathcal{F}$) that determines the job package that can be simultaneously allocated in a set of multi-cluster resources, selected under certain criteria. This function can be expressed by equation 10.

$$PCK = \mathcal{F}(\mathcal{Q}, \mathcal{R}, \mathcal{C}) \tag{10}$$

where $\mathcal{Q}$ is the set of jobs in the system queue, $\mathcal{R}$ is the set of multi-cluster resources and $\mathcal{C}$ denotes the criteria to be met by resources to accommodate the job package.

In this work, under a FCFS schema, the function $\mathcal{F}$ selects the set of jobs in the system queue that fits in the free multi-cluster resources, that is, those computational nodes non-assigned to other parallel applications. This function is formally expresed by equation 11.

$$\exists\ PCK \subseteq \mathcal{Q} \mid \sum_{j \in PCK} \tau^j \leqslant |\mathcal{R}^{'}| \tag{11}$$

where PCK is the subset of jobs from the system queue ($\mathcal{Q}$) whose total number of tasks is less than or equal to the multi-cluster resources in $\mathcal{R}^{'}$, that represents the subset of multi-cluster resources ($\mathcal{R}^{'} \subseteq \mathcal{R}$) that meets the criteria ($\mathcal{C}$), which in our case are those resources non-assigned to other parallel jobs. With this, we attempt to minimize the job waiting times, and reduce the execution times by applying the MIP model.

It must be taken into account that this expression can be defined in many other ways, adapted to the multi-cluster environment, both in the point of view of the resources or the parallel jobs nature to be executed. Any strategy or heuristic is suitable to be implemented.

3 Experimentation

In this section we assess the performance of the proposed *Package Allocation Scheduling* strategy (PAS) for heterogeneous multi-cluster environments and compare the obtained results with two other techniques in the literature. The first strategy presented by Jones in [4], named CBS for *Chunk Big Small* tries to co-allocate a "large chunk" (75% of the job tasks) into a single cluster in an attempt to avoid inter-cluster link saturation. The

 ISBN 13: 978-84-613-5510-5

second strategy presented by Naik in [8], named JPR for *Job Preferences on Resources*, allocates parallel jobs depending on their processing or communication requirements, selecting the most powerful resources when the jobs are computational intensive and minimizing the communication saturation when jobs are highly communicative, co-allocating or not the jobs as needed.

The experimental environment was a multi-cluster made up of 4 clusters, interconnected by a dedicated Giga-Ethernet network. Each cluster was made up of 16 nodes with the same characteristics. Heterogeneity was implemented assigning different Effective Power weight to each individual cluster, with values of $\{0.4, 0.6, 0.8, 1.0\}$ respectively, from lesser to higher capability.

In order to evaluate the performance of each strategy under different workload conditions, three different kinds of workloads were defined. Each workload was made up of 35 parallel jobs with different processing and communications requirements with an inter-arrival time chosen from a Poisson distribution with a mean of 40 seconds. The Highly Processing workload consisted of cpu-intensive jobs, with a weighting factor (σ^j) randomly selected in the range of $[0.75, 1]$. The Highly Communicative workload consist of communication-intensive jobs, with a weighting factor randomly selected in the range of $[0.05, 0.35]$. Finally, the Mixed workload consisted of a mix of cpu- and communication- intensive jobs.

The parallel jobs had sizes 10, 20 and 30 of tasks, and appeared in the workload with an exponential probability distribution, with higher frequency of small jobs than large ones, as is common in real systems. The execution time of parallel jobs in a dedicated environment (or base execution time, $\overline{T_e}$), were randomly selected from the range of $[60, 180]$ seconds. The average per-node communication requirements $PNBW^j$ were randomly selected from the range of $[0.05, 0.1]$ Gbps, obtaining jobs with low bandwidth requirements and other with high bandwidth requirements.

Different kinds of metrics were defined to measure the performance of the scheduling strategies. The *Average Response Time* measures the mean elapsed time in the system, by the jobs in the workload. The *Average Overhead* measures the delay in base time produced by the allocated resources.

Both metrics help us to measure the goodness of the allocation strategy for the reduction of the response times and its efficiency from the parallel application point of view. *Makespan* measures the total time spent by each scheduling strategy to execute the workload, and helps us to evaluate the goodness of the allocation strategy for improving the overall system performance. The results were obtained by using the $CPLEX$ solver package.

Figure 3 shows the average response time obtained for each scheduling strategy for different kinds of workload. The average response time is shown divided into its three basic components, the average base execution times, the average overhead produced by the allocated resources and the average waiting time.

As can be seen, the PAS scheduling strategy had the lowest average response time. The fact that this strategy is able to allocate many jobs at a time, minimizes the overall execution time and reduces considerably the waiting times. This allows the system to free resources earlier, and thus, improve the response time of the whole workload.

 ISBN 13: 978-84-613-5510-5

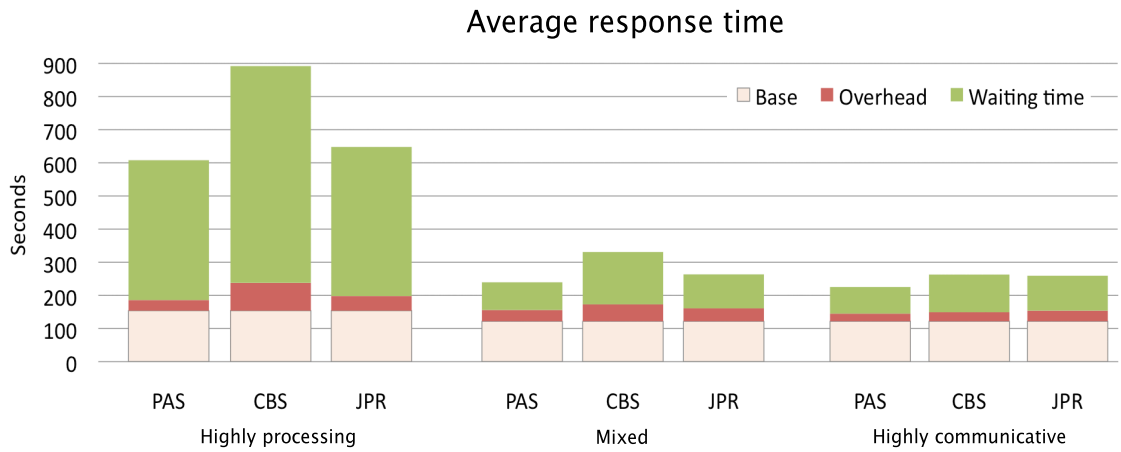


Figure 3: Comparison of average response times

The *JPR* strategy attempts to find the best resources by taking into account the job characterization, i.e. for the *Highly Communicative* workload, the strategy selects the best communication resources, while for the *Highly Processing* workload it minimizes the processing time. The strategy obtained an 8% higher average response time compared with the *PAS* strategy. The *CBS* groups job tasks in order to reduce the communications, but does not take into account the processing characteristics of the jobs or the environment. Due to this, the strategy performs well in the *Highly Communicative* workload, but has poor performance in the other two. Overall, it obtains a 38% higher response time than *PAS*.

Figure 4 shows the makespan obtained for the three kinds of workload. The x-axis represents the workloads, while the y-axis represents the makespan of the full workload, in seconds. A lower makespan implies that the workload finishes its execution earlier.

As can be seen, the *PAS* obtained the lowest makespan in the three kinds of workload. The values obtained were overall 3% and 13% lower than those for *JPR* and *CBS* respectively.

Finally, we studied how the small jobs were treated by each strategy. In [2], it was shown that systematic co-allocation yields the poorest performance, because jobs with conflicting requirements can make the performance worse than in the absence of co-allocation. With this, the small jobs are desirable to be allocated without crossing the cluster boundaries. With lower co-allocation on small jobs it will be more feasible to maintain free the inter-cluster links for those jobs that must be co-allocated and then improving the system performance. By this reason, we evaluated the number of co-allocated small jobs, those composed of 10 tasks, for each one of the compared strategies.

Figure 5 shows the number of small jobs co-allocated by the strategies for the three workloads. The total number of small jobs is indicated, between parentheses, in the label of the workload. The results show that the *PAS* strategy was able to co-allocate the lowest number of small jobs on average. In the *Highly Communicative*, the *CBS*

 ISBN 13: 978-84-613-5510-5

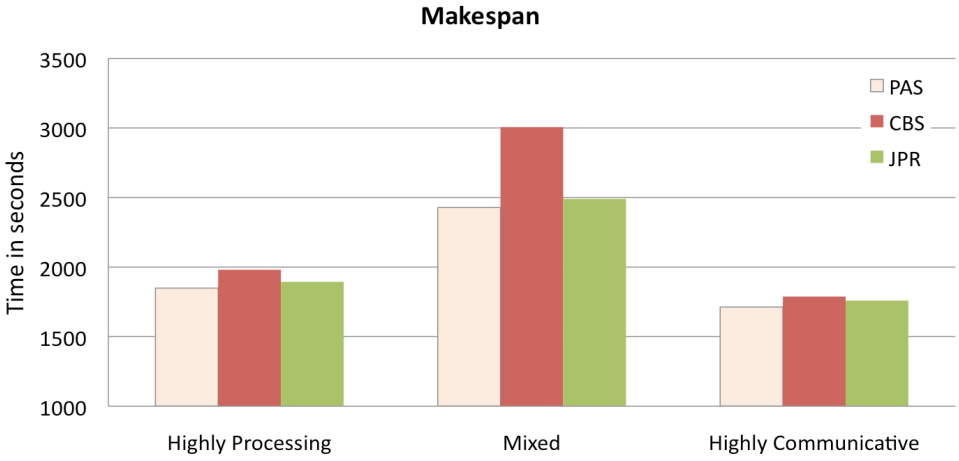


Figure 4: Comparison of Makespans

had the lowest value. This was because, in order to minimize the waiting time, the *PAS* co-allocates even the smallest jobs taking advantage of the free resources in different clusters, as can be seen in Figure 3.

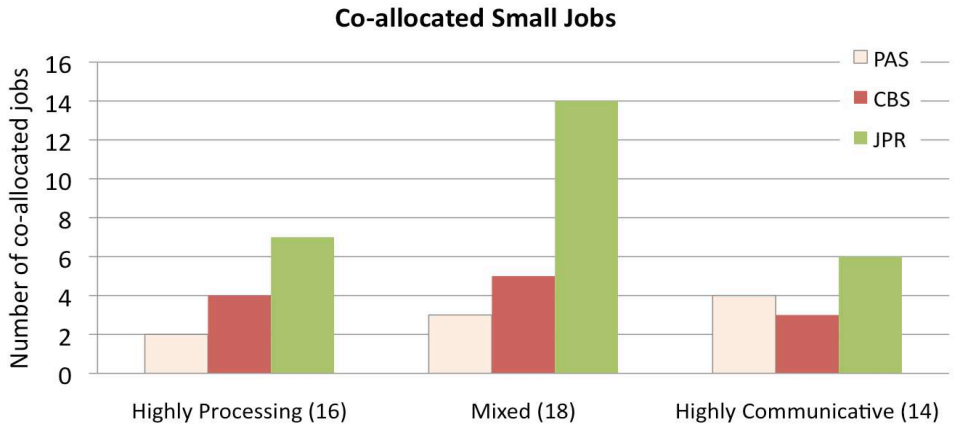


Figure 5: Degree of co-allocation for small jobs

In all workloads, *JPR* co-allocated the higher amount of jobs. In the case of the *Highly Processing* and *Mixed* workloads, their allocations were based just only on obtain the better Effective Power resources, instead the co-allocation. For the *Highly Communicative* workload, it only treated to not saturate the inter-cluster links without taking into account to maintain the tasks at the same cluster.

To summarize, under all workloads, the *PAS* strategy demonstrated its ability to reduce the response times. This was because evaluating multiple jobs simultaneously

 ISBN 13: 978-84-613-5510-5

allowed a fairness allocation for the jobs, and thus a great reduction in the waiting times was possible. Furthermore, the MIP model obtained the lower *Slowdown* (SD), the average execution overhead being the lowest.

4 Conclusions

In the present work a multiple job schedulling strategy, named PAS for *Package Allocation Scheduling* strategy, is presented, based on a *Mixed-Integer Programming* model (MIP). The MIP model minimizes the global execution time for a package of jobs, selected by the PAS strategy from the system queue, taking into account both processing and communication requirements. Our strategy was tested against others in the literature, and the results of the experimentation show that we are able to produce solutions with the lowest execution and waiting times for all the jobs, and also the makespan.

In the future work, we plan to extend our model in a stochastic, to take into account temporal scenarios where the allocations will be done considering the future jobs in the queue.

Acknowledgements

This work was supported by the MEC-Spain under contract TIN2008-05913

References

[1] B. Javadi, M.K. Akbari, J.H. Abawajy, *A performance Model for Analysis of Heterogeneous Multi-Cluster Systems*, Parallel Computing,vol.32(11-12),pp.831–851, 2006

[2] A.I.D. Bucur, D.H.J. Epema, *Schedulling Policies for Processor Coallocation in Multicluster Systems*, IEEE TPDS, vol.18(7), pp.958–972, 2007

[3] J.L. Lérida, F. Solsona, F. Giné, J.R. García, P. Hernández, *Resource Matching in Non-dedicated Multicluster Environments*, In VECPAR'08, pp.160–173, 2008

[4] W. Jones, W. Ligon, L. Pang, D. Stanzione, *Characterization of Bandwidth-Aware Meta-Schedulers for Co-Allocating Jobs Across Multiple Clusters*, The Journal of Supercomputing, vol.34(2), pp.135–163, 2005

[5] J.Abawajy, S.Dandamudi, *Parallel Job Scheduling on Multicluster Computing Systems*, CLUSTER'03: Proc. IEEE Int. Conf. on Cluster Computing,pp.11–18, 2003

[6] E.M. Heien, N. Fujimoto, K. Hagihara, *Static Load Distribution for Communicative Intensive Parallel Computing in Multiclusters* In 16th Euromicro Conf. on Parallel, Distributed and Network-Based Processing, pp.321–328, 2008

 ISBN 13: 978-84-613-5510-5

[7] C. Yang, H. Tung, K. Chou, W. Chu *Well-Balanced Allocation Strategy for Multiple-Cluster Computing* In 12th IEEE Workshop on Future Trends of Distributed Computing Systems, pp.178–184, 2008

[8] V.K. Naik, C. Liu, L. Yang, J. Wagner, *Online Resource Matching for Heterogeneous Grid Environments*, In CCGRID'05: Proc. of the 5th Int. Symp. on Cluster Computing and the Grid, vol.2, pp.607–614, 2005

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

H-Isoefficiency: Scalability Metric for Heterogeneous Systems

Jose Luis Bosque[1], Oscar D. Robles[2], Pablo Toharia[2] and Luis Pastor[2]

[1] *Dpto. de Electrnica y Computadores, Universidad de Cantabria*

[2] *Dpto. de ATC y CCIA , Universidad Rey Juan Carlos*

emails: joseluis.bosque@unican.es, oscardavid.robles@urjc.es, pablo.toharia@urjc.es, luis.pastor@urjc.es

Abstract

Scalability is one of the most important features in exascale computing. Most of this systems are heterogeneous and therefore it becomes necessary to develop models and metrics that take into account this heterogeneity. This paper presents a new expression of the isoefficiency function called H-isoefficiency. This function can be applied for both homogeneous and heterogeneous systems and allows to analyze the scalability of a parallel system. Then, as an example, a theoretical *a priori* analysis of the scalability of Floyd's algorithm is presented. Finally a model evaluation which demonstrate the correlation between the theoretical analysis and the experimental results is showed.

Key words: Heterogeneous Computing, Scalability Analisys, Isoefficiency

1 Introduction

The performance of parallel programs must be evaluated together with the computer system on which they are run. Otherwise, an algorithm that solves a problem well using a fixed number of processors on a particular architecture may perform poorly if the number of processors changes [3]. Common speedup graphs teach us that the speedup of a system does not grow linearly with the number of processors but tend to saturate. On the other hand, a higher speedup can be obtained as the problem size increases on the same number of processors [4]. Then, a system is considered to be scalable if the performance measures remain constant whenever the number of processors is increased by selecting the appropriate problem size. The system's degree of scalability is given by the ratio problem growth to system growth needed to keep those measures constant. It can be said that scalability has been a desired capability that means not just the ability to operate a system, but to operate it efficiently and with an adequate quality of service over the available range of configurations [6].

 ISBN 13: 978-84-613-5510-5

But moreover, in the age of exascale computing, a 21st century attempt to push computing capabilities beyond the existing ones, a quick look to the top500 list reveals 6 machines with more than 100,000 processors (3 of them over 200,000). Since processors are affordable and quite powerful nowadays (currently up to 12 cores and growing), other aspects as performance loose importance while scalability emerges as one of the key concepts in parallel computing.

The study of scalability of homogeneous parallel systems has not come up with a unique and common way of evaluation, although the isoefficiency metric [3] is the most accepted and used. In it, the degree of scalability is given by the *isoefficiency function*, that expresses the dependence of the problem size on the number of processors needed to keep the efficiency constant. The smaller the problem size is, the lower is the isoefficiency value and therefore the higher the scalability of the parallel system is.

This paper presents a new expression of the isoefficiency function, called *heterogeneous isoefficiency*. It is a more general definition than the one presented in [10], since the functions they use to analyze the overhead time are always linear with respect to the problem size, so their definition is only valid for the examples they propose. It also improves other existing extensions, as the ones proposed by Kalinov [7] and Chen et al. [1], as it is explained in section 2. In order to prove the use of this model it has been applied to the Floyd algorithm, obtaining from the experiments results quite close to those predicted by the model.

The rest of the paper is organized as follows: Section 2 presents a brief overview of the related work. Section 3 presents the new definition of the isoefficiency function for heterogeneous systems. Section 4 presents the application of this model to a practical problem. Section 5 shows the experimental results achieved. Finally, conclusions and future work are summarized in Section 6.

2 Related Work

Despite its importance for parallel and distributed systems, there is no unique and commonly accepted metric for scalability evaluation. A number of techniques have been suggested throughout the years. They are typically based on the selection of a metric which is used for characterizing the behavior of an homogeneous system [5, 8, 12, 14, 18]. Among the most representative works, it can be mentioned the use of latency as a metric to do an experimental measurement and evaluation of the scalability of programs and architectures [15, 17]. The model is based on the average latency, a function of the problem size and the number of processors. It determines the average overhead time needed so that each processor finishes its assigned work. Scalability is then defined as a combination of the machine and the implementation of the algorithm.

Another model quite used is the one raised by Sun and Rover [13], who proposed an isospeed scalability metric to describe the scalability of an algorithm-machine combination in homogeneous environments. This model is based on reducing the response time by means of increasing the speed. The execution speed of an algorithm is defined as the amount of work needed to complete its execution divided by the response

 ISBN 13: 978-84-613-5510-5

time. Ideally, this measure should increase linearly with the size of the system. In general communication and synchronization overheads prevent such a behavior. Chen and Wu [1] extended this model to heterogeneous systems. Its main drawback is that isospeed is an *a posteriori* measure, so it demands to implement the model and obtain the measures empirically to be able to decide about the system's scalability.

The isoefficiency is the most widespread model [9, 3]. It defines scalability as the ability of a parallel system to keep the parallel efficiency constant when both system and problem sizes increase. Then, parallel efficiency is defined as the speedup over the number of processors. Speedup is defined in turn as the ratio sequential execution time to parallel execution time. Pastor and Bosque [10] proposed an extension to heterogeneous systems, although their approach lacks of generality, since the functions they define to analyze the overhead time are linear with respect to the problem size for every case.

Kalinov [7] has also extended the isoefficiency model to heterogeneous systems. In this work it is imposed that the system has to keep constant the computational power of the slowest processor, the computational power of the fastest processor and also the average computational power of the system. These are indeed three tight restrictions quite difficult to satisfy when a system is upgraded with new nodes, conditioning its real heterogeneity.

The model presented in this paper has several advantages over the isospeed model and the extension to the isoefficiency model presented by Kalinov, since as it is explained in the following, it is an *a priori* model that successfully predicts the scalability of a system and also deals with both power and physical scalability.

3 The Isoefficiency Function for Heterogeneous Clusters

The isoefficiency function depends only on the number of processors assuming all of them have the same computational power [9, 3]. This is not the case for heterogeneous systems where the performance of a single processor can affect the overall performance of the system. In fact, the response time will depend on the slowest node: $T_R = max_{i=1}^{P} T_i$ — being P the number of processors and T_i the response time for node i.

In this way the computational power of a heterogeneous system (P_T) can be defined as the sum of the computational power of its processors (P_i) [10]:

$$P_T = \sum_{i=1}^{P} P_i \tag{1}$$

$$\tag{2}$$

Assuming W is the size of the problem, in this work the computational power of each node has been computed using the following expression:

$$P_i = \frac{W}{T_i} \tag{3}$$

 ISBN 13: 978-84-613-5510-5

3.1 Heterogeneous Efficiency

As previously mentioned the computational power of a heterogeneous system does not only depend on the number of processors but also depends on each processor's computational power. In order to improve the computational power of this type of systems both the number or the power of some processors have to be increased. The latter is called *physical scalability* while the former is referred as *power scalability* [16].

The efficiency of a parallel (either homogeneous or heterogeneous) system, denoted by ε, can be defined as the ratio between the ideal response time in a single node with the same computational power and the real response time achieved [10]. The best response time is achieved when the workload is evenly distributed and no overhead time is introduced. This is reflected in the following equation:

$$\varepsilon = \frac{Optimal\ achievable\ time}{Actual\ response\ time} = \frac{W}{T_R \cdot P_T} \tag{4}$$

For homogeneous systems, it is easy to demonstrate that ε becomes the traditional efficiency.

3.2 H-Isoefficiency

T_i can be decomposed into computation and overhead times: $T_i = t_c^i + t_o^i$. The isoefficiency function assumes that t_c, is constant for all of the processors and therefore it does not affect the scalability of the system. However, this is not true for heterogeneous systems, where not only the number but also the performance of nodes have a high impact on the scalability of the system.

Based on the definition of the efficiency given in the previous section, an isoefficiency function for heterogeneous systems, H-isoefficiency, can be defined. In the heterogeneous case the key parameter will be the total computational power of the system, instead of the number of processors.

Given a heterogeneous parallel system $S(P, P_T, W)$ with P processors, a total computational power P_T and a total amount of work represented by W and given also $S'(P', P'_T, W')$, a scaled system with $P'_T > P_T$, it can be said that S is a scalable system if, whenever the system is upgraded from S to S', it is possible to select a problem size W' such that the efficiencies of S and S' are kept constant.

Now the *heterogeneous isoefficiency function, H-isoefficiency* for heterogeneous parallel systems can be computed, starting from the heterogeneous efficiency definition for S. Furthermore we define the response time of the parallel algorithm as the sum of the execution time plus the overhead time: $T_R = T_{exe} + T_o$. Assuming that the workload is evenly distributed among the nodes proportionally to each node's computational power, T_{exe} will be the same for all of the nodes, and it can be determined as $T_{exe} = \frac{W}{P_T}$. Then the response time will be given by the following expression: $T_R = \frac{W}{P_T} + T_o$. Hence, the H-Isoefficiency function is defined as:

$$\varepsilon = \frac{W}{T_R \cdot P_T} = \frac{W}{W + T_o \cdot P_T} = \frac{1}{1 + \frac{T_o \cdot P_T}{W}}$$

 ISBN 13: 978-84-613-5510-5

For scalable heterogeneous parallel systems, the efficiency can be maintained at a desired value if the ratio $\frac{T_o \cdot P_T}{W}$ in the expression of the efficiency is maintained at a constant value. To maintain a certain efficiency we can do:

$$\frac{T_p \cdot P_T}{W} = \frac{1-\varepsilon}{\varepsilon} \Rightarrow W = \frac{\varepsilon}{1-\varepsilon} T_o \cdot P_T$$

Let $K = \frac{\varepsilon}{1-\varepsilon}$ be a constant depending on the efficiency. Then the H-isoefficiency function can be write as:

$$W = K \cdot T_o(P) \cdot P_T(P) \tag{5}$$

From this expression can be pointed out that the scalability of heterogeneous environments depends on both, the number of nodes and the total computational power of the scaled system.

This expression is similar to the one proposed for homogeneous isoefficiency. The main difference between both is that instead of using a single t_c parameter, which remains constant for the whole set of nodes of the system and which is included in the K parameter, a new P_T parameter is introduced to represent the total aggregated computational power of the system.

Therefore, when scaling a system, the computational power of new nodes has to be taken into account in order to increase the size of the problem in a proportional way. This problem can be seen from a qualitative point of view: if a system is scaled using nodes more powerful than the system's, the total response time would be lower than the time achieved if the nodes had the same computational power. In this way the total overhead (T_o) would be a bigger percentage of the response time and the efficiency would be decreased. This fact makes necessary to increase the size of the problem in order to achieve the same efficiency as is represented in Eq. 5.

A big advantage of the proposed approach compared to Kalinov's generalization of the isoefficiency function [7] is that our approach allows to study the behavior of the system both when scaling by the number of processors *(physical scalability)* and when scaling increasing the power of some of them *(power scalability)* Kalinov's work forces to maintain the average computational power and therefore a power scalability study can not be done. With our approach *a priori* studies can be carried out in order to analyze if a specific algorithm is more suitable for its execution using a large number of less powerful processors or a lower number of more powerful ones. This is demonstrated in the experimental results section.

4 Scalability of the Floyd Algorithm

Once the H-isoefficiency has been defined it becomes necessary to experimentally validate and verify it. In this way, as an example, Floyd's algorithm has been chosen. This algorithm solves the all-pairs shortest-path problem. In this Section the performance of a parallel implementation of this algorithm is analyzed in order to obtain both the isoefficiency and H-isoefficiency. In Section 5 experimental results are presented.

 ISBN 13: 978-84-613-5510-5

4.1 Performance Analysis

Let's assume a model for communication cost in parallel programs. The time spent in a single point-to-point communication over an uncontested interconnection network can be well approximated in terms of *startup latency* (λ) and *bandwidth* (β). Then the time to communicate a m word message can be approximated by: $T_M = \lambda \cdot \frac{m}{\beta}$.

A broadcast function to p processors requires $\lceil logp \rceil$ message-passing steps. Hence the time spent in broadcasting a m word message can be approximated by $T_B = \lambda \cdot \frac{m}{\beta} \cdot logp$.

The sequential implementation of the Floyd algorithm is composed by three nested loops, from 0 to $n-1$, where n is the dimension of the adjacency matrix. Therefore the complexity of the sequential Floyd algorithm is $\Theta(n^3)$.

With respect to the parallel algorithm the innermost loop has complexity $\Theta(n)$. Given a row-wise block-striped decomposition of the adjacency matrix, each process executes at most $\lceil \frac{n}{p} \rceil$ iterations of the middle loop. Hence the complexity of the inner two loops is $\Theta(\frac{n^2}{p})$. Immediately before the middle loop is the broadcast step. Passing a single message of length n from one processor to another has time complexity $\Theta(n)$. Since broadcasting to p processors requires $\lceil logp \rceil$ message-passing steps, the overall time complexity of broadcasting each iterations is $\Theta(nlogp)$. The outermost loop executes n times. Hence the overall time complexity of the parallel algorithm is:

$$\Theta(n(nlogp + \frac{n^2}{p})) = \Theta(\frac{n^3}{p} + n^2 logp)$$

Now let's come up with a prediction of the response time of the parallel algorithm. The parallel Floyd program requires n broadcasts, each of them with $\lceil logp \rceil$ steps. Each step involves passing messages that are $4n$ bytes long. Hence the expected communication time of the parallel program is:

$$n\lceil logp \rceil(\lambda + \frac{4n}{\beta})$$

If t_c is the average time needed to update a single cell (a basic algorithm operation), then the expected computational time of the parallel algorithm is:

$$n^2 \left\lceil \frac{n}{p} \right\rceil t_c$$

However it is possible to overlap communication and computation operations. The computation time per iteration exceeds the time needed to pass messages. For this reason after the first iteration each process spends the same amount of time waiting for or setting up messages: $\lceil logp \rceil \lambda$. If $\lceil logp \rceil \frac{4n}{\beta} < \lceil \frac{n}{p} \rceil n t_c$, the message transmission time after the first iteration is completely overlapped by the computational time and should not be counted toward the total execution time. Hence a better expression for the execution time of the parallel program is:

$$T_R = n^2 \left\lceil \frac{n}{p} \right\rceil t_c + n\lceil logp \rceil \lambda + \lceil logp \rceil \frac{4n}{\beta}) \quad (6)$$

 ISBN 13: 978-84-613-5510-5

4.2 Isoefficiency Function

Let's determine the isoefficiency function for the parallel implementation of the Floyd's algorithm. The sequential algorithm has time complexity $\Theta(n^3)$. Each of the p processes executing the parallel algorithm spends $\Theta(nlogp)$ time performing communications. Therefore the isoefficiency relation is:

$$n^3 \geq K(nplogp) \Rightarrow n \geq K(plogp)^{\frac{3}{2}}$$

where K is a constant.

4.3 H-isoefficiency of the Floyd Algorithm

Now let's determine the H-isoefficiency function of the Floyd's algorithm. For obtaining the execution time in a heterogeneous environment, we have to take into account that the workload is evenly distributed according to each node's computational power. Then each node has a computational workload given by $w_i = \frac{W}{P_T} \cdot P_i$.

In such a way, the middle loop executes w_i times per each node, but all the nodes spend the same amount of time $\frac{W}{P_T}$. A performance analysis similar to the one presented in Section 4.1 gives the following response time and complexity expressions:

$$T_R^H = \frac{n^3}{P_T} + n \cdot \lceil logp \rceil \cdot (\lambda + \frac{4n}{\beta}) \Rightarrow T_o = n \cdot \lceil logp \rceil \cdot (\lambda + \frac{4n}{\beta}) \quad (7)$$

$$\Theta(\frac{n^3}{P_T} + n^2 logp) \quad (8)$$

Then the H-isoefficiency function is:

$$W = KP_T T_o \Rightarrow n^3 = KP_T(n\lambda\lceil logp \rceil + \frac{4n}{\beta}\lceil logp \rceil) = KP_T n\lambda\lceil logp \rceil + KP_T\frac{4n}{\beta}\lceil logp \rceil \quad (9)$$

Analyzing each term independently we reach the same expression for the H-isoefficiency:

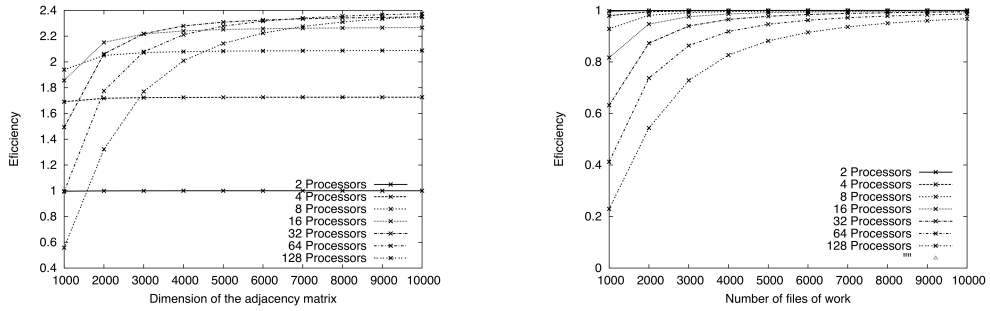
$$\Theta((P_T logp)^{\frac{3}{2}}). \quad (10)$$

5 Model Evaluation

A practical experiment was set up in order to test the analytical results presented in the previous sections and to show how the heterogeneous isoefficiency model could be applied. The tests were performed on a heterogeneous HP Cluster with up to 142 processors, interconnected with Gigabit Ethernet network. The cluster is composed of the following resources:

- 20 HP Proliant DL 145 with two 1.8 GHz Dual Core AMD Opteron 265 processors, labeled as Node Slow (NS) in this paper.

 ISBN 13: 978-84-613-5510-5



(a) Efficiency in a heterogeneous cluster

(b) Heterogeneous efficiency in a heterogeneous cluster. eps

Figure 1: Comparison of classic and proposed efficiency figures in a heterogeneous cluster

- 25 HP Proliant DL 160 with two 3.0 GHz Quad Core INTEL Xeon 5472 processors, labeled ad Node Fast (NF).

In this cluster the parameters of the model have been measured and they are listed in Table 1. The application was developed using GNU tools and the MPICH 1.2.7 library [2, 11].

Label	P_i	tc_i	λ	β
Node Fast (NF)	83.988.126	0,0000000119	25	2.000.000.000
Node Slow (NS)	34230899	0,0000000292		

Table 1: Parameters of the cluster

The first experiment compared the values of classical and proposed efficiency in a heterogeneous cluster under variable node count and workload conditions (figures 1(a) and 1(b)). In all the configurations, the cluster is composed of 2 NS processors while the rest of processors are NF. It can be seen that the efficiency figures computed using the classical efficiency definition on the heterogeneous cluster are not consistent. On the other hand the proposed efficiency yields results very close to those obtained in the homogeneous cluster.

In the second experiment we have measured the heterogeneous efficiency value for a cluster composed by two nodes and a dimension of the adjacency matrix of 128. Then we have estimated the workload needed to maintain constant this efficiency when the number of nodes is increased in a heterogeneous cluster and compared with the real measured values. Table 2 presents the results achieved, where P is the number of nodes, P_T is the total computational power, N is the measured workload and H-isoefficiency is the theoretical computed value for the workload. Additionally the values of heterogeneous and classical efficiency are shown.

 ISBN 13: 978-84-613-5510-5

Table 2 presents the workload computed through Equation 10 (labeled as H-isoefficiency), necessary to keep the heterogeneous cluster's efficiency constant at a value around 0.842. The column N shows the workload actually measured. Keeping the heterogeneous efficiency constant requires that the problem size must be increased according to the H-isoefficiency expression rather than with respect to homogeneous one. The errors between the estimated and measured workload values are very small, validating the assumption that the H-isoefficiency function provides an accurate, *a priori* method for analyzing scalability in heterogeneous clusters.

Table 2 also presents the values of efficiency obtained measuring the sequential time in both the fast and the slow processors. It has to be noted that these results are not consistent with the efficiency definition, in the NF processor. On the other hand the heterogeneous system is not scalable which is not reasonable.

P	P_T	N	ε	E (NF)	E (NS)	H-isoeff.
2	68461798	128	0,842	0,842	0,343	128
4	236438050	336	0,841	1,453	0,592	336,40
8	572390554	640	0,841	1,758	0,717	641,05
16	1244295562	1088	0,841	1,910	0,779	1091,38
32	2588105578	1760	0,842	1,989	0,811	1759,79
64	5275725610	2752	0,842	2,027	0,826	2752,34
128	10650965674	4216	0,841	2,045	0,833	4224,05

Table 2: H-isoefficiency for a E=0.842

Finally Table 3 shows the H-isoefficiency and the H-efficiency values for different cluster configurations with the same number of nodes. It has been to highlight that H-isoefficiency predicts different values of workload to different configurations, depends on the total computational power. This predictions are consistent with the measured values and the H-efficiency can be maintained constant. Again, the figures presented by classical efficiency are not consistent.

Configuration	P_T	H-isoeff.	E(NF)	E(NS)	ε
128 NF	10750480128	3680,60	0,800	1,964	0,800
114 NF 16 NS	10122340748	3571,45	0,754	1,849	0,800
96 NF 32 NS	9158248864	3397,12	0,682	1,673	0,801
64 NF 64 NS	7566017600	3087,72	0,563	1,382	0,800

Table 3: H-isoefficiency for a E=0.80, P=128 with different cluster configurations

6 Conclusions

This paper presents a new expression of the isoefficieny function, called H-isoefficiency, which can be applied to homogeneous and heterogeneous systems and which can be used for predicting algorithm scalability without needing the actual implementation of the algorithm in the selected architecture. Comparing this model with others previously described in the literature, it presents several advantages, just like the isoefficiency

 ISBN 13: 978-84-613-5510-5

model proposed by Kumar and Rao [9] does. Its most remarkable advantages are that, on one hand, being an *a priori* method it does not require the implementation of the algorithm to be studied in the selected architecture. On the other hand, it deals with both power and physical scalability without imposing any restriction on the system's setup.

The experiments performed have shown that the proposed method yields results quite close to the values theoretically predicted. This shows that the H-isoefficiency function is an accurate model that allows performing scalability analysis both for homogeneous and heterogeneous systems. The results have also verified the strong impact that different configurations have on the scalability of a heterogeneous environment.

Future work includes the development of more systematic and precise methods for estimating both overhead and relative node computational power. Additionally the assumption of the "workload evenly distributed according to each's node computational power" will be removed from the scalability theorems.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Education and Science under the contracts TIN2007-68023-C02-01, TIN2007-67188 and CSD2007-00050, by the HiPEAC European Network of Excellence as well as by the Cajal Blue Brain project.

References

[1] Yong Chen, Xian-He Sun, and Ming Wu. Algorithm-system scalability of heterogeneous computing. *Journal of Parallel and Distributed Computing*, 68(11):1403–1412, 2008.

[2] MPI Forum. A message-passing interface standard. 1995. http://www.mpi-forum.org.

[3] Ananth Y. Grama, Anshul Gupta, and Vipin Kumar. Isoefficiency: measuring the scalability of parallel algorithms and architectures. *IEEE parallel and distributed technology: systems and applications*, 1(3):12–21, August 1993.

[4] John L. Gustafson. Reevaluating amdahl's law. *Communications of the ACM*, 31(5):532–533, May 1988. Sandia NL.

[5] John L. Gustafson, Gary R. Montry, and Robert E. Benner. Development of Parallel Methods for a 1024-Processor Hypercube. *SIAM Journal on Scientific and Statistical Computing*, 9(4):609–638, 1988.

[6] P. Jogalekar and M. Woodside. Evaluating the scalability of distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, 11(6):589–603, June 2000.

 ISBN 13: 978-84-613-5510-5

[7] A. Ya. Kalinov. Scalability of heterogeneous parallel systems. *Programming and Computer Software*, 32(1):1–7, 2006.

[8] Alan H. Karp and Horace P. Platt. Measuring parallel processor performance. *Communications of the ACM*, 22(5):539–543, May 1990.

[9] Vipin Kumar and V. Nageshwara Rao. Parallel depth-first search on multiprocessors: Part II: Analysis. *International Journal of Parallel Programming*, 16(6):501–519, December 1987.

[10] Luis Pastor and Jose L. Bosque. Efficiency and scalability models for heterogeneous clusters. In *Third IEEE International Conference on Cluster Computing,*, pages 427–434, Los Angeles, California, Octubre 2001. IEEE Computer Society Press.

[11] Marc Snir, Steve W. Otto, Steven Huss-Lederman, David W. Walker, and Jack Dongarra. *MPI: The Complete Reference*. The MIT Press, 1996.

[12] Xian-He Sun and John L. Gustafson. Sizeup: a new parallel performance metric. In *Proceedings of the 1991 International Conference on Parallel Processing*, volume II, Software, pages II–298–II–299. CRC Press, August 1991.

[13] Xian-He Sun and Diane T. Rover. Scalability of parallel algorithm-machine combinations. *IEEE Transactions on Parallel and Distributed Systems*, 5(6):599–613, June 1994.

[14] Frederic A. Van-Catledge. Towards a general model for evaluating the relative performance of computer systems. *International Journal of Supercomputer Applications*, 3(2):100–108, 1989.

[15] Y. Yan, X. Zhang, and Q. Ma. Software support for multiprocessor latency measurement and evaluation. *IEEE transactions on Software Engineering*, 23(1):4–16, 1997.

[16] X. Zhang and Y. Yan. Modelling and characterizing parallel computing performance on heterogeneous networks of workstations. *Proc. 7th IEEE Symp. on Parallel and Distributed Processing*, pages 25–35, 1995.

[17] Xiaodong Zhang, Yong Yan, and Keqiang He. Latency metric: An experimental method for measuring and evaluating parallel program and architecture scalability. *Journal of Parallel and Distributed Computing*, (3), February 1994.

[18] J. R. Zorbas, D. J. Reble, and R. E. VanKooten. Measuring the scalability of parallel computer systems. In *Proceedings, Supercomputing '89: November 13–17, 1989, Reno, Nevada*, pages 832–841. ACM Press, 1989.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27--30 June 2010.

A counterexample showing the semi-explicit Lie-Newmark algorithm is not variational

Nawaf Bou-Rabee[1], Giulia Ortolan[2] and Alessandro Saccon[3]

[1] *Department of Mathematics, Courant Institute of Mathematical Sciences, New York University*

[2] *Department of Information Engineering, University of Padova*

[3] *Instituto de Sistemas e Robótica, Instituto Superior Técnico, Universidade Técnica de Lisboa*

emails: nawaf@cims.nyu.edu, ortolang@dei.unipd.it, asaccon@isr.ist.utl.pt

Abstract

This paper presents a counterexample to the conjecture that the semi-explicit Lie-Newmark algorithm is variational. As a consequence the Lie-Newmark method is not well-suited for long-time simulation of rigid body-type mechanical systems. The counterexample consists of a rigid body in a static potential field.

Key words: rigid body, long-time simulation, Newmark algorithm
MSC 2000: 65P10

1 Introduction

In this paper we will focus on the dynamics of a rigid body in a static potential field. To describe this system, denote by $Q(t) \in SO(3)$, $W(t) = [W_1(t)\ W_2(t)\ W_3(t)]^T \in \mathbb{R}^3$, and $\mathbb{I} = \mathrm{diag}(I_1, I_2, I_3) \in \mathbb{R}^{3\times 3}$ the configuration, body angular velocity and inertia matrix of the body, respectively. Let $\tau : SO(3) \to \mathbb{R}^3$ be the torque acting on the body and $\widehat{\ }: \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ the hat map

$$\widehat{W} = \begin{bmatrix} 0 & -W_3 & W_2 \\ W_3 & 0 & -W_1 \\ -W_2 & W_1 & 0 \end{bmatrix}.$$

 ISBN 13: 978-84-613-5510-5

In terms of this notation, the governing equations are

$$\begin{cases} \dot{Q} &= Q\widehat{W} & \text{(1a)} \\ \mathbb{I}\dot{W} &= \mathbb{I}W \times W + \tau(Q), & \text{(1b)} \end{cases}$$

with initial conditions $Q(0) = Q_0 \in SO(3)$ and $W(0) = W_0 \in \mathbb{R}^3$. This rigid body corresponds to a mechanical system whose Lagrangian is of the form

$$L(Q, W) = T(W) - U(Q) \tag{2}$$

where $T(W) = \frac{1}{2}W^T\mathbb{I}W$ and $U(Q)$ are the potential and kinetic energy of the body, respectively. Notice that the total energy is separable and $T(-W) = T(W)$. The flow of (1) possesses certain structures such as total energy preservation, time-symmetry, and symplecticity. Moreover, the path Q lies on a configuration manifold $SO(3)$ which possesses a Lie-group structure.

This paper investigates the long-run behavior of two integrators for (1): the Lie-Newmark [1] and Lie-Verlet methods [2]. Both methods are semi-explicit, second-order accurate and symmetric. They are also 'Lie group methods' because they respect the Lie group structure of the configuration manifold [3]. The main difference between the integrators is that the Lie-Verlet method is designed to be variational, whereas the Lie-Newmark method is not.

Variational integrators are time-integrators adapted to the structure of mechanical systems [4, 5, 6]. They are symplectic, and in the presence of symmetry, momentum preserving. The theory of variational integrators includes discrete analogues of Hamilton's principle, Noether's theorem, the Euler-Lagrange equations, and the Legendre transform. The variational nature of Lie-Verlet guarantees its excellent long-time behavior. In fact, one can prove this. The basic idea of the proof is to show that a trajectory of a variational integrator is interpolated by a level set of a 'modified' energy function nearby the true energy [7, 8, 9]. This implies that a trajectory of the variational integrator is confined to these level sets for the duration of the simulation. As a consequence variational integrators nearly preserve the true energy and exhibit linear growth in global error. For these reasons variational integrators are well-suited for long-time simulation.

Even though the Lie-Newmark integrator is not designed to be variational, this does not rule out the possibility that the algorithm is variational in a subtle way like vector space Newmark [10]. Specifically Kane et al. prove that a trajectory of the vector space Newmark method is shadowed by a trajectory of a variational algorithm. In other words the Newmark integrator is not directly symplectic, but a so-called conjugate symplectic method [9]. This possibility was supported by numerical evidence showing that the Lie-Newmark algorithm exhibits good behavior analogous to vector space Newmark [11]. In that paper Krysl and Endres conjecture that the Lie-Newmark algorithm is variational.

This paper disproves this conjecture. In particular, the paper presents a simple numerical counterexample showing that the Lie-Newmark method exhibits systematic energy drift. In contrast, the Lie-Verlet method nearly preserves the true

 ISBN 13: 978-84-613-5510-5

energy and exhibits the qualitative properties one expects of a variational integrator. In summary, the Lie-Verlet method is well-suited for long-time simulation of rigid body-type mechanical systems while the Lie-Newmark method is not.

2 Integrators

Lie-Newmark methods were originally proposed in [1]. The methods consist of a Newmark style discretization of (1b) and a discretization of (1a) that ensures the configuration update remains on $SO(3)$. This paper focuses on the following semi-explicit member of the Lie-Newmark family tested in [11].

Given (Q_k, W_k) and time-stepsize h, the algorithm determines (Q_{k+1}, W_{k+1}) by the following iteration rule:

$$\begin{cases} W_{k+\frac{1}{2}} = W_k + \dfrac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\,W_k \times W_k + \tau(Q_k)\right) & \text{(3a)} \\ Q_{k+1} = Q_k\,\mathrm{cay}(hW_{k+\frac{1}{2}}) & \text{(3b)} \\ W_{k+1} = W_{k+\frac{1}{2}} + \dfrac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\,W_{k+1} \times W_{k+1} + \tau(Q_{k+1})\right), & \text{(3c)} \end{cases}$$

where $\mathrm{cay} : \mathbb{R}^3 \to SO(3)$ denotes the Cayley map:

$$\mathrm{cay}(\xi) = \left(I - \frac{\widehat{\xi}}{2}\right)^{-1}\left(I + \frac{\widehat{\xi}}{2}\right) = I + \frac{4}{4 + \|\xi\|}\widehat{\xi} + \frac{2}{4 + \|\xi\|}\widehat{\xi}^2. \tag{4}$$

This integrator is semi-explicit because (3a) and (3b) involve explicit updates, and (3c) is only implicit in the angular velocity, not in the torque. Hence, the implicitness of the Lie-Newmark method is not severe. In fact, this numerical algorithm is called explicit Newmark in [11]. In the Appendix we check for the reader's convenience that this algorithm is symmetric and reversible. It is also second-order accurate. There are other maps one can use in place of the Cayley map in (3b) (see, e.g., §5.4 of [12]), but the Cayley map is known to be the most computationally efficient in practice.

The velocity Lie-Verlet integrator was proposed in [2] and inspired by the theory of discrete and continuous Euler-Poincaré systems [13, 14]. The method is closely related to, but different from the RATTLE method for constrained mechanical systems.

Given (Q_k, W_k) and time-stepsize h, the algorithm determines (Q_{k+1}, W_{k+1}) by the following iteration rule:

$$\begin{cases} W_{k+\frac{1}{2}} = W_k + \dfrac{h}{2}\mathbb{I}^{-1}\left[\mathbb{I}\,W_{k+\frac{1}{2}} \times W_{k+\frac{1}{2}} - \dfrac{h}{2}\left(W_{k+\frac{1}{2}}^T \mathbb{I}\,W_{k+\frac{1}{2}}\right) W_{k+\frac{1}{2}} + \tau(Q_k)\right] & \text{(5a)} \\ Q_{k+1} = Q_k\,\mathrm{cay}(hW_{k+\frac{1}{2}}) & \text{(5b)} \\ W_{k+1} = W_{k+\frac{1}{2}} + \dfrac{h}{2}\mathbb{I}^{-1}\left[\mathbb{I}\,W_{k+\frac{1}{2}} \times W_{k+\frac{1}{2}} + \dfrac{h}{2}\left(W_{k+\frac{1}{2}}^T \mathbb{I}\,W_{k+\frac{1}{2}}\right) W_{k+\frac{1}{2}} + \tau(Q_{k+1})\right]. & \text{(5c)} \end{cases}$$

 ISBN 13: 978-84-613-5510-5

Similar to the Lie-Newmark method, this algorithm is symmetric, semi-explicit and second-order accurate. In particular, the updates in (5b) and (5c) are explicit, and the implicitness in (5a) does not involve the torque. We emphasize the Lie-Verlet integrator is variational and refer the reader to [2] for a proof of this result.

3 Numerical Counterexample

This section describes a numerical experiment in which a trajectory of the semi-explicit Lie-Newmark integrator (3) exhibits systematic drift in total energy. Such drift proves that the method is *not* a conjugate symplectic integrator for (1), and hence, is not variational. The numerical counterexample we discuss is strongly inspired by a numerical experiment reported in [15, §4.4]. In that paper a systematic drift in the total energy of a spring pendulum (with exterior forces) was found when using a fourth-order accurate, implicit, and symmetric Lobatto IIIB integrator.

Consider the function $\mathrm{dist} : SO(3) \times SO(3) \to \mathbb{R}$ defined as

$$\mathrm{dist}(Q_1, Q_2) := \sqrt{2\,\mathrm{tr}(Q_2 - Q_1)}\,.$$

Let $\|\cdot\|_F$ denote the Frobenius matrix norm. We recall that $\|A\|_F := \sqrt{\mathrm{tr}(A^T A)}$ for $A \in \mathbb{R}^{n\times n}$. It is straightforward to verify that $\mathrm{dist}(\cdot,\cdot)$ is a *distance* function in $SO(3)$ induced by the Frobenius norm using the identity $2\,\mathrm{tr}(Q_2 - Q_1) = = \|Q_2 - Q_1\|_F^2$.

For the numerical experiment, consider a single rigid body in a static potential field. Let $I \in SO(3)$ be the identity element. The potential energy $U : SO(3) \to \mathbb{R}$ is the sum of two contributions and is defined as

$$U(Q) = \big(\mathrm{dist}(Q, I) - 1\big)^2 - \frac{\alpha}{\mathrm{dist}(Q, Q_m)}\,. \tag{6}$$

The first term in the right hand side of (6) is a bounded potential which attains its minimum value at $Q \in SO(3)$ satisfying $\mathrm{dist}(Q, I) = 1$. The second term is an unbounded potential that generates an attraction toward the configuration $Q_m \in SO(3)$. The parameter α is a tuning parameter.

For $\alpha = 0$, the potential U attains its minimum value on the set

$$S := \{Q \in SO(3) : \mathrm{dist}(Q, I) = 1\},$$

a two-dimensional surface in $SO(3)$. The set $(S, 0) \subset SO(3) \times \mathbb{R}^3$ is a stable set (in the sense of Lyapunov). If we choose the initial condition $(Q_0, W_0) \in SO(3) \times \mathbb{R}^3$ so that Q_0 is close to S and W_0 is small, the resulting trajectory $(Q(t), W(t))$, $t \geq 0$, stays close to the set S, in the sense that $\mathrm{dist}(Q(t), I) \approx 1$. Furthermore, if we choose W_0 to have a component 'tangential' to the surface S at Q_0, then the rigid body will wander along S reaching configurations quite distant from the

 ISBN 13: 978-84-613-5510-5

initial condition Q_0 while staying close to the set S. This latter fact will be key to the numerical experiment we will describe.

For $\alpha > 0$, the unbounded attractive potential will cause a distortion of the two-dimensional surface S. On this distorted energy landscape, the rigid body experiences an attraction toward the configuration Q_m. For $\mathrm{dist}(I, Q_m) \neq 1$ and $\alpha > 0$ sufficiently small, the set S gets distorted into a set, that we label S_α, with similar stability properties as previously discussed.

Let the inertia matrix be $\mathbb{I} = \mathrm{diag}(2.0,\, 2.0,\, 4.0)$. Select the potential energy tuning parameter to be $\alpha = 0.3$. Place the attraction point at $Q_m = \exp(\hat{v}_m)$, where $v_m = [2.5\ \ 0\ \ 2.5]^T \in \mathbb{R}^3$. Now select the initial configuration to be $Q_0 = \exp \hat{v}_0$, where $v_0 = [0\ 0.7227\ 0]^T$, and the initial angular velocity to be $W_0 = [0\ 0\ 0.625]^T$. Notice that the initial condition (Q_0, W_0) is selected so that $\mathrm{dist}(Q_0, I)$ is nearly one.

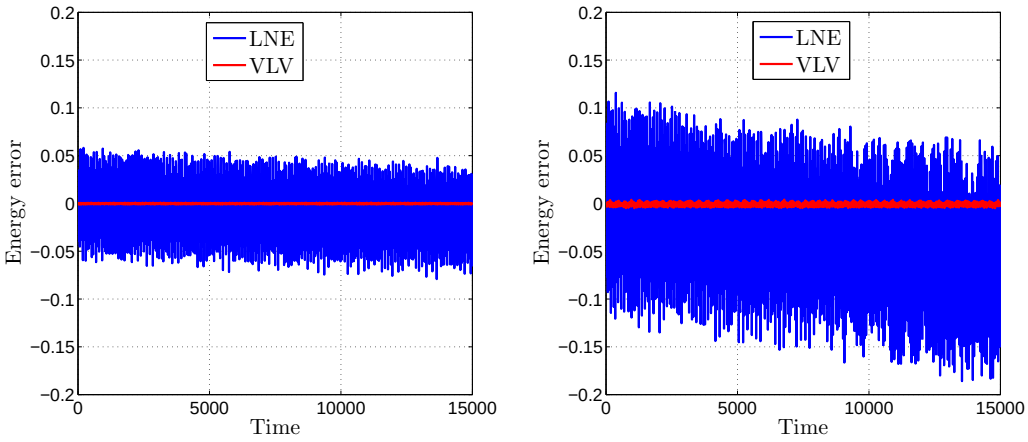
In the numerical experiment we test the two integrators, Lie-Newmark (LNE) and velocity Lie-Verlet (VLV), on a long time interval $[0, 15000]$. The energy error obtained with the time-stepsize $h = 0.125$ is shown in Figure 1(a). The experiment was repeated with a time-stepsize $h = 0.25$ and results are reported in Figure 1(b). A systematic drift for the LNE scheme can be observed in both cases. The drift appears linear in the time span T and quadratic in the time-stepsize h. We abbreviate this fact by saying the total energy error behaves like $\mathcal{O}(Th^2)$. No energy drift is observed for the VLV scheme. The trajectory generated by Lie-Newmark for time-stepsize $h = 0.25$ is shown in the axis/angle representation of $SO(3)$ in Figure 2. The semi-transparent surfaces correspond to isosurfaces of the potential energy (6).

The time-precision diagrams, shown in Figures 3(a) and 3(b) confirm that LNE and VLV are second-order accurate. Observe from the figures that the slope of the two lines denoting the global error is $\mathcal{O}(h^2)$. The diagrams have been generated by computing the global error in the configuration and angular velocity evaluated at $T = 5$. The simulations have been performed for a variety of time-stepsizes as shown in the figures. The reference solution was computed using the function `ode45` in MATLAB, with an absolute tolerance 10^{-14} and relative tolerance $2 \cdot 10^{-14}$.

4 Conclusion

The Lie-Newmark method was proposed as a generalization of the vector-space Newmark algorithm to Lie groups [1]. However, unlike its counterpart on vector spaces, this paper shows that the Lie-Newmark method does not possess excellent long-time behavior when applied to a rigid body in a potential force field. In particular, the paper presents a numerical experiment which shows systematic energy drift along a Lie-Newmark trajectory that behaves like $\mathcal{O}(Th^2)$. The experiment consisted of simulating a simple rigid body system in a static force field. On the other hand, the Lie-Verlet method which is designed to be variational does not ex-

 ISBN 13: 978-84-613-5510-5



(a) h = 0.125 (b) h = 0.25

Figure 1: This figure shows the energy error of the Lie-Newmark (LNE) and velocity Lie-Verlet (VLV) algorithms for the rigid body in the potential energy landscape defined by (6) for two different timesteps. LNE exhibits a systematic energy drift. On the other hand, the energy error of VLV method remains bounded as predicted by theory. The initial conditions and parameters used are provided in the text.

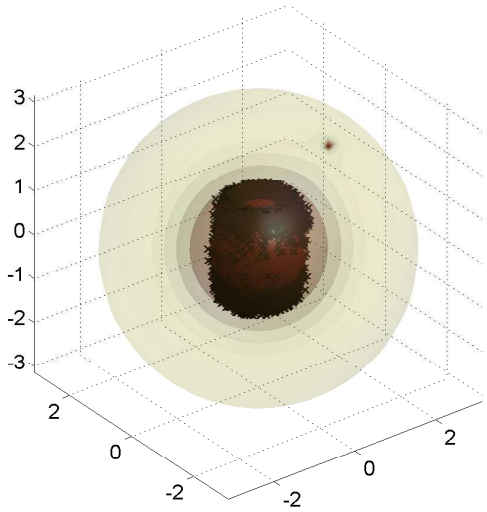
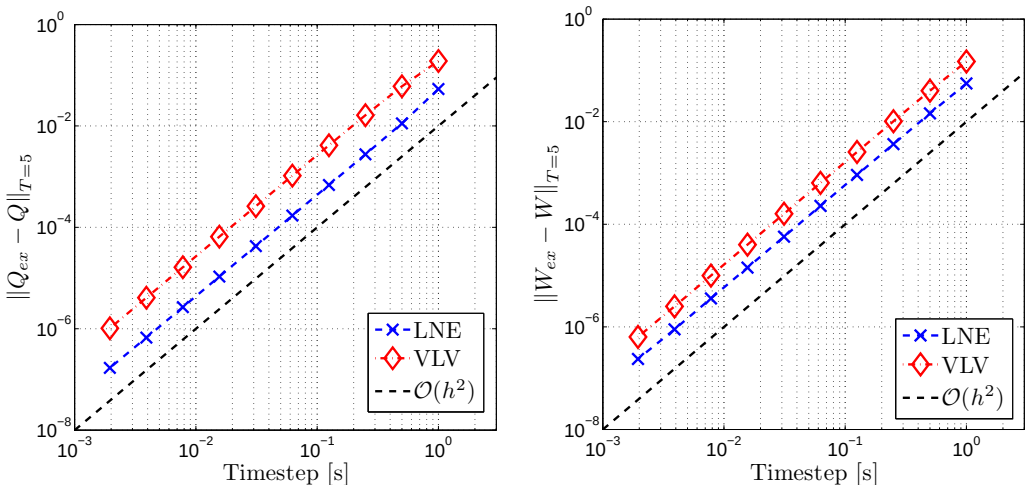


Figure 2: This figure shows the Lie-Newmark trajectory using the axis/angle representation of $SO(3)$ for the initial conditions and parameters provided in the text. The semi-transparent surfaces are level sets of the potential energy (6). The dot in the figure corresponds to the attraction point Q_m of the potential energy.

 ISBN 13: 978-84-613-5510-5



(a) Configuration.

(b) Body angular velocity.

Figure 3: This figure shows the global error of the Lie-Newmark (LNE) and velocity Lie-Verlet (VLV) algorithms. The global error is evaluated in configuration and body angular velocity at a physical time of $T = 5$ for a variety of time-stepsizes. We use as a reference solution an integration of (1) using the MATLAB function `ode45` with low tolerance. Observe that both integrators are second-order accurate.

hibit energy drift as theory predicts. Since the two methods are semi-explicit and computationally similar to implement, we conclude that the Lie-Verlet method is better suited for long-time simulations of rigid body-type systems.

Acknowledgements

We are extremely grateful to Jerry Marsden for suggesting this topic, providing encouragement, excellent teaching, and many good ideas. We would also like to thank Melvin Leok for stimulating discussions.

N. B-R. acknowledges the support of the United States National Science Foundation through NSF Fellowship # DMS-0803095. A. S. acknowledges the support of projects DENO/FCT-PT (PTDC/EEA-ACR/67020/2006), FCT-ISR/IST pluriannual funding program, and the CMU-Portugal program.

Appendix A Properties of Lie-Newmark Algorithm

Let us call

$$\Phi_h : (Q_k, W_k) \mapsto (Q_{k+1}, W_{k+1}) \,,$$

the map defined by the Lie-Newmark algorithm (3). For the reader's convenience we provide a standard proof that Lie-Newmark is symmetric and reversible.

 ISBN 13: 978-84-613-5510-5

Proposition 1. *The Lie-Newmark algorithm* (3) *is symmetric and reversible.*

Proof. Exchanging $h \leftrightarrow -h$ and $(Q_k, W_k) \leftrightarrow (Q_{k+1}, W_{k+1})$, it is straightforward to see that the method is unaltered.

Define the involution $\rho : (Q, W) \mapsto (Q, -W)$. Recall [9] that a numerical algorithm is $\rho-$reversible if

$$\Phi_{-h} \circ \rho = \rho \circ \Phi_h.$$

Evaluating $\Phi_{-h} \circ \rho(Q_k, W_k) = \Phi_{-h}(Q_k, -W_k)$, we obtain

$$\begin{cases} \overline{W}_{k+1/2} & = -W_k - \frac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\, W_k \times W_k + \tau(Q_k)\right) \\ \overline{Q}_{k+1} & = Q_k \operatorname{cay}(-h\overline{W}_{k+1/2}) \\ \overline{W}_{k+1} & = \overline{W}_{k+1/2} - \frac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\,\overline{W}_{k+1} \times \overline{W}_{k+1} + \tau(\overline{Q}_{k+1})\right). \end{cases} \tag{7}$$

Comparing (7) with (3), it can be seen that $\overline{W}_{k+1/2} = -W_{k+1/2}$, which also implies that $\overline{Q}_{k+1} = Q_{k+1}$. We can thus rewrite the last equation of (7) as

$$\begin{aligned} \overline{W}_{k+1} &= -W_{k+1/2} - \frac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\,\overline{W}_{k+1} \times \overline{W}_{k+1} + \tau(Q_{k+1})\right) \\ &= -\left[W_{k+1/2} + \frac{h}{2}\mathbb{I}^{-1}\left(\mathbb{I}\,\overline{W}_{k+1} \times \overline{W}_{k+1} + \tau(Q_{k+1})\right)\right] \end{aligned}$$

It is straightforward to see that $-W_{k+1}$ is a solution for the last equation; moreover, if h is sufficiently small the implicit function theorem assures that the solution is unique, that is, $\overline{W}_{k+1} = -W_{k+1}$. □

References

[1] J. C. Simo and L. Vu-Quoc, ''On the dynamics in space of rods undergoing large motions - a geometrically exact approach,'' *Computer Methods in Applied Mechanics and Engineering*, vol. 66, pp. 125--161, 1988.

[2] N. Bou-Rabee and J. E. Marsden, ''Hamilton-Pontryagin integrators on Lie groups,'' *Foundations of Computational Mathematics*, vol. 9, pp. 197--219, 2008.

[3] A. Iserles, S. P. Munthe-Kaas, H. Z. andNørsett, and A. Zanna, ''Lie-group methods,'' *Acta Numerica*, vol. 9, pp. 1--148, 2000.

[4] J. E. Marsden and M. West, ''Discrete mechanics and variational integrators,'' *Acta Numerica*, vol. 10, pp. 357--514, 2001.

[5] A. Lew, J. E. Marsden, M. Ortiz, and M. West, ''An overview of variational integrators,'' in *Finite Element Methods: 1970s and Beyond*, 2004, pp. 98--115.

 ISBN 13: 978-84-613-5510-5

[6] ------, ''Variational time integrators,'' *Int. J. Numer. Methods Eng.*, vol. 60, pp. 153--212, 2004.

[7] G. Benettin and A. Giorgilli, ''On the Hamiltonian interpolation of near to the identity symplectic mappings with applications to symplectic integration algorithms,'' *J. Statist. Phys.*, vol. 74, pp. 1117--1143, 1994.

[8] S. Reich, ''Backward error analysis for numerical integrators,'' *SIAM J. Num. Anal.*, vol. 36, pp. 1549--1570, 1999.

[9] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, ser. Springer Series in Computational Mathematics. Springer, 2006, vol. 31.

[10] C. Kane, J. E. Marsden, M. Ortiz, and M. West, ''Variational integrators and the Newmark algorithm for conservative and dissipative mechanical systems,'' *Int. J. Numer. Methods Eng.*, vol. 49, pp. 1295--1325, 2000.

[11] P. Krysl and L. Endres, ''Explicit Newmark/Verlet algorithm for time integration of the rotational dynamics of rigid bodies,'' *International Journal for Numerical Methods in Engineering*, vol. 62, pp. 2154--2177, 2005.

[12] N. Bou-Rabee, ''Hamilton-Pontryagin integrators on Lie groups,'' Ph.D. dissertation, California Institute of Technology, 2007.

[13] J. E. Marsden, S. Pekarsky, and S. Shkoller, ''Discrete Euler-Poincaré and Lie-Poisson equations,'' *Nonlinearity*, vol. 12, pp. 1647--1662, 1998.

[14] J. E. Marsden and J. Scheurle, ''The reduced Euler-Lagrange equations,'' *Fields Inst. Comm.*, vol. 1, pp. 139--164, 1993.

[15] E. Faou, E. Hairer, and T. Pham, ''Energy conservation with non-sympletic methods: examples and counter-examples,'' *BIT Numerical Mathematics*, vol. 44, pp. 699--709, 2004.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Multi-authority attribute based encryption with honest-but-curious central authority

Vladimir Božović[1], Daniel Socek[2], Rainer Steinwandt[3] and Viktória I. Villányi

[1] *Department of Science and Mathematics, University of Montenegro, Montenegro*

[2] *CoreTex Systems LLC, 2851 S Ocean Blvd. 5L, Boca Raton, FL 33432, U.S.A.*

[3] *Department of Mathematical Sciences, Florida Atlantic University, Boca Raton, U.S.A.*

emails: vladobozovic@yahoo.com, dsocek@gmail.com, rsteinwa@fau.edu, vvillanyi@gmail.com

Abstract

An attribute based encryption scheme capable of handling multiple authorities was recently proposed by Chase. The scheme is built upon a single-authority attribute based encryption scheme presented earlier by Sahai and Waters. Chase's construction uses a trusted central authority that is inherently capable of decrypting arbitrary ciphertexts created within the system. We present a multi-authority attribute based encryption scheme in which only the set of recipients defined by the encrypting party can decrypt a corresponding ciphertext. The central authority is viewed as "honest-but-curious": on the one hand it honestly follows the protocol, and on the other hand it is curious to decrypt arbitrary ciphertexts thus violating the intent of the encrypting party. The proposed scheme, which like its predecessors relies on the Bilinear Diffie-Hellman assumption, has a complexity comparable to that of Chase's scheme. We prove that our scheme is secure in the selective ID model and can tolerate an honest-but-curious central authority.

Key words: pairing-based cryptography, attribute based encryption
MSC 2000: 94A60

1 Introduction

In both standard *public key encryption* and *identity based encryption* a message is to be transmitted to a single recipient known at the time of encryption. Similarly, *broadcast encryption* addresses scenarios where a sender explicitly specifies a set of receivers (or revoked users) when encrypting a plaintext. In contrast, in an *attribute based encryption*

 ISBN 13: 978-84-613-5510-5

scheme, the sender does not provide an explicit list of recipients or revoked users when encrypting a plaintext, but instead, the recipient of a ciphertext is specified through a set of credentials, also referred to as the *attributes*, which are sufficient to decrypt a ciphertext. Fuzzy identity based encryption proposed by Sahai and Waters [7] can be used to address such a setting, if all attributes are controlled by a single authority.

The starting point of the current paper is a recent proposal of Chase [4] which considers *multi-authority attribute based encryption*, therewith solving an open problem from [7]. Chase's scheme is capable of handling disjoint sets of attributes that are distributed among multiple authorities. In this setting, an encrypting party specifies a set of attributes $\mathcal{A}_C$ with the attributes in $\mathcal{A}_C$ being controlled by several authorities. Let $\mathcal{A}_k$ be the set of attributes controlled by authority k. Then the ciphertext C associated with the attribute set $\mathcal{A}_C$ can only be decrypted by those users u with a set of attributes $\mathcal{A}_u$ for which the cardinality of the intersection $\mathcal{A}_u \cap \mathcal{A}_k \cap \mathcal{A}_C$ exceeds the respective threshold d_k, for each authority k.

As pointed out in [4], one of the primary challenges in implementing such a multi-authority attribute based encryption scheme is the prevention of collusion attacks among users that obtain secret key components from different authorities. Moreover, it is desirable that there be no communication between the individual authorities. To overcome these difficulties, Chase's scheme relies on a trusted central authority. The resulting scheme is capable of tolerating multiple corrupted authorities, but the honesty of the central authority remains of vital importance since, by the constriction from [4], the trusted authority has the capability of decrypting every ciphertext.

Our contribution. Building on Chase's proposal, we construct a threshold scheme for multi-authority attribute based encryption which offers the same security guarantees provided by Chase's construction, but in addition can tolerate an honest-but-curious central authority. Assuming the central authority is honest during the initialization phase, the indistinguishability of encryptions is guaranteed. As in [4], our security analysis is in the selective ID model and builds on the Decisional Bilinear Diffie Hellman assumption.

Related work. Since Shamir posed the problem of identity based encryption [8], various proposals have been made, a very partial list being the work in [6, 9, 10, 2, 5]. Building on the Bilinear Diffie Hellman assumption and the selective ID model [3, 1], at EUROCRYPT 2005 Waters presented an identity based encryption scheme in the standard model [11]. Sahai and Water's proposal for a fuzzy identity based encryption [7] provides an attribute based encryption with a single authority. Here, *fuzzy* refers to an identity id' being able to decrypt a ciphertext encrypted by an identity id if and only if id and id' are close to each other in the "set overlap" distance metric. This is of interest when dealing with noisy inputs, such as biometric templates. Building on the ideas from [7], Chase proposed a solution for multi-authority attribute based encryption, provided that a trusted central authority is available [4]. Our proposal aims at improving Chase's construction by imposing a weaker assumption on the central authority without paying a high cost in terms of efficiency.

 ISBN 13: 978-84-613-5510-5

2 Notation and preliminaries

As already mentioned, our proposal relies on the Decisional Bilinear Diffie Hellman assumption. For the sake of clarity, the next sections review the relevant terminology related to bilinear maps and multi-authority attribute based encryption. Section 2.3 discusses the security model where, like in [4], we make use of the selective ID model.

2.1 Bilinear maps and the Bilinear Diffie Hellman assumption

Let G_1, G_2 be groups of prime order p, and let P a generator of G_1. We assume p to be superpolynomial in the security parameter ℓ and that all group operations in G_1 and G_2 can be computed efficiently, i. e., in probabilistic polynomial time. We use additive notation for G_1 and multiplicative notation for G_2. By $e : G_1 \times G_1 \longrightarrow G_2$ we denote an admissible bilinear map, i. e., all of the following hold [2]:

- For all $P, Q \in G_1$ and for all $\alpha, \beta \in \mathbb{Z}$ we have $e(\alpha P, \beta Q) = e(P, Q)^{\alpha\beta}$.
- We have $e(P, P) \neq 1$, i. e., $e(P, P)$ is a generator of G_2.
- There is a probabilistic polynomial time algorithm that for arbitrary $P, Q \in G_1$ computes $e(P, Q)$.

In the above setting, the Decisional Bilinear Diffie Hellman (D-BDH) problem in (G_1, G_2, e) is the problem of distinguishing between the challenger's possible outputs in the following experiment: The challenger chooses $\alpha, \beta, \gamma, \eta \leftarrow \{0, 1, \ldots, p-1\}$ independently and uniformly at random, flips a fair binary coin $\delta \leftarrow \{0, 1\}$, and then outputs the tuple

$$(P, \alpha P, \beta P, \gamma P, e(P, P)^{\delta \cdot \alpha\beta\gamma + (1-\delta)\cdot\eta}).$$

In other words, with probability 1/2 the last component of the challenger's output is $e(P, P)^{\alpha\beta\gamma}$, and with probability 1/2 the last component is a uniformly at random chosen element from G_2. We define the *advantage* of algorithm $\mathcal{A}$ in solving the D-BDH problem as

$$\mathrm{Adv}_{\mathcal{A}}^{\mathsf{bdh}}(\ell) := \Pr(\delta' = \delta) - \frac{1}{2}$$

where δ' is the output of $\mathcal{A}$ when trying to guess the value of the fair binary coin δ. We say that an algorithm $\mathcal{A}$ has a *non-negligible advantage* in solving the D-BDH problem, if $\mathrm{Adv}_{\mathcal{A}}^{\mathsf{bdh}}$ is not negligible[1] where the probability is over the randomly chosen $\alpha, \beta, \gamma, \eta$ and the random bits consumed by $\mathcal{A}$.

Definition 1 (Decisional Bilinear Diffie Hellman assumption) *The Decisional Bilinear Diffie Hellman assumption holds for (G_1, G_2, e) if there exists no probabilistic polynomial time algorithm having non-negligible advantage in solving the above* D-BDH *problem.*

[1] We refer to a function $f : \mathbb{N}_{>0} \longrightarrow \mathbb{R}$ as negligible, if $|f| = |f(\ell)| \in \frac{1}{\ell^{o(1)}}$.

 ISBN 13: 978-84-613-5510-5

2.2 Authorities, attributes and users

Let $\mathcal{K}$ be the polynomial size set of authorities and $\mathcal{U}$ the polynomial size set of users we consider, and denote by $\mathcal{A}_k$ the polynomial size set of attributes handled by authority $k \in \mathcal{K}$. We impose that the sets $\mathcal{A}_k$ are pairwise disjoint, i. e., the *universal attribute set*

$$\mathcal{A} := \biguplus_{k \in \mathcal{K}} \mathcal{A}_k$$

is the disjoint union of the $\mathcal{A}_k$. In addition to the authorities $k \in \mathcal{K}$, there is one central authority $k_{\mathsf{CA}} \notin \mathcal{K}$ which we will model as honest-but-curious—the central authority k_{CA} honestly follows the protocol, but will try to decrypt ciphertexts sent by users in the system. During an initialization phase we allow communication between k_{CA} and k for each authority $k \in \mathcal{K}$, but thereafter no communication between the central authority and the authorities $k \in \mathcal{K}$ is possible: while the central authority k_{CA} is involved in setting up the system, we do not want to rely on k_{CA} being available throughout the complete lifetime of the system. Also, we do not allow any communication among the authorities in $\mathcal{K}$.

To distinguish different users, we follow [4] and assume that each user $u \in \mathcal{U}$ has a unique identifier. Depending on the application, the identifier could refer to a social security number or a passport number, for instance. We denote the set of those attributes in $\mathcal{A}$ that are available to user $u \in \mathcal{U}$ by $\mathcal{A}_u$. Similarly, we write $\mathcal{A}_C$ for the set of attributes that is associated with a ciphertext C. This set $\mathcal{A}_C$ is chosen by the encrypting party as part of the input to the encryption algorithm, the other part of the input being the plaintext. We associate with each authority $k \in \mathcal{K}$ a threshold $d_k \in \mathbb{N}_{>0}$. The goal is that exactly those users u satisfying

$$|\mathcal{A}_u \cap \mathcal{A}_k \cap \mathcal{A}_C| \geq d_k \text{ for every } k \in \mathcal{K}$$

are able to decrypt the ciphertext C. In other words, for each authority k, user u must have at least d_k of the attributes that have been specified at the time of encryption. To decrypt a ciphertext, user $u \in \mathcal{U}$ uses the secret keys obtained during the initialization phase from the authorities $k \in \mathcal{K}$. Figure 1 lists the main components of a multi-authority attribute based encryption scheme (cf. [4]).

Remark 1 *Unlike [4] we do not make use of a* central key generation algorithm*, run by the central authority k_{CA} to generate secret keys for users u. Without loss of generality, in the security model we therefore will not give the adversary the possibility to query k_{CA} for private user keys. In the scheme we discuss, private user keys are generated by the attribute authorities $k \in \mathcal{K}$ only.*

A crucial feature of a multi-authority attribute based encryption scheme is the prevention of collusions among users: we want to prevent that any set of users, each of which is not able to decrypt a ciphertext C, can combine their information to decrypt C. The security definition discussed next tries to capture this design goal.

 ISBN 13: 978-84-613-5510-5

Setup. A probabilistic polynomial time algorithm[a] that given the security parameter 1^ℓ, a list of pairwise disjoint sets of attributes $[\mathcal{A}_k]_{k\in\mathcal{K}}$ and thresholds $[d_k]_{k\in\mathcal{K}}$ generates

- a (public key, secret key)-pair for each attribute authority $k \in \mathcal{K}$
- public system parameters.

Attribute key generation. A probabilistic polynomial time algorithm that given an attribute authority k's secret key, the corresponding threshold d_k, a (unique identifier of a) user u and a subset $\mathcal{A}_u \subseteq \mathcal{A}_k$ outputs decryption keys for user u.

Encryption. A probabilistic polynomial time algorithm that given a plaintext, attributes $\mathcal{A}_C \subseteq \mathcal{A}$ and the public system parameters, outputs a ciphertext C.

Decryption. A deterministic polynomial time algorithm that given a set of decryption keys for a set of attributes $\mathcal{A}_u$ and a ciphertext C encrypted with attribute set $\mathcal{A}_C$, outputs the corresponding plaintext M if $|\mathcal{A}_u \cap \mathcal{A}_k \cap \mathcal{A}_C| \geq d_k$ for all attribute authorities $k \in \mathcal{K}$; otherwise it outputs an error symbol $\perp$.

[a] It may be preferable to realize this computation in a distributed fashion, involving individual attribute authorities and some central authority. Below we will use such a distributed realization.

Figure 1: Algorithms in a multi-authority attribute based encryption scheme.

2.3 Security model

Like [4], we use a selective ID model for the security analysis. The adversary $\mathcal{H}$ has to specify the set of attributes that he wants to attack before receiving any public keys of the system. Figure 2 shows the game an adversary has to win to defeat the security of our scheme. As in [4], for our security analysis we impose the technical restriction that the adversary does not query the same attribute authority twice for private keys of the same user.

For a multi-authority attribute based encryption scheme to be secure, we require that there is no efficient algorithm achieving a non-negligible advantage in the game in Figure 2. More specifically, we define the advantage of an adversary $\mathcal{H}$ in the game in Figure 2 as

$$\mathrm{Adv}^{\mathsf{sid}}_{\mathcal{H}}(\ell) := \Pr(\delta' = \delta) - \frac{1}{2}$$

and make the following definition.

Definition 2 (Security in the selective ID model) *A scheme for multi-authority attribute based encryption is* secure in the selective ID model*, if for all probabilistic polynomial time adversaries $\mathcal{H}$, the advantage $\mathrm{Adv}^{\mathsf{sid}}_{\mathcal{H}}(\ell)$ is negligible.*

 ISBN 13: 978-84-613-5510-5

Setup. 1. Given the security parameter 1^ℓ, the adversary $\mathcal{H}$ outputs

- a non-empty list $\mathcal{U}$ of (unique identifiers of) users
- a non-empty list $\mathcal{K}$ of (unique identifiers of) attribute authorities
- a list $[(\mathcal{A}_k, \mathsf{corrupted}, d_k)]_{k\in\mathcal{K}}$ of non-empty, pairwise disjoint attribute sets, each along with a threshold $d_k \in \mathbb{N}_{>0}$ and a flag indicating if the respective authority is corrupted. There must be at least one uncorrupted authority.[a]
- a non-empty set of attributes $\mathcal{A}_C \subseteq \biguplus_{k\in\mathcal{K}} \mathcal{A}_k$ that will be associated with the challenge ciphertext.

2. The public and secret keys are generated, and $\mathcal{H}$ learns

- the public keys of all attribute authorities
- the public system parameters
- the complete history of all those authorities $k \in \mathcal{K}$ that are corrupted.

Secret key queries. The adversary can query the authorities $k \in \mathcal{K}$ for private user keys for attributes in $\mathcal{A}_k$ for user u. Whenever the adversary queries k for a secret key for attribute $a \in \mathcal{A}_k$ for user u, the attribute a is added to the (initially empty) set $\mathcal{A}_u$. The only restrictions for secret key queries are the following:

- at any time, for each user u there is at least one uncorrupted authority $\hat{k} = \hat{k}(u)$ with $|\mathcal{A}_u \cap \mathcal{A}_{\hat{k}} \cap \mathcal{A}_C| < d_{\hat{k}}$[b]
- for each user u, no authority $k \in \mathcal{K}$ is queried more than once for private keys of u.

Challenge. 1. The adversary $\mathcal{H}$ outputs two equal length messages M_0, M_1.

2. The challenger flips a fair binary coin $\delta \leftarrow \{0, 1\}$ and then applies the encryption algorithm to M_δ and the attribute set $\mathcal{A}_C$.
3. The resulting ciphertext C is given to the adversary $\mathcal{H}$.

Further secret key queries. The adversary can query for further private keys of users, subject to the same restrictions as before: for each user u there is at least one uncorrupted authority $\hat{k} = \hat{k}(u)$ with $|\mathcal{A}_u \cap \mathcal{A}_{\hat{k}} \cap \mathcal{A}_C| < d_{\hat{k}}$, and for each user u, no authority $k \in \mathcal{K}$ is queried more than once for private keys of u.

Guess. The adversary $\mathcal{H}$ outputs a guess δ' for the challenger's secret coin δ.

[a] Note that the central authority k_{CA} is not included in this list and in particular cannot be corrupted.

[b] The uncorrupted authority $\hat{k} = \hat{k}(u)$ may be different for each user u.

Figure 2: Attacking multi-authority attribute based encryption in the selective ID model.

 ISBN 13: 978-84-613-5510-5

The security requirement in Definition 2 does not address the question which information is available to the central authority. Specifically, in Chase's scheme [4], the central authority has the capability of reading arbitrary ciphertexts constructed by the users within the system. To express a requirement that limits the possibilities of an honest-but-curious central authority, we take a more detailed look at the setup phase, which is combined into a single algorithm in Figure 1. More precisely, this step can be seen as a simple protocol where the central authority k_{CA} securely communicates with the attribute authorities.

Remark 2 *From a practical perspective, it is desirable to have no communication among attribute authorities, and only very limited interaction of the central authority with each attribute authority. In the protocol in Section 3.1, the central authority sends one message to each attribute authority and derives the public system parameters from the replies.*

The game in Figure 3 captures a setting where an honest-but-curious central authority tries to violate the indistinguishability of ciphertexts. We introduce a "curious" algorithm $\mathcal{B}$ which, similarly as the "outside adversary" $\mathcal{H}$ in Figure 2, fixes the attribute sets and their distribution among the attribute authorities. Further on, $\mathcal{B}$ specifies the set of attributes that will be associated with the challenge ciphertext. At the end of the setup phase, $\mathcal{B}$ learns the complete state of the central authority, and based on this knowledge then tries to violate the indistinguishability of ciphertexts.
For an algorithm $\mathcal{B}$, we define the advantage in the game in Figure 3 as

$$\mathrm{Adv}^{\mathsf{ca}}_{\mathcal{B}}(\ell) := \Pr(\delta' = \delta) - \frac{1}{2} \quad .$$

Definition 3 (Tolerating an honest-but-curious central authority) *A scheme for multi-authority attribute based encryption* can tolerate an honest-but-curious central authority*, if for all probabilistic time algorithms $\mathcal{B}$, the advantage $\mathrm{Adv}^{\mathsf{ca}}_{\mathcal{B}}(\ell)$ is negligible.*

Remark 3 *Unlike for the adversary $\mathcal{H}$ in Figure 2, we do not require that an honest-but-curious central authority specifies the challenge attributes $\mathcal{A}_C$ in advance: algorithm $\mathcal{B}$ in Figure 3 does not have to provide this set before the challenge phase.*

We are now in the position to describe our suggestion for a multi-authority attribute based encryption scheme and to discuss its security in the sense of both Definition 2 and Definition 3.

3 Proposed protocol

We adopt the notation from Section 2 with G_1, G_2 being groups of prime order p, P a generator of G_1 and $e : G_1 \times G_1 \longrightarrow G_2$ an admissible bilinear map. We assume the unique identifiers for users u and for the attribute authorities $k \in \mathcal{K}$ to be public.

 ISBN 13: 978-84-613-5510-5

Setup

1. Given the security parameter 1^ℓ, the algorithm $\mathcal{B}$ outputs
   - a non-empty list $\mathcal{U}$ of (unique identifiers of) users
   - a non-empty list $\mathcal{K}$ of (unique identifiers of) attribute authorities
   - a list $[(\mathcal{A}_k, \mathsf{corrupted}, d_k)]_{k\in\mathcal{K}}$ of non-empty, pairwise disjoint attribute sets, each along with a threshold $d_k \in \mathbb{N}_{>0}$ and a flag indicating if the respective authority is corrupted. There must be at least one uncorrupted authority.[a]
2. The public and secret keys of all authorities $k \in \mathcal{K}$ are generated, and $\mathcal{B}$ learns
   - all public keys
   - the public system parameters
   - the complete history of all those authorities $k \in \mathcal{K}$ that are corrupted
   - the complete history of the central authority k_{CA}.

Challenge

1. The algorithm $\mathcal{B}$ outputs two equal length messages M_0, M_1 and a non-empty set of attributes $\mathcal{A}_C \subseteq \biguplus_{k\in\mathcal{K}} \mathcal{A}_k$.
2. The challenger flips a fair binary coin binary $\delta \leftarrow \{0, 1\}$ and then applies the encryption algorithm to M_δ and the attribute set $\mathcal{A}_C$.
3. The resulting ciphertext C is given to $\mathcal{B}$.

Guess

The algorithm $\mathcal{B}$ outputs a guess δ' for the challenger's secret coin δ.

[a]Note that the central authority k_{CA} is not included in this list and in particular cannot be corrupted.

Figure 3: Dealing with an honest-but-curious central authority.

Similarly, we assume the sets of attributes $\mathcal{A}_k$ and the corresponding threshold d_k to be public—in particular, all these values are known to the central authority k_{CA}, which we invoke (only) in the setup phase. In order to generate secret keys for users, we assume that each attribute $a \in \mathcal{A}$ can be identified with a number $\iota(a) \in \{1, \ldots, p-1\}$—for practical purposes, $\iota(a)$ could be based on a hash value, for instance.

3.1 The proposed protocol

3.1.1 Setup.

The setup phase requires one message to be sent from the central authority to each of the attribute authorities. It is assumed that the adversary has no possibility to interfere with or to access this communication:

 ISBN 13: 978-84-613-5510-5

The central authority k_{CA} chooses, for each pair $(k,u) \in \mathcal{K} \times \mathcal{U}$, uniformly at random a secret value $s_{k,u} \leftarrow \{0,\ldots,p-1\}$. In additon, k_{CA} chooses $\sigma \in \{0,\ldots p-1\}$ uniformly at random, and for each $u \in \mathcal{U}$ computes a "dummy secret" $s_{k_{\mathsf{CA}},u} := \sigma - \sum_{k\in\mathcal{K}} s_{k,u}$. The sequence

$$\underbrace{[s_{k,u} \cdot P]}_{=:S_{k,u}}{}_{u\in\mathcal{U}}$$

is sent to attribute authority k ($k \in \mathcal{K}$), and k_{CA} publishes the public system parameters

$$\Big([s_{k_{\mathsf{CA}},u} \cdot P]_{u\in\mathcal{U}}, \underbrace{e(P,P)^{\sigma}}_{=:\mathsf{pk}}\Big).$$

Remark 4 *The value $s_{k_{\mathsf{CA}},u} \cdot P$ is only needed by user u. To decrease the size of the public parameters, instead of publishing the sequence $[s_{k_{\mathsf{CA}},u} \cdot P]_{u\in\mathcal{U}}$, alternatively a scenario could be considered where $s_{k_{\mathsf{CA}},u} \cdot P$ is transmitted to u (only).*

Attribute authority $k \in \mathcal{K}$ receives the corresponding sequence of $S_{k,u}$-values from k_{CA} and chooses a value $r_k \leftarrow \{0,\ldots,p-1\}$ uniformly at random. Moreover, for each of its attributes $a \in \mathcal{A}_k$, a secret value $t_{k,a} \leftarrow (\mathbb{Z}/p\mathbb{Z})^*$ is chosen uniformly at random by k, and the pair

$$\Big(e(P,P)^{r_k}, \underbrace{[t_{k,a} \cdot P]}_{=:T_{k,a}}{}_{a\in\mathcal{A}_k}\Big)$$

forms k's public key. The secret key of k contains the aforementioned values r_k, $[S_{k,u}]_{u\in\mathcal{U}}$, and $[t_{k,a}]_{a\in\mathcal{A}_k}$. Finally, for each user $u \in \mathcal{U}$, attribute authority k chooses uniformly at random a secret polynomial $f_{k,u} \in \mathbb{F}_p[X]$ of degree $< d_k$.

Remark 5 *The value $e(P,P)^{r_k}$ is only used during encryption and decryption to compute the product $\mathsf{pk} \cdot \prod_{k\in\mathcal{K}} e(P,P)^{r_k}$—which is ciphertext-independent. If one allows the attribute authorities to contribute to the generation of the public system parameters, the $e(P,P)^{r_k}$-component in the attribute authorities' public keys can be omitted. To do so, the public system parameter $\mathsf{pk} = e(P,P)^{\sigma}$ can be replaced with $e(P,P)^{\sigma+\sum_{k\in\mathcal{K}} r_k}$.*

3.1.2 Attribute key generation.

To extract the secret decryption key associated with an attribute $a \in \mathcal{A}_k \cap \mathcal{A}_u$ for a user $u \in \mathcal{U}$, attribute authority k proceeds as follows:

- The secret value $X_{k,u} := S_{k,u} + (r_k - f_{k,u}(0)) \cdot P$, which depends on k and u, but not the specific attribute a, is computed and given to u.
- The attribute-specific value $D_{k,u,a} := \frac{f_{k,u}(\iota(a))}{t_{k,a}} \cdot P$ is computed and given to u.

 ISBN 13: 978-84-613-5510-5

3.1.3 Encryption.

To encrypt a plaintext $M \in G_2$ with associated attribute set $\mathcal{A}_C \subseteq \mathcal{A}$, the encrypting party chooses $s \leftarrow \{0, \ldots, p-1\}$ uniformly at random and computes the ciphertext

$$\left(\left(\mathsf{pk} \cdot \prod\nolimits_{k\in\mathcal{K}} e(P,P)^{r_k}\right)^s \cdot M, s \cdot P, [s \cdot T_{k,a}]_{a\in\mathcal{A}_C}\right).$$

3.1.4 Decryption.

Let $C = ((\mathsf{pk} \cdot \prod_{k\in\mathcal{K}} e(P,P)^{r_k})^s \cdot M, s \cdot P, [s \cdot T_{k,a}]_{a\in\mathcal{A}_C})$ be a ciphertext with associated attribute set $\mathcal{A}_C$, and suppose that user u's attribute set $\mathcal{A}_u$ satisfies $|\mathcal{A}_u \cap \mathcal{A}_k| \geq d_k$ for all $k \in \mathcal{K}$. Then u can recover the plaintext M as follows.

1. For each $k \in \mathcal{K}$, he chooses d_k attributes $a \in \mathcal{A}_u \cap \mathcal{A}_k$, and computes

   $$e(s \cdot T_{k,a}, D_{k,u,a}) = e(P,P)^{f_{k,u}(\iota(a))\cdot s}.$$

   Then, using Lagrange polynomial interpolation, u computes

   $$e(P,P)^{f_{k,u}(0)\cdot s}.$$

2. Further on, for each $k \in \mathcal{K}$, user u can use the $X_{k,u}$-component of his secret key to compute $e(X_{k,u}, s \cdot P) = e(P,P)^{(s_{k,u}+r_k-f_{k,u}(0))\cdot s}$.

3. Multiplying $e(s \cdot P, s_{k_{\mathsf{CA}},u} \cdot P)$ with all of the above values yields

   $$\begin{aligned} & e(s \cdot P, s_{k_{\mathsf{CA}},u} \cdot P) \cdot \prod_{k\in\mathcal{K}} e(P,P)^{f_{k,u}(0)\cdot s} \cdot e(P,P)^{(s_{k,u}+r_k-f_{k,u}(0))\cdot s} \\ = \; & e(P,P)^{s\cdot s_{k_{\mathsf{CA}},u}} \cdot e(P,P)^{s\cdot\sum_{k\in\mathcal{K}}(s_{k,u}+r_k)} \\ = \; & e(P,P)^{s\cdot(\sigma+\sum_{k\in\mathcal{K}} r_k)} \\ = \; & \left(\mathsf{pk} \cdot \prod_{k\in\mathcal{K}} e(P,P)^{r_k}\right)^s. \end{aligned}$$

   By inverting this element and multiplying the result with the first component of the ciphertext, the plaintext M can be recovered.

3.2 Adding new authorities

The "dummy secrets" $s_{k_{\mathsf{CA}},u}$ facilitate the introduction of new authorities to a previously established protocol. To add a new authority k^*, the central authority k_{CA} replaces the old value σ with a new uniformly at random chosen σ', and replaces each $s_{k_{\mathsf{CA}},u}$ with $\sigma' - \sum_{k\in\mathcal{K}\cup\{k^*\}} s_{k,u}$. Then the updated "dummy public keys" $s_{k_{\mathsf{CA}},u} \cdot P$ have to be communicated to the users, and the new authority k^* can compute its secret and public key as before.

 ISBN 13: 978-84-613-5510-5

4 Security analysis

The protocol proposed in Section 3 can be shown to be secure both both in the sense of Definition 2 and Definition 3. Proofs for the subsequent two theorems are given in the extended version of this paper.

Theorem 1 *Suppose there exists a probabilistic polynomial time adversary $\mathcal{H}$ against the protocol in Section 3.1 having a non-negligible advantage in the game in Figure 2. Then there is a probabilistic polynomial time algorithm $\mathcal{S}$ having a non-negligible advantage in solving the* D-BDH*-problem.*

Our proof of Theorem 1 builds on the analysis of Chase's scheme in [4], and it is worth noting that the reduction to a D-BDH adversary $\mathcal{S}$ in the proof is tight: Essentially, the advantage of the adversary $\mathcal{H}$ violating security in the selective ID model is only halved at the cost of simulating the attribute authorities k and the central authority k_{CA}.

Theorem 2 *Let $\mathcal{B}$ be a probabilistic polynomial time adversary against the protocol in Section 3.1 having a non-negligible advantage in the game in Figure 3. Then there is a probabilistic polynomial time algorithm $\mathcal{S}$ having a non-negligible advantage in solving the* D-BDH*-problem.*

To prove Theorem 2, i. e., that the proposed scheme can tolerate an honest-but-curious central authority in the sense of Definition 3, a similar argument as in the proof of Theorem 1 can be used. It turns out that again there is a tight security reduction: Essentially, for the price of simulating the central authority and the attribute authorities, from an adversary $\mathcal{B}$ described in the game from Figure 3, we obtain a D-BDH adversary whose advantage is half the advantage of $\mathcal{B}$.

5 Conclusion

Building on the proposal for multi-authority based attribute based encryption from [4], we constructed a scheme where the central authority is no longer capable of decrypting arbitrary ciphertexts created within the system. In addition to providing security in the selective ID model, the proposed system can tolerate an honest-but-curious central authority. Since both Chase's scheme and the proposed scheme rely on the same hardness assumption, and have a comparable complexity, the new scheme seems a viable alternative to Chase's construction. However, since only the proposed method is capable of handling a curious yet honest central authority, the proposed scheme is recommended in applications where security against such a central authority is required.

 ISBN 13: 978-84-613-5510-5

References

[1] D. Boneh and X. Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Springer-Verlag, 2004.

[2] D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag, 2001.

[3] R. Canetti, S. Halevi, and J. Katz. A Forward-Secure Public-Key Encryption Scheme. In E. Biham, editor, *Advances in Cryptology – EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Computer Science*, pages 255–271. Springer-Verlag, 2003.

[4] M. Chase. Multi-authority Attribute Based Encryption. In S.P. Vadhan, editor, *Theory of Cryptography – TCC 2007*, volume 4392 of *Lecture Notes in Computer Science*, pages 515–534. Springer-Verlag, 2007.

[5] C. Cocks. An Identity Based Encryption Scheme Based on Quadratic Residues. In B. Honary, editor, *Cryptography and Coding, 8th IMA International Conference*, volume 2260 of *Lecture Notes in Computer Science*, pages 360–363. Springer-Verlag, 2001.

[6] Y. Desmedt and J-J. Quisquater. Public-key Systems Based on the Difficulty of Tampering (Is there a difference between DES and RSA?). In A. M. Odlyzko, editor, *Advances in Cryptology – CRYPTO '86*, volume 263 of *Lecture Notes in Computer Science*, pages 111–117. Springer-Verlag, 1987.

[7] A. Sahai and B. Waters. Fuzzy Identity-Based Encryption. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer-Verlag, 2005.

[8] A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology – CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1985.

[9] H. Tanaka. A Realization Scheme for the Identity-Based Cryptosystem. In C. Pomerance, editor, *Advances in Cryptology – CRYPTO '87*, volume 293 of *Lecture Notes in Computer Science*, pages 340–349. Springer-Verlag, 1988.

[10] S. Tsujii and T. Itoh. An ID-Based Cryptosystem Based on the Discrete Logarithm Problem. *IEEE Journal on Selected Areas in Communications*, 7(4), May 1989.

[11] B. Waters. Efficient Identity-Based Encryption Without Random Oracles. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer-Verlag, 2005.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Convergence of an Adaptive Approximation Scheme for the Wiener Process

Mats Brodén[1] **and Magnus Wiktorsson**[1]

[1] *Centre for Mathematical Sciences, Lund University*

emails: matsb@maths.lth.se, magnusw2@maths.lth.se

Abstract

The problem of approximating/tracking the value of a Wiener process is considered. The discretization points are placed at times when the value of the process differs from the approximation by some amount, here denoted by η. It is found that the limiting difference, as η goes to 0, between the approximation and the value of the process normalized with η converges in distribution to a triangularly distributed random variable.

Key words: Discretization error, convergence in distribution, triangular distribution
MSC 2000: 60F05, 60G15

1 Introduction and preliminaries

An adaptive approximation scheme of the Wiener process is considered. The discretization points are placed at times when the value of the true process differs from the approximation by some amount, here denoted by η. This can be seen as a control problem where we want to track the true value of the process with our approximation, and where both the process and its approximation are fully observable. The approximation strategy presented here may be feasible when discretization is associated with some cost that should be kept low. Examples of related problems is that of discrete time hedging of derivative contracts in financial markets (see e.g. [3]) and certain space-time discretization schemes of stochastic differential equations (see e.g. [5]).

Let X be a diffusion process defined by $X_t = \sigma W_t$, where W denotes a one dimensional standard Wiener process. Define, for some $\eta > 0$, a sequence of stopping times $\{t_i^\eta\}_{i\geq 0}$ by $t_{i+1}^\eta = \inf\{t > t_i^\eta \,|\, |X_t - X_{t_i^\eta}| = \eta\}$, where $t_0^\eta = 0$. The components of the sequence t^η may be seen as epochs of the renewal process N^η defined by $N_t^\eta = \sup\{i : t_i^\eta \leq t\}$. Furthermore, let the sequence $\{\tau_i^\eta\}_{i\geq 1}$ of interarrival times be defined by $\tau_i^\eta = t_i^\eta - t_{i-1}^\eta$, and define the renewal-reward process φ by $\varphi_t^\eta := \sum_{i=1}^{N_t^\eta} \tau_i^\eta$.

 ISBN 13: 978-84-613-5510-5

The process $X_{\varphi_t^\eta}$ may also be seen as a renewal-reward process, but with a reward that takes the values $-\eta$ and η with equal probability.

The aim of this work is to investigate the asymptotic behavior of $(X_t - X_{\varphi_t^\eta})/\eta$ as η approaches 0. It will be seen that this quantity converges, pointwise for each $t > 0$, in distribution to a stochastic variable which is triagularly distributed.

Before we end this section we will state some resluts regarding barrier crossings and renewal processes. The main result is presented in Section 2. In Section 3 we perform a simulation study and investigate the transition to the limiting distribution.

1.1 The Wiener process with two absorbing barriers

Since the components of the sequence $\{\tau_i^\eta\}_{i\geq 1}$ are independent and identically distributed, we will let τ^η denote a stochastic variable with the same properties as these τ_i^η's, and which may be characterized by $\tau^\eta = \inf\{t > 0 \,|\, |X_t| = \eta\}$.

Now, consider the process X absorbed in $-\eta$ and η, that is $X_{t\wedge\tau}$. The transition density of this process, from $X_0 = 0$, may be represented by (see [1])

$$p^\eta(t,x) = \sum_{k=1}^{\infty} \frac{1}{\eta^2} e^{-\frac{1}{2}\left(\frac{k\sigma\pi}{2\eta}\right)^2 t} \sin\left(\frac{k\pi}{2}\right) \sin\left(\frac{k\pi(x+\eta)}{2\eta}\right),\ (t,x) \in (0,\infty)\times[-\eta,\eta]. \quad (1)$$

This transition density may also be expressed as an infinite sum over Gaussian kernels (see [1])

$$p^\eta(t,x) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2 t}} \left(e^{-\frac{(x-4k\eta)^2}{2\sigma^2 t}} - e^{-\frac{(x-2\eta+4k\eta)^2}{2\sigma^2 t}}\right),\ (t,x) \in [0,\infty)\times[-\eta,\eta]. \quad (2)$$

Lemma 1. *The integral of $p^1(t,x)$*

a) with respect to t over the interval $[a,b] \subset [0,\infty)$ may be represented as

$$\int_a^b p^\eta(t,x)\,dt = \sum_{k=1}^{\infty} \int_a^b \frac{e^{-\frac{1}{2}\left(\frac{k\sigma\pi}{2\eta}\right)^2 t}}{\eta^2} \sin\left(\frac{k\pi}{2}\right) \sin\left(\frac{k\pi(x+\eta)}{2\eta}\right) dt,\ x \in [-\eta,\eta].$$

b) with respect to x over the interval $[a,b] \subset [-\eta,\eta]$ may be represented as

$$\int_a^b p^\eta(t,x)\,dx = \sum_{k=-\infty}^{\infty} \int_a^b \frac{e^{-\frac{1}{2}\left(\frac{k\sigma\pi}{2\eta}\right)^2 t}}{\eta^2} \sin\left(\frac{k\pi}{2}\right) \sin\left(\frac{k\pi(x+\eta)}{2\eta}\right) dx\,, t > 0. \quad (3)$$

or as

$$\int_a^b p^\eta(t,x)\,dx = \sum_{k=-\infty}^{\infty} \int_a^b \frac{1}{\sqrt{2\pi\sigma^2 t}} \left(e^{-\frac{(x-4k\eta)^2}{2\sigma^2 t}} - e^{-\frac{(x-2+4k\eta)^2}{2\sigma^2 t}}\right) dx,\ t \geq 0. \quad (4)$$

 ISBN 13: 978-84-613-5510-5

Proof. a) Define the functions g_k^F and $G_n^F$by

$$g_k^F(t,x) = \frac{e^{-\frac{1}{2}\left(\frac{k\sigma\pi}{2\eta}\right)^2 t}}{\eta^2} \sin\left(\frac{k\pi}{2}\right) \sin\left(\frac{k\pi(x+\eta)}{2\eta}\right),$$

and $G_n^F(t,x) = \sum_{k=1}^n g_k^F(t,x)$ then $\lim_{n\uparrow\infty} G_n^F(t,x) = p^\eta(t,x)$. Since $g_k^F(t,0) \geq 0$ it follows that $0 \leq G_n^F(t,0) \leq G_{n+1}^F(t,0)$, and consequently by Lebesgues monotone convergence thorem $\int_a^b \lim_{n\uparrow\infty} G_n(t,0)\mathrm{d}t = \lim_{n\uparrow\infty} \int_a^b G_n(t,0)\mathrm{d}t$. Extending the integral we get $\lim_{n\uparrow\infty} \int_a^b G_n(t,0)\mathrm{d}t \leq \lim_{n\uparrow\infty} \int_0^\infty G_n(t,0)\mathrm{d}t$. Moving the integral inside of the sum in $G_n(t,0)$ and performing the integration over $\mathbb{R}_+$ we get the sum $\lim_{n\uparrow\infty} \sum_{k=1}^n 8/(k^2\pi^2\sigma^2) = 4/(3\sigma^2)$, and hence $\lim_{n\uparrow\infty} \int_0^\infty G_n(t,0)\mathrm{d}t < \infty$. Since $|\sin(k\pi/2)\sin(k\pi(x+\eta)/(2\eta))| \leq 1$ it holds that $|g_k^F(t,x)| \leq g_k^F(t,0)$ which implies that $|G_n^F(t,x)| \leq G_n^F(t,0)$. Since $G_n^F(t,0)$ is bounded by $\lim_{n\uparrow\infty} G_n^F(t,0)$ the function $G_n^F(t,x)$ is dominated by the integrable function $\lim_{n\uparrow\infty} G_n^F(t,0)$ and by the dominated convergence theorem it follows that $\int_a^b \lim_{n\uparrow\infty} G_n(t,x)\mathrm{d}t = \lim_{n\uparrow\infty} \int_a^b G_n(t,x)\mathrm{d}t$. Moving the integral inside of the sum on the right hand side the claim is proved.

b) Eqn. (3) From the proof of *a)* we know that $G_n^F(t,x) \leq p^\eta(t,0) = \lim_{n\uparrow\infty} G_n(t,0)$ which is bounded for every $t > 0$. Since the set $[a,b]$ is bounded (i.e. $[a,b] \subset [-\eta,\eta]$), the claim now follows from the bounded convergence thorem.

Eqn. (4) Define the functions g_k^G and G_n^G by

$$g_k^G(t,x) = \frac{e^{-\frac{(x-k)^2}{2\sigma^2 t}}}{\sqrt{2\pi\sigma^2 t}} \qquad \text{and} \qquad G_n^G(t,x) = \sum_{k=-n}^{n} (g_{4k\eta}^G(t,x) - g_{2-4k\eta}^G(t,x)),$$

then $\lim_{n\uparrow\infty} G_n^G(t,x) = p^\eta(t,x)$. The function G_n^G may be decomposed as

$$G_n^G(t,x) = G_n^{G,1}(t,x) + G_n^{G,2}(t,x),$$

where

$$G_n^{G,1}(t,x) = \sum_{k=0}^{n} (g_{4k\eta}^G(t,x) - g_{4k\eta+2}^G(t,x)),$$
$$G_n^{G,2}(t,x) = \sum_{k=-1}^{-n} (g_{4k\eta}^G(t,x) - g_{4k\eta+2}^G(t,x)).$$

Since each term in $G_n^{G,1}$ is positive and each term in $G_n^{G,2}$ is negative it holds that

$$0 \leq G_n^{G,1}(t,x) \leq G_{n+1}^{G,1}(t,x) \qquad \text{and} \qquad 0 \geq G_n^{G,2}(t,x) \geq G_{n+1}^{G,2}(t,x).$$

The claim now follows by Lebesgues monotone convergence theorem. □

Lemma 2. *It holds that*

$$\sigma^2 \int_0^\infty p^1(t,x)dt = (1-|x|)^+.$$

 ISBN 13: 978-84-613-5510-5

Proof. From Lemma 1 *a)* we have that

$$\int_0^\infty p^1(t,x)\mathrm{d}t = \frac{8}{\pi^2\sigma^2}\sum_{k=1}^\infty \frac{1}{k}\sin\left(\frac{k\pi}{2}\right)\frac{1}{k}\sin\left(\frac{k\pi(x+1)}{2}\right).$$

The idea is to find a function that can be expressed as a series which corresponds to the above sum. Let $s_1 = 1/2$ and $s_2 = (x+1)/2$, then

$$\frac{\pi^2\sigma^2}{8}\int_0^\infty p^1(t,x)\mathrm{d}t = \sum_{k=1}^\infty \frac{1}{k}\sin(k\pi s_1)\frac{1}{k}\sin(k\pi s_2).$$

Define the function h_s by

$$h_s(x) = \begin{cases} 0\,, & 0 \le |x| \le s\,, \\ 1\,, & s < |x| \le 1\,. \end{cases}$$

The Fourier Cosine coefficients of h_s are given by

$$\begin{aligned} c_0 &= \int_0^1 h_s(x)\mathrm{d}x = 1-s\,, \\ a_k &= 2\int_0^1 \cos(k\pi x)h_s(x)\mathrm{d}x = -\frac{2\sin(\pi k s)}{\pi k}\,. \end{aligned}$$

Applying Parseval's formula yields

$$\int_0^1 h_{s_1}(x)h_{s_2}(x)\mathrm{d}x = 2\sum_{k=1}^\infty \frac{\sin(\pi k s_1)}{\pi k}\frac{\sin(\pi k s_2)}{\pi k} + (1-s_1)(1-s_2)\,.$$

Assume that $x \in [0,1]$, then $0 \le s_1 \le s_2 \le 1$ and

$$\sum_{k=1}^\infty \frac{2}{\pi^2k^2}\sin(\pi k s_1)\sin(\pi k s_2) = 1 - s_2 - (1-s_1)(1-s_2) = s_1(1-s_2).$$

Thus

$$\sigma^2\int_0^\infty p^1(t,x)\mathrm{d}t = 4\sum_{k=1}^\infty \frac{2}{k^2\pi^2}\sin\left(\frac{k\pi}{2}\right)\sin\left(\frac{k\pi(x+1)}{2}\right) = (1-x).$$

Repeating the argument with $x \in [-1,0]$ yields the result. □

One important property in the theory of renewal processes is that of direct Riemann integrability of a function. A function function $H(\cdot)$ is said to be *directly Riemann integrable* over $[0,\infty)$ if for any $h > 0$, the normalized sums

$$h\sum_{n=1}^\infty \inf_{0\le\delta\le h} H(nh-\delta) \qquad \text{and} \qquad h\sum_{n=1}^\infty \sup_{0\le\delta\le h} H(nh-\delta),$$

converge to a common finite limit as $h \downarrow 0$ (see chapter 4.4 in [2]).

 ISBN 13: 978-84-613-5510-5

Lemma 3. *The function $p^1(t,x)$ is directly Riemann integrable with respect to t for each $x \in [-1,1]$.*

Proof. We will start by considering the case when $x = 0$. The function $p^1(t,0)$ is directly Riemann integrable if $p^1(t,0)$ is nonegative, monotonically decreasing and Lebesgue integrable (see chapter 4.4 in [2]). Since each term in the representation (3) is nonegative and monotonically decreasing for $x = 0$ so is $p^1(t,0)$, and by Lemma 2 the integral of $p^1(t,0)$ over $[0,\infty]$ is given by $\int_0^\infty p^1(t,0)\mathrm{d}t = 1$ and thus $p^1(t,0)$ is Lebesgue integrable which proves that $p^1(t,0)$ is directly Riemann integrable.

Next let $x \in [-1,1] \setminus \{0\}$. The function $p(t,x)$ is directly Riemann integrable with respect to t if $p^1(t,x) \geq 0$, $p(t,x)$ is uniformly continuous in t and bounded from above by a monotonically decreasing integrable function (see chapter 4.4 in [2]). Since $p(t,x)$ is a probability distribution for each t it is clear that $p(t,x) \geq 0$. To show uniform continuity we will split the interval $[0,\infty)$ into two parts, say $[0,1]$ and $[1,\infty)$, and show that $p^1(t,x)$ is uniformly continuous on each part. For the interval $[0,1]$ we will use the representation (4). Let g_k^G and G_n^G be defined as in the proof of Lemma 1. It is clear that each g_k^G is uniformly continuous in t and thus also G_n^G is uniformly continuous for each $n < \infty$. If we can shown that $G_n^G(t,x)$ for each $x \in [-1,1] \setminus \{0\}$ converges uniformly with respect to t over $[0,1]$ as $n \uparrow \infty$, then also the limit $p(t,x)$ will be uniformly continuous. Rewrite G_n^G as $G_n^G(t,x) = \sum_{k=0}^n \tilde{g}_k^G(t,x)$ where $\tilde{g}_0^G(t,x) = g_0^G(t,x) - g_2^G(t,x)$ and

$$\tilde{g}_k^G(t,x) = g_{4k}^G(t,x) - g_{2-4k}^G(t,x) + g_{-4k}^G(t,x) - g_{2+4k}^G(t,x), \text{ for } k \geq 1.$$

According to Weierstrass M-test, if there is a series of constants M_k such that $\sum_{k=0}^\infty M_k$ is convergent and $|\tilde{g}_k^G(t,x)| \leq M_k$ for all $t \in [0,1]$ then G_n^G converges uniformly in $[0,1]$ as $n \uparrow \infty$. The functions $g_k(t,x)$ attains its maximum at $t = (x-k)^2/\sigma^2 \wedge 1$ for $t \in [0,1]$, and thus $g_k(t,x) \leq g_k((x-k)^2/\sigma^2 \wedge 1, x)$. The function $g_0(x^2/\sigma^2 \wedge 1, x)$ is bounded and it is easily seen that the functions g_k^G may be bounded by $C/(1+k^2)$, for some bounded constant C, and which is clearly convergent. Hence, for each $x \in [-1,1] \setminus \{0\}$, $p(\cdot,x)$ is uniformly continuous in $[0,1]$. To show uniform continuity in $[1,\infty)$ we will use the representation (3). Let $t \geq 1$, then

$$|p^1(t+\delta,x) - p^1(t,x)| \leq \sum_{k=1}^\infty e^{-\frac{k^2\sigma^2\pi^2}{8}t} |e^{-\frac{k^2\sigma^2\pi^2}{8}\delta} - 1| \leq \sum_{k=1}^\infty \frac{8^2}{k^4\sigma^4\pi^4} \frac{k^2\sigma^2\pi^2}{8}\delta = \delta\frac{3}{4\sigma^2}$$

where we used the inequalites $e^{-y} \leq y^{-2}$ and $|e^{-y}-1| \leq y$ which holds for $y \geq 0$. Hence for every $\epsilon > 0$ we may chose δ such that $\delta < 4\sigma^2\epsilon/3$ which holds for every t in $[1,\infty)$. Hence $p^1(\cdot,x)$ is also uniformly continuous in $[1,\infty)$, which together with the previous result yields that $p(\cdot,x)$ is uniformly continuous in $[0,\infty)$. In the proof of Lemma 1 we showed that $p(t,x) \leq p(t,0)$, and that $p(t,0)$ is a monotonically decreasing Lebesgue integrable function. Hence, $p(t,x)$ is also directly Riemann integrable with respect to t for $x \in [-1,1] \setminus \{0\}$, which together with the first result of this proof yeilds that $p(t,x)$ is directly Riemann integrable for $x \in [-1,1]$. □

 ISBN 13: 978-84-613-5510-5

The next two lemmas regards properties of the random variable τ^η defined earlier in this section. Let F_{τ^η} denote the distribution function of τ^η. Lemma 5 states that that τ^η has a density, which we will denote by f_{τ^η}.

Lemma 4. *The expectation of τ^η is given by $E[\tau^\eta] = \eta^2/\sigma^2$.*

Proof. Let $g(x_0) = E[\tau^\eta]$, where x_0 denotes the initial point of the process. The function g satisfies the following ordinary differential equation (see [1])

$$\frac{\sigma^2}{2}\frac{d^2g}{dx_0^2}(x_0) = -1, \quad m_1(-\eta) = m_1(\eta) = 0\,.$$

The solution to this problem, with $x_0 = 0$, is given by $g(0) = \eta^2/\sigma^2$, as was to be shown. □

Lemma 5. *The random variable τ^η has a density, denoted by f_{τ^η}, that may be represented as*

$$f_{\tau^\eta}(t) = \sum_{k=-\infty}^{\infty} \frac{1}{2t\sqrt{2\pi\sigma^2 t}}\Bigg((\eta+4k\eta)e^{-\frac{(\eta+4k\eta)^2}{2\sigma^2 t}} - (\eta+2-4k\eta)e^{-\frac{(\eta+2-4k\eta)^2}{2\sigma^2 t}} + (\eta-4k\eta)e^{-\frac{(\eta-4k\eta)^2}{2\sigma^2 t}} - (\eta-2+4k\eta)e^{-\frac{(\eta-2+4k\eta)^2}{2\sigma^2 t}}\Bigg),\ t \geq 0.$$

Proof. In this proof we will use the representation (4). Let g_k^G and G_n^G be defined as in the proof of Lemma 1. By the use of Lemma 1 for $t \in [0,\infty)$

$$P(\tau^\eta \leq t) = 1 - \sum_{k=-\infty}^{\infty} \int_{-\eta}^{\eta} (g_{4k\eta}^G(t,x) - g_{2-4k\eta}^G(t,x))\mathrm{d}x.$$

If each term in the sum above is differentiable on $[0,\infty)$ and

$$\sum_{k=-\infty}^{\infty} \frac{d}{dt}\int_{-\eta}^{\eta} (g_{4k\eta}^G(t,x) - g_{2-4k\eta}^G(t,x))\mathrm{d}x \tag{5}$$

converges uniformly on $[0,\infty)$ then

$$\frac{d}{dt}P(\tau^\eta \leq t) = -\sum_{k=-\infty}^{\infty} \frac{d}{dt}\int_{-\eta}^{\eta} (g_{4k\eta}^G(t,x) - g_{2-4k\eta}^G(t,x))\mathrm{d}x.$$

Calculating the integral and differentiating with respect to t we get for each term in (5)

$$\begin{aligned}&\frac{d}{dt}\int_{-\eta}^{\eta} (g_{4k\eta}^G(t,x) - g_{2-4k\eta}^G(t,x))\mathrm{d}x\\ &= \frac{1}{2t\sqrt{2\pi\sigma^2 t}}\Bigg((\eta+4k\eta)e^{-\frac{(\eta+4k\eta)^2}{2\sigma^2 t}} - (\eta+2-4k\eta)e^{-\frac{(\eta+2-4k\eta)^2}{2\sigma^2 t}}\\ &\qquad + (\eta-4k\eta)e^{-\frac{(\eta-4k\eta)^2}{2\sigma^2 t}} - (\eta-2+4k\eta)e^{-\frac{(\eta-2+4k\eta)^2}{2\sigma^2 t}}\Bigg),\end{aligned} \tag{6}$$

 ISBN 13: 978-84-613-5510-5

The maximum of the function $\exp\{-\frac{(x-k)^2}{2\sigma^2 t}\}/t^{3/2}$ in $[0,\infty)$ is attained at $t = (x-k)^2/(3\sigma^2)$. For the first term in the expression above we get that

$$\frac{(\eta+4k\eta)}{2t^{3/2}\sqrt{2\pi\sigma^2}}e^{-\frac{(\eta+4k\eta)^2}{2\sigma^2 t}} \leq \left(\frac{3}{2}\right)^{3/2}\frac{\sigma^2 e^{-\frac{3}{2}}}{\eta^2\sqrt{\pi}}\frac{1}{(1+4k)^2},$$

which may be bounded by $C/(1+k^2)$, where C is a bounded constant. In a similar manner it can be shown that the rest of the terms in (6) may also be bounded by $C/(1+k^2)$, and thus

$$\left|\frac{d}{dt}\int_{-\eta}^{\eta}(g^G_{4k\eta}(t,x) - g^G_{2-4k\eta}(t,x))\mathrm{d}x\right| \leq \frac{4C}{1+k^2}. \tag{7}$$

Since $\sum_{k=-\infty}^{\infty} 4C/(1+k^2)$ is a convergent series by Wierstrass M-test the sum (5) converges uniformly on $[0,\infty)$, and hence, the density, f_{τ^η}, may be represented by the sum (6). Since the terms in the sum of (6) could be bounded by $4C/(1+k^2)$ we have that $|f_{\tau^\eta}(t)| \leq 4C\sum_{k=-\infty}^{\infty} 1/(1+k^2) < \infty$ which shows that $f_{\tau^\eta}(t)$ is bounded in $[0,\infty)$. □

1.2 Renewal processes

In this paragraph we will focus on a renewal process denoted by N with idenpendent and identically distributed interarrival times $\{\tau_i\}_{i\geq 1}$. Define the renewal function M by $M_t = E[N_t]$, and let μ denote the mean time between renewals, that is $\mu = E[\tau_i]$, which holds for all $i \geq 1$. Next, we will state the well known key renewal theorem that will be needed later on. For a proof see e.g. [2].

Lemma 6 (Key renewal theorem)**.** *If $H(\cdot)$ is a directly Riemann-integrable function then*

$$\lim_{t\to\infty}\int_0^t H(t-x)dM(x) = \frac{1}{\mu}\int_0^\infty H(x)dx\,.$$

Let F_τ denote the common distribution function of the stochastic variables τ_i. Since the components of $\{\tau_i\}_{i\geq 1}$ are idependent and identically distributed the distribution function of the sum $\sum_{i=1}^k \tau_i$ may be represented by the k-fold convolution of F_τ (here denoted F_τ^{*k}), i.e. $P(\sum_{i=1}^k \tau_i < t) = F_\tau^{*k}(t)$.

Lemma 7 (Theorem 5.4 in [4])**.** *There exists a one-to-one correspondence between F_τ and M, and M has the representation $M_t = \sum_{k=1}^\infty F_\tau^{*k}(t)$ ..*

Under the assumption that F_τ has a density (here denoted f_τ) we have that $f_\tau^{*k}(t) = \frac{d}{dt}F_\tau^{*k}(t)$, where f_τ^{*k} is the k-th convolution of the density function f_τ. We may now define the renewal density m by

$$m_t := \frac{d}{dt}M_t = \sum_{k=1}^\infty f_\tau^{*k}(t)\,. \tag{8}$$

 ISBN 13: 978-84-613-5510-5

2 Main result

In this section we state and prove the main result of this paper. To ease the notation we will let $Z_t^\eta = X_t^\eta - X_{\varphi_t^\eta}^\eta$.

Theorem 1. *Fix a point $t > 0$, then*

$$\frac{1}{\eta}\left(X_t - X_{\varphi_t^\eta}\right) \xrightarrow{d} \Lambda \quad \text{as } \eta \to 0\,,$$

where Λ is a stochastic variable with density function $f_\Lambda(z) = (1-|z|)^+$.

Proof. Denote by $Y_t^\eta(u)$ the quantity $Y_t^\eta(u) = X_t - X_{\varphi_t^\eta}|\{t - \varphi_t^\eta = u\}$. Because of the time homogeneity of the process X the following equality in distribution holds

$$X_t - X_{\varphi_t^\eta}|\{t-\varphi_t^\eta = u\} \stackrel{d}{=} X_u|\{|X_s| < \eta,\, 0 \le s \le u\}\,.$$

Consequently the density function of $Y_t^\eta(u)$ can be expressed as

$$f_{Y_t^\eta(u)}(y) = \frac{p^\eta(u,y)}{P(\tau^\eta > u)}\,,$$

The distribution function of Z_t^η is given by

$$f_{Z_t^\eta}(z) = \int_0^t f_{Y_t^\eta(u)}(z)\mathrm{d}F_{t-\varphi_t^\eta}(u)\,,$$

where

$$\mathrm{d}F_{t-\varphi_t^\eta}(u) = \left\{\delta(u-t)P(\tau^\eta > t) + \sum_{k=1}^\infty \frac{\partial}{\partial u}P(t-\varphi_t^\eta \le u,\, N_t^\eta = k)\right\}\mathrm{d}u\,.$$

The probability in the last term of the above expression can be rewritten as

$$\begin{aligned} P(t-\varphi_t^\eta \le u,\, N_t^\eta = k) &= P\left(t - \sum_{j=1}^k \tau_j^\eta \le u,\, \sum_{j=1}^k \tau_j^\eta < t < \sum_{j=1}^k \tau_j^\eta + \tau_{k+1}^\eta\right) \\ &= P\left(t - \sum_{j=1}^k \tau_j^\eta \le u,\, 0 < t - \sum_{j=1}^k \tau_j^\eta < \tau_{k+1}^\eta\right) = \int_{t-u}^\infty \int_{t-v}^\infty f_{\tau^\eta}^{*k}(v) f_{\tau^\eta}(z)\mathrm{d}z\,\mathrm{d}v\,, \end{aligned}$$

where f_{τ^η} (which exists due to Lemma 5) is the density function of τ^η, and $f_{\tau^\eta}^{*k}$ denotes the k-th convolution of f_{τ^η}. Differentiating the above expression with respect to u yields

$$\frac{\partial}{\partial u}\left(\int_{t-u}^\infty \int_{t-v}^\infty f_{\tau^\eta}^{*k}(v) f_{\tau^\eta}(z) dz\, dv\right) = \int_u^\infty f_{\tau^\eta}^{*k}(t-u) f_{\tau^\eta}(z)\mathrm{d}z = f_{\tau^\eta}^{*k}(t-u)P(\tau^\eta > u)\,.$$

 ISBN 13: 978-84-613-5510-5

This gives us that

$$\mathrm{d}F_{t-\varphi_t^\eta}(u) = \left\{ \delta(u-t)P(\tau^\eta > t) + \sum_{k=1}^{\infty} f_{\tau^\eta}^{*k}(t-u)P(\tau^\eta > u) \right\} \mathrm{d}u\,.$$

Using the scaling property of the Brownian motion the following two relations are easily deduced $P(\tau^\eta > t) = P(\tau^1 > t/\eta^2)$ and $Y_t^\eta(u)/\eta \stackrel{d}{=} Y_t^1(u/\eta^2)$. The first of these two relations yields

$$f_{\tau^\eta}(t) = -\frac{\mathrm{d}}{\mathrm{d}t}P(\tau^\eta > t) = -\frac{\mathrm{d}}{\mathrm{d}t}P(\tau^1 > t/\eta^2) = \frac{1}{\eta^2} f_{\tau^1}(t/\eta^2)\,,$$

and consequently

$$\mathrm{d}F_{t-\varphi_t^\eta}(u) = \left\{ \delta(u-t)P(\tau^1 > t/\eta^2) + \sum_{k=1}^{\infty} \frac{1}{\eta^2} f_{\tau^1}^{*k}\left(\frac{t-u}{\eta^2}\right) P(\tau^1 > u/\eta^2) \right\} \mathrm{d}u\,.$$

The relation $Y_t^\eta(u)/\eta \stackrel{d}{=} Y_t^1(u/\eta^2)$ yields

$$\int_0^t f_{Y_t^\eta(u)/\eta}(y)\mathrm{d}F_{t-\varphi_t^\eta}(u) = \int_0^t f_{Y_t^1(u/\eta^2)}(y)\mathrm{d}F_{t-\varphi_t^\eta}(u)\,,$$

and thus

$$\begin{aligned} f_{Z_t^\eta/\eta}(z) &= \int_0^t \frac{p^1(u/\eta^2, z)}{P(\tau^1 > u/\eta^2)} \delta(u-t)P(\tau^1 > t/\eta^2)\mathrm{d}u \\ &\quad + \int_0^t \frac{p^1(u/\eta^2, z)}{P(\tau^1 > u/\eta^2)} \sum_{k=1}^{\infty} \frac{1}{\eta^2} f_{\tau^1}^{*k}\left(\frac{t-u}{\eta^2}\right) P(\tau^1 > u/\eta^2)\mathrm{d}u \\ &= p^1(t/\eta^2, z) + \int_0^t p^1(u/\eta^2, z) \sum_{k=1}^{\infty} \frac{1}{\eta^2} f_{\tau^1}^{*k}\left(\frac{t-u}{\eta^2}\right) \mathrm{d}u\,. \end{aligned}$$

Now, by a change of variables ($v = (t-u)/\eta^2$)

$$f_{Z_t^\eta/\eta}(z) = p^1(t/\eta^2, z) + \int_0^{t/\eta^2} p^1\left(\frac{t}{\eta^2} - v, z\right) \sum_{k=1}^{\infty} f_{\tau^1}^{*k}(v)\,\mathrm{d}v\,.$$

Since $|p^1(t/\eta^2, x)| \le \sum_{k=1}^{\infty} \frac{8\eta^2}{k^2\sigma^2\pi^2} = \frac{4\eta^2}{3\sigma^2}$ we have that $\lim_{\eta\to 0} p^\eta(y, t/\eta^2) = 0$. For the second term we have using (8), Lemma 4 and Lemma 6

$$\lim_{\eta\to 0} \int_0^{t/\eta^2} p^1\left(y, \frac{t}{\eta^2} - v\right) \sum_{k=1}^{\infty} f_{\tau^1}^{*k}(v)\,dv = \sigma^2 \int_0^{\infty} p^1(y, u)\mathrm{d}u\,.$$

Now by Lemma 2 $\lim_{\eta\to 0} f_{Z_t^\eta/\eta}(z) = (1-|z|)^+$, as was to be shown. □

Remark 1. Note that the limiting distribution does not depend on σ. This is unlike the case when discretization takes place on an equidistant grid, where σ affects the variance of the limiting distribution. Instead, in the case of adaptive approximation, σ is related to the expected number of discretization points.

 ISBN 13: 978-84-613-5510-5

3 Numerical results

In this section the transition of $f_{Z_t^\eta/\eta}$ as η goes from some large value towards zero is investigated. We will argue that for large values of η the stochastic variable Z_t^η/η is approximately normally distributed, and thus as η approaches zero we will see that $f_{Z_t^\eta/\eta}$ goes from the density of a normally distributed random variable to the density of a triangularly distributed random variable.

A total of 50000 trajectories of the process X was simulated, over a period from $t = 0$ to $t = 0.5$, with $\sigma = 1$, on a time grid with 200001 equally spaced points. Trajectories of the approximation $X_{\varphi_t^\eta}$ were calculated for a number of different values of η in the range $[0.5, 4.0]$.

Recall, from the proof of Theorem 1, the expression of the density

$$f_{Z_t^\eta/\eta}(z) = p^1(t/\eta^2, z) + \int_0^{t/\eta^2} p^1\left(\frac{t}{\eta^2} - v, z\right) \sum_{k=1}^{\infty} f_{\tau^1}^{*k}(v)\, dv\,. \tag{9}$$

It is clear that for large values of η it is the first term in (9) that is the dominant one. Thus, in this case the density is approximately the same as the absorbed Wiener process. Furthermore, since η was assumed to be large the density of the absorbed Wiener process is approximately the same as the Wiener process without absorbing barriers. Hence, for large η we have that

$$f_{Z_t^\eta/\eta}(z) \approx \frac{\eta}{\sigma\sqrt{t}}\phi\left(z\frac{\eta}{\sigma\sqrt{t}}\right)\,, \tag{10}$$

where ϕ denotes the standard normal density function.

In Figure 1 the density of $f_{Z_t^\eta/\eta}$, at $t = 0.5$, as we let η go from 4.0 to 0.5 is depicted. It is seen that when $\eta = 4.0$ the distribution is quite close to the normal distribution. For $\eta = 0.5$ the distribution on the other hand is quite close to the triangular distribution.

To further illustrate the transition from the normal distribution to the triangular distribution we measured the distance in therms of the Wasserstein metric between the, from the Monte Carlo simulation, estimated distribution and these two distributions. The distance between two distributions, with distribution functions F and G, in terms of the Wasserstein metric is defined by $d_W(F, G) = \int_{\mathbb{R}} |F(x) - G(x)|dx$.

In Figure 2 the Wasserstein distance between the empirical distribution and the triangular distribution as well as the distance between the empirical distribution and the normal distribution (10), at $t = 0.5$, as a function of η is depicted. Note that in the case of the normal distribution (10) not only the empirical distribution but also the normal distribution that we compare with is dependent of η. It is seen that for η smaller than 1.25 the empirical distribution is relatively close to the triangular distribution whereas for values over 2.25 it is close to the normal distribution (10). For η in the interval $(1.25, 2.25)$ the distribution is probably better explained by a mixture of the two distributions. The small offset from zero for small values of the distance is due to the variance of the monte carlo simulation.

 ISBN 13: 978-84-613-5510-5

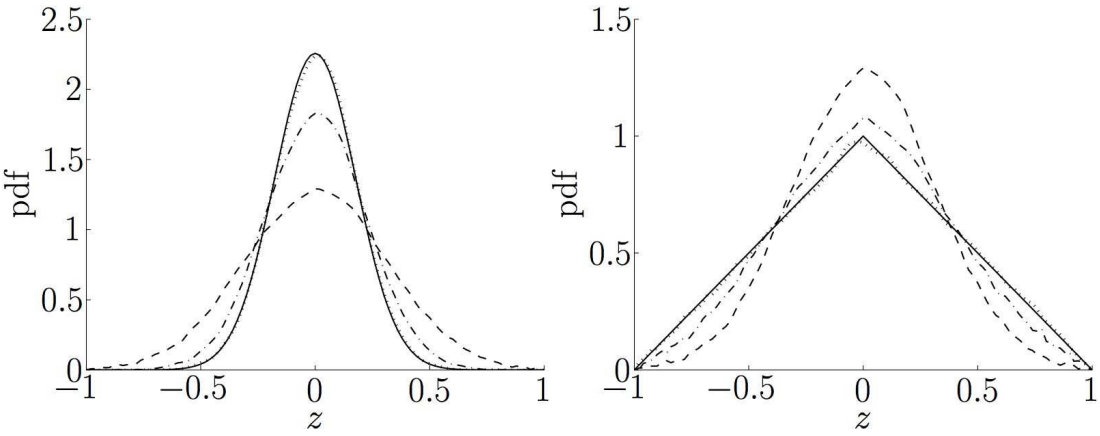


Figure 1: Left (large values of η): kernel estimates of $f_{Z_t^\eta/\eta}(z)$ where $\eta = 4.0$ (dotted), $\eta = 3.25$ (dash-dotted) and $\eta = 2.5$ (dashed), and the Gaussian distribution (solid). Right (small values of η): kernel estimates of $f_{Z_t^\eta/\eta}(z)$ where $\eta = 2.5$ (dashed), $\eta = 2.0$ (dash-dotted) and $\eta = 0.5$ (dotted), and the triangular distribution (solid).

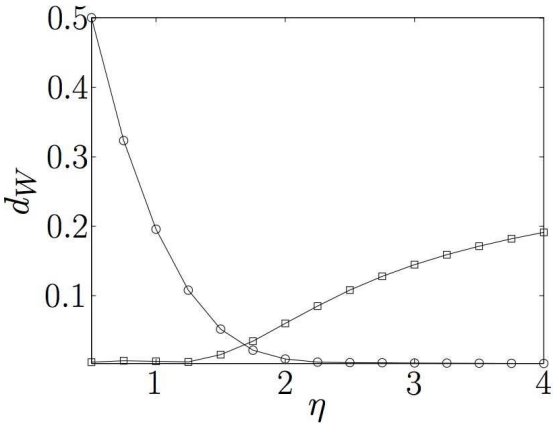


Figure 2: Distance in terms of the Wasserstein metric between the triangular distribution and the empirical distribution (squares), and the normal distribution (10) and the empirical distribution (circles).

 ISBN 13: 978-84-613-5510-5

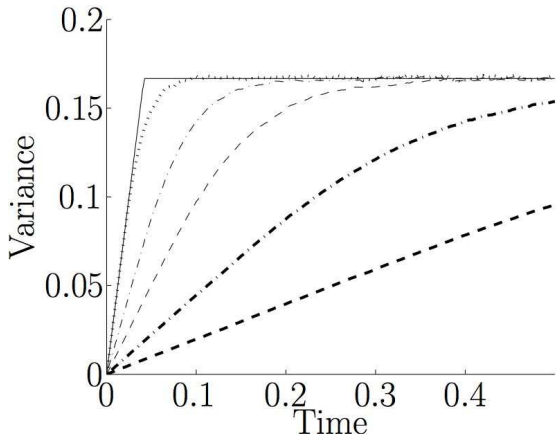


Figure 3: The variance of Z_t^η/η as a function of time where $\eta = 0.50$ (dotted), $\eta = 0.75$ (thin dash-dotted), $\eta = 1.00$ (thin dashed), $\eta = 1.50$ (thick dash-dotted) and $\eta = 2.25$ (thick dashed), together with the function $(t/0.5^2) \wedge (1/6)$ (solid).

From (9) it is clear that it is possible to fix η and instead of letting η approach zero let t approach infinity. To capture this we have plotted the variance of Z_t^η/η as a function of t for a couple of different values of η (see Figure 3). The constant $1/6$, that is the value of the variance of the triangularly distributed random variable, is also plotted in the figure. As expected it is seen that for low values of η the limiting variance of $1/6$ is attained much faster than for higher values of η. From the argumentation above regarding high values of η it is also clear that for low values of t the distribution is approximately normal. Hence, the slope of the lines near zero is given by $1/\eta^2$, as is seen in the figure.

References

[1] D.R. Cox and H.D. Miller. *The Theory of Stochastic Processes.* Methuen and CO LTD, 1965.

[2] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes.* Springer Series in Statistics. Springer-Verlag, New York, 1988.

[3] C. Geiss and S. Geiss. On an approximation problem for stochastic integrals where random time nets do not help. *Stochastic Processes and their applications*, 116(3): 407–422, 2006.

[4] Daniel P. Heyman and Matthew J. Sobel. *Stochastic models in operations research.*, volume I. McGraw-Hill., 1982.

[5] G. N. Milstein and M. V. Tretyakov. Simulation of a space-time bounded diffusion. *Ann. Appl. Probab.*, 9(3):732–779, 1999.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Combinatorial Optimization of Stencil-based Jacobian Computations

H. Martin Bücker[1] and Michael Lülfesmann[1]

[1] *Institute for Scientific Computing (SC), Center for Computational Engineering Science (CCES), RWTH Aachen University, 52056 Aachen, Germany*

emails: buecker@sc.rwth-aachen.de, luelfesmann@sc.rwth-aachen.de

Abstract

The computation of all nonzero entries of a sparse Jacobian matrix using either divided differencing or the forward mode of automatic differentiation is considered. Throughout this article, we assume that the sparsity of the Jacobian stems from a stencil-based computation of the underlying function which is typical for numerical applications in computational science and engineering involving partial differential equations. The minimization of the time needed to compute all nonzero Jacobian entries is formulated as a combinatorial optimization problem. We present three different, yet equivalent, representations of that problem and discuss each of its advantages and disadvantages. Broadly speaking, the three representations belong to the areas of linear algebra, grid discretization, and graph theory.

Key words: sparsity, derivative computation, graph coloring, partial differential equations, discretization

MSC 2000: 05C15, 05C50, 90C27, 65D25, 65F50, 68N19

1 Introduction

Various algorithms for the solution of problems arising from scientific computing require the evaluation of derivatives of some underlying function. Prominent examples include Newton-type algorithms for the solution of nonlinear systems and continuous optimization problems. It is not uncommon that the underlying mathematical functions are given in the form of computer programs rather than as formulæ. In practice, real-world problems from science or engineering give rise to complicated programs written in C/C++, Fortran, MATLAB or any other high-level programming language. In realistic applications from computational science and engineering, the evaluation of a function f corresponds to the execution of a corresponding program implementing f that requires substantial amount of computing time. The derivatives of f are either approximated by divided differencing involving truncation error or computed exactly, i.e.,

 ISBN 13: 978-84-613-5510-5

without truncation error, using automatic differentiation. In both cases, the computing time to evaluate the Jacobian matrix of f is a multiple of the time to evaluate f. For derivative computations in large-scale problems, it is therefore crucial to exploit any available structure of the problem at hand to reduce time and/or storage requirement.

If the Jacobian is sparse it is well-known that a given sparsity pattern can be exploited to reduce the time to evaluate the Jacobian [11]. In this article, we focus on the special case where the function

$$f : \mathbb{R}^{MN} \longrightarrow \mathbb{R}^{MN} \tag{1}$$

is computed by a stencil operation on a regular $M \times N$ grid. That is, the value of a quantity on a grid point is updated by the weighted values of the quantity on neighboring grid points. We consider only neighbors in space rather than in time. The neighborship relation for a grid point (m, n) is defined by the stencil $\mathcal{N}(m, n)$, the set of all neighboring grid points whose values influence the new value at (m, n). We assume that the update of a grid point involves its old value so that $(m, n) \in \mathcal{N}(m, n)$. The grid point (m, n) is called the *center* of the stencil $\mathcal{N}(m, n)$.

Since stencil operations perform global sweeps over a possibly large data structure that exceeds the capacity of available data cache, they typically achieve only a low fraction of the theoretical peak performance on today's processors. It is therefore no surprise that researchers extensively studied the reorganization of stencil operations in an attempt to better exploit deep memory hierarchies. These performance optimizations are based on improving locality in both space and time; see [9] and the references therein. Rather than considering any such performance optimization, the focus of this paper is on combinatorial optimization problems arising from computing the Jacobian of some stencil-based function of the form (1). Because of the spatial locality of a stencil, the Jacobian of such a function is sparse. In general, the exploitation of sparsity in derivative computations leads to a rich set of hard combinatorial optimization problems [11]. The combinatorial problem considered in the present article consists of minimizing the number of function evaluations in divided differencing or, equivalently, the number of automatic differentiation passes for the computation of all nonzero elements of a sparse Jacobian of an underlying function that is based on stencil computations. This combinatorial problem is NP-complete for general nonzero patterns [6] and can be formulated using different representations. The new contribution of this article is to bring together three different representations for stencil-based Jacobian computations where, in contrast to general nonzero patterns, explicit solutions for various special stencils are known [12, 17].

The organization of this article is as follows. In Section 2, we give an outline of the combinatorial optimization problem using the language of linear algebra. In particular, matrix-vector and matrix-matrix multiplications are crucial in this context. In Section 3, the same combinatorial problem is described in terms of the underlying grid where the spatial neighborhood is extensively addressed. Another contribution of this paper is presented in Section 4 where a third representation of the combinatorial problem based on a suitable graph model is sketched for stencil-based Jacobian computations. A discussion of the three different representations is given in Section 5.

 ISBN 13: 978-84-613-5510-5

2 Linear Algebra Representation

To describe the combinatorial problem associated with stencil-based Jacobian computations in terms of matrices and vectors, we first introduce an ordering of the grid points to map a two-dimensional index of a grid point into a one-dimensional index. To this end, let $\psi(m,n)$ denote the one-dimensional index used for the grid point (m,n) in such a numbering scheme. Given a regular $M \times N$ grid and a stencil $\mathcal{N}(m,n)$ describing the neighborhood relationship for all grid points $1 \leq m \leq M$ and $1 \leq n \leq N$, the sparsity pattern of the $MN \times MN$ Jacobian $A = (a_{ij})$ of the function f defined in (1) is determined and given as follows. A nonzero Jacobian entry is characterized by

$$a_{ij} \neq 0 \quad \Longleftrightarrow \quad i = \psi(m,n), j = \psi(k,l), \text{ and } (k,l) \in \mathcal{N}(m,n).$$

An example of a nonzero pattern is illustrated in Fig. 1. Here, we consider a 3×3 grid and a five-point stencil defined by

$$\mathcal{N}_{5\text{pt}}(m,n) = \{(m+1,n),(m-1,n),(m,n),(m,n+1),(m,n-1)\} \tag{2}$$

for any grid point (m,n) that is not located on the boundary of the grid. That is, the neighbors of a grid point are immediately adjacent in the north, south, west, and east directions. Grid points on the boundary have less neighbors. We assume a natural ordering where the grid points are numbered starting from left to right and from bottom to top:

$$\psi(1,1) = 1, \quad \psi(1,2) = 2, \quad \psi(1,3) = 3, \quad \psi(2,1) = 4, \quad \ldots$$

As depicted in Fig. 1, the only non-boundary grid point $(2,2)$ of this small grid induces five nonzero entries, denoted by crosses, in row/column $\psi(2,2) = 5$ of the nonzero pattern.

The derivative of a vector-valued function f with respect to some vector x into the direction of a vector s is defined by

$$\frac{\partial f}{\partial x} s = \lim_{h \to 0} \frac{f(x+hs) - f(x)}{h}.$$

Let $A := \partial f / \partial x$ denote the Jacobian whose columns are given by

$$A = [a_1 a_2 \cdots a_{MN}].$$

Then, by choosing $s \in \{0,1\}^{MN}$ as a binary vector, any sum of columns a_j can be computed where the jth entry of s is nonzero, i.e.,

$$As = \sum_{j \text{ with } s_j = 1} a_j.$$

Moreover, the product of the Jacobian A and some $MN \times p$ matrix S can be approximated by $p+1$ evaluations of the function f using divided differencing. Similarly, the forward mode of automatic differentiation is capable of computing that product, $A \cdot S$,

 ISBN 13: 978-84-613-5510-5

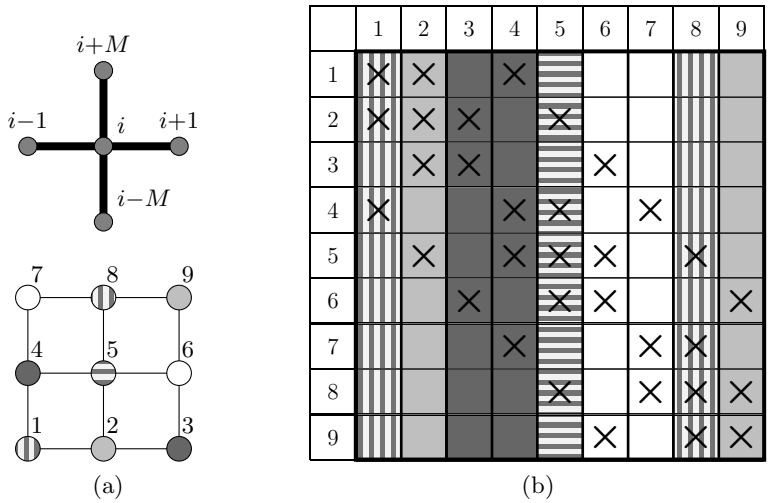


Figure 1: (a) Five-point stencil (top) and regular 3×3 grid (bottom). (b) Nonzero pattern of Jacobian matrix resulting from a five-point stencil using a natural ordering of grid points on a 3×3 grid. Background padding indicates $p = 5$ groups of structurally orthogonal columns.

without truncation error using $p+1$ times the time needed to evaluate f. Therefore, p indicates a rough measure of the time needed to compute the Jacobian.

The idea to reduce p—and hence the time to compute all nonzero entries of a sparse Jacobian—consists of partitioning the columns of the Jacobian into groups of those columns whose sum contains all the nonzero elements of the columns in that group [8]. The property characterizing such a group is introduced in the following definition.

Definition 1. Two columns a_j and a_k are *structurally orthogonal* if and only if they do not have any nonzero element in the same row, i.e.,

$$a_j \perp a_k \quad :\Longleftrightarrow \quad \nexists i : \; a_{i,j} \neq 0 \wedge a_{i,k} \neq 0.$$

In the example given in Fig. 1, the columns a_3 and a_4 are structurally orthogonal since there is no row in which both columns have a nonzero element. So, the sum $a_3 + a_4$ contains all nonzero elements of these two columns. The combinatorial optimization problem is now formulated as follows.

Problem 1. *Given a Jacobian matrix A, partition its columns into a minimal number of groups of structurally orthogonal columns. More precisely, find a binary $MN \times p$ matrix S such that all nonzero elements of A are contained in the matrix-matrix product $A \cdot S$ and the number of columns p, representing the number of groups, is minimized.*

 ISBN 13: 978-84-613-5510-5

There is a solution to that problem. However, the solution may not be unique. For the example illustrated in Fig. 1, a solution is given by

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Any solution satisfies $p = 5$ since, from inspection of the fifth row which involves 5 nonzero elements, there is no matrix S with $p < 5$.

3 Grid Representation

Rather than considering the linear algebra representation we now focus on the underlying regular $M \times N$ grid and the stencil $\mathcal{N}(m,n)$. The combinatorial problem can then be reformulated in terms of the underlying grid. Since a grid point corresponds to a row/column of the Jacobian matrix we arrive at the following definition that characterizes the property needed to partition the grid points into groups.

Definition 2. Two grid points (i,j) and (k,l) are *structurally orthogonal* if and only if their stencils do not overlap, i.e.,

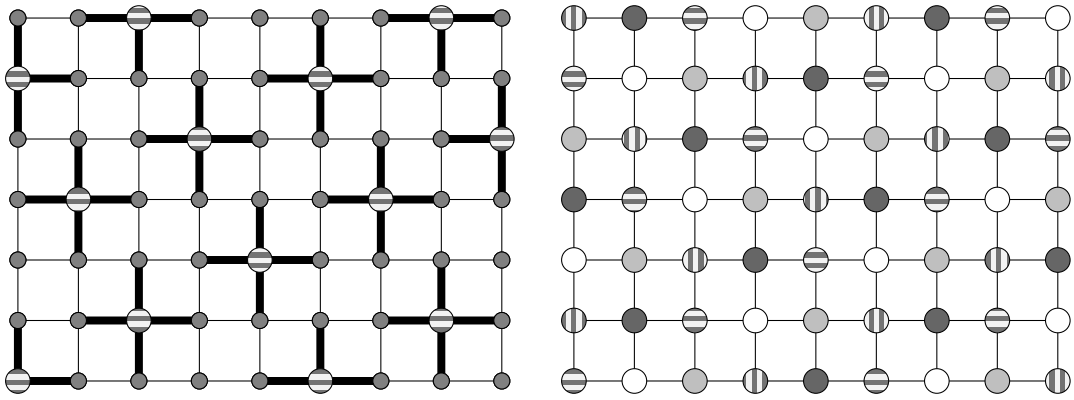
$$\begin{aligned} (i,j) \perp (k,l) \quad &:\Longleftrightarrow \quad \nexists (m,n) : \ (i,j) \in \mathcal{N}(m,n) \ \wedge \ (k,l) \in \mathcal{N}(m,n) \\ &\Longleftrightarrow \quad \mathcal{N}(i,j) \cap \mathcal{N}(k,l) = \emptyset. \end{aligned}$$

To illustrate this definition, we resume the example of the five-point stencil given in the previous section but vary the grid size. The centers of all stencils depicted in Fig. 2(a) are structurally orthogonal. A group of structurally orthogonal center grid points is called a *cover*. In general, there are grid points that are not structurally orthogonal so that multiple covers are needed to contain all grid points. Therefore, the corresponding combinatorial optimization problem is given as follows.

Problem 2. *Given a grid, partition its grid points into a minimal number of groups of structurally orthogonal grid points. More precisely, find a sequence of covers containing all grid points such that stencils within a cover do not overlap and the number of covers p, representing the number of groups, is minimized.*

To reduce the number of covers it is reasonable to construct "compact" covers, meaning that the non-overlapping stencils are placed close to each other. In this sense, the cover depicted in Fig. 2(a) is compact since any placement of stencils attempting to reduce the distance between two stencils would violate the structural orthogonality of its

 ISBN 13: 978-84-613-5510-5



(a) (b)

Figure 2: (a) Cover corresponding to a group of structurally orthogonal center points for the five-point stencil. (b) Sequence of covers obtained from using $p = 5$ covers of the form given in (a).

centers. In Fig. 2(b), a solution is shown that was constructed by taking $p = 5$ covers of the form shown in Fig. 2(a) and arranging them so as to contain all grid points. The different covers are depicted in that figure using different background padding. Since the stencil involves five grid points, there is no solution with $p < 5$. Hence, the sequence of covers shown in Fig. 2(b) is indeed a solution to the combinatorial optimization problem. For this five-point stencil, the open literature [12, 17] gives the explicit formula to construct the sequence of covers given in Fig. 2(b).

Goldfarb and Toint [12] also present solutions for various other stencils showing that, in general, the solution does not consist of a sequence of identical covers. For instance, the sequence of covers used for a nine-point stencil is constructed using different covers. More recently, grids with periodic boundary conditions are analyzed [7].

4 Bipartite Graph Representation

Coleman and Moré [6] were the first authors who modeled the computation of sparse Jacobians by graphs. In particular, they introduced the column intersection graph. Since then, various graph models were used to describe different sparsity-related derivative computations [3, 5, 13–15]. In the present article, we follow the approach taken in [11] where a bipartite graph $G(V_r, V_c, E)$ is introduced. To every grid point, a row and a column vertex is associated leading to the vertex sets

$$V_r = \{r_i \mid 1 \leq i \leq MN\} \quad \text{and} \quad V_c = \{c_j \mid 1 \leq j \leq MN\}.$$

There is an edge $(r_i, c_j) \in E$ if there is a stencil to which the grid points represented by r_i and c_j belong. This graph model is illustrated using the five-point stencil again. In Fig. 3(a), the subgraph representing a single stencil is depicted. There is an edge from the row vertex r_i corresponding to the center of the stencil to the column vertices c_{i-M},

 ISBN 13: 978-84-613-5510-5

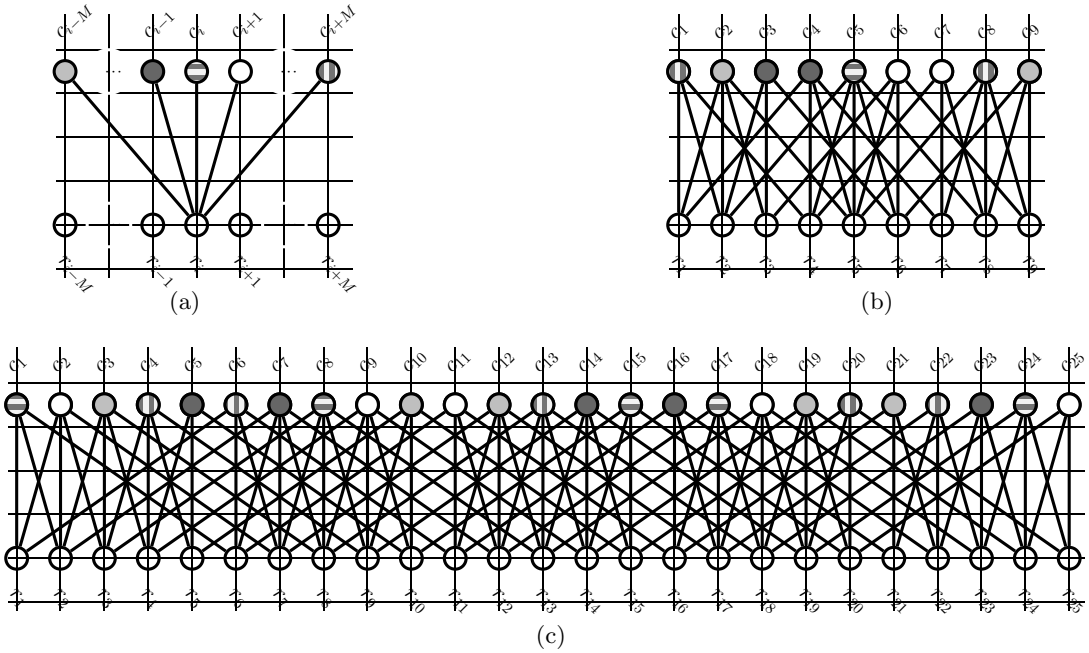


Figure 3: (a) Bipartite subgraph representing a five-point stencil. (b) Bipartite graph corresponding to a 3×3 grid with a coloring representing $p = 5$ groups of structurally orthogonal column vertices. (c) Corresponding graph for a 5×5 grid.

c_{i-1}, c_i, c_{i+1}, and c_{i+M} that belong to the stencil. The bipartite graph representing a regular 3×3 grid is shown in Fig. 3(b). The graph depicted in Fig. 3(c) corresponds to a larger grid displaying the structure of the graph more clearly. In particular, the graph given in Fig. 3(b) contains only a single subgraph of the form displayed in Fig. 3(a) whereas the larger graph given in Fig. 3(c) contains nine of them.

The following definition is used to partition the column vertices into different groups.

Definition 3. Two column vertices c_i and c_j are *structurally orthogonal* if and only if they are not connected by a path of length two, i.e.,

$$c_i \perp c_j \quad :\Longleftrightarrow \quad \nexists r_k \in V_r : \, (r_k, c_i) \in E \, \wedge \, (r_k, c_j) \in E.$$

An illustration of this definition is given in Fig. 3(b) and Fig. 3(c) where groups of structurally orthogonal vertices are indicated using the same color. The combinatorial optimization problem in terms of the bipartite graph is then as follows.

Problem 3. *Given a bipartite graph $G = (V_r, V_c, E)$, partition its column vertices into a minimal number of groups of structurally orthogonal vertices. More precisely, find a coloring of V_c such that all vertices connected by paths of length 2 are colored differently and the number of colors p, representing the number of groups, is minimized.*

 ISBN 13: 978-84-613-5510-5

Recall from the previous sections that a solution of this problem for the five-point stencil satisfies $p = 5$. In Fig. 3(b) and Fig. 3(c), a solution is given with $p = 5$ colors. There is no solution with $p < 5$ because the subgraph corresponding to a stencil consists of 5 column vertices that are pairwise not structurally orthogonal.

5 Discussion

A few comments on the three equivalent representations to describe the combinatorial optimization problem for stencil-based Jacobian computations are in order. The linear algebra representation is the one closest to the algorithms that make use of the Jacobian values. Consider, for instance, Newton's method for the solution of a nonlinear system. Here, the algorithm involves the subproblem of solving a system of linear equations whose coefficient matrix is given by the Jacobian of the underlying function. From that perspective, the linear algebra representation of the combinatorial problem is therefore intimately connected to the linear algebra view of the Jacobian in Newton's method. However, the linear algebra representation is inherently based on some ordering of the grid points. A different ordering corresponds to a permutation of the rows and columns of the Jacobian. Although the minimal number of groups of structurally orthogonal columns is invariant under such permutations, it could be difficult to expose the problem structure using an ordering that is not carefully chosen. Moreover, the ordering may influence the number of groups of structurally orthogonal columns that is computed by some heuristic.

The advantage of the grid representation is that it offers a clear view on the origin of the problem including its structure. This could be important for efficient data handling trying to capture the data locality available in the implementation of the underlying function [9]. A disadvantage of the grid representation is that it tends to be more difficult to address stencils in three and more space dimensions. In applications with a high number of space dimensions, it is less intuitive to analyze structural orthogonality for stencils. In contrast, the linear algebra and the bipartite graph model both handle higher spatial dimensions using a one-dimensional representation of all spatial dimensions which reduces the human effort for the analysis of structural orthogonality.

The bipartite graph representation offers an abstraction on a high level. It also provides a unified scheme to describe all sorts of different coloring problems associated with derivative computations [11]. Another advantage is the availability of a rich number of related results in the matured field of graph theory. For instance, preordering techniques used in various areas of combinatorial scientific computing can also be used for graph coloring [1, 16, 18]. In the graph representation, the sparsity is directly available and does not need to be encoded in some data structure. In contrast, an implementation of the linear algebra representation is based on some sparsity-exploiting matrix data structure geared toward efficient numerical computations on that matrix but may lead to less efficient data accesses when carrying out algorithms involving neighborship relations. Moreover, software tools implementing graph coloring heuristics are available [2–5, 10].

 ISBN 13: 978-84-613-5510-5

Stencil	Grid Size	p	CPR			
			NO	LFO	SLO	IDO
$\mathcal{N}_{5\text{pt}}(m,n)$	50×50	5	7	7	7	6
	100×100	5	7	7	7	6
	200×200	5	7	7	7	6
$\mathcal{N}_{9\text{pt}}(m,n)$	50×50	10	16	17	14	14
	100×100	10	17	17	14	14
	200×200	10	17	17	14	14

Table 1: Number of groups of structurally orthogonal grid points for five-point and nine-point stencil and different grid sizes.

To illustrate the discussion, we consider again the five-point stencil $\mathcal{N}_{5\text{pt}}(m,n)$ from (2) and, in addition, the nine-point stencil defined by

$$\mathcal{N}_{9\text{pt}}(m,n) = \mathcal{N}_{5\text{pt}}(m,n) \cup \{(m+2,n),(m-2,n),(m,n+2),(m,n-2)\}.$$

In [12], an assignment of every grid point to a group of structurally orthogonal grid points is derived for both stencils. The explicit formulæ given therein present a solution of the combinatorial optimization problem with a minimal number of groups. This minimal number is $p = 5$ for $\mathcal{N}_{5\text{pt}}(m,n)$ and $p = 10$ for $\mathcal{N}_{9\text{pt}}(m,n)$. We compare these numbers with the corresponding values obtained from applying the standard CPR heuristic [8]. This greedy heuristic is designed to solve the general NP-complete problem. The results of the comparison applied to a set of instances varying the grid size is summarized in Table 1. In that table, the minimal number of groups p is given in the third column. The fourth column states the number of groups computed by the CPR heuristic using the natural ordering (NO). The following three columns are obtained by CPR with three different preordering algorithms: Largest First Ordering (LFO) [18], Smallest Last Ordering (SLO) [16], and Incident Degree Ordering (IDO) [1]. For $\mathcal{N}_{5\text{pt}}(m,n)$, the number of groups resulting from any CPR heuristics is no larger than $p+2$. The lowest number of groups is computed by IDO. The difference between the optimal number of groups p and the ones computed by heuristics is larger for $\mathcal{N}_{9\text{pt}}(m,n)$. Here, the heuristics compute 14, 16 or 17 groups rather than $p = 10$. In summary, the results indicate that there is an influence of the ordering on the number of groups. This is consistent with the observations of other authors who studied this effect for general nonzero patterns [1, 6] and shows the potential of using graph-theoretical elements in that context.

6 Concluding Remarks

The evaluation of derivatives of given functions is important for various techniques in computational science and engineering. We address the problem of combinatorially optimizing the number of function evaluations to approximate a Jacobian matrix with a given sparsity pattern using divided differencing. In automatic differentiation where

 ISBN 13: 978-84-613-5510-5

the derivatives are computed exactly, the same combinatorial optimization problem occurs in the forward mode. Throughout this article we focus on the Jacobian of a function defined on a regular grid using a stencil operation.

The main contribution of the present article is to present the combinatorial optimization problem using a consistent description of three different representations: linear algebra, grid discretization, and graph theory. We also compare these representations with their advantages and disadvantages. This collection of representations is beneficial for researchers who are familiar with the grid representation of their underlying mathematical function and want to explore the potential of exploiting the sparsity in derivative computations using a different representation. It is also interesting for educational purposes when teaching combinatorial problems in connection with scientific computing. Furthermore, it paves the way for research directions that involve more than one of the three representations, for instance, graph-based preconditioning of Jacobian matrices.

Acknowledgements

We thank Simon Leßenich for his help in preparing the figures and Armin Jäger for discussions on stencil-based Jacobian computations. This research is partially supported by the German Federal Ministry of Education and Research (BMBF) under the contract 03SF0326A "MeProRisk: Novel methods for exploration, development, and exploitation of geothermal reservoirs—a toolbox for prognosis and risk assessment." The Aachen Institute for Advanced Study in Computational Engineering Science (AICES) provides a stimulating research environment for our work.

References

[1] D. Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22:251–256, 1979.

[2] T. F. Coleman, B. S. Garbow, and J. J. Moré. Algorithm 618: Fortran subroutines for estimating sparse Jacobian matrices. *ACM Transactions on Mathematical Software*, 10(3):346–347, 1984.

[3] T. F. Coleman, B. S. Garbow, and J. J. Moré. Software for estimating sparse Jacobian matrices. *ACM Transactions on Mathematical Software*, 10(3):329–345, 1984.

[4] T. F. Coleman, B. S. Garbow, and J. J. Moré. Algorithm 636: FORTRAN subroutines for estimating sparse Hessian matrices. *ACM Transactions on Mathematical Software*, 11(4):378–378, 1985.

[5] T. F. Coleman, B. S. Garbow, and J. J. Moré. Software for estimating sparse Hessian matrices. *ACM Transactions on Mathematical Software*, 11(4):363–377, 1985.

 ISBN 13: 978-84-613-5510-5

[6] T. F. Coleman and J. J. Moré. Estimation of sparse Jacobian matrices and graph coloring problems. *SIAM Journal on Numerical Analysis*, 20(1):187–209, 1983.

[7] D. W. Cranston and P. D. Hovland. Colorings for efficient derivative computation on grids with periodic boundaries. Preprint of the Mathematics and Computer Science Division ANL/MCS–P1557–1108, Argonne National Laboratory, Argonne, IL, USA, 2008.

[8] A. R. Curtis, M. J. D. Powell, and J. K. Reid. On the estimation of sparse Jacobian matrices. *Journal of the Institute of Mathematics and Applications*, 13:117–119, 1974.

[9] K. Datta, S. Kamil, S. Williams, L. Oliker, J. Shalf, and K. Yelick. Optimization and performance modeling of stencil computations on modern microprocessors. *SIAM Review*, 51(1):129–159, 2009.

[10] A. H. Gebremedhin. The enabling power of graph coloring algorithms in automatic differentiation and parallel processing. In U. Naumann, O. Schenk, H. D. Simon, and S. Toledo, editors, *Combinatorial Scientific Computing*, number 09061 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2009. Schloss Dagstuhl — Leibniz-Zentrum für Informatik, Germany.

[11] A. H. Gebremedhin, F. Manne, and A. Pothen. What color is your Jacobian? Graph coloring for computing derivatives. *SIAM Review*, 47(4):629–705, 2005.

[12] D. Goldfarb and P. L. Toint. Optimal estimation of Jacobian and Hessian matrices that arise in finite difference calculations. *Mathematics of Computation*, 43(167):69–88, 1984.

[13] S. Hossain and T. Steihaug. Computing a sparse Jacobian matrix by rows and columns. *Optimization Methods and Software*, 10:33–48, 1998.

[14] S. Hossain and T. Steihaug. Sparsity issues in the computation of Jacobian matrices. In *ISSAC '02: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, pages 123–130, New York, NY, USA, 2002. ACM Press.

[15] S. Hossain and T. Steihaug. Graph coloring in the estimation of sparse derivative matrices: Instances and applications. *Discrete Applied Mathematics*, 156(2):280–288, 2008.

[16] D. Matula, G. Marble, and J. Isaacson. Graph coloring algorithms. In R. Read, editor, *Graph Theory and Computing*, pages 109–122. Academic Press, New York, 1972.

[17] D. K. Melgaard and R. F. Sincovec. General software for two-dimensional nonlinear partial differential equations. *ACM Transactions on Mathematical Software*, 7(1):106–125, 1981.

 ISBN 13: 978-84-613-5510-5

[18] D. J. A. Welsh and M. J. D. Powell. An upper bound for the chromatic number of a graph and its applications to timetabling problems. *The Computer Journal*, 10:85–87, 1967.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Bit parallel circuits for arithmetic operations in composite fields $GF(2^{nm})$

A.A.Burtsev[1], R.A.Khokhlov[2], S.B.Gashkov[2] and I.B.Gashkov[3]

[1] *Moscow Institute of Physics and Technology, Moscow, Russia*

[2] *Department of Mechanics and Mathematics, Moscow State University, Russia*

[3] *Department of Mathematics, Faculty of Technology and Science , Karlstad University, Sweden*

emails: a-burtsev@yandex.ru, khokhlov@mail.ru, gashkov@lsili.ru, igor.gachkov@kau.se

Abstract

It was constructed circuits with low depth and complexity for multiplication and inversion in some finite fields of characteristic two.

Key words: : finite field, circuits, complexity .

1 Introduction

We investigate the depth an the complexity of realization of operations of multiplication and inversion by circuits consisting of two-input logic elements.

Our goal is the minimization of the depth of arithmetic circuits. The depth is the maximal number of elements in any chain connecting inputs of given circuit and its outputs. Also we were aimed to minimize the complexity of this circuits. The complexity is the number of elements in a circuit.

Circuits for finite field arithmetical operations are used in coding (see, for example, [1], [3], [2]), cryptography (see, for example, [4], [5]), digital signal processing (see, for example, [6]) etc. In these applications usually the fields of characteristic two are used.

In public key cryptography large dimensional fields are applied. For security it is necessary to use fields of dimension 1000 and greater. But in ECC — elliptic curve cryptography (see, for example, [7]) fields of dimension less than 200 are used.

The multipliers in standard bases generated by irreducible trinomials and pentanomials were constructed in the Ph.D. thesis of E.D. Mastrovito [8]. For example, in [8] the complexity of $GF(2^n)-$multipliers for $n = 2, 3, \ldots, 16$ is equal

$$7, 17, 31, 49, 71, 97, 148, 161, 199, 241, 351, 371, 478, 449, 537$$

 ISBN 13: 978-84-613-5510-5

and the depth is equal

$$3, 4, 4, 6, 5, 5, 6, 7, 7, 7, 8, 7, 8, 6, 7.$$

The general bounds of the complexity (and the depth) of these multipiers are equal $O(n^2), O(\log n)$.

By Karatsuba's method (see, for example, [9]) it is possible to construct the multipliers with the complexity $O(n^{log_2 3})$.

Problems of practical application of Karatsuba's method for multiplication in a field $GF(2^n)$ are considered in the Ph.D. thesis of C.Paar [10].

In [11] some architectures of multipliers are given for a field $GF(2^{4n})$ Best multipliers for $n = 8, 12, 16, 20, 24, 28, 32$ have the complexity

$$117, 216, 390, 546, 813, 1020, 1569,$$

and the depth $10, 9, 11, 13, 11, 14$. For multiplication in normal bases are known the methods [14], [15], [16].

The best bounds of the depth of inversion in a field $GF(2^n)$ is equal $O(\log n)$ [17], [18]. However, this result have only theoretical significance.

In [19] is given the $GF(2^n)$ inversion algorithm with the complexity $O(n^3) \log n$ and depth $O(\log^2 n)$. It is based on the identity

$$x^{2^n-1} = \left(x^{2^{\lceil n/2 \rceil}-1}\right)^{2^{\lfloor n/2 \rfloor}} x^{2^{\lfloor n/2 \rfloor}-1}.$$

High bound of the complexity has the reason in the high complexity of normal basis multiplication [14] (it is generally equal $O(n^3)$.) Main idea of the method [19] is fast computation of powers x^{2^m-1}. This algorithm of powers computation is known since the thirtieth years and belongs to A.Brauer and A.Scholz (see [20]).

We propose for inversion in some fields of composite dimension circuits of smaller depth and smaller complexity than well- known. These fields are considered as the extension of the subfields and also are called composite fields. For multiplication in these fields is known the method [15]. For a completeness we give in some special cases this method with simpler proof, than in [15].

2 Low depth circuits for operations in normal bases of composite fields

Further we need some definitions and notations

2.1 Normal and optimal normal bases

Finite field of the order q^n is denoted by $GF(q^n)$. Elements of a field $GF(q^n)$ is represented as polynomials of a degree no more $n-1$ with coefficients from a field $GF(q)$. If the polynomials are represented *in a standard base*

$$B_\alpha = \{\alpha^0, \alpha^1, \ldots, \alpha^{n-1}\}$$

 ISBN 13: 978-84-613-5510-5

(an element α is called *the generator* of a base), then a multiplication of elements of a field is a multiplication of polynomials modulo of some irreducible over the field $GF(q)$ polynomial $g(x)$.

Sometimes instead of a standard base it is more convenient to use *a normal base*, namely a base

$$B^{\alpha} = \{\alpha^{q^0}, \alpha^{q^1}, \ldots, \alpha^{q^{n-1}}\},$$

generated by an generator α of a standard base (α is a root of a polynomial $g(x)$). *Exponentiation* in the degree q (and in any degree q^m) in a normal base is a shift of coefficients, because

$$\zeta^q = x_{n-1}\alpha + x_0\alpha^q + x_1\alpha^{q^2} + \cdots + x_{n-2}\alpha^{q^{n-1}}.$$

$$B = \{\alpha, \alpha^q, \alpha^{q_2}, \ldots, \alpha^{q^{n-1}}\}$$

Let T be the matrix, in which $i-$th line is the vector of coefficients of element $\alpha\alpha^{q^i}$ fields $GF(q^n)$ concerning base B The number of nonzero elements in the T is called *the complexity C_B of a given normal base.* This definition explain by the Massey-Omura algorithm of multiplication in a normal base B (see, for example, [1]): Let

$$\xi = \sum_{i=0}^{n-1} x_i\alpha^{q^i} \quad \zeta = \sum_{j=0}^{n-1} y_j\alpha^{q^j}$$

be any elements of the field $GF(q^n)$, then product of this elements may be calculated by the formula

$$\pi = \sum_{m=0}^{n-1} p_m\alpha^{q^m},\ p_m = \sum_{i,j=0}^{n-1} t_{i-j,m-j}x_iy_j = \sum_{i,j=0}^{n-1} a_{i,j}S^m(x_i)S^m(y_j)$$

where S^m is a shift of coordinates of the given vector on m positions, and

$$A(x,y) = \sum_{i,j=0}^{n-1} a_{i,j}x_iy_j$$

is a bilinear form with a matrix A, defined by the equalities $a_{i,j} = t_{i-j,-j}$, where $i-j$ and $-j$ are calculated modulo n.

The complexity of multiplication over normal base of a field $GF(q^n)$ is less or equal to $n(2C_B + n - 1)$ operations in the subfield $GF(q)$.

It is known, that the complexity C_B of any normal base B of a field $GF(q^n)$ is not less than $2n-1$. Normal bases with the complexity $2n-1$ are called *optimal normal bases.* These bases were found in [22].

Optimal normal bases of *the first type* there exist if $n+1 = p$ is a prime number, and q is a primitive element modulo p. A generator of a base is a primitive root of degree p from 1 in a field $GF(q^n)$. Bases of *the second type* there exist if $2n+1 = p$ is a prime, and q is a primitive element modulo p. Bases of *the third type* there exist if $2n+1 = p$ is a prime, $p \equiv 3 (\mod 4)$, and the order of q modulo p is equal n. Generators in the last two cases are $\alpha = \zeta + \zeta^{-1}$, where ζ is a primitive root of degree p from 1 in a field $GF(q^{2n})$.

 ISBN 13: 978-84-613-5510-5

2.2 Reduction of inversion in a field $GF(n_1n_2)$ to inversion in a field $GF(n_2)$.

Let's present a field $GF((2^{n_1})^{n_2})$ as the extension of degree n_2 of fields $GF(2^{n_1})$. In the field $GF(2^{n_1})$ we choose any base B_1. Suppose the circuits for multiplication and inversion for this base in this field are constructed with the complexity and the depth $L(M(n_1)) = L(M_{B_1}(n_1))$, $D(M(n_1)) = D(M_{B_1}(n_1))$, $L(I(n_1)) = L(I_{B_1}(n_1))$, $D(I(n_1)) = D(I_{B_1}(n_1))$. In the considered extension we choose any base B_2 over the subfield $GF(2^{n_1})$. Then in the field $GF(2^n)$ we choose the base

$$B = B_1 \otimes B_2 = \{\alpha_i\beta_j : \alpha_i \in B_1, \beta_j \in B_2\}.$$

It is possible according to the following lemma (for example, see [1, Lemma 3.3.12], [23])

Lemma 1 *Let $A = \{\alpha_0, \ldots, \alpha_{m-1}\}$ and $B = \{\beta_0, \ldots, \beta_{n-1}\}$ be a bases of a fields $K = GF(q^m)$ and $L = GF(q^n)$ over a field $F = GF(q)$ and $g.c.d(m, n) = 1$. Then $C = \{\alpha_i\beta_j : i = 0, \ldots, m-1; j = 0, \ldots, n-1\}$ is the base in the field $GF(q^{mn})$ over the F.*

It is valid also (for example, see [1, Theorem 3.3.13], [23])

Theorem 1 *Let α and β be generators of normal bases A and B fields $K = GF(q^m)$ and $L = GF(q^n)$ over a field $F = GF(q)$ and and $g.c.d(m, n) = 1$. Then $\gamma = \alpha\beta$ is a generator of a normal base of the field $E = GF(q^{mn})$ over the F.*

Suppose the constructed circuit for multiplication for base B_2 consist of m elements of multiplication in a subfield $GF(2^{n_1})$ and a elements of addition, where $m + a = L(M_{B_2}(n_2))$ and have depth $D(M_{B_2}(n_2))$. Suppose in any chain of elements connecting inputs and outputs of the circuit there is only one element of multiplication. Then this circuit generates the circuit of multiplication in base B with the total complexity

$$L(M_B(n)) = mL(M_{B_1}(n_1)) + an_1 \leq L(M_{B_2}(n_2))L(M_{B_1}(n_1)),$$

with the multiplicative complexity (the number of elements of multiplication)

$$M(M_B(n)) = m(M_{B_1}(n_1))m(M_{B_2}(n_2)),$$

and with the depth

$$D(M_B(n)) = D(M_{B_1}(n_1)) + D(M_{B_2}(n_2)) - 1.$$

It is known, that the set of all automorphisms of a field $GF((2^{n_1})^{n_2})$ over a subfield $GF(2^{n_1})$ is a cyclical group . This group (is called usually *the Galois group* of the given extension) may be represented as the group $G = \{\sigma, \ldots, \sigma^{n_2}\}$ of powers of the automorphism $\sigma : x \to x^q$, $q = 2^{n_1}$, such that $\sigma^{n_2} = e$. (for example, see [1, Theorem 1.2.2]). The automorphism $\sigma : x \to x^q$ is *the Frobenius automorphism* . For an

 ISBN 13: 978-84-613-5510-5

extension $E = GF(q^n)$ of a field $F = GF(q)$ with the Galois group generated by an automorphism $\sigma : x \to x^q$ the function

$$N_{E/F}(x) = x\sigma(x)\dots\sigma^{n-1}(x) = x^{q^0}x^{q^1}\dots x^{q^{n-1}},$$

of any $x \in E$ is called *the norm* of this element.

Further instead of $N_{E/F}(x)$ we write $N_n(x)$, if the field E is the extension of degree n of a field F or $N(n_2/n_1)$, if a field E are the extension of degree n_2 of a field $GF(2^{n_1})$.

The norm of any element of extension of a field always belongs to this field (see, for example, [1, Lemma 1.6.1]).

Using the notation $Z_{n_2}(x) = \sigma(x)\dots\sigma^{n_2-1}(x)$, a inverse element can be computed by the formula $x^{-1} = Z_{n_2}(x)(N_{n_2}(x))^{-1}$. The multiplication in this formula is multiplication an element of the field $GF((2^{n_1})^{n_2})$ on an element of the subfield $GF(2^{n_1})$, therefore the complexity of multiplication is equal $n_2L(M_{B_1}(n_1))$, and the depth is equal $D(M_{B_1}(n_1))$.

Let us the circuit computing simultaneously $N_{n_2}(x)$ and $Z_{n_2}(x)$ denote by $NZ_{n_2}(x)$. Suppose the complexity of this circuit is equal $L(NZ_{n_2}(n_1))$, the depth of the subcircuit computing $N_{n_2}(x)$ is equal $D(N_{n_2}(n_1))$, and depth of the subcircuit computing $Z_{n_2}(x)$ is equal $D(Z_{n_2}(n_1))$. Then the circuit $I_{n_1n_2}$ for inversion in the field $GF(2^{n_1n_2})$ can be constructed from the circuit NZ_{n_2}, the circuit for inversion I_{n_1} in the subfield $GF(2^{n_1})$ and circuit of multiplication $M(n_1n_2, n_1)$. The complexity of this circuit is equal

$$L(I_{n_1n_2}) = L(I_{n_1}) + L(NZ_{n_2}(n_1)) + n_2L(M(n_1)), \tag{1}$$

and the depth is equal

$$D(I_{n_1n_2}) = \max\{D(I_{n_1}) + D(N_{n_2}(n_1)), D(Z_{n_2}(n_1))\} + D(M(n_1)). \tag{2}$$

We need a nontrivial methods of constructions of the circuit $NZ_{n_2}(n_1)$. The trivial methods give bounds $L(NZ_{n_2}(n_1)) \le (n_2 - 1)L(M(n_1))$,

$$D(NZ_{n_2}(n_1)) \le \lceil \log_2(2n_2 - 2)\rceil D(M(n_1)).$$

3 Towers of fields.

3.1 Reduction of inversion in a field $GF^{n_1n_2n_3}$ to inversion in a field GF^{n_3}.

Applying two times the construction 2.2, it is possible to reduce inversion in the field $GF(((2^{n_1})^{n_2})^{n_3})$ to inversion in the subfield $GF(2^{n_1})$. The complexity of the obtained circuit is equal

$$L(I_n) = L(I_{n_1n_2}) + L(NZ_{n_3}(n_1n_2)) + n_3L(M(n_1n_2)) =$$

$$L(I_{n_1}) + L(NZ_{n_1}(n_2)) + n_2L(M(n_1)) + L(NZ_{n_3}(n_1n_2)) + n_3L(M(n_1n_2))$$

 ISBN 13: 978-84-613-5510-5

and the depth is equal

$$D(I_n) = \max\{D(I_{n_1n_2}) + D(N_{n_3}(n_1n_2)), D(Z_{n_3}(n_1n_2))\} + D(M(n_1n_2)) =$$

$$\max\{\max\{D(I_{n_1}) + D(N_{n_2}(n_1)), D(Z_{n_2}(n_1))\} + D(M(n_1)) +$$
$$+D(N_{n_3}(n_1n_2)), D(Z_{n_3}(n_1n_2))\} + D(M(n_1n_2)).$$

It is possible to construct the circuit with greater complexity but smaller depth as follows. Let's consider the field $GF(((2^{n_1})^{n_2})^{n_3})$ as the extension of degree n_2n_3 of field $GF(2^{n_1})$ and once apply the construction of the previous section. Then we obtain the circuit for inversion with the complexity

$$L(I_n) = L(I_{n_1}) + L(NZ_{n_2n_3}(n_1)) + n_2n_3L(M(n_1)),$$

and the depth

$$D(I_n) = \max\{D(I_{n_1}) + D(N_{n_2n_3}(n_1)), D(Z_{n_2n_3}(n_1))\} + D(M(n_1)).$$

Nontrivial constructions for a circuit $NZ_{n_1n_2}(n_3)$ are given in the sequel.

3.2 Inversion in towers of fields and circuits $NZ_{n_1n_2}(n_3)$.

Let's consider the extension of degree n_1n_2 of field $GF(2^{n_3})$ as a field tower consisting of the extension of degree n_2 of field $GF(2^{n_3})$ and the extension of degree n_1 of field $GF(2^{n_2n_3})$. It is known, that the Galois group of the field $GF(2^n)$ over the subfield $GF(2^{n_3})$ is cyclical group of the order n_1n_2. We present this group as

$$\{\sigma, \dots, \sigma^{n_1n_2}\},$$

where σ is a generating automorphism. Let's consider the subgroup of automorphisms of the field $GF(2^n)$ over the subfield $GF(2^{n_2n_3})$. This subgroup has the order n_1, therefore it is equal to the subgroup

$$\{\sigma^{n_2}, \sigma^{2n_2}, \dots, \sigma^{n_1n_2}\},$$

generated by the automorphism σ^{n_2}. Restriction this automorphism(and all other automorphisms from this subgroup) on the subfield $GF(2^{n_2n_3})$ is equal to the identical automorphism.

Let's obtain in a convenient kind a well-known property of norm (see, for example, [1, the theorem 1.6.8]).

As the norm of the extension of degree n_1n_2 is equal

$$N_{n_1n_2}(x) = x\sigma(x)\dots\sigma^{n_1n_2-1}(x),$$

and the norm in this field, considered as the extension of degree n_1, is equal

$$N_{n_1}(x) = x\sigma^{n_2}(x)\sigma^{2n_2}(x)\dots\sigma^{(n_1-1)n_2}(x) = x\tau(x)\dots\tau^{n_1-1}(x), \tau = \sigma^{n_2},$$

 ISBN 13: 978-84-613-5510-5

therefore, putting $y = N_{n_1}(x)$, we have

$$N_{n_1n_2}(x) = x\sigma(x)\dots\sigma^{n_1n_2-1}(x) = y\sigma(y)\dots\sigma^{n_2-1}(y), y \in GF(2^{n_2n_3}).$$

The automorphism σ maps the subfield $GF(2^{n_2n_3})$ in itself(so there are no other subfields of the same order), automorphism σ^{n_2} is equal the identical automorphism on this subfield, and other automorphisms $\sigma^k, k < n_2$ are not identical(otherwise it belong to the subgroup automorphisms of thr field $GF(2^n)$ over the subfield $GF(2^{n_2n_3})$, therefore the set of restrictions of automorphisms

$$\{\sigma, \dots, \sigma^{n_2}\}$$

on the subfield $GF(2^{n_2n_3})$ form the group of automorphisms this field over the subfield $GF(2^{n_3})$. Therefore for norm

$$N_{n_2}(y) = y\sigma(y)\dots\sigma^{n_2-1}(y), y \in GF(2^{n_2n_3}),$$

the equality is valid

$$N_{n_1n_2}(x) = x\sigma(x)\dots\sigma^{n_1n_2-1}(x) = y\sigma(y)\dots\sigma^{n_2-1}(y) = N_{n_2}(y) = N_{n_2}(N_{n_1}(x)).$$

Deleting in this product the first multiplicand, we obtain the identity

$$Z_{n_1n_2}(x) = \sigma(x)\dots\sigma^{n_1n_2-1}(x) = Z_{n_1}(x)\sigma(y)\dots\sigma^{n_2-1}(y) =$$

$$Z_{n_1}(x)Z_{n_2}(y) = Z_{n_1}(x))_Z n_2(N_{n_1}(x)),$$

where $Z_{n_1}(x) = \sigma^{n_2}(x)\sigma^{2n_2}(x)\dots\sigma^{(n_1-1)n_2}(x) = \tau(x)\dots\tau^{n_1-1}(x), \tau = \sigma^{n_2}$, $Z_{n_2}(y) = \sigma(y)\dots\sigma^{n_2-1}(y)$. Let's denote by $M(n, n_2n_3)$ the circuit of multiplication of any element of a field $GF(2^n)$ on any element of a subfield $GF(2^{n_2n_3})$ and denote by $NZ_{n_2}(y)$ the circuit for simultaneous computation $N_{n_2}(y), Z_{n_2}(y)$. Also we denote by $NZ_{n_1}(x)$ the circuit for simultaneous compution $N_{n_1}(x), Z_{n_1}(x)$. From these four circuits it is possible to construct the circuit $NZ_{n_1n_2}(x)$ with the complexity

$$L(NZ_{n_1n_2}(n_3)) = L(NZ_{n_1}(n_2n_3)) + L(NZ_{n_2}(n_3)) + L(M(n, n_2n_3))$$

and the depth

$$D(N_{n_1n_2}(n_2n_3)) = D(N_{n_1}(n_2n_3)) + D(N_{n_2}(n_3)),$$

$$D(Z_{n_1n_2}(n_2n_3)) =$$

$$\max\{D(Z_{n_1}(n_2n_3)), D(Z_{n_2}(n_3)) + D(N_{n_1}(n_2n_3))\} + D(M(n, n_2n_3)).$$

 ISBN 13: 978-84-613-5510-5

3.3 On the circuits $M(n, n_2n_3)$.

If a base B of the extensions of degree n_1n_2 of field $GF(2^{n_3})$ is represented as a product of a base B_2 of the extensions of degree n_2 of the same field and a base B_1 of the extensions of degree n_1 of the field $GF(2^{n_2n_3})$, then are valid the equalities

$$L(M(n, n_2n_3)) = n_1L(M(n_2n_3)), D(M(n, n_2n_3)) = D(M(n_2n_3)),$$

$$L(M(n)) \leq L(M_{B_1}(n_1))L(M(n_2n_3)),$$
$$D(M(n)) = D(M_{B_1}(n_1)) + D(M(n_2n_3)) - 1,$$

where $M_{B_1}(n_1)$ is the circuit of multiplication in the base B_1 consisting of elements realizing arithmetic operations in the subfield $GF(2^{n_2n_3})$. If $g.c.d(n_1, n_2) = 1$, then in the subfield $GF(2^{n_1n_3})$ it is possible to choose arbitrary base B_1.

As an example we consider Application to the biquadratic extension

It is valid

Theorem 2 *If in the field tower $GF(((2^n)^2)^2)$ were chosen the optimal normal base $\{\alpha_1, \alpha_1^2\}$ and the base $\{1, \alpha_2\}$, $\alpha_2^2+\alpha_2 = \alpha_1$ then is valid the following recurrent relations for the complexity and the depth of multiplication*

$$L(M(4n)) \leq 9L(M(n)) + 20n,$$

$$D(M(4n)) \leq D(M(n)) + 4,$$

and for the complexity and the depth of inversion

$$L(I(4n)) \leq 14L(M(n)) + 16n + L(I(n)),$$

$$D(I(4n)) \leq 3D(M(n)) + 2 + \max\{D(I(n)), 2\}$$

4 Table

In the first column the dimension n of a field $GF(2^n)$ is given. If the second column is empty, then the circuit is constructed by computing an explicit inversion formula and optimizing of the obtained circuit. If the second column is not empty, for example, it contains the sign $\xrightarrow{2(st)}$, then it means that the circuit was constructed using of the quadratic extension and in the field $GF(2^2)$ was choosen the standard base. By badges *opt* and *norm* denote an optimal normal or an ordinary normal base. The absence of a badge means that the base is neither normal nor standard. In the fourth column are given the complexity and the depth of inversion circuits as $I_n = (\text{number}, \text{number})$, the complexity and the depth of multipliers as $M_n = (\text{number}, \text{number})$, the complexity and the depth of squaring circuits as $K_n = (\text{number}, \text{number})$. In the last column the badges *st*, *norm*, *opt* means that a base in a field was chosen standard, normal, optimal normal.

 ISBN 13: 978-84-613-5510-5

Table 1: Table

2 :		$I_2 = (0,0),\ M_2 = (7,3)$	opt
3 :		$I_3 = (6,2),\ M_3 = (18,4)$	opt
4 :		$I_4 = (24,3),\ M_4 = (31,4)$	opt
4 :		$I_4 = (21,4),\ M_4 = (31,4),\ K_4 = (2,1)$	st
5 :		$I_5 = (55,4),\ M_5 = (55,5)$	opt
6 :		$I_6 = (156,6),\ M_6 = (81,5)$	opt
6 :	$\overset{2(opt)}{\longrightarrow}$	$I_6 = (60,10),\ M_6 = (66,6)$	$norm$
6 :	$\overset{3(opt)}{\longrightarrow}$	$I_6 = (84,8),\ M_6 = (66,6)$	$norm$
8 :	$\overset{2(st)}{\longrightarrow}$	$I_8 = (125,11),\ M_8 = (112,6),\ K_8 = (7,2),\ M_{r,4} = (3,1)$	
10 :	$\overset{2(opt)}{\longrightarrow}$	$I_{10} = (220,14),\ M_{10} = (230,15)$	$norm$
10 :	$\overset{2(opt)}{\longrightarrow}$	$I_{10} = (213,15),\ M_{10} = (229,16)$	
10 :	$\overset{5(opt)}{\longrightarrow}$	$I_{10} = (871,16),\ M_{10} = (185,7)$	$norm$
12 :	$\overset{3(opt)}{\longrightarrow}$	$I_{12} = (335,16),\ M_{12} = (234,7)$	$norm$
12 :	$\overset{4}{\longrightarrow}$	$I_{12} = (306,16),\ M_{12} = (222,8)$	
15 :	$\overset{3(opt)}{\longrightarrow}$	$I_{15} = (590,20),\ M_{15} = (390,8)$	$norm$
15 :	$\overset{3(opt)}{\longrightarrow}$	$I_{15} = (556,21),\ M_{15} = (354,8)$	
16 :	$\overset{2(st)}{\longrightarrow}$	$I_{16} = (502,24),\ M_{16} = (378,9),\ K_{16} = (32,5),\ M_{r,8} = (10,2)$	
16 :	$\overset{2}{\longrightarrow}$	$I_{16} = (475,25),\ M_{16} = (382,12),\ M_{r,8} = (14,3)$	
20 :	$\overset{4opt}{\longrightarrow}$	$I_{20} = (905,21),\ M_{20} = (595,9)$	
24 :	$\overset{2}{\longrightarrow}$	$I_{24} = (1078,30),\ M_{24} = (767,11)$	
30 :	$\overset{5(opt)}{\longrightarrow}$	$I_{30} = (8229,18),\ M_{30} = (1455,9)$	$norm$
30 :	$\overset{6(norm)}{\longrightarrow}$	$I_{30} = (1925,28),\ M_{30} = (1230,10)$	$norm$
32 :	$\overset{2(st)}{\longrightarrow}$	$I_{32} = (3275,39),\ M_{32} = (1772,13),\ M_{r,16} = (26,3)$	
48 :	$\overset{3(opt)}{\longrightarrow}$	$I_{48} = (4128,52),\ M_{48} = (2460,12)$	
96 :	$\overset{3(opt)}{\longrightarrow}$	$I_{96} = (19707,79),\ M_{96} = (11016,16)$	
120 :	$\overset{3(opt)}{\longrightarrow}$	$I_{120} = (73930,74),\ M_{120} = (12480,14)$	$norm$

 ISBN 13: 978-84-613-5510-5

References

[1] Jungnickel D., *Finite fields. Structure and arithmetic*, Wissenschaftsverlag, Mannheim, Leipzig, Wien, Zurich (1993).

[2] Blahut R.E., *Theory and practice of error control codes*, Addison-Wesley publishing company (1984).

[3] Berlekamp E.R., *Algebraic coding theory*, McGraw Hill(1968).

[4] Menezes A., van Oorshot P., Vansotone S. *Handbook of applied cryptography*, CRC Press (1997).

[5] Schneier B., *Applied cryptography*, John Wiley and Sons, Inc. (1996).

[6] McCLellan J.H., Rader C.M., *Number theory in digital signal processing*, Prentice-Hall (1979).

[7] Rosing M., *Implementing elliptic curve cryptography*, Manning,(1998).

[8] Mastrovito E.D., *VLSI architectures for computation in Galois fields*, Ph. D. Thesis, Linkoping University, Dept. Electr. Eng., Sweden (1991).

[9] von zur Gathen J., Gerhard J. Modern computer algebra, Cambridge University Press, 1999.

[10] Paar C., *Effective VLSI architectures for bit paralel computation in Galois fields*, Ph. D. Thesis, Universitat GH Essen, Germany, (1994).

[11] Paar C., Fleischmann P., Roelse P., *Effective multiplier architectures for Galois fields* $GF(2^{4n}$, *IEEE Trans. Comp.*, bf 47 2 (1998), 162-70.

[12] Paar C., Fan J.L., *Efficient inversion in tower fields of characteristic two*, *ISIT*, Ulm, Germany, (1997).

[13] Afanasyev V.B., *Complexity of VLSI implementation of finite field arithmetic*, in *II Intern. Workshop on algebraic and combinatorial coding theory*, Leningrad, USSR (Sep. 1990), 6-7.

[14] Massey J.L., Omura J.K., *Apparatus for finite fields computation*, *US patent 4587627* (1986).

[15] Reyhani-Masoleh A., Hasan M.A., *On effective normal basis multiplication*, *IndiaCRYPT*,(2000).

[16] Bolotov A.A., Gashkov S.B., *Fast multiplication in normal bases of finite fields* , *Discrete mathematics and applications*, **13** 3, (2001), 3-31.

[17] Litow B.E., Davida G.I., $O(\log n)$ *time for finite field inversion*, *Lecture notes in computer sciences*, **319** (1988), Springer-Verlag, 74-80.

 ISBN 13: 978-84-613-5510-5

[18] von zur Gathen J.,*Inversion in finite fields*, *J. Symbplic Comput.*, **9** (1990), 175-183.

[19] Itoh T., Tsujii S. *A fast algorithm for computing multiplicative inverses in $GF(2^n)$ using normal bases*, *Inform. And Comp.*, **78** (1988), 171-177.

[20] Knut D. The art of computer programming, third edition. Addison-Wesley, 1998.

[21] Morii M., Kasahara M. *Efficient construction of gate circuit for computing multiplicative inverses in $GF(2^n)$*, *Trans. Of the IEICE*, E **72**, 1 (1989), 37-42.

[22] Mullin R.C., Onyszchuk I.M., Vanstone S.A., Wilson R.M *Optimal normal bases in $GF(p^n)$*, *Discrete Applied Mathematics*, **22** (1988/89), 149-161.

[23] Seguin J.E.,*Low complexity normal bases*, *Discrete Applied Mathematics*, **28**, (1990), 309-312.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Mathematical Problems of Traffic Flow Theory

Alexander P. Buslaev[1], Alexander V. Gasnikov[2] and Marina V. Yashina[3]

[1] *Department of Mathematics, Moscow Automobile and Road Technical University*

[2] *Department of Applied Mathematics, Moscow Institute of Physics and Technology*

[3] *Department of Mathematical Cybernetics, Moscow Technical University of Communication and Informatics*

emails: apal2006@yandex.ru, gasnikov@yandex.ru, yash-marina@yandex.ru

Abstract

We have considered some mathematical models of traffic flow. There are formulations of basic modern approaches to description of traffic jam characteristics on a network. We also lead a few open problems.

Key words: traffic flow, traffic jam, shock waves, macroscopic and microscopic models, leader's following models, conservation laws.
MSC 2000: AMS codes (optional) 34D05, 3405, 35B05, 35B40, 35Q58

1 Introduction

Modelling of traffic flows on megacities network is as actual, extremely difficult problem. And the exact formalization of these questions gives the contensive mathematical problems.

The question of modelling complexity has two aspects: how much wide is the flow support and how much the flow components are synchronized. Really, as only the support is reduced to "two rails", and synchronization of flow component are as iron connection, than the flow kinematics becomes trivial. It is more than freedom at a military column, Which, nevertheless, should keep steady regular distribution. At last, on multilane road with possibility of transition from one lane to another (non channel movement) and with the various qualifying Skills and purposes of drivers movement becomes incomparable More difficult. So more difficult, that since thirtieth years the twentieth century there is a search of adequate models, [1].

 ISBN 13: 978-84-613-5510-5

Growth of quantity of cars on roads, capacity and high-speed properties lead to the aggravated a problem of safety of the traffic inadequate behaviour of a part of drivers on the road, the limited possibilities human body according to road conditions.

The car becomes more and more smart, it means that share of formalizable actions in traffic increases. For example, unlike the driver, the technical device can more precisely estimate the dynamic dimension, i.e. the safe distance to ahead following car. Thus, transition to the formalized algorithms of car control reduces a share of unpredictable actions of drivers and gives the chance to fashion designers of traffic flow. At last after we hope to construct the general theory of traffic flows the lapse of many tens years, also we can describe the behavior of a separate car, a flow of cars on a road section and a set of cars on a complex city network.

2 Models for the leader following

For movement on one lane the position of points can be presented as of the sequence of the functions

$$... < x_1(t) < x_2(t) < ... < x_n(t)... \tag{2.1}$$

We consider necessary the following conditions for functions $x_i(t), i = 0, \pm 1, ...$
a) the functions satisfy (2.1);
b) they strictly monotonously increase;
c) the functions are continuously differentiable and

$$||\dot{x_i}||_{L_\infty(R+)} \leq M_1. \tag{2.2}$$

d) $\dot{x}$ are absolutely continuous and almost everywhere differentiable, and

$$||\ddot{x_i}||_{L_\infty(R+)} \leq M_2. \tag{2.3}$$

Let $d = d(x)$ be a positive and monotonously growth function of nonnegative argument, which defines safe distance between the next points, i.e. dynamic dimension.

For example,

$$x_{n+1} - x_n = \dot{x_i^2} + x_n + 1. \tag{2.4}$$

The movement of the pair (x_n, x_{n+1}) by the law (2.4) we call connected. If the relation (2.4) are executed for sequence $n = 1..., N-1$ then we receive a connected chain.

For a correctness of problem formulation it is necessary to add initial conditions, i.e. the positions of points at the time moment $t = 0$, and boundary conditions, for example, behaviour of the leader (or the outsider).

We note physical statements of model of following for the leader concern the sixtieth years of the last century [8]. The further development is received in many researches, for instant [9]. We consider following questions.

Question 1: Whether there is a connected chain of the set length N? What are necessary conditions of existence? What are sufficient conditions?

 ISBN 13: 978-84-613-5510-5

Question 2: How can be the description of qualitative behaviour of a chain depending on behaviour of the leader (Or the outsider)?

Question 3: What is an asymptotics of system solutions for a circle.

The following cycle of questions concerns to descriptions of flow dynamics in case of nonconnected a component. For a statement correctness we will enter function $g(x)$, strictly monotonously decreasing, $g(0) = 0$ and smooth.

Dynamics of nonconnected pair is described by the following law

$$\ddot{x}_n = g(\dot{x}_n - f(x_{n+1} - x_n)), \tag{2.5}$$

where $x_{n+1} - x_n = d(x_n) \Leftrightarrow \dot{x_n} = f(x_{n+1} - x_n)$.

Questions 4. What are the necessary and sufficient conditions on function , g, for which the system (2.5) would describe non-critical (without collisions) the movement converging to connected state.

3 Intensity, a plane and continual traffic models

Real numerical characteristics of a flow of following for the leader are the quantity of particles passing through fixed section (intensity) q and quantity of particles, on length unit at fixed moment (density) ρ. Long-term theoretical and experimental researches of experts in the traffic show, that there does not exist a simple dependence between these values. *Moreover, in some researches formal definitions of density and intensity do not every often consider necessary to show. But every technology of these values measurement have themselves nuances and details.*

Nevertheless, at certain conditions, about which it is supposed to mention, dependence $v = F(\rho)$ (speed on density) and, as consequence, $q = \rho F(\rho)$ (intensity on density) adequately reflects real behaviour of a chain. Then the model of following for the leader can to be reduced to the mathematical physics equation.

If we fulfill automodel reduction for the leader following model we obtained (in first order) conservation law (Lighthill - Whitham - Richards model), [9] ,

$$\frac{\partial \rho}{\partial t} + \frac{\partial q(\rho)}{\partial x} = 0,$$

where ρ is density; $v(\rho) = f(1/\rho)$ is velocity and $q(\rho) = \rho v(\rho)$ is intensity, "fundamental diagram".

And in second order conservation law with diffusion $D(\rho) = -v'(\rho)/2$ (Whitham model, 1974) , [9]

$$\frac{\partial \rho}{\partial t} + \frac{\partial q(\rho)}{\partial x} = \frac{\partial}{\partial x}\left(D(\rho)\frac{\partial \rho}{\partial x}\right).$$

This models (and also their future generalization) are rather convenient for investigation because of the hydrodynamic analogues. In this text we restrict ourselves only these macroscopic models, for more detail see [1], [2]. First of all, it is rather interesting to comprehend how these equations can describe traffic jam. Mathematical theory of shock and training waves was developed in works of A.N. Kolmogorov, I.G.

 ISBN 13: 978-84-613-5510-5

Petrovsky, N.S. Piskunov, Ya.B. Zel'dovich, I.M. Gel'fanfd, E.A. Hopf, P.D. Lax, A.M. Ilijn, O.A. Oleijnik, S.N. Kruzhkov, N.S. Petrosyan, G.M. Henkin, A.A. Shananin and other (see, for example, [10]). We plane to describe the contemporary state-of-the-art of this branch.

The other interesting question is how to stay the initially-boundary problem, say, for LWR model on the graph of the transport network. The main difficulties are boundary conditions in the nodes of the graphs (see, for example, [15]).

4 Deterministic-stochastic model of movement

We consider multilane movement of a considerable quantity of cars (particles). In a case when velocities of all particles are identical, it is possible to consider a flow configuration invariable. In this case intensity is configuration function, and averagings on various intervals of time lead to various estimations of intensity. It is natural to consider, that particles are not crossed by the dynamic dimensions and all motion field (multilane road) can be broken into corresponding cells in which or there is a particle, or there is no it. We receive a set connected clusters with an invariable configuration.

As in the field there are free cells as soon as the mode of velocities becomes non tense, there are particles which try to increase own velocity for the account changes of a configuration of own position in the flow. This situation generates a stochastic (individual) component of the flow.

Thus particle movement is summarized from collective and individual components.

Research of individual components of traffic flow is reduced to so-called models of integer automatic automata in traffic or to an percolation problem in physics [7], [13-14] or to problems of random walk in probability theory [2], [7].

As Blank notes [3] the first not trivial results, basically numerical, are received by Nagel K, Schreckenbag, Herman, Simon, Krug, [7], [10]. In these researches it was found, for example, that average particles velocity of movement on a cellular ring with probability 1

$$v(\rho) = \{1,\ \rho \in [0, 1/2];\ \rho^{-1} - 1,\ \rho \in (1/2, 1)\}.$$

Exact formulations and statements in the elementary model and generations is received by the Blank, [3], [4].

Problems on numerical characteristics for moving on a multilane cellular field are open. With the fixed probability p, $0 < p \leq 1$ and on the closed path with knots (crossings), for example, a path with shape as "eight".

It is offered to discuss some results in this topics.

5 Network dynamic system

Let G be a the plane oriental graph with vertices rate n and with the ends of rate 1, "n-th ended star", fig. 1.

The system state is characterized by a vector of density, flow mass on each edge. Flow process is defined by a mixed matrix in a vertex (Markov's matrix of the size $n \times n$)

 ISBN 13: 978-84-613-5510-5

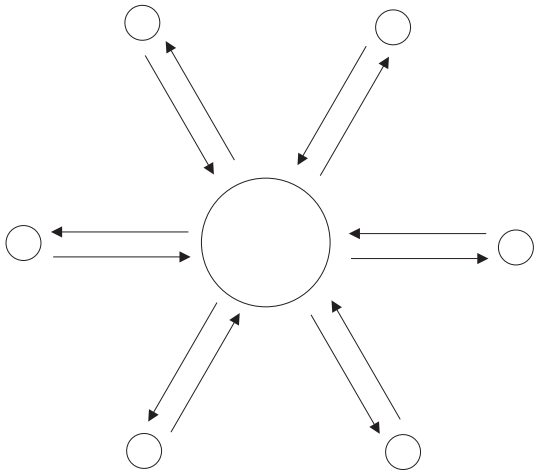


Figure 1:

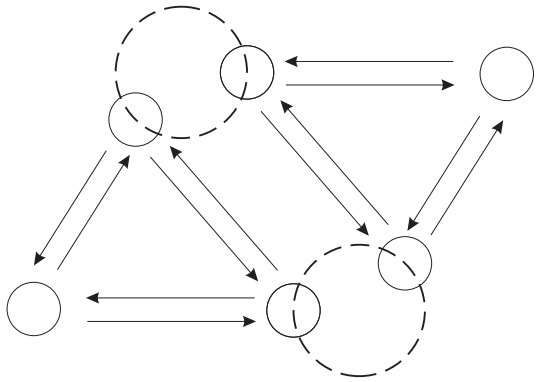


Figure 2:

and dependence of flow velocity on density (state function of an edge). The variant of dynamic system with control assumes two operating modes in vertex on input(" closed" and "opened") and also in case of open system there are an input and output of mass.

Already on an example " n-th ended star" there are substantial the problems of the stationary states description, of stability, critical modes when the flow mass on any edge reaches a maximum, flow control.

The considered elementary scheme allows to create more difficult traffic graphs by means of stars pasting, including regular lattices. For example, a triangulation (fig. 2) and a quadrature (fig. 3).

Except above questions there are actual mathematical problems are recovery of dynamic system states on a network under the information on its behaviour on a part network, i.e. approximate information. In what points of the network and with what accuracy it is necessary to measure information, that it will be possible to restore a dynamic system state on whole network, to estimate time of approach of a critical

 ISBN 13: 978-84-613-5510-5

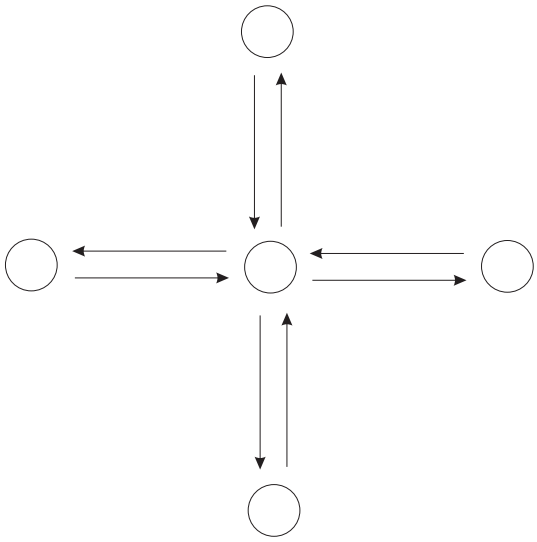


Figure 3:

mode. It is offered to give mathematically exact formalizations, to give some qualitative results and to show of computer simulations on above topics.

Acknowledgements

This work has been supported by Russian Foundation for Basic Research, Grant No. 08-01-00959.

References

[1] Rothery R. W. *Car following models.* Traffic Flow Theory, Transportation Research Board, 165, ed. by Gartner N., Messer C.J. Rathi A.K., Chap.4, 1992.

[2] Belyaev Yu. K. *On simplest model of movement without overtaking.* Izvest. AN SSSR, Tech.Kibern. **No.3** (1969) 17–21. (in Russian)

[3] Blank M.L. *Exact analyze of dynamic system applied in traffic flow models.* Uspehi Matem.Nauk. **55, 3** (2000) 167–168. (in Russian)

[4] Blank M.L. *Dynamics of traffic jams: order and chaos.* Mosc.Math.J.**V.1, No. 1** (2001) 1–26.

[5] Buslaev A.P., Novikov , V.M. Prikhodko, A.G. Tatashev, Yashina M.V. *Stochastic and simulation approach to traffic.* Moscow, Mir (2003), 286 p. (in Russian)

 ISBN 13: 978-84-613-5510-5

[6] Helbing D. *Traffic and related self-driven many particle systems* Reviews of modern physics. **V. 73, No. 4** (2001) 1067–1141. arXiv:cond-mat0012229

[7] Nagel K., Schreckenberg M. *A cellular automation model for freeway traffic* Phys. I France. **V. 2.** (1992) 2221–2229.

[8] Newell G.F. *Nonlinear effects in the dynamics of car - following* Oper. Res. **V. 9** (1961) 209–229.

[9] Treiber M., Hennecke A., Helbing D. *Congested traffic states in empirical observations and microscopic simulation* Phys. Rev. E. **V. 62.** (2000) 1805–1824.

[10] Whitham G.B. *Linear and Nonlinear Waves* John Wiley & Sons, 1974. http://en.wikipedia.org

[11] Gasnikov A.V. *Time-asymptotic behaviour of a solution of the Cauchy initial-value problem for a conservation law with non-linear divergent viscosity* Izv. RAN. Ser. Mat. **V. 73, No. 6** (2009) 39–76.

[12] Garavello M., Piccoli B. *Traffic Flow on Networks* Volume 1 of AIMS Series on Applied Mathematics. AIMS, 2006.

[13] Dangazo C.F. *The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory* Transp.Res. B. **V.28, No.4** (1994) 269–287.

[14] Dangazo C.F. *The cell transmission model, Part II: Network traffic* Transp.Res. B. **V.29, No.2** (1995) 79–93.

[15] Blank M. *Ergodic properties of a simple deterministic traffic flow model* J. Stat. Phys. **V. 111, No. 3-4** (2003) 903–930. arXiv:math.DS/0206194

[16] Blank M. *Hysteresis phenomenon in deterministic traffic flows* J. Stat. Phys. **V. 120, No. 3-4** (2005) 627–658. arXiv:math.DS/0408240

[17] Kurjhansky A.A. PhD thesis, Berkeley: University of California, 2007. http://lihodeev.com/pubs.html

[18] Chowdhury D., Santen L., Schadschneider A. *Statistical physics of vehicular traffic and some related systems* Phys. Rep. **V. 329** (2000) 199–329. arXiv:cond-mat/0007053v1

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Modelization of Turbo Encoder from Linear System Point of View

Pedro Campillo[1], Antonio Devesa[1], Victoria Herranz[1] and Carmen Perea[1]

[1] *Departamento de Estadstica, Matemticas e Informtica. Centro de Investigacin Operativa, Universidad Miguel Hernndez*

emails: pcampillo@umh.es, antonio.devesa@umh.es, mavi.herranz@umh.es, mavi.herranz@umh.es

Abstract

In this work we introduce de input-state-output representation of turbo codes and we present conditions for that the obtained representation was observable and minimal. *Key words: convolutional code, turbo code, linear system*

1 Introduction

A turbo encoder is formed by parallel concatenation of two recursive systematic convolutional encoders separated by a random interleaver. Turbo codes were first introduced in 1993 by Berrou, Glavieux, and Thitimajshima [1]. Actually are one of the most effective methods of generating codes with high error correction capability. In this paper, using the input-state-output representation of convolutional codes introduced by Rosenthal Schumacher and E.V. York [5] and, in a similar way that Climent, Herranz and Perea [3, **?**], we introduce the modelization of turbo codes from linear system point of view. The structure of the paper is as follows. In the next section we introduce the preliminary results and in Section 3 we present the main results.

2 Preliminary results

In this paper, we denote by $\mathbb{F} = \mathbb{F}_q$ the Galois field of q elements, $\mathbb{F}[z]$ the polynomial ring on the variable z with coefficients in $\mathbb{F}$ and $\mathbb{F}(z)$, the field of rational functions over $\mathbb{F}$ and $\overline{\mathbb{F}}$, the algebraic closure of $\mathbb{F}$.

Following [4] and [5], we define a convolutional code as a submodule $\mathcal{C} \subseteq \mathbb{F}^n[z]$. Since $\mathbb{F}[z]$ is a principal ideal domain and since $\mathcal{C}$ is a submodule of the free submodule $\mathbb{F}^n[z]$, the code $\mathcal{C}$ is free and it has a well defined rank k. Let $\{g_1(z), \ldots, g_k(z)\} \subseteq \mathbb{F}^n[z]$ be a basis of the free module $\mathcal{C}$ and let $G(z)$ be the $n \times k$ polynomial matrix whose ith column is the polynomial vector $g_i(z)$, for $i = 1, \ldots, k$. Then, $\mathcal{C}$ is defined as

 ISBN 13: 978-84-613-5510-5

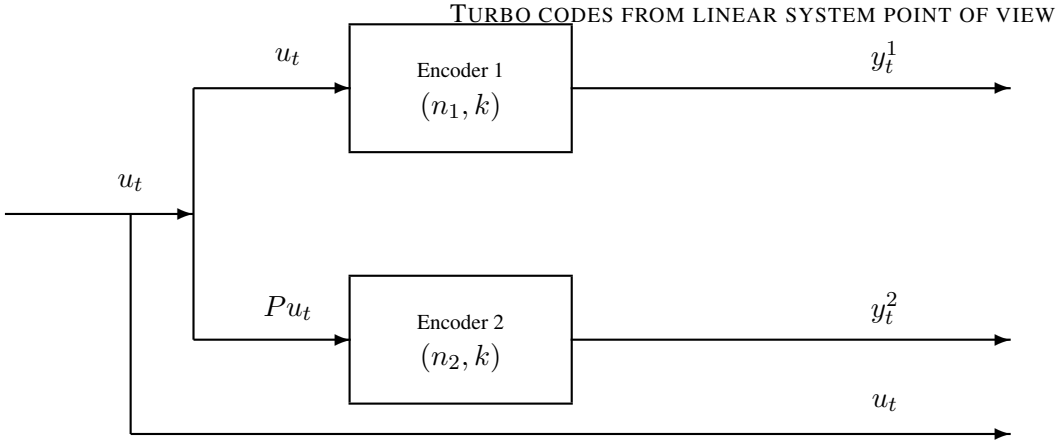


Figure 1: Turbo Code

$$\mathcal{C} = \{v(z) \in \mathbb{F}^n[z] \ : \ v(z) = G(z)u(z), \quad \text{with} \quad u(z) \in \mathbb{F}^k[z]\}$$

where $G(z)$ is a *generator matrix* of $\mathcal{C}$. We say that $\mathcal{C}$ has *rate* k/n if k is the rank of the module $\mathcal{C}$. The *free distance* of a convolutional code is given by

$$d_{free}(\mathcal{C}) = \min\{\mathrm{wt}(v(z)) \ : \ v(z) \in \mathcal{C}, \quad \text{with} \quad v(z) \neq 0\},$$

where wt denotes the Hamming weight of a codeword. Another important parameter of a convolutional code is the *degree* or *complexity*, which is defined as the largest degree δ of the $k \times k$ full size minors of any generator matrix $G(z)$.

$$\begin{array}{rcl} x_{t+1} & = & Ax_t + Bu_t, \\ y_t & = & Cx_t + Du_t, \\ v_t & = & \begin{pmatrix} y_t \\ u_t \end{pmatrix}, \qquad x_0 = 0, \end{array} \tag{1}$$

where for each instant t, $x_t \in \mathbb{F}^\delta$ is the *state vector*. The set of all codeword sequences $v_t \in \mathbb{F}^n$ is the convolutional code $\mathcal{C}$ and we say that $\mathcal{C}$ is generated by (A, B, C, D). By abuse of notation we denote it by $\mathcal{C}(A, B, C, D)$. In systems literature, representation (1) is known as the input-state-output representation. The integer δ describes the McMillan degree of the linear system (1). That is, the McMillan degree is equal to the dimension of the state-space realization of a rational and systematic convolutional encoder.

3 Turbo Code

Let $\mathcal{C}_1$ and $\mathcal{C}_2$ two convolutional codes of rate k/n_1 and k/n_2, respectively. In the turbo code $\mathcal{TC}^{(1)}$, the first encoder, $\mathcal{C}_1$, operates directly on the input information and the second one, $\mathcal{C}_2$, encodes the interleaved input information, denoted by Pu_t. Thus the codeword of the turbo code consists of the parity vectors of both encoders following by the information vector.

Next theorem shows the input-state-output representation for the Turbo code $\mathcal{TC}^{(1)}$ from the input-state-output representation of the constituent encoders.

 ISBN 13: 978-84-613-5510-5

Theorem 1 *Let $\mathcal{C}_1(A_1, B_1, C_1, D_1)$ be a (n_1, k, δ_1)-encoder. Let $\mathcal{C}_2(A_2, B_2, C_2, D_2)$ be a (n_2, k, δ_2)-encoder. Then the input-state-output representation for the $(n_1 + n_2 - k, k, \delta)$-turbo code is given by (1), where*

$$\begin{aligned} A &= \begin{pmatrix} A_1 & O \\ 0 & A_2 \end{pmatrix}, \quad B = \begin{pmatrix} B_1 \\ B_2P \end{pmatrix}, \\ C &= \begin{pmatrix} C_1 & O \\ O & C_2 \end{pmatrix}, \quad D = \begin{pmatrix} D_1 \\ D_2P \end{pmatrix}, \end{aligned} \tag{2}$$

Proof. Let u_t be the information vector of the turbo code. Let x_t^l and y_t^l be the state vector and the parity vector of the encoders $\mathcal{C}_l$, for $l = 1, 2$.

Now, from equation (1), we have, for $\mathcal{C}_1$,

$$\begin{aligned} x_{t+1}^1 &= A_1x_t^1 + B_1u_t \\ y_t^1 &= C_1x_t^1 + D_1u_t \end{aligned}$$

and for $\mathcal{C}_2$,

$$\begin{aligned} x_{t+1}^2 &= A_2x_t^2 + B_2Pu_t \\ y_t^2 &= C_2x_t^2 + D_2Pu_t \end{aligned}$$

Since the state space of the turbo code $\mathcal{TC}^{(1)}$ is the union of the state spaces of the constituent encoders, that is, $x_t = \begin{pmatrix} x_t^2 \\ x_t^1 \end{pmatrix}$, we obtain:

$$x_t = \begin{pmatrix} x_t^2 \\ x_t^1 \end{pmatrix} = \begin{pmatrix} A_1 & O \\ O & A_2 \end{pmatrix} x_t + \begin{pmatrix} B_1 \\ B_2P \end{pmatrix} u_t. \tag{3}$$

Now, the parity vector of the turbo code is $y_t = \begin{pmatrix} y_t^1 \\ y_t^2 \end{pmatrix}$, so

$$y_t = \begin{pmatrix} y_t^1 \\ y_t^2 \end{pmatrix} = \begin{pmatrix} C_1 & O \\ O & C_2 \end{pmatrix} x_t + \begin{pmatrix} D_1 \\ D_2P \end{pmatrix} u_t, \tag{4}$$

where $x_t = \begin{pmatrix} x_t^2 \\ x_t^1 \end{pmatrix}$.

Finally, from relations (3) and (4), we get the input-state-output representation of the Turbo code.

□

Our gain now is to obtain an observable Turbo code with minimal input-state-output representation. The following theorem gives the conditions for this.

Theorem 2 *Let $\mathcal{C}_1(A_1, B_1, C_1, D_1)$ be an (n_1, k, δ_1)-code and let $\mathcal{C}_2(A_2, B_2, C_2, D_2)$ be an (n_2, k, δ_2)-code. Let $\mathcal{TC}^{(1)}(A, B, C, D)$ be the Turbo code described by (2). Assume that the following conditions hold*

1. *rank$(B) = \delta_1 + \delta_2$.*
2. *The pair (A_l, C_l) is observable for $l = 1, 2$.*

 ISBN 13: 978-84-613-5510-5

Then (A, B, C, D) is a minimal representation with complexity $\delta = \delta_1 + \delta_2$. Furthermore, $\mathcal{TC}^{(1)}$ is an observable convolutional code.

Proof. From condition 1, we have

$$\operatorname{rank}\begin{pmatrix} A - zI & B \end{pmatrix} = \delta_1 + \delta_2 \qquad \text{for all } z \in \overline{\mathbb{F}}.$$

So, the pair (A, B) is controllable and consequently (A, B, C, D) is a minimal representation of $\mathcal{TC}^{(1)}$.

Now, for all $z \in \overline{\mathbb{F}}$,

$$\operatorname{rank}\begin{pmatrix} A - zI \\ C \end{pmatrix} = \operatorname{rank}\begin{pmatrix} A1 - zI_{\delta_1} & O \\ O & A_2 - zI_{\delta_2} \\ C_1 & O \\ O & C_2 \end{pmatrix} = \delta_1 + \delta_2$$

since $\operatorname{rank}\begin{pmatrix} A_l - zI_{\delta_l} \\ C_l \end{pmatrix} = \delta_l$, for $l = 1, 2$, from condition 2. So the turbo code $\mathcal{TC}^{(1)}$ is an observable convolutional code. □

Acknowledgements

This work has been partially supported by spanish grant MTM2008-06674-C02-02

References

[1] C. Berrou, A. Glavieus and P. Thitimajshima. *Near Shannon limit error-correcting coding and decoding: turbo codes (1)*. International Conference on Communication. 1064-1070 IEEE, Geneva, Switzerland, 1993.

[2] J. Climent, V. Herranz, and C. Perea *A first approximation of concatenated convolutional codes from linear systems theory viewpoint*, Linear Algebra and its Applications, 425: 673-699 (2007).

[3] J.-J. Climent, V. Herranz and C. Perea. *Linear system modelization of concatenated block and convolutional codes*. Linear Algebra and its Applications, 429: 1191-1212 (2008).

[4] J. Rosenthal. *Connections between linear systems and convolutional codes*. In B. Marcus and J. Rosenthal editors, "Codes, Systems and Graphical Models", IMA 123,39-66, Springer Verlag, 2000.

[5] J. Rosenthal, J.M. Schumacher and E.V. York. *On behaviors and convolutional codes*. IEEE Transactions on Information Theory, 42(6) 1881-1891, 1996.

[6] J. Rosenthal and R. Smarandache. *Construction of Convolutional Codes using Methods from Linear Systems Theory* Proceedings of the 35th Allerton Conference on Communication, Control and Computing, 953-960, 1997.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

An algebraic description of correlation attacks

S. D. Cardell[1], G. Maze[2], J. Rosenthal[2] and U. Wagner[2]

[1] *Departament d'Estadística i Investigació Operativa, Universitat d'Alacant*

[2] *Institut für Mathematik, Universität Zürich*

emails: s.diaz@ual.es, gerard.maze@math.uzh.ch, rosenthal@math.uzh.ch, urs.wagner@math.uzh.ch

Abstract

We discuss correlation attacks in a setting that extends classical LFSR, showing the relation with a coding theory problem and types of autonomous behavior that shoud be avoided because of existing fast correlation attacks.

Key words: stream cipher, autonomous system, correlation attack

Additive stream ciphers are an important class of stream ciphers. Hereby a pseudorandom bitstream is XORed to the message. The generation of the bitstream is often based on *linear feedback shift registers* (LFSRs). An LFSR stream $r = (r_i)_{i\geq 0}$ satisfies a linear recurrence relation of the form

$$c_0 r_t + c_1 r_{t+1} + c_2 r_{t+2} + \cdots + c_{l-1} r_{t+l-1} + r_{t+l} = 0, t \geq 0.$$

The sequence r is completely determined by its l initial values $(r_0, \ldots, r_{l-1})^T =: R_0$, which in most cases equals the secret key of the corresponding stream cipher.

An LFSR can be described by the linear system:

$$R_t = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ c_0 & c_1 & c_2 & \ldots & c_{l-1} \end{pmatrix} R_{t-1}$$

$$r_t = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \end{pmatrix} R_t$$

It is clear that the output of such a system cannot directly be used as keystream, since the linearity of the system allows to recover the initial state (key). Hence, the stream generated by this linear system is fed into a nonlinear function f to destroy linearity. Hence recovery of the initial state in principle comes down to solving systems

 ISBN 13: 978-84-613-5510-5

of nonlinear equations, which is known to be hard. However often f is not correlation immune, i.e. there exists a linear sequence $\{s_i\}_{i\geq 0}$ such that $\Pr(f(x_{1+i},\ldots,x_{l+i}) = s_i) = \frac{1}{2} + \epsilon$, with $|\epsilon| > 0$. Now it can be shown that the observation of

$$N \approx \frac{1}{\epsilon^2}$$

streambits theoretically allows the recovery of the initial state.

We would like to generalize this result to general autonomous systems in $\mathbb{F}_q$, of the form:

$$\begin{aligned} x_{t+1} &= Ax_t \\ y_t &= Cx_t \end{aligned}$$

Let $\sigma : (\mathbb{F}_q^n)^{\mathbb{Z}} \to (\mathbb{F}_q^n)^{\mathbb{Z}}$, $s_t \to s_{t+1}$ be the shift operator. Given an $n \times n$ polynomial matrix $P \in \mathbb{F}_q[z]^{n\times n}$, P defines an autonomous behavior in the sense of Willems [5]:

$$\mathcal{B} := \ker P(\sigma) \subset (\mathbb{F}_q^n)^{\mathbb{Z}}$$

Assume $\tilde{P} := U(z)P(z)$, with $U(z)$ an $n \times n$ unimodular matrix, has row degrees $\delta_1 \geq \ldots \geq \delta_n$ and high order coefficient matrix I_n. Let $X(z)$ be the $n \times r$ basis matrix with $r := \sum_{i=0}^{n} \delta_i$. Then $\ker P(\sigma) = \ker \tilde{P}(\sigma)$ and there exist an $r \times r$ matrix A and an $n \times r$ matrix C such that:

$$\ker(X(z)|\tilde{P}(z)) = \operatorname{im} \begin{pmatrix} zI_r - A \\ C \end{pmatrix}$$

The autonomous behavior of P is equivalently described by a system as above. Moreover, the matrices A and C can be computed 'by Inspection' [3].

Let $(y_i)_{i\geq 0}$ be a sequence in some behavior $\ker P(\sigma)$. This sequence can be also computed as:

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{pmatrix} x_1$$

Given a highly noisy sequence $\tilde{y}_1, \ldots, \tilde{y}_{N-1}$, how can we obtain the initial state (key)? We face a decoding problem.

In this paper we explain correlation attacks in this general setting and we show types of autonomous behavior which should be avoided because of existing fast correlation attacks.

Acknowledgements

This work was supported in part by Swiss National Science Foundation under grant number 200020-126948. The work of S. D. Cardell was supported by a grant for research students from the Generalitat Valenciana with reference BFPI/2008/138, by Spanish grant MTM2008-06674-C02-01 and by grant ACOMP/2010/039 of the Generalitat Valenciana.

 ISBN 13: 978-84-613-5510-5

References

[1] V. Chepyzhov and B. Smeets, *On a fast correlation attack on stream ciphers*, Lecture Notes in Computer Science, vol. 547, Springer Verlag, Berlin, 1991, EUROCRYPT 91, pp. 176-185.

[2] W. Meier and O. Staffelbach, *Fast correlation attacks on stream ciphers*, Lecture Notes in Computer Science, vol. 330, Springer Verlag, Berlin, 1988, EUROCRYPT 88, pp. 301-316.

[3] J. Rosenthal and J. M. Schumacher, *Realization by inspection*, IEEE Trans. Automat. Contr. **AC-42** (1997), no. 9, 1257-1263.

[4] U. Wagner, *Detection and exploition of small correlations in stream ciphers*, Master's thesis, University of Zürich, Zürich, 2008.

[5] J. C. Willems, *Models for dynamics*, Dynamics Reported (U. Kirchgraber and H. O. Walther, eds.), vol. 2, John Wiley and Sons Ltd, 1989, pp. 171-269.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Improving communication tasks with heterogeneous architectures including network processors

Pablo Cascón[1], Julio Ortega[1], Yan Luo[2], Antonio Díaz[1] and Ignacio Rojas[1]

[1] *Dept. of Computer Architecture and Technology, University of Granada*

[2] *Dept. of Electrical and Computer Engineering, University of Massachusetts Lowell*

emails: pcascon@atc.ugr.es, julio@atc.ugr.es, yan_luo@uml.edu, afdiaz@atc.ugr.es, ignacio@atc.ugr.es

Abstract

In this paper we study the possibilities that the parallelism implemented by network processors offers to accelerate the network interface, and thus, to improve the performance of applications that need communication (nowadays almost all applications). To achieve adequate communication performance levels efficient parallel processing of network tasks and interfaces should be considered. This way, we considered the use of network processors as heterogeneous microarchitectures with several cores, that implement multithreading and are suited for packet processing, to investigate on the use of parallel processing to accelerate the network interface and thus the network applications developed above it. More specifically, we have implemented an intrusion prevention system (IPS) and an OpenFlow switch in an heterogenous node that includes such a network processor. We describe the IPS we have developed that after its offloaded implementation allows faster packet processing of both normal and corrupted traffic. We also describe a complementary design to previous OpenFlow reference designs that takes advantage of the parallel processing allowed by network processors.

Key words: network processors, heterogeneous processors

1 Introduction

The availability of high bandwidth links and the scale up of network I/O bandwidths to multiple gigabits per second have shifted the communication bottleneck towards the network nodes. Therefore, the network interface (NI) performance is getting decisive in the overall communication path performance and it is determinant to reduce the communication protocol overhead due to context switching, multiple data copies, and interrupt mechanisms.

 ISBN 13: 978-84-613-5510-5

Network processors (NP) are programmable circuits that provide fast and flexible resources for high-speed communication functions processing as they are composed of multiple cores that makes it possible to take advantage of parallel processing in these tasks and to leave free clock cycles for the CPU to process the applications. Network processors (NP) usually have heterogeneous microarchitectures with cores optimized for packet processing.

NP-based acceleration cards such as [12] have been designed to change the way in which packets are handled in network systems by offloading packet processing from host CPU level to Network Interface Card (NIC) level.

These network processor cores usually have multithreading capabilities that can be used to improve the performance of network applications by offloading parts of them to the NP in order to take advantage of the parallelism and the memory hierarchy implemented at the NP microarchitecture. This way, not only the networking applications can be accelerated thanks to their implementation in the proximity to the network but also the CPU load is reduced and the free cycles can be allocated to other tasks thus improving the server throughput.

In this same research line, the paper [22] describes how a content-aware switch implemented in a NP can reduce the latency for HTTP processing, improve the packet throughput, and optimize the cluster server architectures, by processing requests and distributing them to the server according to the application-level information. The authors have used the IXP2400 NP [1] and compare their NP-based switch with a Linux-based one. The latency on the NP-based switch is reduced between 83% (for small file sizes) and 89% (at 1024 Kbytes sizes) and the throughput is improved between 5.7 and 2.2 times.

In this work we discuss the protocol offloading approach as an optimization of the communication subsystem. We also propose and analyze an offloaded implementation of a network intrusion prevention system and a converged architecture to accelerate OpenFlow switching using network processors.

The demand of firewalls and network intrusion detection and prevention systems has grown with the increasing importance of network services and infrastructure along with the difficulty of designing end-system security strategies [21]. Another application that can be improved by using network processors is OpenFlow switching, which enables flexible management of enterprise network switches and to accomplish experimental work on regular network traffic. We also apply network processor based acceleration cards to perform OpenFlow switching.

In Section 2 the protocol offloading optimization for the network interface is explained. Later, in Section 3 our network interface using network processors is detailed. It is used as the base for the applications described in Section 4. After that in Section 5 the main conclusions of this research work are provided.

 ISBN 13: 978-84-613-5510-5

2 Protocol Offloading

The protocol offloading moves part of the protocol processing to the NIC, leaving more cycles to be used by the CPU for the application computational needs. With this approach a processor located in the NIC (different from the CPU) is used for the communication tasks.

Basically, in this work we analyze the effect of offloading in NP-based NICs. The potential benefits of protocol offloading are an increase in the CPU cycles available for the application, a latency reduction, an increase in DMA efficiency (as small messages can be gathered into groups) and a reduction in the number of I/O collisions.

Nevertheless there are papers such as [10, 16, 4, 14] that defend that protocol offloading, specially in the TCP case, does not give any benefit for the application performance. The difficulties to achieve some benefits for the application performance come from the implementation, maintenance and *test* of the offloaded protocols [10]. Moreover the protocol between the NIC and the host CPU can have the same level of complexity as the protocol being offloaded [14]. Moreover, as a consequence of Amdahl's law, to offload the protocol processing to processors that are slower than the CPU will not give a better performance [16].

Nevertheless, the reasons detailed in these papers have to be balanced with the possible benefits from protocol offloading listed above. This way, in [20] the authors emulate a NIC connected to the I/O bus and controlled by one the processors in a SMP. The improvements are between a 600% and 900% with the TCP offloaded emulation. Moreover, the models proposed in [5] [17] exposes the possible benefits from protocol offload. It can be also taken into account that, although a processor runs at lower clock frequencies, it is possible to take advantage from its microarchitecture. For example, in the case of networking processors, although they have lower clock frequency than the CPU, they provide resources that allows us to take advantage of multithreading processing of packets, asynchronous memory accesses, multilevel memory hierarchy and parallelism and multithreading.

It is important to take into account that the actual processors provide many cores of processing (multicore) both for the host CPU and for the network processor at the NIC. Every core might have multiple threads. This high parallelism motivates to change the techniques used for offloading as well as the placement of the different communication tasks.

The study provided in [18] shows that parallelizing the reception path can deliver benefits for unidirectional as well as bidirectional traffic. In fact, this scheme allows the authors to reach the theoretical throughput of the medium. Therefore, offloading can be improved by using parallelism.

Another approach to take advantage of parallel processing in protocol offloading is the use of network processors. In [8] communication tasks are distributed between the central processor (CPU) and a network processor (NP) included in the network card. This way, the NP executes part or all of the communication tasks, and even part or the whole application. The NP and the CPU cooperates in the application execution that runs in the node and communicates by means of the buses existing on the node.

 ISBN 13: 978-84-613-5510-5

Through those buses the data among the main memory and the memory included in the network card (i.e. local memory of the NP) is also transferred. Therefore, even if it is called offloading due that part of the code that use to be run in the CPU is moved to a processor that is in a NIC connected to a I/O bus, actually it tries to implement a distribution of the tasks to be completed by the node according to the location of the NP close to the network, and its micro-architecture oriented to the packet processing.

3 A network interface using network processors

Because of the importance of the network interface (NI) in the communication performance we have decided to create a network interface using network processors. This network interface uses a NP-based NIC and allows us to have the base to offload communications tasks to this NP.

The network processors (NP) are programmable circuits optimized for the processing of communication functions at high speed. It plays a very important role in the design of current routers. Moreover, these processors also include hardware components to accelerate common operations in communication functions (CRC processing, hash calculation, etc.) [15, 19]. They are located midway between the ASICs (more speed but less flexibility) and a general purpose processor. Usually, they are compound of a central processor for control and several packet processors (usually multithreaded processors).

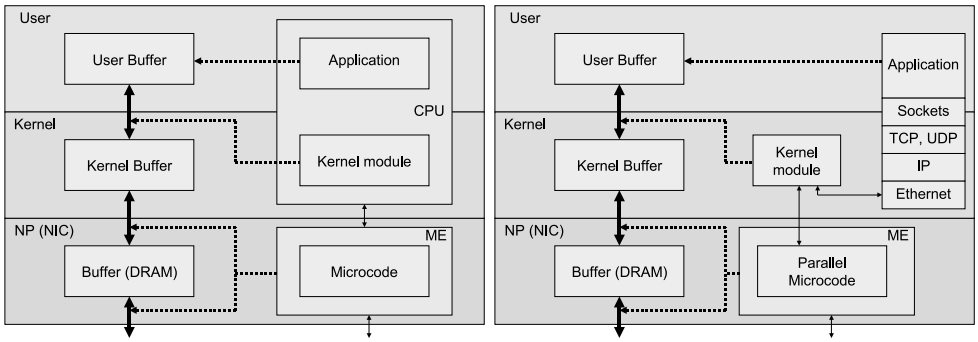
Families of network processors have been marketed with diverse flexibility characteristics, prices and performance [2]. Among the different alternatives, we have chosen Intel IXP processors [7]. The IXP series implement micro-architectures with a high level of parallelism, including several programmable processors: a general-purpose Intel XScale processor (RISC architecture compatible with the ARM architecture), and up to 16 optimized co-processors (called MicroEngines or MEs) for packet processing.

This network interface is created as a platform to offload part or all of the protocol processing from the host CPU to the MicroEngines. The network interface is a Linux kernel device driver as explained in detail in our work [3].It is configured to use the communication API host-MicroEngines, our card provides a complete network interface as shown in Figure 1. This figure compares the software distribution of the interface directly available with the card (Figure 1(a)) and the elements of the network interface developed by us (Figure 1(b)). In these two figures the arrows with dotted lines indicate the need for synchronization between the corresponding modules; the bold arrows indicate the data transfers; and the thinnest arrows show the interactions between the modules that control each transfer. In Figure 1(b) the bold rectangles indicate the software modules developed by us.

As it has been said, this network interface is used as a platform to study the offloading of communication tasks.

Figure 2 provides measures corresponding to the parallelism that can be obtained by devoting more MicroEngines to a specific communication task. In our experimental configuration we have considered that one MicroEngine is dedicated to packet trans-

 ISBN 13: 978-84-613-5510-5



(a) Base software distribution [11]. (b) Network interface.

Figure 1: Complete offloaded network interface.

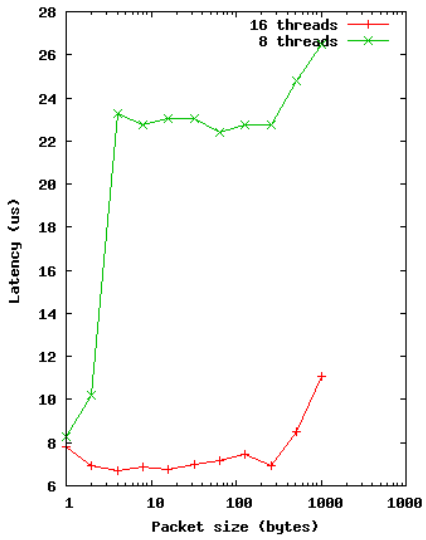


Figure 2: IP latency.

mission, other MicroEngine to packet reception, and other two MicroEngines to the PCI Express bus transfers. Nevertheless, the best results have been obtained using two MicroEngines (up to 16 threads) for transmission and other two MicroEngines for reception.

4 Applications and experimental results

In the next subsections two applications developed in top of our network interface are explained and evaluated.

 ISBN 13: 978-84-613-5510-5

4.1 OpenFlow

OpenFlow tries to address the needs of researchers to control and experiment with production networks and not only with simulators or special networks. OpenFlow was started in Stanford by [9] and soon was followed by many other Universities. Also the industry has developed some OpenFlow devices [13].

In a regular switch both the data and control plane are done in the same physical device. The OpenFlow Switch initiative separates these two paths. In an NP usually the data plane runs at the MicroEngines and the control plane at the XScale main core. With the OpenFlow idea, the data plane stays at the switch while the control plane is implemented in an external entity called the *controller*. Translating the OpenFlow idea to the IXP architecture means that the MicroEngines run the data plane while the control plane is in another computer, usually external to the switch.

An OpenFlow switch can easily help to *share* the network into several pieces, each for every kind of flow. This was done previously by using Ethernet VLAN codes for every different subnet. The main objective that can be achieved with OpenFlow is to manage in an easy way a complete network. A flow is determined by ten fields that can be found in packets. These fields range from Layers 1 to 4 in the OSI model, or from the physical layer to transport layer.

Once a specific flow is detected an OpenFlow switch has to determine which action to take on this flow. The switch needs a *Flow table* where a list of *flows* and actions to take on the packets belonging to a flow is stored. The typical actions are to forward packets to a specific output port, to drop the packet, or to forward to the *controller*. The *controller* in an OpenFlow switch is the external entity that runs the control plane. It decides the flows and associated actions of the switch. In case a new packet arrives to the switch and it does not belong to any flow in the *Flow table* as created by the *controller*, the switch will encapsulate the new packet into a message and send it to the *controller*, the *controller* will decide what to do and inserts a flow-action row into the *Flow table*.

The baseline OpenFlow design is briefly described here. As it uses a Linux based PC equipped with several regular NICs (or at least one dual port NIC) running the OpenFlow software we will call it the software reference design. In Figure 3 the architecture is illustrated. Even if this reference design consists of two different sub-designs, (one for user-level and one kernel-level) the architecture is the same. The switch creates a *secure channel*[1] to the OpenFlow *controller* through an out-of-band connection, i.e., a dedicated NIC port is allocated for communication with the *controller*. The flow table is maintained in the host memory, and the host CPU looks up flows entries, modifying them upon receiving the OpenFlow control packets from the *controller*. The packets received by the switch are forwarded to either the *controller* or the destination port, depending on the result of the lookup.

In Figure 4 the details of our OpenFlow switch accelerated with a network processor are shown. The Flow table resides at the host memory and at the NP memory and both are synchronized. As in the case with the FPGA the first packet is send to the host,

[1] A communication channel using cryptography

 ISBN 13: 978-84-613-5510-5

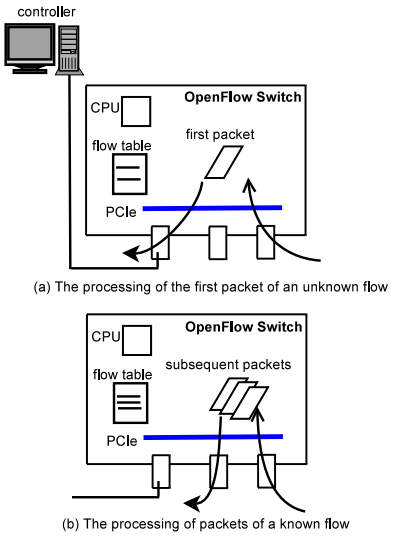


Figure 3: PC-based OpenFlow switch

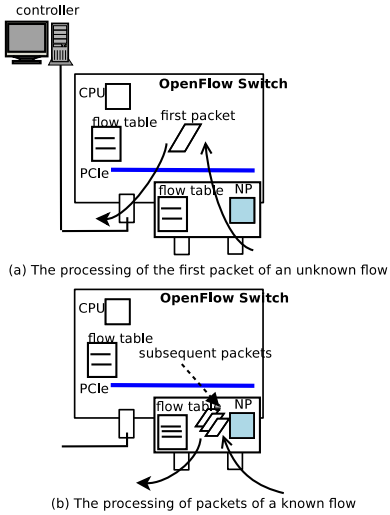


Figure 4: NP-based OpenFlow switch

that in turn will query the *controller* about what to do, and the rest of the packets in the flow will be processed at the NP directly.

The results for packet forwarding throughput are shown in Figure 5 with the differences for the three different architectural options: user-level (x86), kernel-level (x86-kernel) and NP accelerated (NFD).

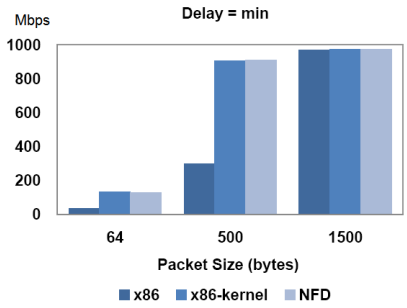


Figure 5: Packet forwarding throughput. (Inter-packet delay = minimal)

First of all, the kernel-level switch and NFD-based switch perform consistently better than the user-level switch when dealing small (64B) or medium (500B) packets. This result can be explained by the data copying from kernel to user space in user-level switch software. Such overheads (buffer allocation/deallocation, copying, interrupt) are more significant for small packets since they occur on a per packet basis. Second, the forwarding rates are more or less the same for all the explained three designs when the packet is of maximal size. The data-copying overhead from kernel to user space is amortized in the case of large packets. The question is why the NP does not out

 ISBN 13: 978-84-613-5510-5

perform the other two setups since it does not transfer packets to host CPU via PCIe bus. There are several reasons. The network links used in the experiments are *only* 1 Gbps, which might not be enough to show any difference between the processing power of the CPU and the NP.

We also analyze the round trip time reported by *ping* on the measurement node. This number reflects the delay incurring at the OpenFlow switch of a packet. In this set of experiments we compare the packet delay in three scenarios: OpenFlow with regular NIC, OpenFlow with NP-enabled virtual NICs where the flow tables still reside in the host server, and OpenFlow with NP-accelerated flow table manipulation. The results show that NP based OpenFlow switch can reduce the packet delay by up to 35% (from 0.157ms to 0.102ms).

4.2 Intrusion prevention system

An Intrusion Prevention System (IPS) is a system that prevents the network attacks. An IPS needs to analyze the headers and the content of the packet at higher-level protocols to detect undesired behavior. It is also required that the function implemented by the intrusion prevention system to be updated with new detection procedures due to the evolving characteristics of the attacks. As this application requires high-performance processing capabilities and flexibility it is a good candidate to be implemented in a network processor.

In our research work we focus on monitoring the network. The most common setup for IPS is to monitor all the network traffic entering and exiting the network of an organization by placing a computer running an IPS software at the main Internet connection of the organization. Packets coming from the Internet enter the organization through this computer that process them before eventually let the packets reach the organization network and systems. Usually packets are received into this computer through a regular NIC, processed at the CPU to decide whether to stop the packet or let it reach the computer where is destined through another regular NIC. This approach has the disadvantage that all packets have to reach and be processed at the CPU. The processing can be very CPU resources consuming, ant therefore if there are too many packets force the IPS software to discard some packets that will affect the computers of the organization.

Our approach is to move the IPS, partial or completely, from the host CPU to the network processor (NP) at the NIC by using our network interface (Section 3. Thanks to this network interface we can control where to place the different parts of the IPS and evaluate how it affects the overall performance of applications using the network. It is possible to place the IPS at the MicroEngines or at the host. Using Figure 1(b) the IPS could be either at "Parallel microcode" box or at the "Application" box. When running in the microcode at MEs it will be placed to check every received packet that it is to be sent to the host. The corrupted traffic will be stopped and the legitimate traffic will follow the path to the host. In the case the IPS is executed by the host CPU, it will receive all packets, including the normal and corrupted traffic. The closer position to the network and the specialized hardware of the MicroEngines makes it the

 ISBN 13: 978-84-613-5510-5

candidate alternative to give better results. This will be checked in the experimental results section. Moreover, in this chapter, it will be studied how the location of our IPS affects the processing of corrupted traffic with respect to the legitimate one.

The microcode for this IPS is based in the one used for the network interface. One MicroEngine is devoted to reception of packets, another one to transmission (with the help of two others), two MicroEngines to communicate with the host through the PCIe bus, and one MicroEngine for processing. It is this MicroEngine that is modified to act as an IPS and drop packets that matches some rules and not let them reach the CPU. As it is explained for the network interface there are not shared data structures for this IPS since the rules are *written* in the code (i.e. new rules require the modification of the code) and not in a memory structure shared by all the threads (eight in our prototype). With only one MicroEngine used for the processing task it can handle the traffic injected.

The obtained results with the intrusion prevention system test have been very successful. In our experiment both corrupted and legitimate traffic are sent through the MicroEngines to the host. In the first case the corrupted traffic is dropped at the MicroEngines while in the second one the activity of the IPS is done in the host. The corrupted traffic matches a set of snort rules, implemented in both the host and the ME, while the legitimate traffic is the one used in the communications benchmark *Netgauge* [6]. We measure how the processing associated to the detection of the corrupted traffic affects the performance of this benchmark. The legitimate traffic is *raw* Ethernet because it can achieve a higher throughput than using TCP/UDP. The corrupted traffic is an HTTP header.

In Figure 6 a latency comparison among two setups is shown. The corrupted traffic rate goes from 100 Mbps to 1000 Mbps. When corrupted traffic is stopped at the MicroEngines the legitimate traffic latency is lower. This is, it is almost not affected by sharing the same communication path than the corrupted traffic. When the IPS is run at the host (snort) and the corrupted traffic rate goes up to 600 Mbps the latency behaves similarly to the case when the IPS runs at the network processor. But from 700 Mbps to the 1000 Mbps of corrupted traffic rate the IPS running at the host does not perform well in terms of latency and even the benchmark can not be completed due to timeouts. There is a significative higher latency when the IPS runs at the host compared to when it runs at the network processor. If the corrupted traffic rate is higher than 600 Mbps it makes the IPS to drop legitimate packets.

The same results are observed when comparing the performance, in terms of the throughput of the network benchmark, depending on the IPS placement (Figure 7). If the corrupted packets are stopped at the MicroEngines level, the CPU of the host can give a better service to the legitimate traffic. The benefit for this kind of traffic, whenever the IPS is located at the NIC, is that it is not affected after the 700 Mbps corrupted traffic rate limit.

Another experiment we have completed has been to evaluate the performance of the network benchmark when only legitimate traffic is sent to the host through the MicroEngines, compared with the alternative of sending normal and corrupted traffic to the host. As expected, when only legitimate traffic is sent, the performance is better,

 ISBN 13: 978-84-613-5510-5

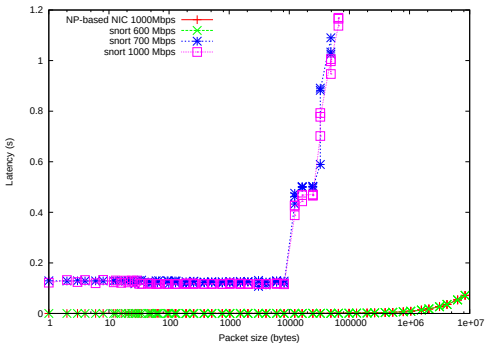


Figure 6: Latency (IPS at the NP vs. IPS at the host)

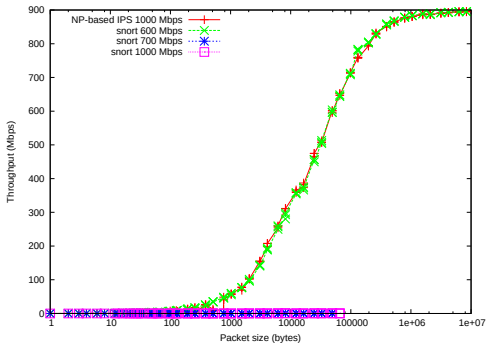


Figure 7: Throughput (IPS at the NP vs. IPS at the host)

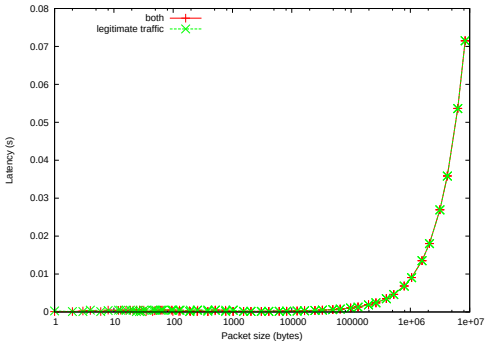


Figure 8: Latencia (IPS en el NP)

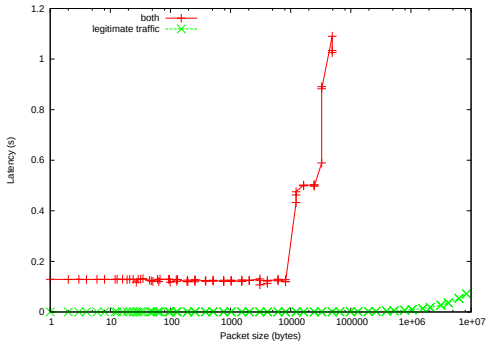


Figure 9: Latencia (IPS en el host)

in terms of both lower latency and higher throughput. Nevertheless, the difference is bigger when the IPS is run at the host CPU, instead of run at the MicroEngines. It can be concluded from Figure 8 where it is shown that the difference in throughput is not important compared with Figure 9. The processing of corrupted packets affects more significative to the processing of normal packets if it is done at the host than at the MicroEngines.

Thus the conclusions are clear for the IPS processing: running it on a network processor does not affect much the rest of the traffic. If it is implemented in the host, it takes a lot of cycles that the CPU can be available to be used by other functions (as processing of legitimate traffic or computation).

5 Conclusions

The IPS based on a multi-threaded network interface here proposed makes possible to take advantage of the parallelism implemented in network processors to improve not only the latency, but also the bandwidth of legitimate traffic that shares the same communication path with the corrupted traffic. The benefits from placing the IPS close to the network, by using specialized network processors, gives up to many times lower

 ISBN 13: 978-84-613-5510-5

latency and higher bandwidth available to the legitimate traffic. The analysis of the possible optimizations to the IPS processing, along with the evaluation of the effect of the improvements in real communication applications are the main tasks for our future work. With respect to OpenFlow switching, the NP implementation provides up to 35% of delay reduction. Precisely our future work in OpenFlow switching is to develop more actions for packets, and not only dropping or forwarding to reach more performance improvements.

Acknowledgements

This work has been supported by *Ministerio de Educación y Ciencia* project TIN2007-60587.

References

[1] Intel network processors. `http://www.intel.com/design/network/products/npfamily/`.

[2] L Byrne, J.; Gwennap. *A Guide to Network Processors*. The Linley Group, 2005.

[3] Pablo Cascón, Julio Ortega, Waseem M. Haider, Antonio F. Díaz, and Ignacio Rojas. A multi-threaded network interface using network processors. In *17th Euromicro February 2009*.

[4] David D. Clark, Van Jacobson, John Romkey, and Howard Salwen. An analysis of tcp processing overhead. *IEEE Communications Magazine*, 27:23–29, 1989.

[5] P. Gilfeather and A.B. Maccabe. Modeling protocol offload for message-oriented communication. In *Cluster Computing, 2005. IEEE International*, pages 1–10, 2005.

[6] T. Hoefler, T. Mehlan, A. Lumsdaine, and W. Rehm. Netgauge: A Network Performance Measurement Framework. In *High Performance Computing and Communications, Third International Conference, HPCC 2007 Proceedings*.

[7] Intel. Intel network processors. `http://www.intel.com/design/network/products/npfamily/`, 2009.

[8] Hyun-Wook Jin, Pavan Balaji, Chuck Yoo, Jin-Young Choi, and Dhabaleswar K. Panda. Exploiting NIC architectural support for enhancing IP-based protocols on high-performance networks. *J. Parallel Distrib. Comput.*, 65(11):1348–1365, 2005.

[9] N. Mckeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: Enabling innovations in college networks. OpenFlow Consortium, `http://www.openflowswitch.org`, 2008.

 ISBN 13: 978-84-613-5510-5

[10] Jeffrey C. Mogul. Tcp offload is a dumb idea whose time has come. In *HOTOS'03: Proceedings of the 9th conference on Hot Topics in Operating Systems*, pages 5–5, Berkeley, CA, USA, 2003. USENIX Association.

[11] Netronome. `http://www.netronome.com/files/file/tech-docs/NFD\_1.3\_Users'\_Guide.pdf`, 2006.

[12] Netronome. Product Brief - NFE-i8000 Network Acceleration Card, 2006. http://www.netronome.com/.

[13] Open Flow Consortium. Openflow blog. http://www.openflowswitch.org/wp/, 2009.

[14] M. ODell. Re: how bad an idea is this? TSV mailing list, November 2002.

[15] I. Papaefstathiou, N.A. Nikolaou, B. Doshi, and E. Grosse. Guest editors' introduction: Network processors for future High-End systems and applications. *Micro, IEEE*, 24(5):7–9, 2004.

[16] G. Regnier, S. Makineni, I. Illikkal, R. Iyer, D. Minturn, R. Huggahalli, Newell, L. Cline, and A. Foong. TCP onloading for data center servers. *Computer*, 37(11):48–58, 2004.

[17] Piyush Shivam and Jeffrey S. Chase. On the elusive benefits of protocol offload. In *Proceedings of the ACM SIGCOMM workshop on Network-I/O convergence: experience, lessons, implications*, pages 179–184, Karlsruhe, Germany, 2003. ACM.

[18] Piyush Shivam, Dhabaleswar Panda, and Pete Wyckoff. Can user-level protocols take advantage of multi-cpu nics? *Parallel and Distributed Processing Symposium, International*, 1:0008, 2002.

[19] L. Thiele, S. Chakraborty, M. Gries, and S. Knzli. *Design Space Exploration of Network Processor Architectures*. 2002.

[20] R. Westrelin, N. Fugier, E. Nordmark, K. Kunze, and E. Lemoine. Studying network protocol offload with emulation: approach and preliminary results. In *High Performance Interconnects, 2004. Proceedings. 12th Annual IEEE Symposium on*, pages 84–90, 2004.

[21] Konstantinos Xinidis, Kostas Anagnostakis, and Evangelos Markatos. *Design and Implementation of a High-Performance Network Intrusion Prevention System*. 2005.

[22] Li Zhao, Yan Luo, L.N. Bhuyan, and Ravi Iyer. A network Processor-Based, Content-Aware switch. *IEEE Micro*, 26(3), 2006.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Mehler–Heine type formulas for some Sobolev orthogonal polynomials

Laura Castaño–García[1] **and Juan J. Moreno–Balcázar**[1]

[1] *Departamento de Estadística y Matemática Aplicada, University of Almería*

emails: lcastano@ual.es, balcazar@ual.es

Abstract

In this survey we pay attention to some formulas for Sobolev orthogonal polynomials obtained in two recent papers. They are known as Mehler–Heine type formulas and allow us to obtain interesting consequences about the asymptotic behaviour of the corresponding zeros. We illustrate our results with numerical examples.

Key words: Sobolev orthogonal polynomials; Jacobi polynomials; Gegenbauer polynomials; zeros; asymptotics.
MSC 2000: 33C47, 42C05

1 Introduction

The issue of Sobolev orthogonal polynomials has been considered in the literature from many points of view. These polynomials are not orthogonal with respect to a standard inner product like $(p,q) = \int pqd\mu$ where μ is a positive measure with support on the real axis. As a consequence, the properties of the standard polynomials do not hold any more. Apart from other considerations, the fact is that Sobolev orthogonal polynomials are different from the standard ones and this motivates to study them deeply. Thus, it is natural to make an exhaustive study of their properties such as it has been done for the standard polynomials and is going on at present.

Here, we consider the following Sobolev inner product

$$(p,q)_S = \int_{-1}^{1} p(x)q(x)d\mu(x) + \int_{\mathrm{supp}(\nu)} p'(x)q'(x)d\nu(x), \tag{1}$$

where μ is a Gegenbauer or Jacobi measure and ν is a measure related to μ. Mehler–Heine formulas for the orthogonal polynomials with respect to μ are well known (see, for example, [5]). Our objective is to obtain the corresponding Mehler–Heine type formulas for the polynomials orthogonal with respect to (1). This is an expository paper and the results appearing here have been obtained recently in [1] and [3]. Therefore, we recommend to consult these articles to the reader interested in the analytic tools used. Here we introduce new numerical experiments to illustrate the results.

 ISBN 13: 978-84-613-5510-5

2 Notation and background

We consider the nonstandard inner product

$$(p,q)_S = \int_{-1}^{1} p(x)q(x)d\mu(x) + \int_{\text{supp}(\nu)} p'(x)q'(x)d\nu(x), \tag{2}$$

where μ is a Gegenbauer or Jacobi measure and ν is a measure related to μ. We also denote by $(G_n^{(\alpha)})$ the sequence of monic Gegenbauer polynomials when $d\mu(x) = (1-x^2)^{\alpha-\frac{1}{2}}dx$ and by $(P_n^{(\alpha,\beta)})$ the sequence of monic Jacobi polynomials when $d\mu(x) = (1-x)^\alpha(1+x)^\beta dx$. In fact, we consider two cases:

- Gegenbauer–Sobolev inner product:

$$(p,q)_S = \int_{-1}^{1} p(x)q(x)(1-x^2)^{\alpha-\frac{1}{2}}dx + \int_{\text{supp}(\nu)} p'(x)q'(x)d\nu(x), \tag{3}$$

  where

$$d\nu(x) = \frac{(\kappa_1+\kappa_2+q\,\kappa_1 x^2)(1-x^2)^{\alpha+\frac{1}{2}}}{1+qx^2}dx + \kappa_2 M^{(q)}\left(\delta\left(\frac{-1}{\sqrt{-q}}\right)+\delta\left(\frac{1}{\sqrt{-q}}\right)\right),$$

  with $\alpha > -1/2$, $q \geq -1$, $\kappa_1 \geq 0$, $\kappa_2 > 0$, and

$$M^{(q)} \begin{cases} = 0, & \text{if } \ q \geq 0, \\ \geq 0, & \text{if } \ -1 \leq q < 0. \end{cases}$$

- Jacobi–Sobolev inner product:

$$(p,q)_S = \int_{-1}^{1} p(x)q(x)(1-x)^\alpha(1+x)^\beta dx + \int_{-1}^{1} p'(x)q'(x)d\nu(x), \tag{4}$$

  where

$$d\nu(x) = \frac{\kappa(\kappa_1+\kappa_2)-\kappa_1 x}{\kappa - x}(1-x)^{\alpha+1}(1+x)^{\beta+1}dx + \kappa_2\kappa_3\delta(\kappa)\,,$$

  with $|\kappa| \geq 1$, $\kappa_2 \geq 0$, $\kappa_3 \geq 0$ and $\kappa_1 \geq -\dfrac{|\kappa|}{1+|\kappa|}\,\kappa_2$.

We can observe that in both inner products the measure ν is a rational modification of the type of measure μ with the addition of one or two mass points outside the support of μ.

We know (see [5]) the Mehler–Heine formulas for Gegenbauer and Jacobi orthogonal polynomials. We write them in the following Theorem.

Theorem 1 ([5]) *Let $(G_n^{(\alpha)})$ and $(P_n^{(\alpha,\beta)})$ be the sequences of monic Gegenbauer and Jacobi orthogonal polynomials, respectively. Then,*

$$\lim_{n\to\infty} \frac{2^n G_n^{(\alpha)}(\cos(x/n))}{n^\alpha} = \sqrt{\pi}(2x)^{\frac{1}{2}-\alpha}J_{\alpha-\frac{1}{2}}(x), \tag{5}$$

$$\lim_{n\to\infty} \frac{2^n P_n^{(\alpha,\beta)}(\cos(x/n))}{n^{\alpha+\frac{1}{2}}} = 2^{-\beta}\sqrt{\pi}x^{-\alpha}J_\alpha(x), \tag{6}$$

 ISBN 13: 978-84-613-5510-5

where J_α is the Bessel function of the first kind. Both limits hold uniformly on compact subsets of $\mathbb{C}$.

From the above theorem and using Hurwitz's Theorem in a straightforward way we obtain the asymptotic behaviour of the zeros of the corresponding orthogonal polynomials.

Let m be the number of positive zeros of Gegenbauer or Jacobi polynomials. We denote by $x_{n,i}^{(\alpha)}$ and $x_{n,i}^{(\alpha,\beta)}$ the positive zeros of Gegenbauer and Jacobi polynomials, respectively, ordered as

$$x_{n,m}^{(\alpha)} < x_{n,m-1}^{(\alpha)} < \ldots < x_{n,1}^{(\alpha)},$$

$$x_{n,m}^{(\alpha,\beta)} < x_{n,m-1}^{(\alpha,\beta)} < \ldots < x_{n,1}^{(\alpha,\beta)}.$$

We also denote by $0 < j_1^{(\alpha)} < j_2^{(\alpha)} < \cdots < j_m^{(\alpha)}$ the first m positive zeros of the Bessel function of the first kind J_α.

Corollary 1 *We have*

$$\begin{aligned} \lim_{n\to\infty} n \arccos(x_{n,i}^{(\alpha)}) &= j_i^{(\alpha-\frac{1}{2})}, \\ \lim_{n\to\infty} n \arccos(x_{n,i}^{(\alpha,\beta)}) &= j_i^{(\alpha)}. \end{aligned}$$

From now on, we will denote by (S_n^G) and (S_n^J) the sequence of orthogonal polynomials with respect to the Sobolev inner products (3) and (4), respectively. For the Gegenbauer–Sobolev case we consider the t zeros $s_{n,i}^G$ of S_n^G inside $(0,1)$ ordered as $s_{n,t}^G < s_{n,t-1}^G < \cdots < s_{n,2}^G < s_{n,1}^G$. In the same way, for the Jacobi–Sobolev case we consider the r zeros $s_{n,i}^J$ of S_n^J inside $(-1,1)$ ordered as $s_{n,r}^J < s_{n,r-1}^J < \cdots < s_{n,2}^J < s_{n,1}^J$.

More details about the zeros of these families of Sobolev orthogonal polynomials can be found in [3], [1], and the references therein.

Finally, notice that both Gegenbauer and Gegenbauer–Sobolev orthogonal polynomials are symmetric, that is, $G_n^{(\alpha)}(-x) = (1-)^n G_n^{(\alpha)}(x)$ and $S_n^G(-x) = (-1)^n S_n^G(x)$.

3 Mehler–Heine type formulas

In [1] and [3] we have found the following Mehler–Heine type formulas for the Sobolev polynomials considered here.

Theorem 2 *a) We have for $\alpha > -1/2$,*

$$\lim_{n\to\infty} \frac{2^n S_n^G(\cos(x/n))}{n^\alpha} = \frac{1+4b^{(q)}}{1+4a^{(q,\kappa_1,\kappa_2)}} \sqrt{\pi}(2x)^{\frac{1}{2}-\alpha} J_{\alpha-\frac{1}{2}}(x), \tag{7}$$

where

$$b^{(q)} = \begin{cases} \dfrac{1}{4}\Psi(q), & \text{if } q \geq -1 \text{ and } M^{(q)} = 0, \\ \dfrac{1}{4\Psi(q)}, & \text{if } -1 \leq q < 0 \text{ and } M^{(q)} > 0, \end{cases}$$

 ISBN 13: 978-84-613-5510-5

and

$$a^{(q,\kappa_1,\kappa_2)} = \frac{1}{4}\Psi\left(\frac{q\,\kappa_1}{\kappa_1+\kappa_2}\right),$$

where Ψ *is a real function defined by* $\Psi(x) = x/(1+\sqrt{1+x})^2$, *for* $x \geq -1$. *For* $x > -1$, $|\Psi(x)| < 1$.

b) We have for $\alpha, \beta > -1$ *and* $\kappa_1 \geq 0$,

$$\lim_{n\to\infty} \frac{2^n S_n^J\left(\cos(x/n)\right)}{n^{\alpha+\frac{1}{2}}} = \frac{1+2b(\kappa)}{1+2a(\tilde{\kappa})}\frac{\sqrt{\pi}}{2^\beta}x^{-\alpha}J_\alpha(x), \tag{8}$$

where

$$b(\kappa) = \begin{cases} -\dfrac{\varphi(\kappa)}{2}, & \text{if } \kappa_3 > 0, \\ -\dfrac{1}{2\,\varphi(\kappa)}, & \text{if } \kappa_3 = 0, \end{cases}$$

$$\tilde{\kappa} := \begin{cases} \dfrac{\kappa(\kappa_1+\kappa_2)}{\kappa_1}, & \text{if } \kappa_1 > 0, \\ +\infty, & \text{if } \kappa_1 = 0, \kappa \geq 1, \\ -\infty, & \text{if } \kappa_1 = 0, \kappa \leq -1, \end{cases}$$

and

$$a(\tilde{\kappa}) = -\frac{1}{2\,\varphi(\tilde{\kappa})},$$

being φ *the complex function*

$$\varphi(z) = z + \sqrt{z^2-1}, \quad \textit{for } z \in \mathbb{C}\setminus[-1,1],$$

with $\sqrt{z^2-1} > 0$ *when* $z > 1$, *and* $\varphi(+\infty) = +\infty$.

Both limits hold uniformly on compact subsets of the complex plane.

The proofs of these results can be found in [1] and [3]. Here, we only pay attention to the interpretation of these results. We summarize our conclusions in the following items:

- Except for a constant in each case, formulas (7) and (8) are equal to (5) and (6), respectively. These constants are $\dfrac{1+4b^{(q)}}{1+4a^{(q,\kappa_1,\kappa_2)}}$ in the Gegenbauer case and $\dfrac{1+2b(\kappa)}{1+2a(\tilde{\kappa})}$ in the Jacobi case. If these constants are nonzero, we can say that this type of asymptotic behaviour for the standard and nonstandard polynomials is the same. But, *what happens if the constants are zero?* In this case the previous theorem does not provide any asymptotic information since the value of the limits in (7) and (8) is 0. It very easy to deduce that this situation occurs when

  (a) $b^{(q)} = -1/4$, in the Gegenbauer case. That implies that $q = -1$.

 ISBN 13: 978-84-613-5510-5

(b) $b(\kappa) = -1/2$, in the Jacobi case. That implies that $\kappa = 1$.

Thus, for $q = -1$ and $\kappa = 1$, and taking into account the restrictions on the parameters described in Section 2, the corresponding measures ν in the Sobolev inner products are:

(a) Gegenbauer case.

$$d\nu(x) = (\kappa_1 + \kappa_2 - \kappa_1 x^2)(1-x^2)^{\alpha-\frac{1}{2}}dx + \kappa_2 M^{(-1)}\left(\delta(-1) + \delta(1)\right).$$

(b) Jacobi case.

$$d\nu(x) = (\kappa_1 + \kappa_2 - \kappa_1 x)(1-x)^{\alpha}(1+x)^{\beta+1}dx + \kappa_2\kappa_3\delta(1).$$

Thus, additional efforts should be made to obtain the asymptotic behaviour of the Sobolev orthogonal polynomials in these special cases. Notice that they occur when we put the mass (masses in the Gengebauer case) at the point 1 (at the points -1 and 1 in the Gengebauer case), i.e., the mass or masses are located in the extremes of the support of the classical measure.

- The techniques used to prove this theorem are analytic. They need previous results obtained in other papers (see the references in [1] and [3]) and additionally we obtain other Mehler–Heine type formulas for other standard polynomials related to Gegenbauer or Jacobi polynomials.

3.1 Mehler–Heine type formulas for the special cases

The tools used to obtain the asymptotic results for the special cases mentioned above are more difficult technically than in the general case. The details can be found in [1] and [3].

Theorem 3 *Let us define* $g_\alpha(x) := (2x)^{-\alpha}J_\alpha(x)$.

a) Then, for $q = -1$ *and* $\alpha > 1/2$, *it holds*

$$\lim_{n\to\infty}\frac{2^n S_n^G\left(\cos(x/n)\right)}{n^{\alpha-1}} = \begin{cases} -\frac{\sqrt{\pi}}{1+4a^{(-1,\kappa_1,\kappa_2)}}\left((2x)^2 g_{\alpha+\frac{1}{2}}(x) + 4g_{\alpha-\frac{1}{2}}(x)\right), & M^{(-1)} > 0, \\ \frac{\sqrt{\pi}}{1+4a^{(-1,\kappa_1,\kappa_2)}}\, g_{\alpha-\frac{3}{2}}(x), & M^{(-1)} = 0. \end{cases}$$

b) For $\alpha > 0$, $\beta > -1$ *and* $\kappa_1 \geq 0$, *it holds*

$$\lim_{n\to\infty}\frac{2^n S_n^J\left(\cos(x/n)\right)}{n^{\alpha-\frac{1}{2}}} = \begin{cases} -\frac{1}{1+2a(\tilde{\kappa})}\frac{\sqrt{\pi}}{2^{\beta-\alpha-1}}\left(x^2 g_{\alpha+1}(x) + g_\alpha(x)\right), & \kappa_3 > 0, \\ \frac{1}{1+2a(\tilde{\kappa})}\frac{\sqrt{\pi}}{2^{\beta-\alpha+1}}\, g_{\alpha-1}(x), & \kappa_3 = 0. \end{cases}$$

Both limits hold uniformly on compact subsets of the complex plane. All the constants are given in Theorem 2.

 ISBN 13: 978-84-613-5510-5

Therefore, the presence of the masses in these special cases changes the Mehler–Heine type formulas for the Sobolev polynomials in an essential way. We can observe that this does not occur in the general case when $q \neq -1$ or $\kappa \neq 1$. Some natural questions arise: *Why does it occur?. In the Jacobi case, why are there not any essential changes in the Mehler–Heine type formula when* $\kappa = -1$*?* We leave these questions for the reader to think about them. In our opinion, the answers for these questions are nice. Thus, these cases and their consequences about the zeros are more interesting. Moreover, in the process to obtain Theorem 3 we also get Mehler–Heine type formulas for some cases of so–called Krall type polynomials.

4 Zeros

Mehler–Heine type formulas in the previous Section have immediate consequences on the asymptotic behaviour of the zeros. We only use Hurwitz's Theorem.

In [4] some results for the zeros of Gegenbauer–Sobolev polynomials, S_n^G, have been obtained under the assumptions

$$\alpha \geq 0, \quad \kappa_2 \geq \left(\frac{2\alpha+3}{2\alpha+2} + 2(1+q)\right)\kappa_1 > 0. \tag{9}$$

When $-1 \leq q < 0$, S_n^G has n distinct real zeros and at least $n-2$ of them lie inside the interval $(-1,1)$. If $q > 0$, all the $2m+1$ zeros of S_{2m+1}^G are real, simple and within the interval $(-1,1)$, and S_{2m}^G has at least $2m-2$ distinct real zeros in $(-1,1)$.

For the Jacobi case, in [2] the authors have proved that under the conditions

$$\kappa_2 \geq 2\kappa_1 \geq 0, \quad \kappa_3 \geq 0, \quad \alpha+\beta > 2 \quad \text{and} \quad \begin{cases} \alpha \leq \beta, & \text{if} \quad \kappa \leq -1, \\ \alpha \geq \beta, & \text{if} \quad \kappa \geq 1, \end{cases} \tag{10}$$

the polynomial S_n^J has n distinct real zeros and at least $n-1$ of them lie inside $(-1,1)$

With this information and using Theorem 2 we can deduce that in the general case (i.e, $q \neq -1$ in the Gegenbauer case and $\kappa \neq 1$ in the Jacobi case) the asymptotic behaviour of zeros of the Gegenbauer–Sobolev and Jacobi–Sobolev orthogonal polynomials is the same as the one for Gegenbauer and Jacobi orthogonal polynomials, respectively (see Corollary 1 in [3] and Corollary 4.1 in [1]).

As we have commented, this situation changes for the special cases considered in Theorem 3. We show the results obtained in [1] and [3] about the asymptotic behaviour of the zeros in the following result.

Corollary 2 *a) Let* $\alpha > 1/2$ *and* $q = -1$ *and let* κ_1 *and* κ_2 *which satisfy the restrictions given in (9). We denote by*

$$t = \begin{cases} [n/2]-1, & \text{if } S_n^G \text{ has 2 zeros outside (-1,1),} \\ [n/2], & \text{otherwise.} \end{cases}$$

If $M^{(-1)} > 0$*, then*

$$\lim_{n\to\infty} n \arccos(s_{n,i}^G) = h_i^{(\alpha)}, \quad i = 1, 2, \ldots, t,$$

 ISBN 13: 978-84-613-5510-5

where $0 < h_1^{(\alpha)} < h_2^{(\alpha)} < \cdots < h_t^{(\alpha)}$ *denote the first* t *positive real zeros of the function*

$$h^{(\alpha)}(x) = -\frac{\sqrt{\pi}}{1+4a^{(-1,\kappa_1,\kappa_2)}}\left((2x)^2 g_{\alpha+\frac{1}{2}}(x) + 4g_{\alpha-\frac{1}{2}}(x)\right).$$

If $M^{(-1)} = 0$, *then*

$$\lim_{n\to\infty} n\arccos(s_{n,i}^G) = j_i^{(\alpha-\frac{3}{2})}, \quad i = 1, 2, \ldots, t.$$

b) Let $\kappa = 1$ *be and we assume (10). If* $\kappa_3 > 0$ *then,*

$$\lim_{n\to\infty} n\arccos(s_{n,i}^J) = \hat{h}_i^{(\alpha)}, \quad i = 1, 2, \ldots, r,$$

where $0 < \hat{h}_1^{(\alpha)} < \hat{h}_2^{(\alpha)} < \cdots < \hat{h}_r^{(\alpha)}$ *denote the first* r *positive zeros of the function*

$$\hat{h}^{(\alpha)} = -\frac{1}{1+2a(\tilde{\kappa})}\frac{\sqrt{\pi}}{2^{\beta-\alpha-1}}\left(x^2 g_{\alpha+1}(x) + g_\alpha(x)\right).$$

If $\kappa_3 = 0$ *then,*

$$\lim_{n\to\infty} n\arccos(s_{n,i}^J) = j_i^{(\alpha-1)}, \quad i = 1, 2, \ldots, r.$$

5 Numerical experiments

We illustrate some results about the asymptotic behaviour of the zeros of the Sobolev orthogonal polynomials throughout numerical examples. In fact, we only provide the numerical experiments for the special cases (i.e, $q = -1$ in the Gegenbauer case and $\kappa = 1$ in the Jacobi case) which are the most interesting ones because the asymptotic behaviour of the zeros is different from the one for the classical polynomials considered here.

Table 1: $n\arccos(s_{n,i}^G)$, for $i = 1, 2, 3$, $\alpha = 2$, $q = -1$, $M^{(-1)} = 2$, $\kappa_1 = 5$, $\kappa_2 = 10$. $h_i^{(\alpha)}$ are the positive real zeros of the function $h^{(\alpha)}(x)$ defined in Corollary 2.

$n\arccos(s_{n,i}^G)$	$i=1$	$i=2$	$i=3$
$n=25$	5.301243629	8.637758826	11.818000401
$n=50$	5.370276013	8.755170554	11.986782042
$n=125$	5.415977473	8.830425757	12.091104881
$n=200$	5.427998421	8.850069288	12.118073521
$h_i^{(\alpha)}$	5.448613315	8.883697867	12.164144564

 ISBN 13: 978-84-613-5510-5

Table 2: $n\arccos(s_{n,i}^G)$, for $i=1,2,3$, $\alpha=4$, $q=-1$, $M^{(-1)}=0$, $\kappa_1=6$, $\kappa_2=10$.

$n\arccos(s_{n,i}^G)$	$i=1$	$i=2$	$i=3$
$n=25$	5.276145490	8.305156684	11.223253003
$n=50$	5.484008796	8.652527556	11.720687343
$n=125$	5.643506553	8.905646547	12.066229304
$n=200$	5.687215106	8.974677590	12.159867298
$j_i^{(\alpha-3/2)}$	5.763459197	9.095011331	12.322940970

Table 3: $n\arccos(s_{n,i}^J)$, for $i=1,2,3$, $\alpha=2$, $\beta=5$, $\kappa_1=2$, $\kappa_2=4$, $\kappa_3=3$, $\kappa=1$. $\hat{h}_i^{(\alpha)}$ are the positive real zeros of the function $\hat{h}^{(\alpha)}(x)$ defined in Corollary 2.

$n\arccos(s_{n,i}^J)$	$i=1$	$i=2$	$i=3$
$n=25$	5.410734932	8.489113162	11.417201399
$n=50$	5.727335091	8.985463646	12.085719646
$n=125$	5.941783580	9.321642061	12.537979871
$n=200$	5.998754710	9.410975350	12.658127778
$\hat{h}_i(\alpha)$	6.096997096	9.565066652	12.865374333

Acknowledgements

This work has been partially supported by Ministerio de Educación of Spain under grant MTM2008-06689-C02-01 and Junta de Andalucía (Grupo de investigación FQM229).

References

[1] E.X.L. de Andrade, C.F. Bracciali, L. Castaño–García, J.J. Moreno–Balcázar, *Asymptotics for Jacobi–Sobolev orthogonal polynomials associated with non–coherent pairs of measures*, J. Approx. Theory, accepted.

[2] E.X.L. Andrade, C.F. Bracciali, M.V. Mello, T.E. Pérez, *Zeros of Jacobi–Sobolev orthogonal polynomials following non–coherent pair of measures*, Comput. Appl. Math, to appear.

[3] C.F. Bracciali, L. Castaño–García, J.J. Moreno–Balcázar, *Some asymptotics for Sobolev orthogonal polynomials involving Gegenbauer weights*, submitted.

[4] E.X.L. Andrade, C.F. Bracciali, A. Sri Ranga, *Zeros of Gegenbauer-Sobolev orthogonal polynomials: beyond coherent pairs*, Acta Appl. Math. **105** (2009) 65-82.

 ISBN 13: 978-84-613-5510-5

Table 4: $n \arccos(s^J_{n,i})$, for $i = 1, 2, 3$, $\alpha = 4$, $\beta = 3$, $\kappa_1 = 20$, $\kappa_2 = 5$, $\kappa_3 = 0$, $\kappa = 1$.

$n \arccos(s^J_{n,i})$	$i = 1$	$i = 2$	$i = 3$
$n = 25$	5.652120997	8.776446483	11.729238441
$n = 50$	5.913885478	9.199269025	12.321570872
$n = 125$	6.083696236	9.466024428	12.684408818
$n = 200$	6.128131258	9.535269716	12.777513706
$j_i^{(\alpha-1)}$	6.380161895	9.761023129	13.015200721

[5] G. Szegő, *Orthogonal Polynomials*, vol. 23 of Amer. Math. Soc. Colloq. Publ., 4th ed., Amer. Math. Soc., Providence, RI, 1975.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

An application of the Banach contraction principle on the product of complexity spaces to the study of certain algorithms with two recurrence procedures

F. Castro-Company[1], S. Romaguera[2] and P. Tirado[2]

[1] *Departamento de Matemática Aplicada, Universidad Politécnica de Valencia, 46071 Valencia, Spain*

[2] *Instituto Universitario de Matemática Pura y Aplicada, Universidad Politécnica de Valencia, 46071 Valencia, Spain*

emails: fracasco@doctor.upv.es, sromague@mat.upv.es, pedtipe@mat.upv.es

Abstract

By applying the Banach contraction principle to the product quasi-metric of two complexity spaces we show the existence and uniqueness of solution for the recurrence equations associated to certain algorithms with two recurrence procedures.

Key words: The Banach contraction principle, fixed point, complexity space, quasi-metric space, recurrence, improver.

MSC 2000: 54E50, 54H25, 68Q25, 68Q55.

1 Introduction and preliminaries

Schellekens introduced in [8] the complexity (quasi-metric) space to construct a suitable mathematical model for the complexity analysis of algorithms. In fact, he proved in Section 6 of [8] the existence and uniqueness of solution for the recurrence equations associated to Divide and Conquer algorithms by applying a quasi-metric version of the Banach fixed point theorem to the complexity space. Recently it was shown in [4] and [7] that Schellekens' technique can be systematized to deduce the existence and uniqueness of solution for the recurrence equations associated to Probabilistic Divide and Conquer algorithms, and for the recurrence inequations associated to ExpoDC algorithms, respectively (see [3] and [2, Section 7.7] for a study of such algorithms).

Here we show that such a technique also allow us to prove the existence and uniqueness of solution for the pair of recurrence equations associated to a class of algorithms

 ISBN 13: 978-84-613-5510-5

with two recurrence procedures, as considered by Atkinson in [1]. With the help of the notion of an improver (see Definition 1 in Section 2) we also deduce the well-known fact that if (f_0, g_0) denotes the solution of such recurrences, then $f_0(n) \in \mathcal{O}(e^{2n})$ and $g_0(n) \in \mathcal{O}(e^{2n})$. In order to prove these results, with our approach, we will need to apply the Banach fixed point theorem to the "product complexity space" instead to the original one.

In the rest of this section we recall some pertinent concepts and previous results.

The letters $\mathbb{N}$ and ω will denote the set of positive integer numbers and the set of nonnegative integer numbers, respectively. The supremum of two real numbers x and y will be denoted by $x \vee y$.

By a quasi-metric on a set X we mean a function $d : X \times X \to [0, \infty)$ such that for all $x, y, z \in X$: (i) $x = y \Leftrightarrow d(x, y) = d(y, x) = 0$, and (ii) $d(x, z) \leq d(x, y) + d(y, z)$.

A quasi-metric space is a pair (X, d) such that X is a set and d is a quasi-metric on X.

Each quasi-metric d on X induces a T_0 topology τ_d on X which has as a base the family of open balls $\{B_d(x, r) : x \in X, \varepsilon > 0\}$, where $B_d(x, \varepsilon) = \{y \in X : d(x, y) < \varepsilon\}$ for all $x \in X$ and $\varepsilon > 0$.

Given a quasi-metric d on X, then the function d^{-1} defined by $d^{-1}(x, y) = d(y, x)$, is also a quasi-metric on X, called the conjugate of d, and the function d^s defined by $d^s(x, y) = d(x, y) \vee d^{-1}(x, y)$ is a metric on X.

A quasi-metric space (X, d) is said to be bicomplete if (X, d^s) is a complete metric space.

By a contraction map on a quasi-metric space *(X, d)* we mean a self-map f of X such that $d(fx, fy) \leq kd(x, y)$ for all $x, y \in X$, where k is a constant with $0 < k < 1$. The number k is called a contraction constant for *f*.

It is clear that if f is a contraction map on a quasi-metric space (X, d) with contraction constant k, then f is a contraction map on the metric space (X, d^s) with contraction constant k.

Therefore, the classical Banach contraction principle can be generalized to the quasi-metric setting as follows (see for instance [5, Lemma 2.4])

Theorem 1. *Let f be a contraction map on a bicomplete quasi-metric space (X,d). Then, for each $x \in X$, the sequence of iterations $(f^n x)_{n \in \omega}$ is convergent in (X, d^s) to a point $x_0 \in X$ which is the unique fixed point of f.*

Let us recall that the product quasi-metric space of two quasi-metric spaces (X, d) and (Y, e) is the quasi-metric space $(X \times Y, d \times e)$, where $d \times e$ is defined by

$$(d \times e)((x_1, y_1), (x_2, y_2)) = d(x_1, x_2) \vee e(y_1, y_2),$$

for all $(x_1, y_1), (x_2, y_2) \in X \times Y$.

In this case, $d \times e$ is called the product (or box) quasi-metric of d and e.

 ISBN 13: 978-84-613-5510-5

The so-called complexity space ([8]) is the quasi-metric space $(\mathcal{C}, d_{\mathcal{C}})$, where

$$\mathcal{C} = \left\{ f : \omega \to (0, \infty] : \sum_{n=0}^{\infty} 2^{-n} \frac{1}{f(n)} < \infty \right\},$$

and d_C is the quasi-metric on $\mathcal{C}$ given by

$$d_{\mathcal{C}}(f, g) = \sum_{n=0}^{\infty} 2^{-n} \left((\frac{1}{f(n)} - \frac{1}{g(n)}) \vee 0 \right)$$

for all $f, g \in \mathcal{C}$. (We adopt the convention that $1/\infty = 0$.)

The elements of $\mathcal{C}$ are called complexity functions.

The following useful result is a consequence of [6, Theorem 1, and Remark on p. 317].

Theorem 2. *The complexity space $(\mathcal{C}, d_{\mathcal{C}})$ is bicomplete.*

2 The results

Following Atkinson [1, p. 16-17], consider the two recursive procedure algorithm defined, for two procedures P and Q, and $n \in \omega$, by:

```
function P(n)
    if n > 0 then Q(n-1); C; P(n-1); C; Q(n-1)

function Q(n)
    if n > 0 then P(n-1); C; Q(n-1); C; P(n-1); C; Q(n-1)
```

where C denotes any statements taking time independent of n.

Then, the execution times $S(n)$ and $T(n)$ of $P(n)$ and $Q(n)$, satisfy, at least approximately, the recurrences

$$S(n) = S(n-1) + 2T(n-1) + K_1,$$

and

$$T(n) = 2S(n-1) + 2T(n-1) + K_2,$$

for $n \in \mathbb{N}$, and with K_1, K_2, nonnegative constants. (We assume that $S(0) > 0$ and $T(0) > 0$).

The following extension to our context of Definition 6.2 of [8] will be need.

Definition 1. A functional Φ from $(\mathcal{C} \times \mathcal{C}, d_{\mathcal{C}} \times d_{\mathcal{C}})$ into itself is an improver with respect to an element $(f, g) \in \mathcal{C} \times \mathcal{C}$ if for each $n \in \omega$, $\Phi^{n+1}(f, g) \leq \Phi^n(f, g)$.

 ISBN 13: 978-84-613-5510-5

Note that if Φ is monotone increasing (i.e., $\Phi(f_1, g_1) \leq \Phi(f_2, g_2)$ whenever $f_1 \leq f_2$ and $g_1 \leq g_2$), to show that Φ is an improver with respect to (f, g) it suffices to verify that $\Phi(f, g) \leq (f, g)$.

Intuitively (compare, for instance, [4, p. 348]), an improver is a functional that corresponds to a transformation on algorithms and satisfies the following condition: the iterative applications of the transformation to a given algorithm yield an improved algorithm at each step of the iteration.

In Theorem 3 below (whose proof will be presented in a full version of this paper) we construct a monotone increasing functional Φ, associated with the two recurrences equations T and S given above, which is a contraction on $(\mathcal{C} \times \mathcal{C}, d_{\mathcal{C}} \times d_{\mathcal{C}})$. Then, its unique fixed point (f_0, g_0) will be the solution of the recurrence equations. We can also deduce, with the help of Theorem 1, that $f_0(n) \in \mathcal{O}(e^{2n})$ and $g_0(n) \in \mathcal{O}(e^{2n})$.

(As usual, for $g : \omega \to [0, \infty)$, we write $f(n) \in \mathcal{O}(g(n))$ if $f : \omega \to [0, \infty)$ and there exist $n_0 \in \omega$ and $c > 0$ with $f(n) \leq cg(n)$ for all $n \geq n_0$.)

Theorem 3. *Let Φ be the functional on $\mathcal{C} \times \mathcal{C}$ defined by*

$$\Phi(f, g)(0) = (S(0), T(0)),$$

and

$$\Phi(f, g)(n) = \left(f(n-1) + 2g(n-1) + K_1,\ 2f(n-1) + 2g(n-1) + K_2\right),$$

for $n \in \mathbb{N}$ and $f, g \in \mathcal{C}$. Then:

(1) *Φ is a monotone increasing contraction on $(\mathcal{C} \times \mathcal{C}, d_{\mathcal{C}} \times d_{\mathcal{C}})$ with contraction constant $3/4$.*

(2) *Φ has a unique fixed point (f_0, g_0).*

(3) *$f_0(n) \in \mathcal{O}(e^{2n})$ and $g_0(n) \in \mathcal{O}(e^{2n})$.*

Acknowledgements

The second and third listed authors acknowledge the support of the Spanish Ministry of Science and Innovation, under grant MTM2009-12872-C02-01

References

[1] M.D. Atkinson, *The Complexity of Algorithms*, in: Computing tomorrow: future research directions in computer science. Cambridge Univ. Press, New York (1996) 1–20.

[2] G. Brassard, P. Bratley, *Fundamentals of Algorithms*, Prentice Hall, 1996.

 ISBN 13: 978-84-613-5510-5

[3] P. Flajolet, *Analytic Analysis of Algorithms*, in: 19th Internat. Colloq. ICALP'92, Vienna, July 1992; Automata, Languages and Programming, Lecture Notes in Computer Science **623**, W. Kuich Editor (1992) 186–210.

[4] L.M. García-Raffi, S. Romaguera and M. Schellekens, *Applications of the complexity space to the General Porbabilistic Divide and Conquer Algorithms*, J. Math. Anal. Appl. **348** (2008) 346–355.

[5] S.G. Matthews, *Partial metric topology*, in: Procedings 8th Summer Conference on General Topology and Applications, Ann. New York Acad. Sci. **728** (1994) 183–197.

[6] S. Romaguera and M. Schellekens, *Quasi-metric properties of complexity spaces*, Topology Appl. **98** (1999) 311–322.

[7] S. Romaguera, M. Schellekens, P. Tirado and O. Valero, *Contraction maps on complexity spaces and expoDC algorithms*, in: Proc. International Conference of Computational Methods in Sciences and Engineering ICCMSE 2007. AIP Conference Proceedings. **963** (2007) 1343–1346.

[8] M. Schellekens, *The Smyth completion: a common foundation for denonational semantics and complexity analysis*, Electronic Notes Theoret. Comput. Sci. **1** (1995) 535–556.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Algorithmic Method to Obtain Abelian Subalgebras and Ideals in Lie Algebras

Manuel Ceballos[1], Juan Núñez[1] and Ángel F.Tenorio[2]

[1] *Departamento de Geometría y Topología, Facultad de Matemáticas. Universidad de Sevilla*

[2] *Dpto. de Economía, Métodos Cuantitativos e Historia Económica, Escuela Politécnica Superior. Universidad Pablo de Olavide*

emails: mceballos@us.es, jnvaldes@us.es, aftenorio@upo.es

Abstract

We show an algorithmic method to compute the set of all abelian subalgebras and ideals of any finite-dimensional Lie algebra, starting from the nonzero brackets in its law. To implement this algorithm we use the symbolic computation package MAPLE. It is also shown a brief computational study considering both the computing time and the memory used in the two main routines of the implementation.

Key words: Abelian Lie Subalgebra, Abelian Ideal, α invariant, β invariant, algorithm.
MSC 2000: 17B30, 17B05, 68W40, 68Q25.

1 Introduction

Nowadays, there exists a very extensive research on Lie Theory. However, some aspects of Lie algebras remain unknown. In fact, the classification of solvable Lie algebras is still an open problem, although the classification of other types of Lie algebras (like semi-simple and simple ones) was already obtained in 1890. In order to solve these and other problems, the need of studying other properties of Lie algebras arises. In this way, considering abelian Lie subalgebras of a finite-dimensional Lie algebra constitutes the main goal of this paper.

Indeed, the topic dealt in this paper is the maximal dimension of the abelian subalgebras in a given finite-dimensional Lie algebra $\mathfrak{g}$. Although this concept has been studied in previous papers, most of them (for example [15]) consider abelian ideals instead of abelian subalgebras, which implies that more restrictive hypotheses are needed. However, we do not assume such restrictions, but our work considers all the subalgebras contained in the given Lie algebra $\mathfrak{g}$.

 ISBN 13: 978-84-613-5510-5

Let $\mathfrak{g}$ be a finite-dimensional Lie algebra. We denote by $\alpha(\mathfrak{g})$ the maximal dimension of an abelian subalgebra of $\mathfrak{g}$, and by $\beta(\mathfrak{g})$ the maximal dimension of an abelian ideal of $\mathfrak{g}$. Note that these concepts are proved to be invariant and they are important for many subjects. First of all, they are useful in the study of Lie algebra contractions and degenerations. There exists a large literature, in particular for low-dimensional Lie algebras, see [3, 8, 10] and the references given therein, for example.

Secondly, there are several results concerning the question of how big or small this maximal dimension can be, compared with the dimension of the Lie algebra. Some of them show that a Lie algebra of large dimension contains abelian subalgebras of large dimension. For example, the dimension of a nilpotent Lie algebra $\mathfrak{g}$ satisfying $\alpha(\mathfrak{g}) = \ell$ is bounded by $\dim(\mathfrak{g}) \leq \frac{\ell(\ell+1)}{2}$, see [9, 11]. Another sets that if $\mathfrak{g}$ is a complex solvable Lie algebra with $\alpha(\mathfrak{g}) = \ell$, then we have $\dim(\mathfrak{g}) \leq \frac{\ell(\ell+3)}{2}$, see [7]. To prove these bounds several conditions are also fixed for the value of β invariant.

For a semisimple Lie algebra $\mathfrak{s}$, the invariant $\alpha(\mathfrak{s})$ has been completely determined by Malcev [6]. Since there are no abelian ideals in a simple Lie algebra $\mathfrak{s}$, we have $\beta(\mathfrak{s}) = 0$. Very recently the study of abelian ideals in a Borel subalgebra $\mathfrak{b}$ of a simple complex Lie algebra $\mathfrak{s}$ has drawn considerable attention. We have indeed $\alpha(\mathfrak{s}) = \beta(\mathfrak{b})$, and this number can be computed purely in terms of certain root system invariants, as can be seen in [13]. Let us note that the α invariant can be usefully applied, for example, to characterize Lie algebras in several senses. So Tenorio [14] gave some criteria about properties of Lie algebras starting from this notion. Moreover, this topic has already been studied by different authors, being classical and fundamental the following references: Krawtchouk [4]; Laffey [5], which computed the α invariant of the algebra of $n \times n$ matrices over any field; or Suprunenko and Tyshkevich [12], which dealt with the problem of determining abelian subalgebras of maximal dimension of nilpotent type. However, in some cases like in [15] abelian ideals were considered instead of abelian subalgebras.

Previously, we have already studied abelian subalgebras by considering both points of view: Theoretical and practical. Moreover, the α invariant was computed for two different families of complex Lie algebras: $\mathfrak{g}_n$, of $n \times n$ strictly upper-triangular matrices (see [1]); and $\mathfrak{h}_n$, of $n \times n$ upper-triangular matrices (see [2]). To do it, an algorithmic procedure was introduced in [1]. Now, this paper is devoted to show an algorithmic procedure which works for any arbitrary finite-dimensional complex Lie algebra. The algorithm is given by indicating and commenting each of its steps. Besides, a computational study of its implementation with MAPLE is also shown.

2 Theoretical background

This section is devoted to recall some concepts and results on Lie algebras to be applied later. For a general overview on such subjects, the interested reader can consult [16]. Let us note that, from here on, only finite-dimensional Lie algebras over the field $\mathbb{F}$ are considered, where $\mathbb{F}$ can be $\mathbb{R}$ or $\mathbb{C}$.

Given a finite-dimensional Lie algebra $\mathfrak{g}$, a vector subspace $\mathfrak{h}$ of $\mathfrak{g}$ is an abelian

 ISBN 13: 978-84-613-5510-5

subalgebra if the following conditions hold $[\mathfrak{h},\mathfrak{h}] \subseteq \mathfrak{h}$; and $[u,v] = 0,\ \forall\ u,v \in \mathfrak{h}$.

Moreover, if the subalgebra $\mathfrak{h}$ satisfies the condition $[\mathfrak{h},\mathfrak{g}] \subseteq \mathfrak{h}$, then we say that $\mathfrak{h}$ is an ideal of $\mathfrak{g}$.

To compute the basis of an abelian subalgebra of maximal dimension of $\mathfrak{g}$, we consider a basis $\mathcal{B}_d = \{X_i\}_{i=1}^d$ of $\mathfrak{g}$ and another basis $\mathcal{B} = \{v_h\}_{h=1}^r$ of an arbitrary r-dimensional (abelian) subalgebra $\mathfrak{h}$ (with $r \leq d$). As each vector $v_h \in \mathcal{B}$ is a linear combination of the vectors in $\mathcal{B}_d$, the vectors in $\mathcal{B}$ can be expressed as $v_h = \sum_{i=1}^d a_{h,i}X_i$. Hence, the basis $\mathcal{B}$ can be translated to a matrix in which the h^{th} row records these coordinates of v_h with respect to the basis $\mathcal{B}_d$

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,d} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r,1} & a_{r,2} & \cdots & a_{r,d} \end{pmatrix}. \tag{1}$$

The rank of the matrix (1) is obviously equal to r and, hence, its echelon form is the following by using elementary row and column transformations

$$\begin{pmatrix} b_{1,1} & 0 & \cdots & 0 & b_{1,r+1} & \cdots & b_{1,d} \\ 0 & b_{2,2} & \cdots & 0 & b_{2,r+1} & \cdots & b_{2,d} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_{r,r} & b_{r,r+1} & \cdots & b_{r,d} \end{pmatrix}. \tag{2}$$

So, without loss of generality, we can assume that any given basis $\mathcal{B}$ of $\mathfrak{h}$ can be expressed by (2). Hence, each vector in $\mathcal{B}$ is a linear combination of two different types of vectors X_i: The ones coming from the pivot positions and the remaining ones. The first are called *main vectors* of $\mathcal{B}$ with respect to $\mathcal{B}_d$, being called *non-main vectors* the rest.

3 Algorithm computing abelian subalgebras

Let us consider a n-dimensional Lie algebra $\mathfrak{g}$ with the basis $\mathcal{B}_n = \{Z_1,\ldots,Z_n\}$. If n is lower, the abelian subalgebras and ideals of $\mathfrak{g}$ can be easily computed because the number of nonzero brackets with respect to $\mathcal{B}_n$ is quite greater in proportion with the dimension of $\mathfrak{g}$. To solve this computational problem, we have implemented an algorithmic method which computes a basis of each non-trivial abelian subalgebra of $\mathfrak{g}$. In this algorithm, we will use the main and non-main vectors to express any given basis of the subalgebra in order to determine the existence of nonzero brackets. The vectors in this basis will be expressed as a linear combination of the vectors in $\mathcal{B}_n$.

To implement the algorithm, we have used the symbolic computation package MAPLE. We start loading the libraries `linalg` and `ListTools` to activate commands like `Flatten` and others related to Linear Algebra, since Lie algebras are vector spaces endowed with a second inner structure: The Lie bracket. Besides, the library `combinat` has to be also loaded to apply commands related to Combinatorial Algebra.

 ISBN 13: 978-84-613-5510-5

Now, we show the different steps which constitute the algorithm and its respective implementation. The structure of the algorithm is based on two main routines calling several other subroutines with different functions.

1. Implementing a subroutine which computes the Lie bracket between two arbitrary basis vectors in $\mathcal{B}_n$. This subroutine depends on the law of $\mathfrak{g}$.
2. Programming a subroutine to compute the bracket between two vectors expressed as a linear combination of vectors from the basis $\mathcal{B}_n$ of $\mathfrak{g}$.
3. For each k-dimensional subalgebra $\mathfrak{h}$ of $\mathfrak{g}$, computing the bracket between two arbitrary vectors in the basis of $\mathfrak{h}$. Those vectors are linear combinations of a main vector (with coefficient equal to 1) and the $n-k$ non-main ones. These expressions depend on the dimension of $\mathfrak{h}$.
4. Solving a system whose equations are obtained by imposing the abelian law to the brackects computed in the previous step for the subalgebra $\mathfrak{h}$.
5. Programming a subroutine which determines the existence of abelian subalgebras in a fixed dimension.
6. Computing $\alpha(\mathfrak{g})$ by ruling out dimensions for abelian subalgebras.
7. Computing the basis of an abelian subalgebra of maximal dimension, that is, a subalgebra with dimension $\alpha(\mathfrak{g})$.
8. Computing the basis of an abelian subalgebra for a fixed set of non-main vectors and some restrictions given by the previous subroutines.
9. Programming a subroutine which computes a list with all the abelian subalgebras of $\mathfrak{g}$ with certain dimension k.
10. Implementing the routine to compute a list with the basis of all the non-trivial abelian subalgebras of $\mathfrak{g}$ by using the previous subroutines.
11. Programming a subroutine which determines if there is an abelian ideal associated with a given abelian subalgebra.
12. Computing $\beta(\mathfrak{g})$ from $\alpha(\mathfrak{g})$ and the previous subroutine.
13. Implementing a subroutine which determines the set of abelian ideals of maximal dimension, that is, abelian ideals with dimension $\beta(\mathfrak{g})$.
14. Programming the routine to compute a list with the basis of all the non-trivial abelian ideals of $\mathfrak{g}$ by using the previous subroutines.

The first subroutine, named `law`, receives two natural numbers as inputs. These numbers represent the subindexes of two basis vectors in $\mathcal{B}_n$. The subroutine returns the result of the bracket between these two vectors. Besides, conditional sentences are included to determine nonzero brackets (which are introduced in the subroutine) and the skew-symmetry property.

 ISBN 13: 978-84-613-5510-5

```
> law:=proc(i,j)
> if i=j then return 0; fi; if i>j then return -law(j,i); fi;
> if (i,j)=... then return ...; fi;
> ....
> else return 0; fi; end proc;
```

The first two suspension points are associated with the computation of $[Z_i, Z_j]$: First, the value of the subindexes (i, j) and second, the result of $[Z_i, Z_j]$ with respect to $\mathcal{B}_n$. The third ellipsis denotes the rest of nonzero brackets. For each nonzero bracket, a new sentence `if` has to be included in the cluster.

Then, we implement a subroutine, `bracket`, which computes the bracket between two arbitrary vectors of $\mathfrak{g}$. These vectors are expressed as linear combination of the vectors in $\mathcal{B}_n$. Due to this fact, the subroutine `law` is called in the implementation.

```
> bracket:=proc(u,v,n)
> local exp; exp:=0; for i from 1 to n do for j from 1 to n do
> exp:=exp + coeff(u,Z[i])*coeff(v,Z[j])*law(i,j); od; od; return exp; end proc:
```

After introducing the law of $\mathfrak{g}$, we have to compute the brackets in an arbitrary subalgebra $\mathfrak{h}$. To do it, we implement the subroutine `eq`, which requires four inputs: The dimension `n` of $\mathfrak{g}$; the subindexes `i` and `l`, indicating the main vectors in the bracket to be computed; and a list `M` with the subindexes of the non-main vectors in $\mathfrak{h}$. To do it, three local variables `eqt`, `L` and `P` are defined. For computing the brackets between the vectors in $\mathcal{B}_n$, the subroutine `eq` calls the subroutine `bracket`, which is necessary to obtain each bracket in the law of $\mathfrak{h}$. Whereas the variable `eqt` saves the expression of the bracket belonging to the law of $\mathfrak{h}$, the list `P` takes the elements of `M` two by two and finally, `L` is a list containing all the coefficients in the expression of `eqt` with respect to $\mathcal{B}_n$. Precisely, the list `L` is the first term of the output of the subroutine `eq`. The second is a list with the subindexes `i` and `l` corresponding to `L`. Let us note that the subindexes of the main vectors has to be saved together with the coefficients in order to use them in a later subroutine.

Each vector in the subalgebra $\mathfrak{h}$ can be expressed as a linear combination of one main vector and the $n - k$ non-main ones according to expression (2), where each row represents the coefficients of one vector in the basis of $\mathfrak{h}$. Obviously, we can assume that the coefficient of each main vector is equal to 1, because the row of (2) corresponding to that main vector can be divided by its coefficient. To implement the subroutine `eq`, the coefficients of the non-main vectors are denoted by `a[i,k]`.

```
> eq:=proc(n,i,l,M::list)
> local eqt,L,P; L:=[]; if nops(M)=1 then P:=[[M[1],M[1]]] else P:=choose (M,2);
> end if; eqt:=law(i,l); for k from 1 to nops(M) do
> eqt:=eqt + a[l,M[k]]*law(i,M[k]) + a[i,M[k]]*law(M[k],l);
> end do; for j from 1 to nops(P) do eqt:=eqt+(a[i,P[j][1]]*a[l,P[j][2]]-
> a[i,P[j][2]]*a[l,P[j][1]])*bracket(P[j][1],P[j][2]); od; for m from 1 to n do
> L:=[op(L),coeff(eqt,Z[m])]; end do; return L,[i,l]; end proc;
```

Let us note that it is also possible to program the subroutine `eq` by using the subroutine `bracket`. However, we will consider the previous implementation for the

 ISBN 13: 978-84-613-5510-5

computational study due to the fact that if we consider an implementation of `eq` which calls the subroutine `bracket`, both the computing time and the used memory will increase.

```
> eq:=proc(n,i,l,M::list)
> local eqt,L,u,v; L:=[]; eqt:=0;u:=Z[i];v:=Z[l]; for k from 1 to nops(M) do
> u:=u+a[i,M[k]]*Z[M[k]]; v:=v+a[l,M[k]]*Z[M[k]]; od; eqt:=bracket(u,v,n);
> for m from 1 to n do L:=[op(L),coeff(eqt,Z[m])]; od; return L,[i,l]; end proc:
```

Next, we implement the subroutine `sys`, which receives two inputs: The dimension `n` of $\mathfrak{g}$ and a list `M` with the subindexes of the non-main vectors in the basis of $\mathfrak{h}$. This subroutine solves the system of equations generated by the subroutine `eq`. Four local variables `L`, `P`, `R` and `S` have been defined for its implementation: `L` is a list with the subindexes of the main vectors; the list `R` contains the expressions computed by the subroutine `eq`; `P` is defined as in the previous subroutine; and, finally, `S` is a set where the equations of the system are saved

```
> sys:=proc(n,M::list)
> local L,P,R,S; L:=[]; R:=[]; S:={}; for x from 1 to n do
> if member(x,convert(M,set))=false then L:=[op(L),x]; fi; od;
> if nops(L)=1 then P:=[[L[1],L[1]]] else P:=choose (L,2); fi;
> for j from 1 to nops(P) do r[j]:=[eq(n,P[j][1],P[j][2],M)]; od;
> R:=[seq(r[i][1],i=1..nops(P))]; for y from 1 to nops(R) do
> for k from 1 to n do S:={op(S),R[y][k]=0}; od; od; return {solve(S)}; end proc;
```

The following subroutine, called `absub`, is implemented by introducing two natural numbers `n` and `k`, namely: `n` is the dimension of $\mathfrak{g}$ and `k` is less than `n`. This subroutine determines the existence of abelian subalgebras with dimension `k`. Two local variables are used by the subroutine: `L` and `S`. The first variable, `L`, is a list whose elements are lists with the subindexes of the `n`−`k` non-main vectors. The variable `S` is a set with the solutions given by the subroutine `sys`. In this way, `absub` returns a message indicating the non-existence of `k`-dimensional abelian subalgebras or, if there exist k-dimensional abelian subalgebras, returns the set `S`. Since the coefficient of each main vector is 1, the system given by the subroutine `sys` has not solutions when `S` vanishes. When the system has some solution, the family of computed vectors is linearly independent and forms a basis of the subalgebra. Let us note that, if all the solutions in `S` contain some complex coefficient, there are no real solutions for the system solved by `sys` and there do not exist any abelian subalgebras of dimension `k` for the case $\mathbb{F} = \mathbb{R}$. For this field, it would be necessary to include a conditional sentence for determining if such complex coefficients appear.

```
> absub:=proc(n,k)
> local L,S; L:=choose(n,n-k); S:={ }; for i from 1 to nops(L) do
> if sys(n,L[i])={{}} then S:=S else for j from 1 to nops(sys(n,L[i])) do
> S:={op(S),{convert(L[i],set),sys(n,L[i])[j]}}; od; fi; od;
> if S={} then return "There is no abelian subalgebra"; fi;
> if S={{}} then return "There is no abelian subalgebra"
> else return S; fi; end proc;
```

 ISBN 13: 978-84-613-5510-5

Next, we implement the subroutine `alpha`, which receives the dimension `n` of $\mathfrak{g}$ as its unique input and returns the α invariant of $\mathfrak{g}$. The subroutine starts studying if $\alpha(\mathfrak{g}) = n$ by using the subroutine `absub`. Then, a loop is programmed to stop when `absub` does not find abelian subalgebras.

```
> alpha:=proc(n)
> if type(absub(n,n-1),set)=true then return n-1; fi;
> for i from 2 to n-1 do if absub(n,i)="There is no abelian subalgebra"
> then return i-1; fi; od; end proc;
```

The following subroutine, named `asmd`, receives as input the dimension `n` of $\mathfrak{g}$ and returns the basis of an abelian subalgebra of maximal dimension. To do so, this subroutine calls the subroutines `alpha` and `absub`.

```
> asmd:=proc(n)
> local u,L,R,S,B,k; k:=alpha(n);S:={};L:={}; u:=absub(n,k);
> if k=1 then return {seq({Z[i]},i=1..n)}; fi;
> if type(u[1][1],set(integer))=true then R:=u[1][1]; S:=u[1][2] else
> R:=u[1][2]; S:=u[1][1]; fi; for x from 1 to n do
> if member(x,R)=false then L:={op(L),x}; fi; od; for i from 1 to nops(L) do
> b[i]:=Z[L[i]]; od; for i from 1 to nops(L) do
> for j from 1 to nops(R) do b[i]:=b[i]+a[L[i],R[j]]*Z[R[j]]; od; od;
> B:={seq(b[i],i=1..nops(L))}; return eval(B,S); end proc:
```

Now, we implement the subroutine `basabsub`, which receives three inputs: the dimension `n` of $\mathfrak{g}$ and two sets, `S` and `T`, with the subindexes of the non-main vectors in the basis of $\mathfrak{h}$. We will use this subroutine with the solution given by `sys`. Four local variables `R`, `B`, `M` and `N` have been defined for its implementation. First, a conditional sentence `if`, for the sets `M` and `N`, is introduced in the cluster to find out wether `S` or `T` is the set of non-main vectors. This is due to the fact that MAPLE sometimes returns the solutions in different order. `R` is a set with the subindexes of the main vectors and, in the set `B`, we compute the basis for the abelian subalgebra. In this way, `B` is the output of this subroutine.

```
> basabsub:=proc(n,S::set,T::set)
> local R,B,M,N; R:={};B:={}; if type(S,set(integer))=true then
> M:=S; N:=T else M:=T; N:=S; end if;
> for x from 1 to n do if member(x,M)=false then R:={op(R),x};
> fi; od; for i from 1 to nops(R) do b[i]:=Z[R[i]]; od; for i from 1 to nops(R) do
> for j from 1 to nops(M) do b[i]:=b[i] + a[R[i],M[j]]*Z[M[j]];
> od; od; B:={seq(b[i],i=1..nops(R))}; return eval(B,N); end proc:
```

The following subroutine, named `listabsub`, requires two inputs: The dimension `n` of $\mathfrak{g}$ and a natural number k, less than n and which corresponds with the dimension of the abelian subalgebra. To implement it, two local variables `S` and `L` are considered. This subroutine calls the subroutine `basabsub` for computing the basis for each k-dimensional abelian subalgebra. Whereas this value is saved in the local variable `S`, `L` is a set with the basis of each abelian subalgebra of $\mathfrak{g}$ with dimension k. Precisely, the list `L` is the output of the subroutine `listabsub`.

 ISBN 13: 978-84-613-5510-5

```
> listabsub:=proc(n,k)
> local S,L; S:=absub(n,k);L:={}; if k=1 then return {seq({Z[i]},i=1..n)}; fi;
> if S="There is no abelian subalgebra" then return {}; fi; for i from 1 to nops(S) do
> L:={op(L),basabsub(n,S[i][1],S[i][2])}; od; return L; end proc:
```

Let us note that it is also possible to give an equivalent implementation for the subroutine `asmd` by using the subroutine `listabsub`:

```
> asmd:=proc(n)
> local k; k:=alpha(n); return listabsub(n,k); end proc:
```

Now, we implement the routine `allabsub`, which receives the dimension `n` of $\mathfrak{g}$ as its unique input. The routine `allabsub` returns a set with the basis of all the abelian subalgebras of $\mathfrak{g}$ with dimension less than or equal to $\alpha(\mathfrak{g})$. In this way, the routine starts computing $\alpha(\mathfrak{g})$ and then, the output is defined by using the previous subroutine `listabsub`.

```
> allabsub:=proc(n)
> local B,k; k:=alpha(n);B:={}; for i from 1 to k-1 do B:={op(B),listabsub(n,i)};
> od; return B; end proc:
```

Next, we explain the subroutine `abideal`, which requires two inputs: A set, `S`, with the basis of an abelian subalgebra and the dimension `n` of $\mathfrak{g}$. The subroutine is devoted to determine the existence of an abelian ideal from the basis of an abelian subalgebra, `S`, obtained with the subroutine `listabsub` for a fixed dimension. To do so, we impose that `S` has to be the basis of an abelian ideal. Then, we solve the system: if there is no solution, the output of this subroutine is the message "There is no abelian ideal" and if there is a solution, it returns the basis of an abelian ideal.

```
> abideal:=proc(S,n)
> local w, R, L, Q, M; w:=0; R:=[]; L:=[]; Q:={}; M:={}; N:={};
> for i from 1 to nops(S) do w:=w + a[i]*S[i]; od; for i from 1 to nops(S) do
> for j from 1 to n do if linbracket(S[i],Z[j],n)<>0 then
> L:=[op(L),linbracket(Z[j],S[i],n)]; else L:=L; fi; od; od;
> for i from 1 to nops(L) do r[i]:=0; for j from 1 to nops(S) do
> r[i]:=r[i]+b[i,j]*S[j]; od; od; R:=[seq(r[i],i=1..nops(L))];
> M:={seq(L[k]-R[k], k=1..nops(L))}; for i from 1 to nops(M) do
> Q:={op(Q),seq(coeff(M[i],Z[j])=0,j=1..n)}; od; if {solve(Q)}={} then return
>"There is no abelian ideal" else return eval(S,solve(Q)); fi; end proc:
```

The subroutine `beta` receives the dimension `n` of $\mathfrak{g}$ as its unique input and returns the β invariant of $\mathfrak{g}$. Let us note that this value can be zero (semisimple Lie algebras). The subroutine starts computing the value of α. Then, a loop is programmed by using the previous subroutine and `listabsub`

```
> beta:=proc(n) local r; r:=alpha(n); for k from 0 to r-1 do
> for i from 1 to nops(listabsub(n,r-k)) do
> if abideal(listabsub(n,r-k)[i],n)<>"There is no abelian ideal"
> then return r-k; fi; od; od; return 0; end proc:
```

 ISBN 13: 978-84-613-5510-5

Next, in this subroutine, named `aimd`, we compute the basis of an abelian ideal of maximal dimension; that is, an abelian ideal with dimension $\beta(\mathfrak{g})$. To do so, the routine `aimd` calls the subroutines `beta`, `listabsub` and `abideal`. First, we compute the set of all abelian subalgebras of dimension $\beta(\mathfrak{g})$ and then we apply the subroutine `abideal` to obtain abelian ideals.

```
> aimd:=proc(n)
> local k,S,T; k:=beta(n);S:=listabsub(n,k);T:={};
> for i from 1 to nops(S) do T:={op(T),abideal(S[i],n)}; od; return T; end proc:
```

The routine `allabideal` receives the dimension `n` of $\mathfrak{g}$ as its unique input. This routine returns a set with the basis of all the abelian ideals of $\mathfrak{g}$ with dimension less than or equal to $\beta(\mathfrak{g})$. The output of this routine is defined by using the subroutines `listabsub` and `abideal`.

```
> allabideal:=proc(n)
> local B, k; k:=beta(n); B:={};
> if k=0 then return {}; else for i from 1 to k do
> for j from 1 to nops(listabsub(n,i)) do
> if abideal(listabsub(n,i)[j],n)<>"There is no abelian ideal" then
> B:={op(B),abideal(listabsub(n,i)[j],n)};
> end if; end do; end do; end if; return B; end proc:
```

Now, we show an example with a 4-dimensional Lie algebra with brackets $[Z_1, Z_2] = Z_3, [Z_1, Z_3] = Z_4$.

```
> alpha(4);
                                     3
> listabsub(4,3);
                          {{Z[1], Z[2], Z[3]}}
> allabsub(4);
            {{{Z[2], Z[3], Z[4]}}, {{Z[1]}, {Z[2]}, {Z[3]}, {Z[4]}},
       {{Z[4], Z[1]+a[1, 2]Z[2]+a[1, 3]Z[3]}, {Z[4], Z[2]+a[2,1]Z[1]+a[2, 3]Z[3]},
       {Z[4], Z[3]+a[3, 1]Z[1]+a[3, 2]Z[2]}, {Z[2]+a[2, 3]Z[3], Z[4]+a[4, 3]Z[3]},
       {Z[2]+a[2, 4]Z[4], Z[3]+a[3, 4]Z[4]}, {Z[3]+a[3, 2]Z[2], Z[4]+a[4, 2]Z[2]}}}
> beta(4);
                                     3
> allabideal(4);
               {{Z[4]}, {Z[3], Z[4]}, {Z[2], Z[3], Z[4]}}
```

4 Statistical and computational data

In this section, we show a computational study of the previous algorithm, which has been implemented with MAPLE 12, in an Intel Core 2 Duo T 5600 with a 1.83 GHz processor and 2.00 GB of RAM. Table 2 shows some computational data about both the computing time and the memory used to return the output of `allabsub` according to the value of the dimension n of the algebra.

 ISBN 13: 978-84-613-5510-5

This computational study was done considering a particular family of Lie algebras: The Lie algebras $\mathfrak{s}_n$ generated by $\{e_1, e_2, \ldots, e_n\}$ with the following nonzero brackets:

$$[e_i, e_n] = e_i, \quad \text{for } i < n.$$

This family has been chosen because they constitute a special subclass of non-nilpotent solvable Lie algebras, which allow us to check empirically the computational data given for both the computing time and the used memory.

In Table 1, the set of all non-trivial abelian subalgebras has been computed for the algebras in this family up to dimension $n = 13$ inclusive. Starting from $n = 8$, the computing time is about three times greater when the dimension n is increased in one unit.

Table 1: Computing time and used memory for `allabsub`.

Input	**Computing time**	**Used memory**
$n = 2$	0 s	0 MB
$n = 3$	0 s	0 MB
$n = 4$	0.11 s	3.13 MB
$n = 5$	0.15 s	5.06 MB
$n = 6$	0.43 s	5.38 MB
$n = 7$	1.05 s	5.56 MB
$n = 8$	2.67 s	6.06 MB
$n = 9$	6.98 s	7.06 MB
$n = 10$	20.27 s	8.25 MB
$n = 11$	61.17 s	11.50 MB
$n = 12$	187.89 s	13.87 MB
$n = 13$	804.73 s	51.93 MB

Table 2: Computing time and used memory for `allabideal`.

Input	**Computing time**	**Used memory**
$n = 2$	0 s	0 MB
$n = 3$	0.08 s	3.31 MB
$n = 4$	0.50 s	5.75 MB
$n = 5$	1.98 s	5.88 MB
$n = 6$	8.03 s	6.50 MB
$n = 7$	35.97 s	6.94 MB
$n = 8$	169.54 s	7.56 MB
$n = 9$	779.37 s	8.19 MB
$n = 10$	4118.78 s	9.31 MB

In Table 2, the set of all non-trivial abelian ideals has been computed for the same family of Lie algebras up to dimension $n = 10$ inclusive.

Next we show brief statistics about the relation between the computing time and the memory used by the implementation of the main routines `allabsub` and `allabideal`

 ISBN 13: 978-84-613-5510-5

for the Lie algebras $\mathfrak{s}_n$. In each case, the figure on the left corresponds to the routine `allabsub` and the other to the routine `allabideal`.

Figure 1 shows the behavior of the computing time (C.T.) for both routines according to the dimension n of $\mathfrak{s}_n$ and Figure 2 shows the behavior of the used memory (U.M.) for both routines according to the dimension n of $\mathfrak{s}_n$.

We can observe that the computing time increases more quickly than the used memory in both cases. Besides, whereas the increase of the computing time corresponds to a positive exponential model, the used memory does not follows such a model.

We have also studied the quotients between used memory and computing time. The resulting data can be observed in the frequency diagram shown in Figure 3. In this case, the behavior can be also considered exponential, although this time is negative.

Figure 1: Graphs for the C.T. with respect to dimension.

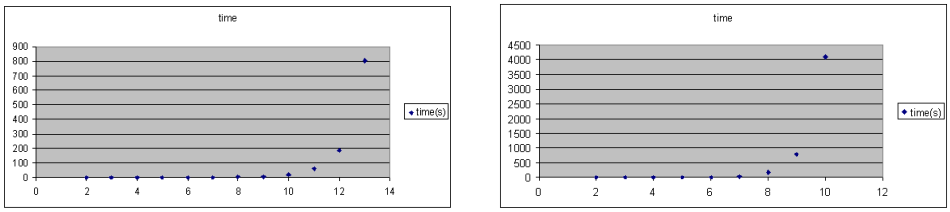


Figure 2: Graphs for the U.M. with respect to dimension.

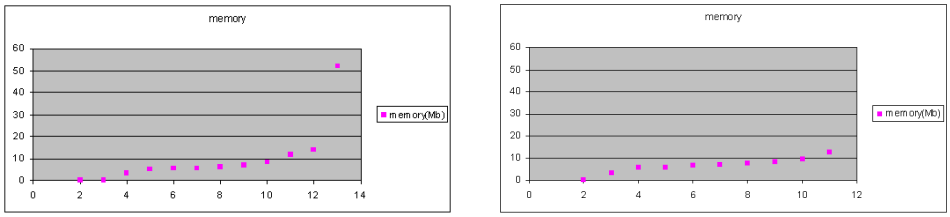
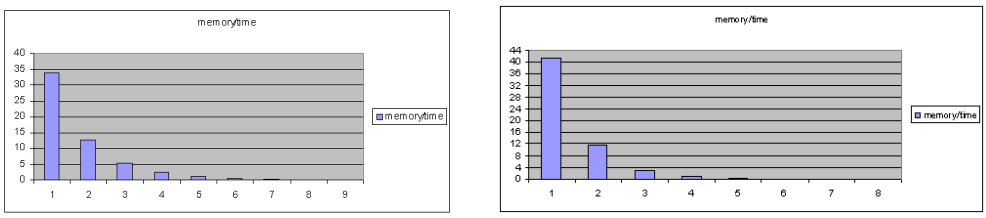


Figure 3: Graphs for the quotients U.M./C.T. with respect to dimension.



 ISBN 13: 978-84-613-5510-5

References

[1] J.C. Benjumea, J. Núñez and A.F. Tenorio, *The Maximal Abelian Dimension of Linear Algebras formed by Strictly Upper Triangular Matrices*, Theor. Math. Phys. **152** (2007) 1225–1233.

[2] M. Ceballos, J. Núñez and A.F. Tenorio, *The Computation of Abelian Subalgebras in the Lie Algebra of Upper-Triangular Matrices*, An. St. Univ. Ovidius Constanta **16** (2008) 59–66.

[3] V. V. Gorbatsevich, *On the level of some solvable Lie algebras*, Siberian Math. J. **39** (1998) 872–883.

[4] M. Krawtchouk, *Über vertauschbare Matrizen*, Rend. Circolo Mat. Palermo Serie I **51** (1927) 126–130.

[5] T.J. Laffey, *The minimal dimension of maximal commutative subalgebras of full matrix algebras*, Linear Alg. Appl. **71** (1985) 199–212.

[6] A. Malcev, *Commutative subalgebras of semi-simple Lie algebras*, Amer. Math. Soc. Transl. **1951** (1951) 15 pp.

[7] M. V. Milentyeva, *On the dimensions of commutative subalgebras and subgroups*, J. Math. Sciences **149** (2008) 1135–1145.

[8] M. Nesterenko and R. Popovych, *Contractions of low-dimensional Lie algebras*, J. Math. Phys. **47** (2006) 123515, 45 pp.

[9] D. M. Riley, *How abelian is a finite-dimensional Lie algebra?*, Forum Math. **15** (2003) 455–463.

[10] C. Seeley, *Degenerations of 6–dimensional nilpotent Lie algebras over* $\mathbb{C}$, Comm. in Algebra **18** (1990) 3493–3505.

[11] I. Stewart, *Bounds for the dimensions of certain Lie algebras*, J. London Math. Soc. (2) **3** (1971) 731–732.

[12] D.A. Suprunenko and R.I. Tyshkevich, *Commutative Matrices*, Academic Press, 1968.

[13] R. Suter, *Abelian ideals in a Borel subalgebra of a complex simple Lie algebra*, Invent. Math. **156** (2004) 175–221.

[14] A.F. Tenorio, *Solvable Lie Algebras and Maximal Abelian Dimensions*, Acta Math. Univ. Comenian. (N.S.) **77** (2008) 141–145.

[15] J.-L. Thiffeault and P.J. Morrison, *Classification and Casimir Invariants of Lie-Poisson Brackets*, Phys. D **136** (2000) 205–244.

[16] V.S. Varadarajan, *Lie Groups, Lie Algebras and Their Representations*, Springer, 1984.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Numerical simulation of a heat transfer problem, via a shape optimization method.

Adbelkrim Chakib[1] **and Mourad Nachaoui**[1,2]

[1] *Laboratoire de Mathématiques et Applications, Universite Soultan Moulay slimane, Faculté des Sciences et Techniques, B.P.523, Béni-Mellal, Maroc.*

[2] *Laboratoire de Mathématiques Jean Leray, UMR 6629, , Université de Nantes/CNRS/ECN, 2 rue de la Houssiniére, BP 92208, 44322 Nantes, France.*

emails: chakib@fstbm.ac.ma, mourad.nachaoui@univ-nantes.fr

Abstract

In this work, we deal with an optimal shape design approach for a problem modelling a process of welding. Our interest is the numerical approximation of the solution of this optimal shape design problem. We consider a discretization of this problem based on linear finite elements. We present a numerical study of genetic algorithms proposed to solve this problem. Then we give some numerical results to demonstrate the accuracy and efficiency of the proposed method.

Key words: Welding problem, Inverse problem, Shape Optimization, Non coercive opertor, Finite element, Bézier curves, Genetic algorithms.

1 Introduction

The Welding remains one of the most common joining processes in manufacturing. The Joining of two workpieces occurs as a result of solidification of the metal molten in the neighborhood of the contact area following application of a heat source, such as a plasma arc, electric current, laser beam, liquid filler droplets, etc.... Thus, the mechanical properties of the resulting joint, such as its strength, uniformity, resistance to fatigue, etc..., are determined by the complex thermo-fluid phenomena occurring in the weld pool. To solve this problem, many models are proposed in literature [1, 3]. We are interested by an approach which deals only with the solid part of the workpiece. Particularly we focus to the numerical approximation of the shape optimization formulation proposed to solve this problem with this approach in [2].

 ISBN 13: 978-84-613-5510-5

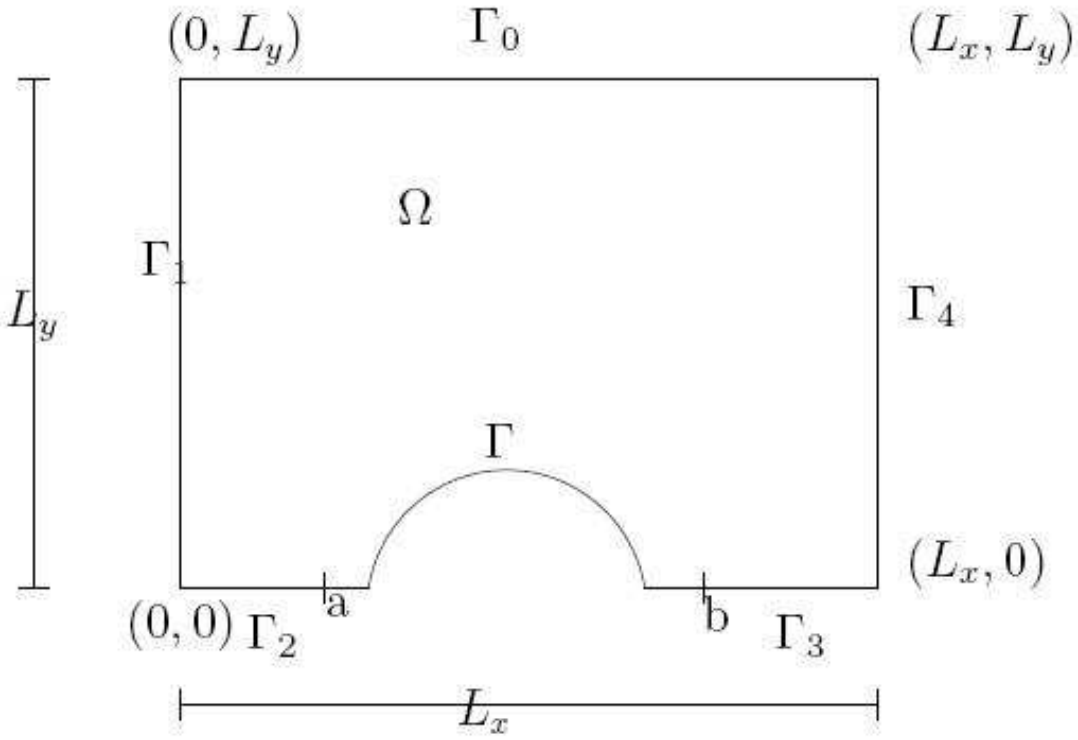


Figure 1: 2-dimensional configuration of the welding process

2 State of the problem

We consider a problem of welding where the solid part of the body denoted by Ω, is illustrated in Figure 1. This problem consist in finding (T, Γ) solution of:

$$\begin{cases} K\,\dfrac{\partial T}{\partial x} = \nabla \cdot (\lambda \nabla T) + f \text{ in } \Omega \\ \lambda \dfrac{\partial T}{\partial \nu} = 0 \text{ on } \Gamma_0 \cup \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \\ T = T_d \text{ on } \Gamma_4 \\ T = T_0 \text{ on } \Gamma_0 \\ T = T_f \text{ on } \Gamma \end{cases} \tag{1}$$

where Γ is the free boundary, K is a constant dependent of the material characteristics (density of the plate and heat capacity,...), λ is the thermal conductivity and f, a source term, is a given function. The quantities T_d, T_0 and T_f are given temperatures.

 ISBN 13: 978-84-613-5510-5

3 Numerical approximation

We consider a discretization of this problem based on linear finite elements. We present a numerical discussion of some new genetic algorithms developed to solve the obtained discrete problem.

3.1 Numerical results

In practice the free boundary is parameterized by the Bézier curves. The corresponding optimal shape discrete problem is solved by the genetic algorithms. The following figures show that the cost decreases with respect to the number of iterations. The obtained numerical results are found to confirm the actual effectiveness of the method proposed.

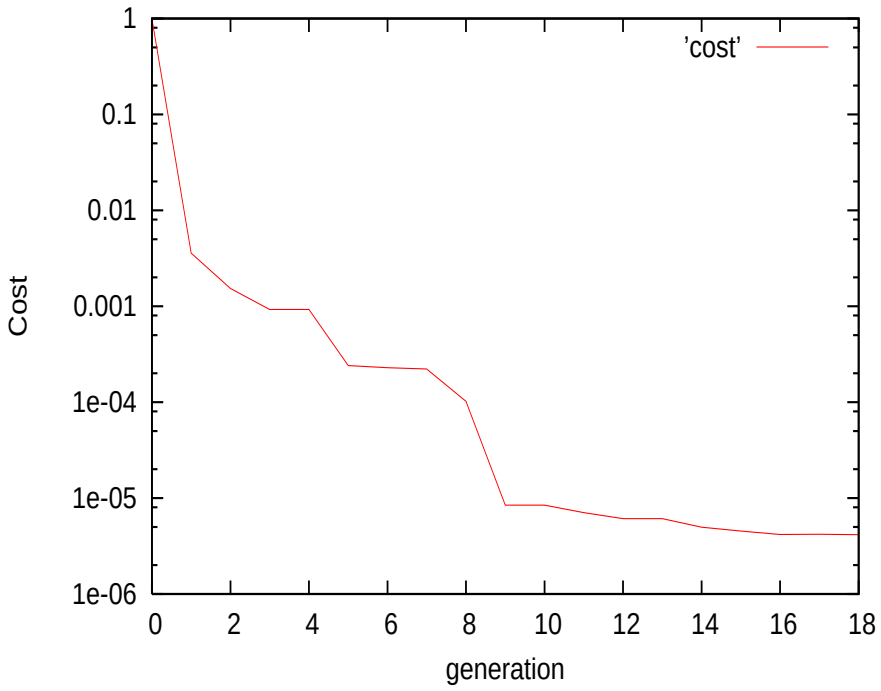


Figure 2: Cost evolutionary

 ISBN 13: 978-84-613-5510-5

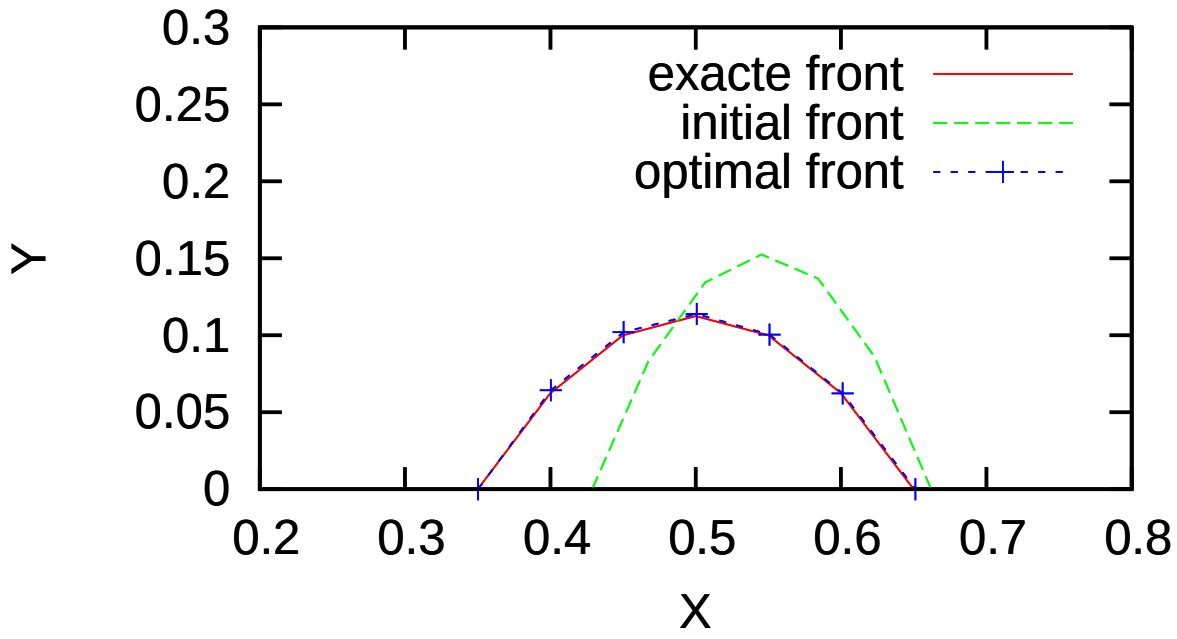


Figure 3: Boundary evolution

References

[1] Bergheau,J. M. Numerical simulation of welding, Revue européenne des éléments finis, volume 13 no 3-4, 2004.

[2] Chakib, A; Ellabib, A; Nachaoui, A; Nachaoui, M. A shape optimization formulation of weld pool determination. Submitted.

[3] Feulvarch, E.; Boitout, F.; Bergheau, J. Simulation thermomécanique du soudage par friction-malaxage, European Journal of Computational Mechanics, Vol 16/6-7 - 2007 - pp.865-887

[4] Haslinger, J.; Makinen, R. A. E. *Introduction to shape optimization. Theory, approximation, and computation.* Advances in Design and Control, 7. SIAM, Philadelphia, 2003.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Pollard's Rho algorithm for ECDLP on Graphic Cards

M.Chinnici[1], S. Cuomo[2], M. Laporta[2], S. Migliori[1] and A. Pizzirani[2]

[1] *ENEA-FIM-INFOPPQ, Casaccia Research Center, S.Maria di Galeria, Italy, ENEA - Research Center*

[2] *Dipartimento di Matematica e Applicazioni R.Caccioppoli Napoli, Italy, Universty of Naples FEDERICO II*

emails: marta.chinnici@enea.it, salvatore.cuomo@unina.it, mlaporta@unina.it, silvio.migliori@enea.it, a.pizzirani@gmail.com

Abstract

The recent introduction by NVidia of Compute Unified Device Architecture (CUDA) libraries for High Performance Computing on Graphic Processing Units has started the trend of video cards for resolution of many computationally hard problems in different areas like fluid dynamics, molecular dynamics, computer vision and astrophysics. In this paper we show how CUDA libraries and hardware can be introduced in cryptography as a cryptoanalytic tool.We describe an implementation of a parallelized Pollard's rho attack on ECDLP, based upon recent results about the optimization of Pollard's rho method and enhanced by some "ad-hoc" choices for CUDA.

Key words: Cryptanalysis, Elliptic curves, High Performce Computing.

1 Introduction

Cryptography is essentially aimed at protecting data from unauthorized access. This is particularly important when data involve sensible informations and are transmitted on insecure channels. Typical examples are business via internet and the payment with a credit card. The data involved in such transactions are usually encrypted to make it harder for an attacker to retrieve secret informations. In the past, the key for encryption was the same for decryption raising a serious problem regarding key distribution. In 1976 W.Diffie and M.Hellman[1] invented an agreement protocol that allows two users to exchange a secret key over an insecure channel without any prior contact. This event is commonly considered the birth of public-key cryptography. Then, relying on some hard mathematical problem, many cryptosystems have been proposed. However, since some attacks to such math problems succeded most of these cryptosystems become insecure or simply impratical. Actually, three mathematical problems

 ISBN 13: 978-84-613-5510-5

are still considered to be hard: the integer factorization problem (IFP), the discrete logarithm problem in the multiplicative group of a finite field (DLP) and in the group of points of an elliptic curve (ECDLP). There is no real proof that the aforementioned problems are intractable. However, a lot of work has been done to try to solve them efficiently (see [Odlyzko] for an overview). All these efforts amount to the development of subexponential-time algorithms for IFP and DLP resolution (index-calculus methods), but these methods are not applyable to ECDLP resolution. Elliptic curve cryptography (ECC) became more and more attractive essentially for such a reason. Moreover, parameters of the ECC are usually much smaller than parameters of cryptosystems based on IFP and DLP. Consequently ECC has lower communication overhead.
In this paper we study how the Rho Pollard algorithm can be implemented on graphics cards. Some experimental results by E.Teske[2][3] showed that the running time of the Pollard's algorithm tends to the expected value $\sqrt{\pi n/2}$ as the number of such subsets increases. Mainly storage can be efficiently reduced to a negligible amount at the cost of some extra computation. By using Floyd's algorithm called "tortoise and hare"([4] exercises 6 and 7, page 7) it can be reduced to a constant. Van Oorschot and Wiener[5] proposed to record only points satisfying a precise condition for a better trade off between space and performances. Moreover, they showed that the algorithm can be efficiently parallelized on an arbitrary number of processors. While each of them generates a random walk from a different starting point, collision detection is completed by another designated computer.
In [6] the authors show a first implementation of a CUDA based code in which a parallel algorithm is reported. In this work an optimized computational code is proposed in order to avoid the so called "divergent threads" as in the following. The paper is organized as follows: in Section 2 we give preliminary notion on elluptic curves; then Section 3 reports mainly argumtens on ECDLP; in Section 4 a Cuda based implementation algorithm and numerical results, and finally, conclusions are provided in Section 5.

2 Overview of elliptic curves

Let $\mathbb{K}$ be a field of characteristic $\neq 2, 3$. The set E of points $(x, y) \in \mathbb{K} \times \mathbb{K}$ satisfying the equation $y^2 = x^3 + ax + b$ with $a, b \in \mathbb{K}$, is called *elliptic curve* whenever $x^3 + ax + b$ has no multiple roots in $\mathbb{K}$. The definition of an elliptic curve is slightly more complicated in when the charateristic of $\mathbb{K}$ is 2 or 3. The set E, enriched with a so-called "point to infinity" O_∞ and a well defined addition $+$, becomes an *elliptic group,* denoted by $E(\mathbb{K})$ where the point O_∞ acts as the group identity. If $\mathbb{K} = \mathbb{R}$, the real field, then the addition can be described geometrically through the method "chord-tangent"(see [7] p.55).The inverse of a point $P = (x, y)$ is $-P = (x, -y)$ by definition.Moreover, one has the following explicit formulas for the sum and the doubling of points on $E(\mathbb{R})$. If $P = (x_1, y_1)$, $Q = (x_2, y_2)$ and $P + Q = (x_3, y_3)$, then $x_1 \neq x_2$ implies

 ISBN 13: 978-84-613-5510-5

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2 \quad , \qquad y_3 = -y_1 + \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) \quad , \qquad (1)$$

while $P = Q$ implies

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1 \quad , \qquad y_3 = -y_1 + \left(\frac{3x_1^2 + a}{2y_1}\right)(x_1 - x_3) \quad . \qquad (2)$$

More remarkable is the fact that these formulas are still valid in $E(\mathbb{K})$ for a generic ground field $\mathbb{K}$, where the so called *elliptic curve discrete logarithm problem* can be formulated as it follows: *given $P, Q \in E(\mathbb{K})$, determine the integer k (if there's one) so that $Q = kP = ktimes\underbrace{P + P + ... + P}$*. This problem is significantly hard if the ground field is finite.

Indeed, V. Miller[8] and N. Koblitz[9] proposed (independently each other) to use the elliptic group $E(\mathbb{F}_p)$, defined on a finite field $\mathbb{F}_p$, as an arithmetic base of a cryptosystem.

3 ECDLP

Altough ECDLP is a particular case of DLP, there is no generic algorithm with subexponential running time that solves it. One reason is that there's a primary difference between the underlying algebraic structures, i.e. the multiplicative group $\mathbb{F}_p^*$ of a finite field $\mathbb{F}_p$ for DLP and the elliptic group $E(\mathbb{F}_p)$. Mainly, while $\mathbb{F}_p^*$ is completed in a structure with two operations, the elliptic group has only its own addition. For example, the *index-calculus methods*, which solves DLP instances with subexponential running time, fail when it comes to elliptic groups (except in very special and well-understood cases). As usual, algorithms for ECDLP are classified as it follows: *generic algorithms* which are appliable to all instances of ECDLP, *special algorithms* which take advantage of the particular instance of the problem. Since all special attacks to the ECDLP can be easily avoided by means of a suitable choice of the parameters, it is more interesting to focus on generic algorithms. The most used generic methods are variations of the Pollard's rho method. Indeed, in 1997 CERTICOM introduced a list of ECDLP challenging problems offering a money prize for each solution[10]. The solved problems got a solution through the use of a parallelized Pollard's rho method.

3.1 Pollard's rho algorithm for ECDLP

Let us consider $P, Q \in E(\mathbb{F}_p)$ and assume that we want to compute k such that $Q = kP$. The main idea of Pollard's rho algorithm is to determine distinct pairs (c', d') and (c'', d'') of integers modulo n such that

$$c'P + d'Q = c''P + d''Q$$

 ISBN 13: 978-84-613-5510-5

where n is the order of the subgroup $\langle P\rangle$ generated by P. Then, one can compute

$$(c' - c'')P = (d'' - d')Q = (d'' - d')kP,$$

which implies

$$(c' - c'') \equiv (d'' - d')k \ (modn).$$

Thus,

$$k \equiv (c' - c'')(d'' - d')^{-1} \ (modn).$$

The naive method requires to generate at random $c, d \in [0, n-1]$, compute $cP + dQ$ and store each triple $(c, d, cP + dQ)$ in a table sorted by the third element until a point $cP + dQ$ is obtained twice (this occurrence is called "collision"). The *birthday paradox*[1] helps to estimate the expected number of iterations (or equivalently the complexity of the algorithm) before a collision is found. This number is approximately $\sqrt{\pi n/2} \approx 1.2533\sqrt{n}$. Instead of randomly generated points Pollard[11] proposed an iteration function acting on $\langle P\rangle$ with a pseudo-random behaviour. If this function is "random enough", then, the algorithm has the same expected running time of the naive method. The original Pollard's function partitions $\langle P\rangle$ into three subsets S_1, S_2, S_3 of approximately equal size. Then, from the starting point P_0 it iterates $P_1 = f(P_0), P_2 = f(P_1), ..., P_{i+1} = f(P_i)$. More precisely, it is defined as:

$$P_{i+1} = f(P_i) = \begin{cases} P_i + a_1P + b_1Q & if \quad P_i \in S_1 \\ 2P_i & if \quad P_i \in S_2 \\ P_i + a_2P + b_2Q & if \quad P_i \in S_3 \end{cases}$$

Some experimental results by E.Teske[2][3] showed that the running time of the Pollard's algorithm tends to the expected value $\sqrt{\pi n/2}$ as the number of subsets S_i increases. Mainly storage can be efficiently reduced to a costant by using Floyd's algorithm ([4] exercises 6 and 7, page 7). Van Oorschot and Wiener[5] proposed to record only points satisfying a precise condition (for example, the last 30bits of the x coordinate have to be equal to zero) for a better trade off between space and performances.

4 CUDA based implementation

CUDA is a general purpose parallel computing architecture developed by NVidia. Programming of CUDA devices is realized mainly through "C for CUDA", an extension of the C language that gives user access to CUDA capabilities. Even if C is the main language in CUDA hardware programming, third party wrappers are available for Python, Fortran, Java and MatLab. Actually, as it is reported by NVidia, there are millions of CUDA-capable gpus which are already installed with prices ranging from a few euros for hardware with limited computing capabilities (20 euros-30 euros for an 8400GS-256mb video card) to thousands of euros for high-end hardware with 4 teraflops (single

[1]The birthday paradox can be formulated as it follows: how large the number of people must be in a room in order to expect at least two of them have the same birthday ? Surprisingly the number is small: $\sqrt{\pi 365/2} \approx 24$.

 ISBN 13: 978-84-613-5510-5

precision) power (Tesla C1060 with 960 cores). Some advantages offered by CUDA architecture are: scattered reads(code has access to all memory addresses), a fast shared memory region (a region that grants really high performances and can be used by all threads together), full support of integer and bitwise operations and fast downloads and readbacks to and from gpu.

CUDA has also some limitation that must be considered while developing software: no support for recursive functions on the device, division and inversion are computationally expensive and the device memory management is difficult(threads using device memory should access it avoiding multiple requests on the same bank).

4.1 Parallelized Pollard's rho algorithm

Considering high processing power of actual gpus, it makes sense to take advantage from them for a parallelized Pollard's rho algorithm. Here we give a brief description of our implementation of this Pollard's rho algorithm for gpu, discussing later some implementation details:

Algorithm 1 *RhoCuda*

1. *The host makes precomputations needed for the Pollard's rho algorithm.*
2. *Precomputed data is sent to the device.*
3. *The host starts threads on gpu.*
4. *Such threads generate pseudorandom points through the iteration function.*
5. *Threads look for distinguished points (DPs)*
6. *Treads reports DPs to the host.*
7. *DPs are stored into a hash table.*
8. *The host looks for collisions.*
9. *Stop procedure if a collision is found.*

Observe that *distinguished points* having last 30 bit of x coordinate all equalto zero so an optimizated storage strategy can be acted. In the following figure we show a simple scheme of the algorithm on GPU.

4.2 *Modular arithmetic and numerical results*

Since we are considering ECDLPs on finite fields $\mathbb{F}_p$ with p prime, first tools that we need are efficient modular arithmetic functions that can be implemented with CUDA. As already said, integer division and modulo operations are really expensive. Hence, one has to find suitable solutions for an efficient modular arithmetic, especially when we handle multiword integers. It is due to Single Instruction Multiple Threads (SIMT)

 ISBN 13: 978-84-613-5510-5

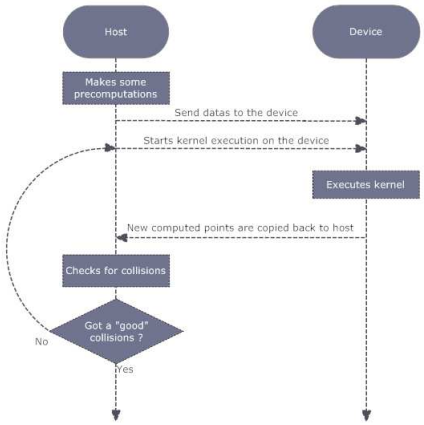


Figure 1: Figure: Cuda based implementation scheme

structure of NVidia GPUs.

Modular addition and difference. If a, b are the operands, and n is modulus, the modular addition is operated computing:

$$a + b \text{ and } a + b - n$$

and then choosing the right result (the one between 0 and n). In this way, we don't use the "**if**" statement avoiding the so called "*divergent threads*" on GPUs. Divergent threads determ an decreasing of the run-time algorithm preformance due to synchronization of threads.
The modular difference is operated in an analogous way computing $a - b$ and $a - b + n$.

Modular multiplication. It is realized through the so called Montgomery product([12] p.395-397).

If n is the modulus, we call k the integer so that $2^{k-1} \leq n < 2^k$ and r is 2^k. Given an integer $a < n$, we define *Montgomery representation* (or *n-residue*) with respecto to r as

$$\overline{a} \equiv a \cdot r (mod n) \ .$$

Observe thata sum and difference of the Mongomery representations of two integers is Montgomery representation of their sum or difference. Given two numbers a, b in their Montgomery representations ($\overline{a}, \overline{b}$ respectively) the Montgomery product is defined as

$$\overline{u} \equiv \overline{a} \cdot \overline{b} \cdot r^{-1} (mod n) \ ,$$

where r^{-1} is the multiplicative inverse of r modulo n.

The result of Montgomery product $\overline{u}$ is the n-residue of the product $u = a \cdot b (mod n)$ since

 ISBN 13: 978-84-613-5510-5

$\overline{u} \equiv \overline{a} \cdot \overline{b} \cdot r^{-1}(modn)$
$= (a \cdot r) \cdot (b \cdot r) \cdot r^{-1}(modn)$
$= (a \cdot b) \cdot r(modn)$.

To describe Montgomery reduction algorithm, we need also the quantity n', that satisfies the property

$$r \cdot r^{-1} - n \cdot n' = 1 \ .$$

Both integers r^{-1} and n' can be easily computed through the extended Euclidean algorithm[inserire citazione].

Given the integers $\overline{a}, \overline{b}$ the Montgomery product is computed by this algorithm:

Algorithm 2 *MonPro($\overline{a}, \overline{b}$)*

1. $t = \overline{a} \cdot \overline{b}$
2. $m \equiv t \cdot n'(modr)$
3. $\overline{u} = (t + m \cdot r)/r$
4. *if* $\overline{u} \geq n$ *then return* $\overline{u} - n$ *else return* $\overline{u}$.

The main feature of this product is that the operations involved are multiplications modulo r and division by r that can be efficiently implemented using bitwise operations.

If n is odd, Montgomery product algorithm can be used to compute (normal) product $u \equiv a \cdot b(modn)$:

Algorithm 3 *ModMul(a, b)*

1. *Compute* n' *using extended Euclidean algorithm*
2. $\overline{a} \equiv a \cdot r(modn)$
3. $\overline{b} \equiv b \cdot r(modn)$
4. $\overline{u}$ =*MonPro($\overline{a}, \overline{b}$)*
5. u =*MonPro($\overline{u}, 1$)*
6. *return* u .

A better algorithm is obtained observing that
MonPro($\overline{a}, b$)=$(a \cdot r) \cdot b \cdot r^{-1}(modn) = a \cdot b(modn)$
Thus we can modify the algorithm above:

Algorithm 4 *ModMul(a, b)*

1. *Compute* n' *using extended Euclidean algorithm, and* $r^2(modn)$

 ISBN 13: 978-84-613-5510-5

2. $\overline{a} = MonPro(a, r^2)$

3. $u = MonPro(\overline{a}, b)$

When a lot of modular multiplication must be performed, all having same modulus, values n' and r^2 can be precomputed.

If we handle multi-word integers, better implementation of Montgomery product is through the coarsely integrated operand scanning method(CIOS)[13]. This method integrate multiplication steps and reduction steps(instead of first computing full product and then making reduction) requiring an additional space for the operation of three words regardless of lenght in words of the operands.

4.3 *The iteration function.*

The starting points are linear combinations of P and Q. Each point is generated with a different multiple of P. If t threads are executed, each of them is associated to a starting point:

$$A_l = a_l P + Q$$

where $0 \leq l < t$ and $0 \leq a_l < n$. Our iteration function is a so-called "**add only**" function which partitions $\langle P \rangle$ into r subsets. Let $A_{l,i} = (x_{l,i}, y_{l,i})$ be the point corresponding to the walk of the l-th thread and the i-th iteration. The iteration function is defined as:

$$f(A_{l,i}) = A_{l,i+1} = A_{l,i} + b_j P + Q$$

if $x_{l,i} \equiv j(mod r), \quad \forall j = 0, ..., r-1, \quad \forall l = 0, ..., t-1$.

The congruence $x_{l,i} \equiv j(mod r)$ can be easily checked through bitwise operators if r is a power of 2. In our application we choose $r = 64$.

4.4 *Points representation and storage.*

The array:

$$(b_0 P + Q, ..., b_j P + Q, ..., b_{r-1} P + Q)$$

is stored by using affine coordinates. More precisely, since the size of these coordinates turns out to be small and since the points do not vary within the program, the array can be fitted into the*constant memory region* of the graphic card. That allows to avoid memory problems due to simultaneous accesses of more than one thread to the same point. Moreover note that all data of the curve are recorded in the constant memory.

Since formulas 1 and 2, all the other points (A_l) are represented with the so-called Jacobian coordinates([14], 3.2).

With this implementation strategy we can avoid division and inverse calculation while adding points on the elliptic curve through mixed addition Jacobian-affine formulas. In order to get coalesced memory access, a single word of each Jacobian coordinate is

 ISBN 13: 978-84-613-5510-5

memorized in a locations multiple of 16 (number of threads that can do a coordinated read from memory).
The following table reports the usage of two middle level N-Vidia gpu, on $\mathbb{F}_{109}$ listed in CERTICOM site. In particular the first test (M1) is curried out on 8800GTS with g92 chip of 80-90euros cost; the second one (M2) is curried out on 8400 GS 256mb of cost 25-30euros cost. Here we analyse the performances on two cases: the code with "**if**" statements that implies a lot of divergent threads (dt); the optimized code where divergent threads are minimized (mdt)

GPU Model	`points/secs code`	`points/secs opitmized code`
M1	320.000	720.000
M2	220.000	425.000

5 Conclusions

CUDA-enabled gpus turn out to form a very interesting platform for high performance computing for many remarkable reasons: the wide diffusion, a rapidly and continuously increasing power, a good performance/price ratio and the possibility of installing more than one gpu into a single workstation.

In this paper we have shown that graphic cards can be useful to improve on performances for ECDLP resolution. We think that all we have done here is new because all CERTICOM problems were attacked only by using cpus (in distributed environment). Now a mixed approach cpu-gpu can also be considered (gpu being a co-processor). At least, computations for our problem can be performed only on gpu while cpu remains free and available for other jobs.

References

[1] W. Diffie, M. E. Hellman, http://dx.doi.org/10.1109/TIT.1976.1055638New Directions in Cryptography, IEEE Transactions on Information Theory IT-22 (6) (1976) 644–654. http://dx.doi.org/http://dx.doi.org/10.1109/TIT.1976.1055638 doi:http://dx.doi.org/10.1109/TIT.1976.1055638.
http://dx.doi.org/10.1109/TIT.1976.1055638

[2] E. Teske, citeseer.ist.psu.edu/teske98speeding.htmlSpeeding up Pollard's rho method for computing discrete logarithms, Lecture Notes in Computer Science 1423 (1998) 541–554.
citeseer.ist.psu.edu/teske98speeding.html

[3] E. Teske, On random walks for Pollard's rho method, Math. Comput. 70 (234) (2001) 809–825. http://dx.doi.org/http://dx.doi.org/10.1090/S0025-5718-00-01213-8 doi:http://dx.doi.org/10.1090/S0025-5718-00-01213-8.

[4] D. E. Knuth, Art of Computer Programming, Volume 2: Seminumerical Algorithms (3rd Edition), Addison-Wesley Professional, 1997.

 ISBN 13: 978-84-613-5510-5

[5] P. C. Van Oorschot, M. J. Wiener, Parallel collision search with cryptanalytic applications, Journal of Cryptology 12 (1999) 1–28.

[6] M. Chinnici, S. Cuomo, M. Laporta and A. Pizzirani, Cuda based implementation of parallelized pollards rho algorithm for ecdlp, in: Proceedings on FINAL WORKSHOP OF GRID PROJECTS, PON RICERCA 2000-2006, AVVISO 1575, Consorzio Comeneta, 2009.

[7] J. H. Silverman, The arithmetic of elliptic curves, Vol. 106 of Graduate Texts in Mathematics, Springer, 1986.

[8] V. S. Miller, http://www.springerlink.com/content/4lfhkd08684v3wylUse of Elliptic Curves in Cryptography, Advances in Cryptology − CRYPTO '85: Proceedings (1986) 417+.
http://www.springerlink.com/content/4lfhkd08684v3wyl

[9] N. Koblitz, Elliptic Curve Cryptosystems, Mathematics of Computation 48 (177) (1987) 203–209.

[10] CERTICOM, Certicom ECC challenge,
http://www.certicom.com/images/pdfs/cert_*ecc_challenge.pdf*.

[11] J. M. Pollard, Monte Carlo methods for index computation mod p, Mathematics of Computation 32 (1978) 918–924.

[12] H. C. A. van Tilborg, Encyclopedia of cryptography and security, Springer-Verlag, 2005.

[13] T. C. K. Ko, B. S. Kalinski Jr., Analyzing and Comparing Montgomery multiplication Algorithms, IEEE Micro 16 (1996) 26–33.

[14] A. J. M. D. Hankerson, S. Vanstone, Guide to Elliptic Curve Cryptography, Springer-Verlag New York, Inc., 2003.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Some algebraic properties related to the degree of a Boolean function

Joan-Josep Climent[1], Francisco J. García[2] and Verónica Requena[1]

[1] *Departament d'Estadística i Investigació Operativa, Universitat d'Alacant*

[2] *Departament de Fonaments de l'Anàlisi Econònica, Universitat d'Alacant*

emails: jcliment@ua.es, francisco.garcia@ua.es, vrequena@ua.es

Abstract

In this paper we establish some algebraic properties of a Boolean function that allow us to introduce a method to determine the degree of the Boolean function from its support.

Key words: Boolean function, support, weight, algebraic normal form, degree.
MSC 2000: 06E30, 94A60.

1 Introduction

Boolean functions are used in several different types of cryptographic applications, including block ciphers, stream ciphers, hash functions [3, 4, 7, 9], and coding theory [2, 6], among others. There is already a well established theory of S-boxes which has sprung from cryptography. This theory concentrates on the design and analysis of Boolean functions which possess desirable cryptographic properties such as balancedness, strict avalanche criterion, correlation immunity, high nonlinearity and high degree. For example, the implementation of an S-box needs nonlinear Boolean functions to guarantee the cryptographic effectiveness in order to resist powerful methods of attack such as the linear and differential cryptanalysis [1, 5, 8, 10].

One of the basic requirements relative to the Boolean functions used in stream ciphers is that they allow to increase the linear complexity [9, 15, 16], which is obtained if these functions have a high algebraic degree.

Both the use of the algebraic normal form or the truth table, have their advantages and disadvantages. For example, the algebraic normal form of a Boolean function directly provide their degree, but not its weight, but if we know the truth table, we know its weight, but do not know its degree.

The complete determination of the algebraic normal form of a Boolean function of which we know its truth table or its support requires simultaneously to compute all the

 ISBN 13: 978-84-613-5510-5

coefficients of the corresponding polynomial, but if we want to know only the degree of the function, it is possible to reduce substantially the number of necessary operations using the properties that we present beforehand here.

The rest of the paper is organized as follows. Firstly, in Section 2 we introduce some basic definitions and notations that are used hereafter. In Section 3, we introduce the main results of this paper; in particular, we present some properties which allow us to determine the degree of a Boolean function of n variables from its support. Finally, in Section 4, we introduce more properties that allow us to improve the process described in Section 3.

2 Preliminaries

We denote by $\mathbb{F}_2$ the Galois field of two elements, 0 and 1, with the addition (denoted by $\oplus$) and the multiplication (denoted by juxtaposition). For any positive integer n, it is well-known that $\mathbb{F}_2^n$ is a linear space of dimension n over $\mathbb{F}_2$ with the usual addition (denoted also by $\oplus$). We denote by $\mathrm{Span}\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k\}$ the linear subspace of $\mathbb{F}_2^n$ generated by the vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k \in \mathbb{F}_2^n$. If $F \subseteq \mathbb{F}_2^n$ and $\boldsymbol{a} \in \mathbb{F}_2^n$, we denote by

$$\boldsymbol{a} \oplus F = \{\boldsymbol{a} \oplus \boldsymbol{u} \mid \boldsymbol{u} \in F\}.$$

When F is a k-dimensional linear subspace of $\mathbb{F}_2^n$ we say that $\boldsymbol{a} \oplus F$ is the k-dimensional affine subspace of $\mathbb{F}_2^n$ passing through $\boldsymbol{a}$ in the direction of F. Finally, if we denote by $\boldsymbol{i}$ the binary expansion of n digits of the integer i, for $i = 0, 1, 2, \ldots, 2^n - 1$, then $\mathbb{F}_2^n = \{\boldsymbol{i} \mid 0 \leq i \leq 2^n - 1\}$.

A **Boolean function** of n variables is a map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$. The set of all Boolean functions of n variables is denoted by $\mathcal{B}_n$; it is well known that $\mathcal{B}_n$, with the usual addition of functions (that we also denote by $\oplus$), is a linear space of dimension 2^n over $\mathbb{F}_2$.

If $f \in \mathcal{B}_n$, we call **truth table** of f (see, for example, [11, 12]) the binary sequence of length 2^n given by

$$\boldsymbol{\xi}_f = (f(\mathbf{0}), f(\mathbf{1}), \ldots, f(\mathbf{2^n - 1})).$$

We call **support** of f, denoted by $\mathrm{Supp}\,(f)$, the set of vectors of $\mathbb{F}_2^n$ whose image by f is 1, that is,

$$\mathrm{Supp}\,(f) = \{\boldsymbol{a} \in \mathbb{F}_2^n \mid f(\boldsymbol{a}) = 1\}.$$

If $f \in \mathcal{B}_n$, we call **weight** of f, and we write $w(f)$, the number of 1s of the truth table of f, therefore, $w(f) = |\mathrm{Supp}\,(f)|$.

Obviously, f is the null function if and only if $\mathrm{Supp}\,(f) = \emptyset$ and then $w(f) = 0$. Analogously, f is the constant function 1 if and only if $\mathrm{Supp}\,(f) = \mathbb{F}_2^n$ and, in this case, $w(f) = 2^n$.

It is easy to check that if $f, g \in \mathcal{B}_n$, then $\mathrm{Supp}\,(f \oplus g) = \mathrm{Supp}\,(f)\,\Delta\,\mathrm{Supp}\,(g)$, where Δ denote the symmetric difference of sets and, as a consequence,

$$w(f \oplus g) \equiv w(f) + w(g) \pmod 2.$$

 ISBN 13: 978-84-613-5510-5

In general, if $f_j \in \mathcal{B}_n$, for $j = 1, 2, \ldots, m$, then

$$\mathrm{Supp}\left(\bigoplus_{j=1}^{m} f_j\right) = \bigtriangleup_{j=1}^{m} \mathrm{Supp}\,(f_j) \tag{1}$$

and, therefore,

$$w\left(\bigoplus_{j=1}^{m} f_j\right) \equiv \sum_{j=1}^{m} w(f_j) \pmod 2. \tag{2}$$

Assume now that $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ where each x_j, for $j = 1, 2, \ldots, n$, is a binary variable. If $f \in \mathcal{B}_n$, we can write $f(\boldsymbol{x})$ uniquely as (see, for example, [6, 11, 12, 13, 14])

$$f(\boldsymbol{x}) = \bigoplus_{\boldsymbol{u} \in \mathbb{F}_2^n} \mu_f(\boldsymbol{u}) \boldsymbol{x}^{\boldsymbol{u}} \tag{3}$$

where $\mu_f(\boldsymbol{u}) \in \mathbb{F}_2$ and if $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$, then

$$\boldsymbol{x}^{\boldsymbol{u}} = x_1^{u_1} x_2^{u_2} \cdots x_n^{u_n} \quad \text{with} \quad x_j^{u_j} = \begin{cases} x_j, & \text{if } u_j = 1, \\ 1, & \text{if } u_j = 0. \end{cases}$$

Expression (3), in which each one of the terms $\boldsymbol{x}^{\boldsymbol{u}}$ is a **monomial**, is known as the **Algebraic Normal Form** (ANF) of $f(\boldsymbol{x})$. We call **degree** of f, denoted by $\deg(f)$, the maximum of the degrees of the monomials of its ANF. So, if $w(\boldsymbol{u})$ denotes the number of components of $\boldsymbol{u}$ equal to 1, then

$$\deg(f) = \max\{w(\boldsymbol{u}) \mid \mu_f(\boldsymbol{u}) = 1\}.$$

We said that f is an **affine function** if $\deg(f) = 1$; in this case expression (3) becomes

$$f(\boldsymbol{x}) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n$$

with $a_j \in \mathbb{F}_2$, for $j = 0, 1, 2, \ldots, n$. In particular, if $a_0 = 0$, we say that f is a **linear function**.

On the other hand, if

$$\boldsymbol{\mu}_f = (\mu_f(\mathbf{0}), \mu_f(\mathbf{1}), \ldots, \mu_f(\mathbf{2^n - 1})),$$

then (see for example [13])

$$\boldsymbol{\mu}_f = \boldsymbol{\xi}_f A_n$$

where

$$A_n = \begin{bmatrix} A_{n-1} & A_{n-1} \\ O & A_{n-1} \end{bmatrix} \text{ for } n \geq 1, \quad \text{with } A_0 = [1].$$

Now, if

$$\boldsymbol{u} = (u_1, u_2, \ldots, u_n) = u_1 \mathbf{2^{n-1}} \oplus u_2 \mathbf{2^{n-2}} \oplus \cdots \oplus u_{n-1} \mathbf{2} \oplus u_n \mathbf{1}$$

 ISBN 13: 978-84-613-5510-5

and $S(\boldsymbol{u}) = \text{Span}\left\{u_1\mathbf{2^{n-1}}, u_2\mathbf{2^{n-2}}, \ldots, u_{n-1}\mathbf{2}, u_n\mathbf{1}\right\}$, then, it is easy to prove by induction over n, that

$$\mu_f(\boldsymbol{u}) = \bigoplus_{\boldsymbol{a}\in S(\boldsymbol{u})} f(\boldsymbol{a}). \tag{4}$$

Finally, if $f \in \mathcal{B}_n$ and for all $\boldsymbol{a} \in \mathbb{F}_2^n$ we consider $g_{\boldsymbol{a}} \in \mathcal{B}_n$ such that $g_{\boldsymbol{a}}(\boldsymbol{x}) = f(\boldsymbol{x}\oplus\boldsymbol{a})$, then, it is difficult to establish the relation between $\boldsymbol{\mu}_f$ and $\boldsymbol{\mu}_{g_{\boldsymbol{a}}}$, nevertheless, it is evident that $\deg(g_{\boldsymbol{a}}) = \deg(f)$, for all $\boldsymbol{a} \in \mathbb{F}_2^n$, and it is not difficult to see that

$$\text{Supp}(g_{\boldsymbol{a}}) = \boldsymbol{a} \oplus \text{Supp}(f), \quad \text{for all } \boldsymbol{a} \in \mathbb{F}_2^n.$$

3 Main Results

Throughout this paper we denote by S_n the set of all the permutations of $\{1, 2, \ldots, n\}$. Moreover, if $\sigma \in S_n$, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n$, $M \subseteq \mathbb{F}_2^n$, and $f \in \mathcal{B}_n$, we write $\boldsymbol{x}^\sigma = \left(x_{\sigma(1)}, x_{\sigma(2)}, \ldots, x_{\sigma(n)}\right)$, $M^\sigma = \{\boldsymbol{x}^\sigma \mid \boldsymbol{x} \in M\}$, and $f^\sigma(\boldsymbol{x}) = f(\boldsymbol{x}^\sigma)$.

The following result, whose proof is immediate, establishes the relation between $\text{Supp}(f^\sigma)$ and $\text{Supp}(f)$ for all $\sigma \in S_n$.

Theorem 1: *Assume that $\sigma \in S_n$. If $f \in \mathcal{B}_n$, then* $\text{Supp}(f^\sigma) = (\text{Supp}(f))^{\sigma^{-1}}$ *and, as a consequence, $w(f^\sigma) = w(f)$.*

The following result, whose proof is also immediate, allows us to determine, explicitly, the support and, therefore, the weight of any monomial.

Theorem 2: *Assume that $1 \le i_1 < i_2 < \cdots < i_k \le n$ and consider $\sigma \in S_n$ such that $\sigma(j) = i_j$, for $j = 1, 2, \ldots, k$. If $f(\boldsymbol{x}) = x_{i_1}x_{i_2}\cdots x_{i_k}$ and $\boldsymbol{u} = (1, 1, \ldots, 1) \in \mathbb{F}_2^k$, then*

$$(\text{Supp}(f))^\sigma = \{\boldsymbol{u}\} \times \mathbb{F}_2^{n-k}$$

and, in particular, $w(f) = 2^{n-k}$.

An immediate consequence of the previous result is that the weight of the monomial formed by all the variables (that is, when $k = n$) is 1, whereas the weight of any other monomial is a power of 2 and, therefore, an even number.

Another immediate consequence that we can deduce of Theorem 2 is that the degree of a Boolean function $f \in \mathcal{B}_n$ is n if and only if $w(f)$ is an odd number, as we establish in the following result.

Theorem 3: *If $f \in \mathcal{B}_n$, then* $\deg(f) = n$ *if and only if $w(f)$ is an odd number.*

Proof: Clearly $f = g \oplus h$, with $g, h \in \mathcal{B}_n$ such that $h(\boldsymbol{x}) = ax_1x_2\cdots x_n$ with $a \in \mathbb{F}_2$ and $\deg(g) \le n-1$. Note that $w(h) = a$, and $\deg(f) = n$ if and only if $a = 1$.

Moreover, if $g(\boldsymbol{x}) = \bigoplus_{j=1}^m g_j(\boldsymbol{x})$, with $g_j(\boldsymbol{x})$ a monomial such that $\deg(g_j) \le n-1$, for $j = 1, 2, \ldots, m$, then, from expression (2) and Theorem 2 we have that

$$w(f) \equiv \sum_{j=1}^{m} w(g_j) + w(h) \pmod 2 \equiv a \pmod 2$$

and so, $\deg(f) = n$ if and only if $w(f)$ is odd. □

 ISBN 13: 978-84-613-5510-5

Another immediate consequence of Theorem 2 is that if the degree of a Boolean function of n variables is less than or equal to $n-2$, then the sum of the elements of its support is the null vector. Before, nevertheless, we introduce the following technical lemma which allows us to simplify the proof of the above mentioned result.

Lemma 1: *Assume that* $1 \leq i_1 < i_2 < \cdots < i_k \leq n$. *If* $f(\boldsymbol{x}) = x_{i_1}x_{i_2}\cdots x_{i_k}$ *and* $1 \leq k \leq n-2$, *then* $\bigoplus_{\boldsymbol{a}\in \mathrm{Supp}(f)} \boldsymbol{a} = \boldsymbol{0}$.

Proof: Assume that $\boldsymbol{u} = (1,1,\ldots,1) \in \mathbb{F}_2^k$. Clearly

$$\bigoplus_{\boldsymbol{a}\in\{\boldsymbol{u}\}\times\mathbb{F}_2^{n-k}} \boldsymbol{a} = \bigoplus_{\boldsymbol{v}\in\mathbb{F}_2^{n-k}} (\boldsymbol{u},\boldsymbol{v}) = \left(\bigoplus_{\boldsymbol{v}\in\mathbb{F}_2^{n-k}} \boldsymbol{u}, \bigoplus_{\boldsymbol{v}\in\mathbb{F}_2^{n-k}} \boldsymbol{v} \right) = (\boldsymbol{0}_k, \boldsymbol{0}_{n-k}) = \boldsymbol{0}$$

because each one of the components of the vectors $\bigoplus_{\boldsymbol{v}\in\mathbb{F}_2^{n-k}} \boldsymbol{u}$ and $\bigoplus_{\boldsymbol{v}\in\mathbb{F}_2^{n-k}} \boldsymbol{v}$ is the sum of an even number of 1s.

The result follows now from the fact that if we consider $\sigma \in S_n$ such that $\sigma(j) = i_j$, for $j = 1,2,\ldots,k$, then, by Theorem 2, the elements of $\mathrm{Supp}\,(f)$ are obtained from the elements of $\{\boldsymbol{u}\}\times\mathbb{F}_2^{n-k}$ permuting their components according to the permutation σ^{-1}.□

Note that the condition $1 \leq k \leq n-2$ of the previous lemma is necessary, because if $k = n$, then $\mathrm{Supp}\,(f) = \{\boldsymbol{u}\} \subseteq \mathbb{F}_2^n$, whereas if $k = n-1$ and $f(\boldsymbol{x})$ is the monomial of degree $n-1$, which does not contain the variable x_j, then $\mathrm{Supp}\,(f) = \{\boldsymbol{u}, \boldsymbol{u}_j\}$ with $\boldsymbol{u}_j = \boldsymbol{u} \oplus \boldsymbol{2^{n-j}}$ and, clearly, $\boldsymbol{u} \oplus \boldsymbol{u}_j \neq \boldsymbol{0}$.

Theorem 4: *Let* $f \in \mathcal{B}_n$. *If* $\deg(f) \leq n-2$, *then* $\bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(f)} \boldsymbol{a} = \boldsymbol{0}$.

Proof: Assume that $f(\boldsymbol{x}) = \bigoplus_{j=1}^m f_j(\boldsymbol{x})$ with $f_j(\boldsymbol{x})$, for $j = 1,2,\ldots,m$, a monomial of degree less than or equal to $n-2$.

We proceed by induction over m. For $m = 1$ the result is true by Lemma 1.

Assume that the result is true for $m-1$ and we will prove that it is true for m. Firstly, note that from expression (1)

$$\mathrm{Supp}\,(f) = \bigtriangleup_{j=1}^{m} \mathrm{Supp}\,(f_j) = \left(\bigtriangleup_{j=1}^{m-1} \mathrm{Supp}\,(f_j) \right) \Delta\, \mathrm{Supp}\,(f_m)\,.$$

To simplify the notation, we denote by

$$A = \mathrm{Supp}\,(f)\,, \quad B = \bigtriangleup_{j=1}^{m-1} \mathrm{Supp}\,(f_j)\,, \quad \text{and} \quad C = \mathrm{Supp}\,(f_m)\,.$$

From the properties of the union, intersection and symmetric difference of sets, and from the induction hypothesis and by Lemma 1, we have,

$$\boldsymbol{0} = \bigoplus_{\boldsymbol{b}\in B} \boldsymbol{b} = \bigoplus_{\boldsymbol{b}\in A\cap B} \boldsymbol{b} \oplus \bigoplus_{\boldsymbol{d}\in B\cap C} \boldsymbol{d} \quad \text{and} \quad \boldsymbol{0} = \bigoplus_{\boldsymbol{c}\in C} \boldsymbol{c} = \bigoplus_{\boldsymbol{c}\in A\cap C} \boldsymbol{c} \oplus \bigoplus_{\boldsymbol{e}\in B\cap C} \boldsymbol{e}.$$

 ISBN 13: 978-84-613-5510-5

Now, adding the two previous expressions, we obtain

$$\mathbf{0} = \bigoplus_{\boldsymbol{b}\in A\cap B} \boldsymbol{b} \oplus \bigoplus_{\boldsymbol{c}\in A\cap C} \boldsymbol{c} = \bigoplus_{\boldsymbol{a}\in A} \boldsymbol{a}$$

because $\bigoplus_{\boldsymbol{d}\in B\cap C} \boldsymbol{d} \oplus \bigoplus_{\boldsymbol{e}\in B\cap C} \boldsymbol{e} = \mathbf{0}$. □

The converse of the previous theorem is not true, as we can see in the following example.

Example 1: If $f \in \mathcal{B}_3$ and $\mathrm{Supp}\,(f) = \{\mathbf{3}, \mathbf{5}, \mathbf{6}\}$. We have that $\mathbf{3} \oplus \mathbf{5} \oplus \mathbf{6} = \mathbf{0}$, but, from Theorem 3, $\deg(f) = 3$. So, the converse of Theorem 4 is not true. □

The following result shows that the situation described in the previous example only can appears when $|\mathrm{Supp}\,(f)|$ is odd, that is, when $\deg(f) = n$.

Theorem 5: *Let $f \in \mathcal{B}_n$ such that $|\mathrm{Supp}\,(f)|$ is an even number. If $\bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(f)} \boldsymbol{a} = \mathbf{0}$, then $\deg(f) \leq n-2$.*

Proof: Since $|\mathrm{Supp}\,(f)|$ is even, from Theorem 3, we have that $\deg(f) \leq n-1$.

If $\deg(f) = n-1$, then f has at least one monomial of degree $n-1$. Assume that

$$f(\boldsymbol{x}) = \bigoplus_{j=1}^{m} g_j(\boldsymbol{x}) \oplus h(\boldsymbol{x})$$

with $g_j(\boldsymbol{x})$, for $j = 1, 2, \ldots, m$, a monomial of degree $n-1$ which does not contain the variable x_{i_j} and $\deg(h) \leq n-2$.

Proceeding as in the proof of Theorem 4, taking into account that from Theorem 4, $\bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(h)} \boldsymbol{a} = \mathbf{0}$, and according to the comment previous to Theorem 4, we have that

$$\mathbf{0} = \bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(f)} \boldsymbol{a} = \bigoplus_{j=1}^{m} \bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(g_j)} \boldsymbol{a} \oplus \bigoplus_{\boldsymbol{a}\in\mathrm{Supp}(h)} \boldsymbol{a} = \begin{cases} \bigoplus_{j=1}^{m} \mathbf{2^{n-i_j}}, & \text{if } m \text{ is even,} \\ \boldsymbol{u} \oplus \bigoplus_{j=1}^{m} \mathbf{2^{n-i_j}}, & \text{if } m \text{ is odd} \end{cases}$$

which is a contradiction. Therefore, $\deg(f) \leq n-2$. □

Let $f \in \mathcal{B}_n$ and assume that we know $\mathrm{Supp}\,(f)$. Since $w(f) = |\mathrm{Supp}\,(f)|$, Theorem 3 allows us to ensure that the monomial $x_1 x_2 \cdots x_n$ is part of the ANF of $f(\boldsymbol{x})$ if and only if $|\mathrm{Supp}\,(f)|$ is an odd number. The following theorem establishes a necessary and sufficient condition in order that the ANF of $f(\boldsymbol{x})$ contains any monomial of degree k with $1 \leq k < n$. Before, nevertheless, we need the following result which will facilitate the proof of the above mentioned theorem.

Lemma 2: *Assume that $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ and consider the map $\varphi : \mathbb{F}_2^k \to \mathbb{F}_2^n$ given by $\varphi(y_1, y_2, \ldots, y_k) = (x_1, x_2, \ldots, x_n)$ with*

$$x_l = \begin{cases} 0, & \text{if } l \notin \{i_1, i_2, \ldots, i_k\}, \\ y_j, & \text{if } l = i_j, \text{ for } j = 1, 2, \ldots, k. \end{cases}$$

If $f \in \mathcal{B}_n$ and consider $h \in \mathcal{B}_k$ such that $h(y_1, y_2, \ldots, y_k) = f(\varphi(y_1, y_2, \ldots, y_k))$, then

$$|\mathrm{Supp}\,(h)| = \left|\mathrm{Supp}\,(f) \cap \mathrm{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_k}}\right\}\right|.$$

 ISBN 13: 978-84-613-5510-5

Theorem 6: *Assume that we know* $\mathrm{Supp}\,(f)$ *for* $f \in \mathcal{B}_n$. *The ANF of* $f(\boldsymbol{x})$ *contains the monomial* $x_{i_1}x_{i_2}\cdots x_{i_k}$ *with* $1 \leq k < n$ *if and only if*

$$\left|\mathrm{Supp}\,(f) \cap \mathrm{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_k}}\right\}\right|$$

is an odd number.

Proof: With the notation of Lemma 2, we have that the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_k}$ if and only if the ANF of $h(\boldsymbol{y})$ contains the monomial $y_1y_2\cdots y_k$. Also, from Theorem 3, the ANF of $h(\boldsymbol{y})$ contains the monomial $y_1y_2\cdots y_k$ if and only if $|\mathrm{Supp}\,(h)|$ is an odd number. Finally, from Lemma 2, the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_k}$ if and only if $\left|\mathrm{Supp}\,(f) \cap \mathrm{Span}\left\{\mathbf{2^{n-i_1}}, \mathbf{2^{n-i_2}}, \ldots, \mathbf{2^{n-i_k}}\right\}\right|$ is an odd number. □

Applying successively the previous result, we can determine the degree and the ANF of a Boolean function of which we know its support.

Furthermore, if $1 \leq k < n$, by a similar argument to that followed to obtain expression (4), we can separate the n variables in two sets with k and $n-k$ variables, respectively, as we describe in the following result. The proof is straightforward but large and, therefore, we omit it.

Theorem 7: *Assume that* $f \in \mathcal{B}_n$. *If* $1 \leq k < n$, *then*

$$f(\boldsymbol{y}, \boldsymbol{x}) = \bigoplus_{\boldsymbol{b} \in \mathbb{F}_2^k} \left(\bigoplus_{\boldsymbol{a} \in S(\boldsymbol{b})} f_{\boldsymbol{a}}(\boldsymbol{x}) \right) \boldsymbol{y}^{\boldsymbol{b}}$$

where $f_{\boldsymbol{a}} \in \mathcal{B}_k$, *for* $\boldsymbol{a} \in \mathbb{F}_2^k$, *satisfies* $f_{\boldsymbol{a}}(\boldsymbol{x}) = f(\boldsymbol{a}, \boldsymbol{x})$. *Furthermore,*

1. $\mathrm{Supp}\,(f_{\boldsymbol{a}}) = \{\boldsymbol{v} \in \mathbb{F}_2^{n-k} \mid (\boldsymbol{a}, \boldsymbol{v}) \in \mathrm{Supp}\,(f)\}$,
2. $\mathrm{Supp}\left(\bigoplus_{\boldsymbol{a} \in S(\boldsymbol{b})} f_{\boldsymbol{a}}\right) = \bigtriangleup_{\boldsymbol{a} \in S(\boldsymbol{b})} \mathrm{Supp}\,(f_{\boldsymbol{a}})$, *for all* $\boldsymbol{b} \in \mathbb{F}_2^k$,
3. $\deg(f) = \max_{\boldsymbol{b} \in \mathbb{F}_2^k} \left\{ \deg\left(\bigoplus_{\boldsymbol{a} \in S(\boldsymbol{b})} f_{\boldsymbol{a}}\right) + w(\boldsymbol{b}) \right\}$.

Before to continue, we show an example that will help us to understand the process to follow.

Example 2: Let $f \in \mathcal{B}_5$ such that

$$\mathrm{Supp}\,(f) = \{\mathbf{6}, \mathbf{7}, \mathbf{12}, \mathbf{13}, \mathbf{16}, \mathbf{17}, \mathbf{18}, \mathbf{20}, \mathbf{21}, \mathbf{23}, \mathbf{24}, \mathbf{25}, \mathbf{26}, \mathbf{30}\}.$$

Since $|\mathrm{Supp}\,(f)|$ is even and

$$\mathbf{6} \oplus \mathbf{7} \oplus \mathbf{12} \oplus \mathbf{13} \oplus \mathbf{16} \oplus \mathbf{17} \oplus \mathbf{18} \oplus \mathbf{20} \oplus \mathbf{21} \oplus \mathbf{23} \oplus \mathbf{24} \oplus \mathbf{25} \oplus \mathbf{26} \oplus \mathbf{30} = \mathbf{0}$$

from Theorem 5, we have that $\deg(f) \leq 5 - 2 = 3$.

 ISBN 13: 978-84-613-5510-5

Now, according to Theorem 7.3, we have that

$$\deg(f) = \max\{\deg(f_0), \deg(f_0 \oplus f_1) + 1\}$$

with $f_0, f_1 \in \mathcal{B}_4$ such that

$$f_0(x_2, x_3, x_4, x_5) = f(0, x_2, x_3, x_4, x_5), \quad f_1(x_2, x_3, x_4, x_5) = f(1, x_2, x_3, x_4, x_5),$$

and, from Theorem 7.1,

$$\mathrm{Supp}(f_0) = \{\mathbf{6}, \mathbf{7}, \mathbf{12}, \mathbf{13}\} \quad \text{and} \quad \mathrm{Supp}(f_1) = \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{7}, \mathbf{8}, \mathbf{9}, \mathbf{10}, \mathbf{14}\}.$$

Therefore,

$$\mathrm{Supp}(f_0 \oplus f_1) = \mathrm{Supp}(f_0) \,\Delta\, \mathrm{Supp}(f_1) = \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{6}, \mathbf{8}, \mathbf{9}, \mathbf{10}, \mathbf{12}, \mathbf{13}, \mathbf{14}\}.$$

In addition, since $\mathbf{6} \oplus \mathbf{7} \oplus \mathbf{12} \oplus \mathbf{13} = \mathbf{0}$ and

$$\mathbf{0} \oplus \mathbf{1} \oplus \mathbf{2} \oplus \mathbf{4} \oplus \mathbf{5} \oplus \mathbf{6} \oplus \mathbf{8} \oplus \mathbf{9} \oplus \mathbf{10} \oplus \mathbf{12} \oplus \mathbf{13} \oplus \mathbf{14} = \mathbf{0},$$

from Theorem 5, we have that

$$\deg(f_0) \leq 4 - 2 = 2 \quad \text{and} \quad \deg(f_0 \oplus f_1) \leq 4 - 2 = 2$$

and, therefore $\deg(f) \leq \max\{2, 2 + 1\} = 3$.

Now, again from Theorem 7.3, we have that

$$\deg(f) = \max\{\deg(f_{\mathbf{0}}), \deg(f_{\mathbf{0}} \oplus f_{\mathbf{1}}) + 1, \deg(f_{\mathbf{0}} \oplus f_{\mathbf{2}}) + 1, \deg(f_{\mathbf{0}} \oplus f_{\mathbf{1}} \oplus f_{\mathbf{2}} \oplus f_{\mathbf{3}}) + 2\}$$

with $f_{\mathbf{0}}, f_{\mathbf{1}}, f_{\mathbf{2}}, f_{\mathbf{3}} \in \mathcal{B}_3$ such that

$$f_{\mathbf{0}}(x_3, x_4, x_5) = f(0, 0, x_3, x_4, x_5), \quad f_{\mathbf{1}}(x_3, x_4, x_5) = f(0, 1, x_3, x_4, x_5),$$
$$f_{\mathbf{2}}(x_3, x_4, x_5) = f(1, 0, x_3, x_4, x_5), \quad f_{\mathbf{3}}(x_3, x_4, x_5) = f(1, 1, x_3, x_4, x_5),$$

and, from Theorem 7.3

$$\mathrm{Supp}(f_{\mathbf{0}}) = \{\mathbf{6}, \mathbf{7}\}, \quad \mathrm{Supp}(f_{\mathbf{1}}) = \{\mathbf{4}, \mathbf{5}\},$$
$$\mathrm{Supp}(f_{\mathbf{2}}) = \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{7}\} \quad \text{and} \quad \mathrm{Supp}(f_{\mathbf{3}}) = \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{6}\}$$

in which case

$$\mathrm{Supp}(f_{\mathbf{0}} \oplus f_{\mathbf{1}}) = \mathrm{Supp}(f_{\mathbf{0}}) \,\Delta\, \mathrm{Supp}(f_{\mathbf{1}}) = \{\mathbf{4}, \mathbf{5}, \mathbf{6}, \mathbf{7}\},$$
$$\mathrm{Supp}(f_{\mathbf{0}} \oplus f_{\mathbf{2}}) = \mathrm{Supp}(f_{\mathbf{0}}) \,\Delta\, \mathrm{Supp}(f_{\mathbf{2}}) = \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{6}\},$$
$$\mathrm{Supp}(f_{\mathbf{0}} \oplus f_{\mathbf{1}} \oplus f_{\mathbf{2}} \oplus f_{\mathbf{3}}) = \mathrm{Supp}(f_{\mathbf{0}}) \,\Delta\, \mathrm{Supp}(f_{\mathbf{1}}) \,\Delta\, \mathrm{Supp}(f_{\mathbf{2}}) \,\Delta\, \mathrm{Supp}(f_{\mathbf{3}}) = \emptyset.$$

Moreover, since

$$\mathbf{6} \oplus \mathbf{7} = \mathbf{1} \neq \mathbf{0}, \quad \mathbf{4} \oplus \mathbf{5} \oplus \mathbf{6} \oplus \mathbf{7} = \mathbf{0} \quad \text{and} \quad \mathbf{0} \oplus \mathbf{1} \oplus \mathbf{2} \oplus \mathbf{4} \oplus \mathbf{5} \oplus \mathbf{6} = \mathbf{4} \neq \mathbf{0}$$

from Theorem 5, we have that

$$\deg(f_{\mathbf{0}}) = 3 - 1 = 2, \quad \deg(f_{\mathbf{0}} \oplus f_{\mathbf{1}}) \leq 3 - 2 = 1, \quad \deg(f_{\mathbf{0}} \oplus f_{\mathbf{2}}) = 3 - 1 = 2$$

and $f_{\mathbf{0}} \oplus f_{\mathbf{1}} \oplus f_{\mathbf{2}} \oplus f_{\mathbf{3}} = 0$, therefore, $\deg(f) = 3$. □

 ISBN 13: 978-84-613-5510-5

4 More results

In this section we introduce some results that allow us to simplify the process described in the previous section.

The following result establishes that any k-dimensional linear subspace of $\mathbb{F}_2^n$ (or the complement of any k-dimensional linear subspace of $\mathbb{F}_2^n$) is the support of a Boolean function of n variables with degree $n-k$.

Theorem 8: *Assume that $1 \leq k \leq n$. If F or $\mathbb{F}_2^n \setminus F$ is a k-dimensional linear subspace of $\mathbb{F}_2^n$, then there exists $f \in \mathcal{B}_n$ such that $\deg(f) = n-k$ and $F = \mathrm{Supp}(f)$.*

Proof: Firstly, assume that F is a k-dimensional linear subspace of $\mathbb{F}_2^n$. Clearly, the map $f : \mathbb{F}_2^n \to \mathbb{F}_2$ given by

$$f(\boldsymbol{x}) = \begin{cases} 1, & \text{if } \boldsymbol{x} \in F, \\ 0, & \text{if } \boldsymbol{x} \notin F, \end{cases}$$

is a Boolean function of n variables whose support is F.

Assume that $n-k+1 \leq l \leq n$ and that the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_l}$; then, by Theorem 6 we have that

$$\left| F \cap \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\} \right|$$

is an odd number. Nevertheless, since

$$\begin{aligned} &\dim\left(F \cap \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\}\right) \\ &\quad = \dim F + \dim \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\} \\ &\qquad - \dim\left(F + \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\}\right) \geq k + l - n \geq 1, \end{aligned}$$

necessarily

$$\left| F \cap \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\} \right| = 2^{\dim(F \cap \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_l}} \right\})}$$

is an even number. So, we have a contradiction. Therefore, the ANF of $f(\boldsymbol{x})$ does not contain any monomial of degree l and, consequently, $\deg(f) \leq n-k$.

Assume now that $\{\boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_k}\}$ is a basis of F and complete such basis, with the vectors of the canonical basis, to obtain a basis

$$\left\{ \boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_k, \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_{n-k}}} \right\}$$

of $\mathbb{F}_2^n$. Clearly $F \cap \mathrm{Span}\left\{ \boldsymbol{2^{n-i_1}}, \boldsymbol{2^{n-i_2}}, \ldots, \boldsymbol{2^{n-i_{n-k}}} \right\} = \{\boldsymbol{0}\}$ and, by Theorem 6, the ANF of $f(\boldsymbol{x})$ contains the monomial $x_{i_1}x_{i_2}\cdots x_{i_{n-k}}$; so $\deg(f) \geq n-k$.

Now, from this inequality and the previous one, we have that $\deg(f) = n-k$.

Assume now that $G = \mathbb{F}_2^n \setminus F$ is a k-dimensional linear subspace of $\mathbb{F}_2^n$. Then, from the above part, there exists $g \in \mathcal{B}_n$ such that $\deg(g) = n-k$ and $G = \mathrm{Supp}(g)$. Let $f \in \mathcal{B}_n$ such that $f = 1 \oplus g$, clearly, $\deg(f) = \deg(g)$ and $\mathrm{Supp}(f) = \mathbb{F}_2^n \setminus \mathrm{Supp}(g)$; that is $\deg(f) = n-k$ and $F = \mathrm{Supp}(f)$. □

 ISBN 13: 978-84-613-5510-5

The converse of the above theorem is not true in general as we can see in Example 3 below. Nevertheless, if $k = n$, then $F = \mathbb{F}_2^n$ is the support of the constant function $f(\boldsymbol{x}) = 1$ whose degree is 0. Furthermore, if $k = n-1$, then the converse of Theorem 8 also holds as we can see in the following result.

Theorem 9: *Assume that $F \subseteq \mathbb{F}_2^n$. Then F or $\mathbb{F}_2^n \setminus F$ is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$ if and only if there exists $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$.*

Proof: If F or $\mathbb{F}_2^n \setminus F$ is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$, by Theorem 8, there exists $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$.

Conversely, let $f \in \mathcal{B}_n$ such that $\deg(f) = 1$ and $F = \mathrm{Supp}(f)$. On the one hand

$$f(\boldsymbol{x}) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n$$

for some $a_0, a_1, a_2, \ldots, a_n \in \mathbb{F}_2$, and clearly

$$S = \{(u_1, u_2, \ldots, u_n) \in \mathbb{F}_2^n \mid a_1 u_1 \oplus a_2 u_2 \oplus \cdots \oplus a_n u_n = 0\}$$

is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$. On the other hand, it is easy to see that $S = F$, if $a_0 = 1$, and $S = \mathbb{F}_2^n \setminus F$, if $a_0 = 0$. □

Next result establishes that Theorem 8 also holds if we change "linear subspace" by "affine subspace". The proof is straightforward and therefore we omit it.

Corollary 1: *Assume that $1 \leq k \leq n$. If F or $\mathbb{F}_2^n \setminus F$ is a k-dimensional affine subspace of $\mathbb{F}_2^n$, then there exists $f \in \mathcal{B}_n$ such that $\deg(f) = n - k$ and $F = \mathrm{Supp}(f)$.*

Theorem 9 also holds if we change "linear subspace" by "affine subspace". But in this case, the proof follows by the fact that if F is an $(n-1)$-dimensional linear subspace of $\mathbb{F}_2^n$, then

$$\mathbb{F}_2^n \setminus F = \boldsymbol{a} \oplus F \quad \text{for all } \boldsymbol{a} \in \mathbb{F}_2^n \setminus F$$

is an $(n-1)$-dimensional affine subspace of $\mathbb{F}_2^n$. This can not be true if $\dim F = k \neq n-1$, because $2^n - 2^k = 2^k(2^{n-k} - 1)$ is not a power of 2.

The following example shows how we can use the above results to improve the process described in the above section.

Example 3: Let $f \in \mathcal{B}_5$ the Boolean function of Example 2. Note that $|\mathrm{Supp}(f)| = 14$ and $\left|\mathbb{F}_2^5 \setminus \mathrm{Supp}(f)\right| = 18$, so neither $\mathrm{Supp}(f)$ nor $\mathbb{F}_2^5 \setminus \mathrm{Supp}(f)$ can be a linear subspace nor an affine subspace of $\mathbb{F}_2^5$.

After some computations, we obtained in Example 2 that

$$\deg(f) = \max\{\deg(f_0), \deg(f_0 \oplus f_1) + 1\}$$

with $f_0, f_1 \in \mathcal{B}_4$ such that

$$f_0(x_2, x_3, x_4, x_5) = f(0, x_2, x_3, x_4, x_5), \quad f_1(x_2, x_3, x_4, x_5) = f(1, x_2, x_3, x_4, x_5),$$

 ISBN 13: 978-84-613-5510-5

and that

$$\mathrm{Supp}\,(f_0) = \{\mathbf{6},\mathbf{7},\mathbf{12},\mathbf{13}\}, \quad \mathrm{Supp}\,(f_0 \oplus f_1) = \{\mathbf{0},\mathbf{1},\mathbf{2},\mathbf{4},\mathbf{5},\mathbf{6},\mathbf{8},\mathbf{9},\mathbf{10},\mathbf{12},\mathbf{13},\mathbf{14}\}.$$

Note that none of the sets $\mathrm{Supp}\,(f_0)$, $\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0)$, $\mathrm{Supp}\,(f_0 \oplus f_1)$ and $\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0 \oplus f_1)$ can be linear subspaces of $\mathbb{F}_2^4$. Nevertheless

$$\mathrm{Supp}\,(f_0) = \mathbf{6} \oplus \{\mathbf{0},\mathbf{1},\mathbf{10},\mathbf{11}\} = \mathbf{6} \oplus \mathrm{Span}\,\{\mathbf{1},\mathbf{10}\},$$
$$\mathbb{F}_2^4 \setminus \mathrm{Supp}\,(f_0 \oplus f_1) = \{\mathbf{3},\mathbf{7},\mathbf{11},\mathbf{15}\} = \mathbf{3} \oplus \{\mathbf{0},\mathbf{4},\mathbf{8},\mathbf{12}\} = \mathbf{3} \oplus \mathrm{Span}\,\{\mathbf{4},\mathbf{8}\}$$

are affine subspaces of dimension 2. So, by Corollary 1,

$$\deg(f_0) = 4 - 2 = 2 \quad \text{and} \quad \deg(f_0 \oplus f_1) = 4 - 2 = 2$$

and, therefore $\deg(f) = \max\{2, 2+1\} = 3$.

Remember that in Example 2 we obtained that

$$\deg(f_0) \leq 4 - 2 = 2 \quad \text{and} \quad \deg(f_0 \oplus f_1) \leq 4 - 2 = 2$$

and, therefore $\deg(f) \leq \max\{2, 2+1\} = 3$. □

Acknowledgements

This work was partially supported by Spanish grant MTM2008-06674-C02-01. The research of Verónica Requena was also supported by a grant of the Vicerectorat d'Investigació, Desenvolupament i Innovació of the Universitat d'Alacant for PhD students.

References

[1] C. M. Adams. Constructing symmetric ciphers using the CAST design procedure. *Designs, Codes and Cryptography*, **12**: 283–316 (1997).

[2] Y. Borissov, A. Braeken, S. Nikova and B. Preneel. On the covering radii of binary Reed-Muller codes in the set of resilient Boolean functions. *IEEE Transactions on Information Theory*, **51(3)**: 1182–1189 (2005).

[3] A. Braeken, V. Nikov, S. Nikova and B. Preneel. On Boolean functions with generalized cryptographic properties. In A. Canteaut and K. Viswanathan (editors), *Progress in Cryptology – INDOCRYPT 2004*, volume 3348 of *Lecture Notes in Computer Science*, pages 120–135. Springer-Verlag, Berlin, 2004.

[4] C. Carlet and Y. Tarannikov. Covering sequences of Boolean functions and their cryptographic significance. *Designs, Codes and Cryptography*, **25**: 263–279 (2002).

[5] K. C. Gupta and P. Sarkar. Improved construction of nonlinear resilient S-boxes. *IEEE Transactions on Information Theory*, **51(1)**: 339–348 (2005).

 ISBN 13: 978-84-613-5510-5

[6] K. Kurosawa, T. Iwata and T. Yoshiwara. New covering radius of Reed-Muller codes for t-resilient functions. *IEEE Transactions on Information Theory*, **50(3)**: 468–475 (2004).

[7] K. Kurosawa and R. Matsumoto. Almost security of cryptographic Boolean functions. *IEEE Transactions on Information Theory*, **50(11)**: 2752–2761 (2004).

[8] M. Matsui. Linear cryptanalysis method for DES cipher. In T. Helleseth (editor), *Advances in Cryptology – EUROCRYPT '93*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer-Verlag, Berlin, 1994.

[9] W. Meier and O. Staffelbach. Nonlinearity criteria for cryptographic functions. In J. Quisquater and J. Vandewalle (editors), *Advances in Cryptology – EUROCRYPT'89*, volume 434 of *Lecture Notes in Computer Science*, pages 549–562. Springer-Verlag, Berlin, 1990.

[10] K. Nyberg. Perfect nonlinear S-boxes. In D. W. Davies (editor), *Advances in Cryptology – EUROCRYPT '91*, volume 547 of *Lecture Notes in Computer Science*, pages 378–386. Springer-Verlag, Berlin, 1991.

[11] D. Olejár and M. Stanek. On cryptographic properties of random Boolean functions. *Journal of Universal Computer Science*, **4(8)**: 705–717 (1998).

[12] E. Pasalic and T. Johansson. Further results on the relation between nonlinearity and resiliency for Boolean functions. In M. Walker (editor), *Crytography and Coding*, volume 1746 of *Lecture Notes in Computer Science*, pages 35–44. Springer-Verlag, Berlin, 1999.

[13] B. Preneel, W. Van Leekwijck, L. Van Linden, R. Govaerts and J. Vandewalle. Propagation characteristics of Boolean functions. In I. B. Damgard (editor), *Advances in Cryptology – EUROCRYPT'90*, volume 473 of *Lecture Notes in Computer Science*, pages 161–173. Springer-Verlag, Berlin, 1991.

[14] C. Qu, J. Seberry and J. Pieprzyk. On the symmetric property of homogeneous Boolean functions. In J. Pieprzyk, R. Safavi-Naini and J. Seberry (editors), *Proceedings of the Australasian Conference on Information Security and Privacy – ACISP'99*, volume 1587 of *Lecture Notes in Computer Science*, pages 26–35. Springer-Verlag, Berlin, 1999.

[15] R. A. Rueppel. *Analysis and Design of Stream Ciphers*. Springer Verlag, New York, NY, 1986.

[16] R. A. Rueppel and O. J. Staffelbach. Products of linear recurring sequences with maximum complexity. *IEEE Transactions on Information Theory*, **33(1)**: 124–131 (1987).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Random generation on order polytopes and fuzzy measures

E. F. Combarro[1], I. Díaz[1] and P. Miranda[2]

[1] *Department of Informatics, University of Oviedo*

[2] *Department of Statistics and Operations Research, Complutense University of Madrid*

emails: elias@aic.uniovi.es, sirene@uniovi.es, pmiranda@mat.ucm.es

Abstract

In this paper we deal with the problem of obtaining a random procedure for generating points in an order polytope. For this, we use the fact that it is easy to make a triangulation in order polytopes in a way such that all simplices have the same hypervolume. As an application, this allows to build a procedure to generate fuzzy measures in a random way.

Key words: Random generation, order polytopes, linear extension, triangulation, fuzzy measures

1 Motivation

Consider $X = \{x_1, ..., x_n\}$ a finite referential set. The set of **non-additive measures** [8], **fuzzy measures** [23] or **capacities** [5] over X, denoted by $\mathcal{FM}(X)$, is the set of functions $\mu : \mathcal{P}(X) \to [0, 1]$ satisfying

- $\mu(\emptyset) = 0, \mu(X) = 1$,
- $\mu(A) \leq \mu(B)$ for all $A, B \in \mathcal{P}(X)$ such that $A \subseteq B$.

Fuzzy measures have been applied to many different fields, as Multicriteria Decision Making, Decision Under Uncertainty and Under Risk, Game Theory, Welfare Theory or Combinatorics (see [11] for a review of theoretical and practical applications of fuzzy measures). Moreover, they are included in the field of Aggregation Operators, that constitutes a major research topic nowadays [10].

An interesting problem arising in the practical use of fuzzy measures is the identification of the fuzzy measure modeling the situation. The problem in this case is that the number of coefficients needed to define a fuzzy measure is $2^n - 2$ for a referential of

 ISBN 13: 978-84-613-5510-5

cardinality n, so that the complexity grows exponentially. The problem of identification has attracted the attention of many researchers in the field; for example, different procedures (in many cases restricted to a subfamily of fuzzy measures) can be found in [1], [18], [6] among many others. The information background used by each method also varies; in some of the previous methods, a sample data is supposed; others used a questionnaire; the data can be numerical or just ordinal, and so on.

Once an algorithm is suggested, it is necessary to test its performance. In many of the previous references a fuzzy measure is considered, some data are generated (possibly with some random noise), and the corresponding procedure is applied. This also serves as an example of applicability. If the procedure works properly, it should obtain a fuzzy measure *near* the initial measure.

However, in order to evaluate the performance of a procedure, it should be tested in many different cases, and the fuzzy measure considered in each case should be chosen randomly. Surprisingly, to our knowledge there is not a method to generate randomly a fuzzy measure. The aim of this paper is to fill this gap.

From the definition of fuzzy measure, it can be seen that the set of fuzzy measures is a polytope. So, the problem reduces to derive a procedure for the uniform random generation in a polytope. However, this is a complex problem and several methods have been suggested to cope with it. Indeed, the uniform random generation in a polytope is a hot problem in Computer Sciences (see for example [9] and [21]).

In our case, we will use the fact that the set of fuzzy measures is a special type of polytope called order polytopes. Then, as explained in the paper, the problem simplifies and it is possible to obtain an efficient procedure.

2 Order polytopes

Let us recall the basic notions about order polytopes. Consider a finite **poset** $(P, \preceq)$ (or P for short) of p elements. We will denote the subsets of P by capital letters $A, B, ...$ and also $A_1, A_2, ...$; elements of P are denoted a, b, and so on. If A is a subset of P, it inherits a structure of poset from the restriction of $\preceq$ to A. In this case, we say that A is a **subposet** of P. If two elements a, b of P satisfy $a \preceq b$ or $b \preceq a$, we say that they are *comparable*. A poset $(P, \preceq)$ is a **chain** if for any $a, b \in P$, either $a \preceq b$ or $b \preceq a$. A poset can be represented by the so-called *Hasse diagram*, where $a \preceq b$ if and only if there is a sequence of connected lines upwards from a to b. An example of Hasse diagram is given in Figure 1.

A subset I of P is an **ideal** or **downset** if for any $a \in I$ and any $b \in P$ such that $b \preceq a$, it follows that $b \in I$. We will denote ideals by $I_1, I_2, ...$ Notice that with this definition the empty set is an ideal. The dual notion of an ideal is a **filter** or **upset**, i.e., a set that contains all upper bounds of its elements. We will denote by $\mathcal{I}(P)$ the set of all ideals of poset P.

Given two ideals I_1 and I_2 of P, we can define $I_1 \cup I_2$ and $I_1 \cap I_2$ as the usual union and intersection of subsets. It is trivial to check that $I_1 \cup I_2$ and $I_1 \cap I_2$ are also ideals in P. In fact, the set of all ideals of P forms a lattice under set inclusion called

 ISBN 13: 978-84-613-5510-5

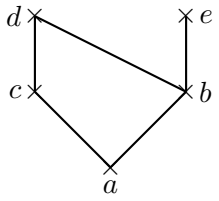


Figure 1: Example of Hasse diagram of a poset.

the **ideal lattice** of P (see [2]). The ideal lattice of the poset presented in Figure 1 is given in Figure 2.

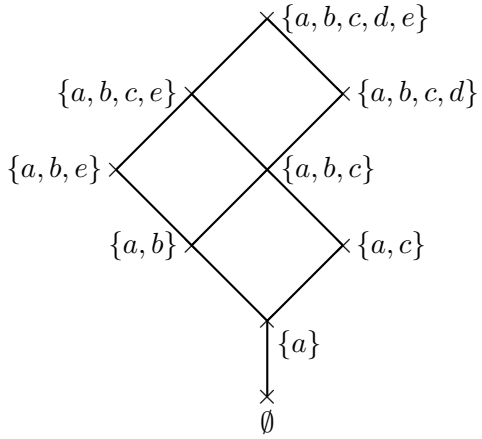


Figure 2: Ideal lattice corresponding to poset of Figure 1.

Let us now turn to order polytopes. Given a poset $(P, \preceq)$, it is possible to associate to P, in a natural way, a polytope $O(P)$ in $\mathbb{R}^p$, called the **order polytope** of P (cf. [22]). The polytope $O(P)$ is formed by the p-uples f of real numbers indexed by the elements of P satisfying

- $0 \leq f(a) \leq 1$ for every a in P,
- $f(a) \leq f(b)$ whenever $a \preceq b$, $a, b \in P$.

Thus, the polytope $O(P)$ consists in (the p-uples of images of) the order-preserving functions from P to $[0, 1]$. It is a well-known fact [22] that $O(P)$ is a 0/1-polytope, i.e. its extreme points are all in $\{0, 1\}^p$. Applied to distributive lattices, the notion of order polytope has been also defined in [13] with the name of **geometric realization**.

 ISBN 13: 978-84-613-5510-5

From the point of view of order polytopes, $\mathcal{FM}(X)$ is the order polytope of the poset $(P, \preceq)$ where $P = \mathcal{P}(X)\backslash\{X, \emptyset\}$ and $\preceq$ is the inclusion between subsets [7]. So, the problem reduces to obtain a random procedure for generating points in an order polytope. This is treated in next section.

3 Algorithm for random generation on order polytopes

There are several procedures to generate random points in a polytope: the grid method [9], sweep-plane method [15], and triangulation methods [9]. In this paper we have chosen the triangulation method for the properties that order polytopes share; more details become apparent below.

Consider $n+1$ affine independent points in $\mathbf{R}^m, m \geq n$, i.e. $n+1$ points of $\mathbf{R}^m$ in general position. The convex hull of these points is called a **simplex**. This notion is a generalization of the notion of triangle for the n-dimensional space.

The *triangulation method* is based on decomposing the polytope into simplices, choosing one of them with probabilities proportional to their volumes and, finally, generating a random tuple on the simplex.

The random generation in simplices is very simple and fast [20]. Indeed, if $x_1, ..., x_{n+1}$ are the vertices of the simplex, it suffices to generate random values $\alpha_1, ..., \alpha_{n+1} \in [0, 1]$ such that $\sum_{i=1}^{n+1} \alpha_i = 1$; the point generated is $\sum_{i=1}^{n+1} \alpha_i x_i$. However, it is not easy to split a polytope in simplices. Moreover, even if we are able to decompose the polytope in a suitable way, we have to deal with the problem of determining the volume of each simplex in order to properly select one of them. This is the Achilles heel of the triangulation method and the reason for which it is not very popular [21].

However, we will see below that for order polytopes the triangulation method can be adapted in a way such that it works properly.

3.1 Step 1: Triangulation of an order polyotpe

Consider a poset $(P, \preceq)$. An **extension** $(P, \preceq')$ of $(P, \preceq)$ is another poset on the same referential such that if $a \preceq b$, then $a \preceq' b$. A **linear extension** is an extension that is a chain. The linear extensions of the poset of Figure 1 are given in Figure 3.

The triangulation of an order polytope that we are going to consider is based on the following result (see [17], pag. 304):

Theorem 1 *Let $(P, \preceq)$ be a poset.*

- *If $\leq$ is a linear ordering on P, then the corresponding order polytope $O(P, \leq)$ is a simplex of volume $\frac{1}{n!}$.*
- *For any partial ordering $\preceq$ on P, the simplices of the order polytope of $(P, \leq)$, where $\leq$ is a linear extension of $\preceq$, cover the order polytope $O(P, \preceq)$ and have disjoint interiors. Hence, $vol(O(P, \preceq)) = \frac{1}{n!}e(\preceq)$, where $e(\preceq)$ is the number of linear extensions that are compatible with $\preceq$.*

 ISBN 13: 978-84-613-5510-5

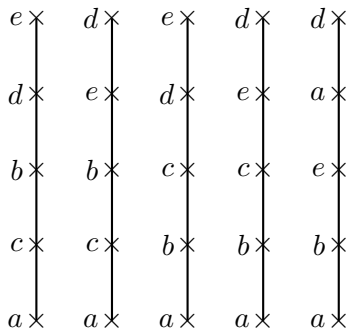


Figure 3: Linear extensions of poset of Figure 1.

These results are also outlined in [22]. From this theorem, we can obtain the following conclusions:

- It suffices to obtain the number of linear extensions of the poset in order to determine the volume of the corresponding order polytope.
- Any linear extension provides a simplex included in the order polytope. Moreover, these simplices have disjoint interiors. As the probability of generating a point in the border of the simplex is zero, we can arbitrarily assign the borders to any simplex, so that this determines a partition of the order polytope in simplices.
- All the simplices generated by linear extensions have exactly the same volume.

Consequently, it suffices to generate randomly a linear extension of $\preceq$ and then generate a point in the corresponding simplex. Therefore, the problem of random generation in an order polytope reduces to generate randomly a linear extension of the poset. This is achieved in next subsection.

3.2 Step 2: Generating linear extensions of a poset

The problem of randomly generating a linear extension of a poset is a $\sharp$P-problem [3] . We will use the algorithm developed in [16]. This algorithm performs in general better than the one developed in [19]; other algorithms that approximate randomness has been suggested in [14] and [4], but we have preferred the previous approach because it provides an exact method.

The idea of the algorithm is the following: first, we construct the ideal lattice of the poset. Next, from the ideal lattice, a random linear extension is generated; for this, it is used the fact that a linear extension consists exactly in a path from the source (the empty ideal) to the sink (the whole poset).

 ISBN 13: 978-84-613-5510-5

The use of the ideal lattice instead of directly enumerating all linear extensions is justified by the fact that the number of linear extensions is in general much larger than the number of ideals, as next table, obtained in [16], shows:

$\|P\|$	Linear extensions	Ideals
5	6	9
10	5.4×10^2	33
15	2.39×10^5	148
20	1.13×10^9	518
25	1.07×10^{12}	953
30	7.83×10^{14}	1406
35	5.57×10^{17}	2637

The posets considered in the table have been obtained by choosing uniformly at random n points out of a two-dimensional grid of size 20 by 20, equipped with the usual ordering. Of course, for other polytopes these figures could vary, but it can be seen that in general the number of linear extensions grows much faster than the number of ideals.

An algorithm for generating the ideal lattice of a poset appears in [12]. The idea is to use the so-called *spanning tree of the lattice*; in [12], an algorithm to build this spanning tree is provided.

When the spanning tree is obtained, we can build the corresponding lattice in an efficient way.

Once the ideal lattice is built, we proceed to generate a linear extension. A such algorithm appears in [16].

4 Conclusions

In this paper we have obtained an algorithm for the random generation of points in order polytopes. The algorithm is based on the fact that order polytopes can be studied in terms of the subjacent poset, thus reducing the complexity of the procedure. The main point is the fact that order polytopes are easy to triangulize using linear extensions. In addition, it is easy to generate linear extensions in a random way from the lattice of ideals.

As a straightforward application, this provides us with a method for generating fuzzy measures in a random way. We think that this could be useful to compare the different approaches of identifying a fuzzy measure from sample data. Moreover, this also applies to any subfamily of fuzzy measures that is an order polytope, as for example p-symmetric measures [7].

As a future work, we intend to compare the different methods of identification of fuzzy measures. We also intend to improve our algorithm by comparing the different procedures that exist in the literature for generating linear extensions.

 ISBN 13: 978-84-613-5510-5

Acknowledgement

This research has been supported in part by grant numbers MTM2007-61193, MTM2009-10072 and BSCH-UCM910707, and by MEC and FEDER grant TIN2007-61273.

References

[1] G. Beliakov, R. Mesiar, and L. Valášková. Fitting generated aggregation operators to empirical data. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 12(2):219–236, 2004.

[2] G. Birkhoff. *Lattice Theory, 3rd ed.* Vol. 25 of AMS Colloquium Publications. American Mathematical Society, 1967.

[3] G. Brightwell and P. Winkler. Counting linear extensions. *Order*, 8(3):225–242, 1991.

[4] R. Bubley and M. Dyer. Faster random generation of linear extensions. *Discrete Mathematics*, 20:81–88, 1999.

[5] G. Choquet. Theory of capacities. *Annales de l'Institut Fourier*, (5):131–295, 1953.

[6] E. F. Combarro and P. Miranda. Identification of fuzzy measures from sample data with genetic algorithms. *Computers and Operations Research*, 33(10):3046–3066, 2006.

[7] E. F. Combarro and P. Miranda. Adjacency on the order polytope with applications to the theory of fuzzy measures. *Fuzzy Sets and Systems*, 180:384–398, 2010.

[8] D. Denneberg. *Non-additive measures and integral*. Kluwer Academic, Dordrecht (the Netherlands), 1994.

[9] L. Devroye. *Non-uniform random variate generation*. Springer-Verlag, New York, 1986.

[10] M. Grabisch, J.-L. Marichal, R. Mesiar, and E. Pap. *Aggregation functions*. Cambridge University Press, 2009.

[11] M. Grabisch, T. Murofushi, and M. Sugeno, editors. *Fuzzy Measures and Integrals-Theory and Applications*. Number 40 in Studies in Fuzziness and Soft Computing. Physica–Verlag, Heidelberg (Germany), 2000.

[12] M. Habib, R. Medina, L. Nourine, and G. Steiner. Efficient algorithms on distributive lattices. *Discrete Applied Mathematics*, 110:169–187, 2001.

[13] G. Koshevoy. Distributive lattices and product of capacities. *Journal of Mathematical Analysis and Applications*, 219:427–441, 1998.

 ISBN 13: 978-84-613-5510-5

[14] D. Lerche and P.B. Sørensen. Evaluation of the ranking probabilities for partial orders based on random linear extensions. *Chemosphere*, (53):981–992, 2004.

[15] J. Leydold and W. Hörmann. A Sweep-Plane Algorithm for Generating Random Tuples in Simple Polytopes. *J. Math. Comp.*, (67):1617–1635, 1998.

[16] Karel De Loof, Hans De Meyer, and Bernard De Baets. Exploiting the lattice of ideals representation of a poset. *Fundam. Inf.*, 71(2,3):309–321, 2006.

[17] Jiri Matousek. *Lectures on Discrete Geometry*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2002.

[18] P. Miranda, M. Grabisch, and P. Gil. Identification of non-additive measures from sample data. In R. Kruse G. Della Riccia, D. Dubois and H.-J. Lenz, editors, *Planning based on Decision Theory*, volume 472 of *CISM Courses and Lectures*, pages 43–63. Springer Verlag, 2003.

[19] G. Pruesse and F. Ruskey. Generating linear extensions fast. *SIAM Journal on Computing*, (23):373–386, 1994.

[20] R.Y. Rubinstein and D.P. Kroese. *Simulation and the Monte Carlo method*. Wiley-Interscience, 2007.

[21] E. Schmerling. An Algorithm for Generating Sequences of Random Tuples on Special Simple Polytopes. In *Proceeding of the International Conference on Computational and Mathematical Methods in Science and Engineering (CMMSE2009)*, pages 989–994, Oviedo (Spain), July 2009.

[22] R. Stanley. Two poset polytopes. *Discrete Comput. Geom.*, 1(1):9–23, 1986.

[23] M. Sugeno. *Theory of fuzzy integrals and its applications*. PhD thesis, Tokyo Institute of Technology, 1974.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

On Commutative Semifields of Dimension 4

Elías F. Combarro[1], I.F. Rúa[2] and José Ranilla[1]

[1] *Departamento de Informática, University of Oviedo, Spain*

[2] *Departamento de Matemáticas, University of Oviedo, Spain*

emails: elias@aic.uniovi.es, rua@uniovi.es, ranilla@uniovi.es

Abstract

Commutative semifields of dimension 4 over the center are considered. Properties of these objects are presented, and a computational based classification for small orders is provided.

Key words: Finite semifield, Finite division ring, Projective planes

1 Introduction

The classification of finite nonassociative division rings (commonly known as **finite semifields** [3]) is not only relevant from an algebraic point of view because of their connections to projective semifield planes [11], coding theory [8, 9], combinatorics [7] and graph theory [12].

A finite semifield (or finite division ring) D is a finite nonassociative ring with identity such that the set $D^* = D \setminus \{0\}$ is closed under the product. In case it has no identity it is known as a **presemifield**. The survey [10] provides a good introduction to the topic together with a state of the art on the classification of these rings. In particular, it is remarked the singular situation of odd characteristic: the number of *different* semifields in such a characteristic is relatively small compared to the actual number of different construction of this type of semifields. So, it seems that different constructions yield the same semifields.

On the other hand, the classification of these objects is far from being an easy task. Recent results on the classification of small semifields ([6, 13, 2]) show the existence of a relatively big number of unknown semifields, that is, semifields which can not be produced by any of the currently known constructions.

However, the situation in the case of commutative semifields of odd characteristic is slightly better, since these semifields benefit from the connections to the theory of planar functions [4].

We consider the concrete case of commutative semifields of dimension 4 over its center. We study the properties of these objects in small orders, and we present a computational based classification of them.

 ISBN 13: 978-84-613-5510-5

2 Commutative semifields of odd order

In this section we collect some definitions and facts on finite semifields, presemifields and planar functions (see, for instance [3, 11, 4, 5]).

We restrict ourselves to the particular case of a semifield D of dimension 4 over a finite field $\mathbb{F}_q$ ($q = p^c$, odd), which is contained in its associative-commutative center $Z(D)$. Other relevant subsets of a finite semifield are the left, right, and middle nuclei (N_l, N_r, N_m), and the nucleus N which have to be field extensions $\mathbb{F}_{q^e}$ ($e \leq 4$).

Classification of presemifields is usually considered up to *isotopy* (since this corresponds to classification of the corresponding projective planes up to isomorphism): If D_1, D_2 are two presemifields of order q^4, an isotopy between D_1 and D_2 is a triple (F, G, H) of bijective $\mathbb{F}_3-$linear maps $D_1 \to D_2$ such that

$$H(ab) = F(a)G(b)\ ,\ \forall a, b \in D_1.$$

Any presemifield is isotopic to a finite semifield.

If $\mathcal{B} = [x_1, \ldots, x_4]$ is a $\mathbb{F}_q$-basis of a presemifield D, then there exists a unique set of constants $\mathbf{A}_{D,\mathcal{B}} = \{A_{i_1 i_2 i_3}\}_{i_1,i_2,i_3=1}^4 \subseteq \mathbb{F}_q$ such that

$$x_{i_1} x_{i_2} = \sum_{i_3=1}^{4} A_{i_1 i_2 i_3} x_{i_3}\ \forall i_1, i_2 \in \{1, \ldots, 4\}$$

This set of constants is known as **cubical array** or **3-cube** corresponding to D with respect to the basis $\mathcal{B}$, and it completely determines the multiplication in D. If D is a presemifield, and $\sigma \in S_3$ (the symmetric group on the set $\{1, 2, 3\}$), then the set

$$\mathbf{A}_{D,\mathcal{B}}^{\sigma} = \{A_{i_{\sigma(1)} i_{\sigma(2)} i_{\sigma(3)}}\}_{i_1,i_2,i_3=1}^4 \subseteq \mathbb{F}_q$$

is the 3-cube of a presemifield. Different choices of bases lead to isotopic presemifields. Up to six projective planes can be constructed from a given finite semifield D using the transformations of the group S_3. So, the classification of finite semifields can be reduced to the classification of the corresponding projective planes up to the action of the group S_3.

Commutative semifields of order q^4 (q odd) can be easily constructed from certain types of **planar functions**, induced by **Dembowski-Ostrom (DO) polynomials**. A planar function is a map

$$f : \mathbb{F}_{q^4} \to \mathbb{F}_{q^4}$$

such that for all nonzero $x \in \mathbb{F}_{q^4}$ the difference mapping

$$\Delta_{f,x} : \mathbb{F}_{q^4} \to \mathbb{F}_{q^4}\ ,\ y \to f(x + y) - f(x) - f(y)$$

is bijective, i.e, induces a permutation of $\mathbb{F}_{q^4}$. Also, it is called a DO polynomial in case it has the form

$$f(x) = \sum_{i,j=0}^{3} a_{i,j} x^{q^i + q^j}$$

 ISBN 13: 978-84-613-5510-5

where $a_{i,j} \in \mathbb{F}_{q^4}$, i.e., if its $q-$weight is at most 2. The construction of a finite presemifield D_f from one of such mappings f is as follows. Take $(D_f, +) = (\mathbb{F}_{q^4}, +)$ and define a multiplication by the following rule:

$$x * y = \Delta_{f,x}(y)$$

Any finite presemifield of order q^4 and center $\mathbb{F}_q \subseteq Z(D)$ is isotopic to one of these constructions and classification of presemifields up to isotopy is equivalent to classification of the corresponding DO polynomials up to **extended affine (EA) equivalence**.

Two DO polynomials $F, G : \mathbb{F}_{q^4} \to \mathbb{F}_{q^4}$ are called EA-equivalent if $G = A_1 \circ F \circ A_2 + A$ for some affine permutations $A_1, A_2 : \mathbb{F}_{q^4} \to \mathbb{F}_{q^4}$ and an affine mapping $A : \mathbb{F}_{q^4} \to \mathbb{F}_{q^4}$. In particular, the presemifield D_f is isotopic to $\mathbb{F}_{q^4}$ if and only if f is EA-equivalent to $g(x) = x^2$, and it is isotopic to Albert's twisted fields [1] if and only if f is EA-equivalent to $h_r(x) = x^{q^r+1}$ $(r \in \mathbb{N})$.

3 4-dimensional commutative semifields of small order

With the help of computational tools and parallel processing we have achieved a complete classification of commutative semifields of dimension 4 over its center $\mathbb{F}_p$, for small values of p.

Our results (which will be presented in full detail in the poster communication) show that, for every order there exists two different isotopy classes of semifields, that also correspond to different S_3-classes. One of them is the class of the finite field $\mathbb{F}_{p^4}$, where as the other corresponds to a non proper semifield of order p^4.

Acknowledgements

This work has been partially supported by MEC-MTM-2007-67884-C04-01, IB08-147, MEC-TIN-2007-61273 and MEC-TIN-2007-29664-E. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the *Centro de Supercomputación y Visualización de Madrid (CeSViMa)* and the Spanish Supercomputing Network.

References

[1] A. A. Albert, *Finite division algebras and finite planes*, Proceedings of Symposia in Applied Mathematics **10** (1960), 53-70.

[2] E. F. Combarro, I. F. Rúa, *New Semifield Planes of order 81*, (2008) (unpublished).

[3] M. Cordero, G. P. Wene, *A survey of finite semifields*, Discrete Mathematics **208/209** (1999), 125-137.

 ISBN 13: 978-84-613-5510-5

[4] R. S. Coulter, R. W. Matthews, *Planar functions and planes of Lenz-Barlotti class II*, Des. Codes and Cryptography **10** (1997), 167-184.

[5] R. S. Coulter, M. Henderson, *Commutative presemifields and semifields*, Adv. Math. **217** (2008), 282-304.

[6] U. Dempwolff, *Semifield Planes of Order 81*, J. of Geometry **89** (2008), 1-16.

[7] D. G. Farmer, K. J. Horadam, *Presemifields bundles over $GF(p^3)$*, Proc. ISIT 2008, Toronto, IEEE (2008), 2613-2616.

[8] S. González, C. Martínez, I. F. Rúa, *Symplectic Spread based Generalized Kerdock Codes*, Designs, Codes and Cryptography **42 (2)** (2007), 213–226.

[9] W. M. Kantor, M. E. Williams, *Symplectic semifield planes and $\mathbb{Z}_4$-linear codes*, Transactions of the American Mathematical Society **356** (2004), 895–938.

[10] W. M. Kantor, *Finite semifields*, Finite Geometries, Groups, and Computation (Proc. of Conf. at Pingree Park, CO Sept. 2005), de Gruyter, Berlin-New York (2006).

[11] D. E. Knuth, *Finite semifields and projective planes*, Journal of Algebra **2** (1965), 182-217.

[12] J. P. May, D. Saunders, Z. Wan, *Efficient Matrix Rank Computation with Applications to the Study of Strongly Regular Graphs*, Proc. of ISSAC 2007, 277-284, ACM, New-York

[13] I. F. Rúa, Elías F. Combarro, J. Ranilla, *Classification of Semifields of Order 64*, J. of Algebra, **322 (11)** (2009), 941-961.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Artificial Satellites Orbit Determination by modified high-order Gauss method

A. Cordero[1], Víctor Arroyo[1], J.R. Torregrosa[1] and M.P. Vassileva[2]

[1] *Instituto de Matemática Multidisciplinar, Universidad Politécnica de Valencia, Valencia, Spain*

[2] *Instituto Tecnológico de Santo Domingo (INTEC), Avda. Los Próceres, Gala, Santo Domingo, República Dominicana*

emails: acordero@mat.upv.es, vicarmar@teleco.upv.es, jrtorre@mat.upv.es, maria.vassilev@gmail.com

Abstract

In recent years, high-order methods have shown to be very useful in many practical applications, in which nonlinear systems arise. In this case, a classical method of positional astronomy have been modified in order to hold a nonlinear system in its establishments (that in the classical method is reduced to a single equation). At this point, high-order methods have been introduced in order to estimate the solutions of this system and, then, determine the orbit of the celestial body. We also have implemented a user friendly application, which will allow us to make a numerical and graphical comparison of the different methods with reference orbits, or user defined orbits.

Key words: orbit determination, Gauss method, nonlinear systems, Newton method, iterative function, order of convergence, efficiency

1 Introduction

Orbit determination is an old problem with new applications: at the early XIX century, Gauss designed a method to predict the future positions of asteroids, as Ceres, or other celestial bodies of our solar system with elliptical orbits. Nowadays, orbit determination methods are an essential tool in order to, by example, correct the position of artificial satellites in their orbits. This kind of methods only determine preliminary orbits, as the motion analyzed is under the premises of the two bodies problem, not taking into account any other force than mutual gravitational attraction between both bodies.

The inertial system which the orbit is placed in is a geocentric system whose fundamental plane is the projection of the terrestrial equator to the celestial sphere, the

 ISBN 13: 978-84-613-5510-5

perpendicular axis points to Celestial North and South Poles (NP and SP, respectively), and the X axis points to the Vernal point γ, in Aries constellation. This is shown in Figure 1.

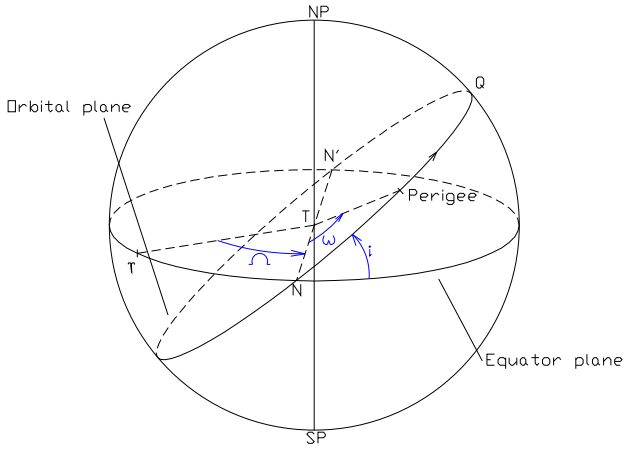


Figure 1: Orientation elements in 3-dimensional coordinate system.

If we focus on the orbital plane, as we can see in Figure 2, it is possible to set a two-dimensional coordinate system, where the X axis points to the perigee of the orbit, the closest point of the elliptical orbit to the focus and center of the system, the Earth. in order to place this orbit in the celestial sphere and determine completely the position of a body in the orbit, some elements (called orbital or keplerian elements) must be determined.

Then, the orbital elements are:

- Ω, (right ascension of the ascending node): defined as the equatorial angle between the Vernal point γ and the ascending node N; it orients the orbit in the equatorial plane.
- ω, (argument of the perigee): defined as the angle of the orbital plane, centered at the focus, between the ascending node N and the perigee of the orbit; it orients the orbit in its plane.
- i, (inclination): Dihedral angle between the equatorial and the orbital planes.
- a, (semi-major axis): Which sets the size of the orbit.
- e, (eccentricity): Which gives the shape of the ellipse.
- T_0, (perigee epoch): Time for the passing of the object over the perigee, to determine a reference origin in time. It can be denoted by a exact date, in Julian Days, or by the amount of time ago the object was over the perigee.

 ISBN 13: 978-84-613-5510-5

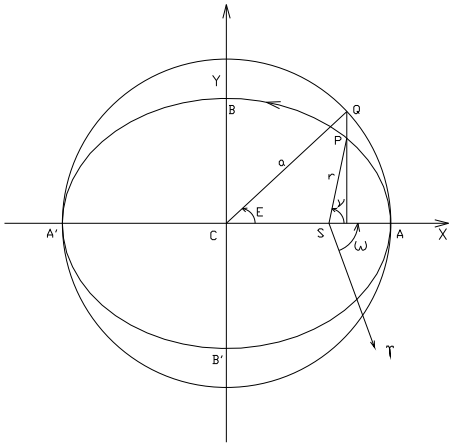


Figure 2: Size, shape and anomalies in orbital plane 2-dimensional coordinate system.

Different methods have been developed for this purpose (see [1, 2, 3]), constituting a fundamental element in navigation control, tracking and supervision of artificial satellites. By using these methods, from position and velocity coordinates for a given time, it is possible to determine those orbital elements for the preliminary orbit, which should be refined with later observations from ground stations, whose geographic coordinates are already known. In order to get this aim, some angles (or anomalies) must be determined on the planar orbit. Firstly, the object position in the ellipse can be determined by an angle, the true anomaly (ν), with center on the focus of the ellipse, which is the covered angle by a position vector, from its last perigee epoch ($\nu = 0$), to the observation instant. Another related angle with the previous one is the eccentric anomaly (E), whose center is on the center of the ellipse. This is the covered angle by a line from this center to the point where a circumference of radius a, the semi-major axis, is cut by a perpendicular line to X axis passing by the coordinates of the position vector, from its last Perigee epoch ($E = 0$) to the observation moment.

Using the Earth as the center of the coordinates system, it is useful to establish related units: the distance unit is the Earth radius (e.r.), approximately 6370 Km, and time unit is the minute, although some dates are described in Julian days (JD).

Now, some fundamental constants must be expressed in terms of these units, as the Earth gravitational constant, $k = \sqrt{G \cdot m_{Earth}} = 0.07436574(e.r.)^{\frac{3}{2}}/min$, deduced from the universal gravitational constant, G, and the Earth mass, m_{Earth}, (see [2]). The objects under study are very light compared with the Earth, like satellites orbiting our planet, so it is possible to relate both body's masses as the unit, $\mu = \frac{1}{m_{Earth}}(m_{Earth} + m_{Object}) \approx 1$. Then, modified time variable is introduced as

$$\tau = k(t_2 - t_1), \tag{1}$$

where t_1 is an initial arbitrary time and t_2 is the observation time. So, τ is considered

 ISBN 13: 978-84-613-5510-5

here as a measure of time difference, which will simplify calculations.

To estimate the velocity we can make use of the closed forms of the f and g series (see [2, 3]),

$$f = 1 - \frac{a}{|\vec{r_1}|}[1 - \cos(E_2 - E_1)] \tag{2}$$

and

$$g = \tau - \frac{\sqrt{a^3}}{\mu}[(E_2 - E_1) - \sin(E_2 - E_1)], \tag{3}$$

so we can express the rate respect two positions vectors and time as

$$\dot{\vec{r_1}} = \frac{\vec{r_2} - f \cdot \vec{r_1}}{g}. \tag{4}$$

So, it is clear that, known two position vectors and its corresponding observational instants, the main objective of the different methods that determine preliminary orbits is the calculation of the semi-major axis, a, and the eccentric anomalies difference, $E_2 - E_1$. When they have been calculated, it is possible to obtain by (4) the velocity vector corresponding to one of the known position vectors and, then, obtain the orbital elements.

Most of these methods have something in common: the need for finding the solution of a nonlinear equation or system, as in Gauss method. Usually, fixed point or secant methods are employed.

From the available input data, two position vectors and times for the observations, τ can be immediately deduced from equation (1), and other intermediate results as the difference in true anomalies, $(\nu_2 - \nu_1)$, deduced by:

$$\cos(\nu_2 - \nu_1) = \frac{\vec{r_1} \cdot \vec{r_2}}{|\vec{r_1}| \cdot |\vec{r_2}|} \tag{5}$$

and

$$\sin(\nu_2 - \nu_1) = \pm \frac{x_1 y_2 - x_2 y_1}{|x_1 y_2 - x_2 y_1|} \sqrt{1 - \cos^2(\nu_2 - \nu_1)}, \tag{6}$$

with positive sign for direct orbits, and negative for retrograde orbits.

Once the difference of true anomalies is obtained from the position vectors and times, the specific orbit determination method is used. In our particular case, we will introduce in the following section the classical Gauss method and, thereafter, we will modify it in order to estimate the value of the semi-major axis and eccentric anomalies by means of high-order iterative methods.

Let us also note that the inverse problem, it is the calculation of ephemeris (position an velocity in a given time) knowing the orbital elements, can be done easily, with direct operations that can be found in related bibliography (see [1, 2, 3]).

2 Gauss method of orbit determination

This method calculate a preliminary orbit of a celestial body by means of only two observations (position vectors). It is based on the relation between the areas of the

 ISBN 13: 978-84-613-5510-5

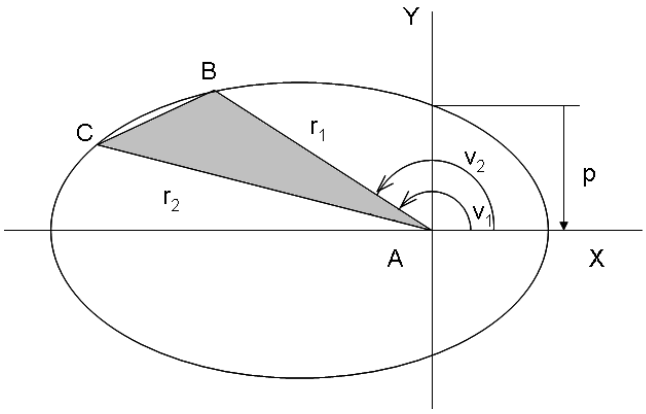


Figure 3: Ratio sector to triangle, in Gauss method.

sector ABC and the triangle ABC, as Figure 3 illustrates, delimited by both position vectors, $\vec{r_1}$ y $\vec{r_2}$, ratio sector- triangle

$$y = \frac{\sqrt{\mu p} \cdot \tau}{r_2 r_1 \sin(\nu_2 - \nu_1)} = \frac{\sqrt{\mu} \cdot \tau}{2\sqrt{a}\sqrt{r_2 r_1} \sin\left(\frac{E_2 - E_1}{2}\right) \cos\left(\frac{\nu_2 - \nu_1}{2}\right)}, \tag{7}$$

(with $(\nu_2 - \nu_1) \neq \pi$), and on the first

$$y^2 = \frac{m}{l + x} \tag{8}$$

and second

$$y^2(y - 1) = mX. \tag{9}$$

Gauss equations, where the constants of the problem (based on the data and the previously made calculations by (4), (5) and (6)), are

$$l = \frac{r_2 + r_1}{4\sqrt{r_2 r_1} \cos\left(\frac{\nu_2 - \nu_1}{2}\right)} - \frac{1}{2} \tag{10}$$

and

$$m = \frac{\mu \tau^2}{\left[2\sqrt{r_2 r_1} \cos\left(\frac{\nu_2 - \nu_1}{2}\right)\right]^2}. \tag{11}$$

Moreover, also must be determined in the process the value of:

$$x = \frac{1}{2}\left[1 - \cos\left(\frac{E_2 - E_1}{2}\right)\right] = \sin^2\left(\frac{E_2 - E_1}{4}\right) \tag{12}$$

and

$$X = \frac{E_2 - E_1 - \sin(E_2 - E_1)}{\sin^3\left(\frac{E_2 - E_1}{2}\right)}. \tag{13}$$

With this equations we present two different schemes to solve the problem: the classical method, which reduces first and second Gauss equations to a unique nonlinear equation, solved by fixed point method, and the modified Gauss scheme, which solve directly the nonlinear system formed by both Gauss equations.

 ISBN 13: 978-84-613-5510-5

2.1 Classical Gauss method scheme

In the classical method, an only nonlinear equation is obtained by, substituting second Gauss equation (8) into first equation (9):

$$y = 1 + X(l + x). \tag{14}$$

Them a fixed-point scheme is used to estimate the solution of (14), making a first estimation of the ratio , $y_0 = 1$, and by using the first Gauss equation to get x_0:

$$x_0 = \frac{m}{y_0^2} - l. \tag{15}$$

From the definition of x in equation (12), it is possible to calculate cosine and sine of the half difference of eccentric anomalies, which is known to be between 0 an π radians, determining uniquely the difference of eccentric anomalies:

$$\cos\left(\frac{E_2 - E_1}{2}\right) = 1 - 2x_0, \tag{16}$$

then

$$\sin\left(\frac{E_2 - E_1}{2}\right) = +\sqrt{4x_0(1 - x_0)}. \tag{17}$$

Then, with equation (13), an estimation of X, X_0, can be calculated and used in the reduced nonlinear equation (14) in order to get a better estimation of the ratio:

$$y_1 = 1 + X_0(l + x_0).$$

This iterative process gets new estimations of the ratio, until a given tolerance condition is satisfied. If there is convergence, the semi-major axis a, can be calculated by means of equation (7), from the last estimations of ratio and difference of eccentric anomalies, and the last phase of the process is then initiated, to determine velocity and orbital elements.

The Gauss method has some limitations, as the critical observation angles spread, in $\nu_2 - \nu_1 = \pi$, where the denominator of equation (7) vanish. Moreover, it is known that this method is only convergent to a coherent solution if the observation angles spread is less than 70^o, where this method has order of convergence 1. The ratio y grows with spread, leading to an invalid solution, if it converge. So this method is suitable for small spreads in observations, that is, observations which are close to each other.

2.2 Modified Gauss schemes

It is possible to make a different approach to the problem, solving the nonlinear system formed by both Gauss equations with different higher order iterative methods, instead of solving a unique nonlinear equation, which have the ratio y as unknown.

Firstly, it is necessary to establish the nonlinear system to be solved. In this case we can use the ratio $u = y$ and the difference of eccentric anomalies, $v = E_2 - E_1$, as

 ISBN 13: 978-84-613-5510-5

our unknowns, and substitute (12) and (13) in first an second Gauss equations, (8) and (9), so the system $F(u,v)=0$ becomes:

$$u^2+\frac{u^2}{2}\left(1-\cos\frac{v}{2}\right)-m=0, \tag{18}$$

$$u^3+u^2-m\frac{v-\sin v}{\sin^3\frac{v}{2}}=0. \tag{19}$$

Let us note that l and m are constants, with the input data, calculated with equations (10) and (11). Moreover, it is easy to check that the jacobian matrix $F'(u,v)$ associated to this system is ill-conditioned, so the iterative methods used to estimate its solutions must be robust enough. General information about iterative methods to solve nonlinear equations and systems can be found in [4].

Firstly, we will use Newton's method. Then, new estimations of the solution can be deduced with the following iterative scheme:

$$x^{(k+1)}=x^{(k)}-F'(x^{(k)})^{-1}F(x^{(k)}), \tag{20}$$

with convergence order up to 2. Also the well-known Jarrat's method (see [5]) will be employed, with forth order of convergence and iterative expression:

$$x^{(k+1)}=x^{(k)}-\frac{1}{2}(3F'(y^{(k)})-F'(x^{(k)}))^{-1}(3F'(y^{(k)})+F'(x_k))F'(x^{(k)}))^{-1}F(x^{(k)} \tag{21}$$

where $y^{(k)}=x^{(k)}-\frac{2}{3}F'(x^{(k)})^{-1}F(x^{(k)})$.

Moreover, a new family of methods is introduced:

$$x^{(k+1)} \;=\; y^{(k)}-A^{-1}BF'(x^{(k)})^{-1}F(y^{(k)}) \tag{22}$$

where $y^{(k)}=x^{(k)}-cF'(x^{(k)})^{-1}F(x^{(k)})$, $A=a_1F'(x^{(k)})+a_2F'(y^{(k)})$ and $B=b_1F'(x^{(k)})+b_2F'(y^{(k)})$ which will be denoted by N_5 methods and whose convergence order will be proved to be five for some values of the parameters.

Theorem 1 *Let $F:D\subseteq\mathbb{R}^n\longrightarrow\mathbb{R}^n$ be sufficiently differentiable at each point of an open neighborhood D of $\bar{x}\in\mathbb{R}^n$, that is a solution of the system $F(x)=0$. Let us suppose that $F'(x)$ is continuous and nonsingular in $\bar{x}$. Then the sequence $\{x^{(k)}\}_{k\geq 0}$ obtained using the iterative expression (22) converges to $\bar{x}$ with order 5 if $c=1$, $a_2\neq 0$, $a_1=-\frac{a_2}{5}$, $b_1=\frac{3a_2}{5}$ and $b_2=-a_1$.*

In the last section, we will use a member of this family in order to compare the precision of the calculated orbit with the other methods. In particular, we will take $a_2=5$ and its iterative expression is:

$$\begin{aligned} y^{(k)} &= x^{(k)}-F'(x^{(k)})^{-1}F(x^{(k)}) \\ x^{(k+1)} &= y^{(k)}-\left(-F'(x^{(k)})+5F'(y^{(k)})\right)^{-1}\left(3F'(x^{(k)})+F'(y^{(k)})\right)F'(x^{(k)})^{-1}F(y^{(k)}). \end{aligned} \tag{23}$$

 ISBN 13: 978-84-613-5510-5

Let us remark that this new uniparametric family of methods has better efficiency index than the well-known Jarrat's method and it is more efficient from the computational point of view, as it gets higher order of convergence with the same number of operations and only one more functional evaluation. In order to measure the efficiency of an iterative method, the efficiency index is defined as $I = p^{1/d}$ (see [6]), where p is the order of convergence and d is the total number of new functional evaluations (per iteration) required by the method. In the particular case of the modified Gauss method, the size of the nonlinear system involved is two; then, the respective efficiency indices are $I_{Newton} = 1.1225$, $I_{Jarrat} = 1.1487$ and $I_{N_5} = 1.1610$. So, the new method is specially appropriated to this problem.

All this Newton's variants applied to the nonlinear system appearing in Gauss method, (18), are expected to be at least so accurate as the classical scheme, but to drastically reduce the number of iterations needed to find a solution to the problem, as it will be seen later.

3 Comparing Gauss method schemes

Now, tests are needed to analyze the previously described schemes. For that purpose a graphical application was developed with Matlab GUIDE (Graphical User Interface Development Environment) to make graphical and numerical comparison. This program, (see Figure 4) lets the user define or load reference positions vector and times (in Julian Days) for two observations and, after setting the desired iterative method to solve the nonlinear system, it determines the velocity vector in the first observation and the orbital elements. Then, the orbit is plotted in both coordinate reference systems. Some other information about iterates is also displayed in graphics and messages. It is possible to choose other details, such as maximum number of iterations and the tolerance of the iterative scheme. It also lets the user to calculate ephemeris from reference or user defined orbital elements and times. The schemes presented will work with 200 digits of mantissa as it uses variable precision arithmetics, so we can set more restrictive tolerances.

The reference or test orbits we will use, found on [2], are:

- Test Orbit I:

$$\vec{r_1} = [2.46080928705339, 2.04052290636432, 0.14381905768815] \text{ e.r.}$$
$$\vec{r_1} = [1.98804155574820, 2.50333354505224, 0.31455350605251] \text{ e.r.}$$
$$t_1 = 0\text{JD} \qquad t_2 = 0.01044412000000 \text{ JD}$$
$$\Omega = 30^o \quad \omega = 10^o \quad i = 15^o \quad a = 4\text{e.r.} \quad e = 0.2 \quad T_0 = 0m$$

- Test Orbit II:

$$\vec{r_1} = [-1.75981065999937, 1.68112802634201, 1.16913429510899] \text{ e.r.}$$
$$\vec{r_2} = [-2.23077219993536, 0.77453561301361, 1.34602197883025] \text{ e.r.}$$
$$t_1 = 0 \text{ JD} \qquad t_2 = 0.01527809000000 \text{ JD}$$
$$\Omega = 80^o \quad \omega = 60^o \quad i = 30^o \quad a = 3 \text{ e.r.} \quad e = 0.1 \quad T_0 = 0m$$

 ISBN 13: 978-84-613-5510-5

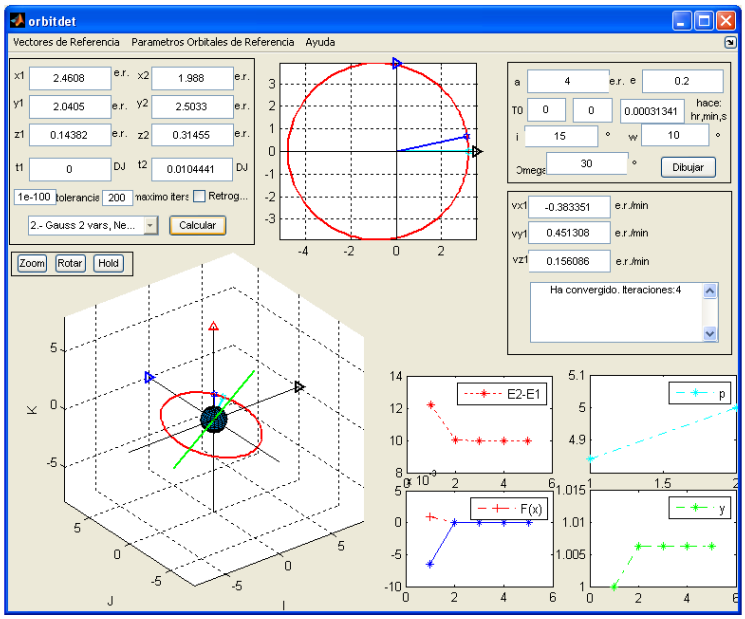


Figure 4: OrbitDet, software developed for testing.

- Test Orbit III:

$$\vec{r_1} = [0.41136206679761, -1.66250000000000, 0.82272413359522] \text{ e.r.}$$
$$\vec{r_2} = [0.97756752977209, -1.64428006097667, -0.04236299091612] \text{ e.r.}$$
$$t_1 = 0 \text{ JD} \qquad t_2 = 0.01316924000000 \text{ JD}$$
$$\Omega = 120^o \quad \omega = 150^o \quad i = 60^o \quad a = 2 \text{ e.r.} \quad e = 0.05 \quad T_0 = 0m,$$

- Test Orbit IV:

$$\vec{r_1} = [0.65241964490697, 3.80258035509303, 2.22750000000000] \text{ e.r.}$$
$$\vec{r_2} = [-1.35626966531604, 2.95849708305651, 3.05100082701246] \text{ e.r.}$$
$$t_1 = 0 \text{ JD} \qquad t_2 = 0.04622903000563 \text{ JD}$$
$$\Omega = 45^o \quad \omega = 45^o \quad i = 45^o \quad a = 4.5 \text{ e.r.} \quad e = 0.01 \quad T_0 = 0m$$

By using the first test positions vectors and times, we can first compare the number of iterations and estimated accuracy of classical (C), Newton (N), Jarrat (J) and new fifth-order (N_5) schemes described in this paper with $a_2 = 5$, described in (23). As we can see in Table 1, with tolerance $= 10^{-100}$, higher order methods reduce significantly the number of iterations, getting even more accuracy than the classical scheme.

 ISBN 13: 978-84-613-5510-5

Scheme	**Iterations**	$\|\|\mathbf{x^{(k+1)}} - \mathbf{x^{(k)}}\|\|$
C	54	1.8364e-101
N	8	7.3258e-133
J	5	7.0e-200
N_5	4	1.0658e-108

Table 1: Comparison of different Gauss method schemes for reference orbit I

Due to limitations in number of digits and format in observations data, and to the last phase of calculations, some accuracy is lost, but it is hard to determine differences in errors in the presented schemes. As far as results can be represented, errors in the final results of the orbital elements, for classical Gauss method, are shown in Table 2, where the exact orbit elements are compared with the calculated ones by means of the classical method.

Errors	C	$N\ J\ N_5$
$\|\mathbf{a'} - \mathbf{a}\|$	3.3032e-070 e.r.	3.3032e-070 e.r.
$\|\mathbf{e'} - \mathbf{e}\|$	6.6064e-071	6.6064e-071
$\|\mathbf{T'_0} - \mathbf{T_0}\|$	2.3162e-048 min	2.0726e-069 min
$\|\mathbf{i'} - \mathbf{i}\|$	3.0000e-198^o	4.0000e-198^o
$\|\omega' - \omega\|$	1.6900e-069^o	1.6900e-069^o
$\|\mathbf{\Omega'} - \mathbf{\Omega}\|$	2.0000e-198^o	1.0000e-198^o

Table 2: Error in classical Gauss method for reference orbit I

Now we can compare the new schemes with the classical, seeing in Table 2 the differences between the calculated orbital elements by the classical method and each one of the modified methods. It can be observed that Jarrat's and new fifth-order methods obtain almost the same estimation of the solution than Newton's method. Nevertheless, the reduction in the number of iterations needed justifies the use of high-order methods.

If we vary tolerance from 10^{-100} up to 10^{-498}, we can compare in Table 3 how number of iterations grows, making it clear that solving the nonlinear system, instead of reducing it to a nonlinear equation, does not increase number of iterations so fast as the classical scheme.

Finally, in Table 4, we can compare the number of iterations needed for different test orbits with different spreads in observations $SP = \nu_2 - \nu_1$, to realize that the limitation of spread is still present, but overall process is made faster, not increasing iterations to find a solution in worse cases, that is, with bigger difference of true anomalies in observation.

 ISBN 13: 978-84-613-5510-5

Scheme	$\mathbf{tol = 10^{-100}}$	$\mathbf{tol = 10^{-198}}$	$\mathbf{tol = 10^{-498}}$
C	54	106	172
N	8	9	10
J	5	5	6
N_5	4	5	5

Table 3: Iterations if varying tolerances, for reference orbit I

Scheme	**Test Orbit I** $SP = 12.23^o$	**Test Orbit II** $SP = 22.06^o$	**Test Orbit III** $SP = 31.46^o$	**Test Orbit IV** $SP = 30.29^o$
C	54	76	101	99
N	8	8	8	8
J	5	5	5	5
N_5	4	4	5	5

Table 4: Iterations needed for different spreads

4 Conclusion

A new approach to the problem of orbit determination is proposed, consisting in solving directly a nonlinear system formed by both Gauss equations, by means of well known iterative functions as Newton's and Jarrat's and a new method which have higher convergence order.

In the test of these variants of the Gauss methods, it is seen that they can reduce significantly the number of iterations, making the process faster, so it is possible to use more limiting tolerances to improve accuracy, without increasing much more the number of iterations. Some limitations of the classical scheme are still present in the alternatives introduced in this paper, such as spread limitation in observations, that is, the difference of true anomalies of observations. As the ratio y grows with spread, bigger spreads mean more iterations to find a solution, but in the proposed modified schemes this increment is very limited. If the difference is greater than 70^o, the process will probably lead to invalid solutions, which makes Gauss method suitable only for observations that are close enough.

References

[1] J. M. A. Danby, *Fundamentals of Celestial Mechanics*, The MacMillan Company, 1962.

[2] P. R. Escobal, *Methods of Orbit Determination*, Robert E. Krieger Publishing Company, 1975.

 ISBN 13: 978-84-613-5510-5

[3] M. J. Sevilla, *Mecánica Celeste Clásica*, Instituto de Astronomía y Geodesia. Facultad de Ciencias Matemáticas. Universidad Complutense de Madrid, 1989.

[4] J. F. Traub, *Iterative methods for the solution of equations*, Chelsea Publishing Company, New York, 1982.

[5] P. Jarrat, *Some fourth order multipoint iterative methods for solving equations*, Math. Comp. **20** (1966) 434–437.

[6] A.M. Ostrowski, *Solutions of equations and systems of equations*, Academic Press, New York-London, 1966.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

High order methods free from derivatives for nonlinear equations

Alicia Cordero[1], José L. Hueso[1], Eulalia Martínez[2] and Juan R. Torregrosa[1]

[1] *Instituto de Matemática Multidisciplinar, Universidad Politécnica de Valencia, Valencia, Spain*

[2] *Instituto de Matemática Pura y Aplicada, Universidad Politécnica de Valencia, Valencia, Spain*

emails: acordero@mat.upv.es, jlhueso@mat.upv.es, eumarti@mat.upv.es, jrtorre@mat.upv.es

Abstract

In the present paper, by approximating the derivatives in the well known order Ostrowski's method and in an sixth order improved Ostrowski's method by central difference quotients, we obtain new free from derivatives modifications of these methods. We prove the important fact that the obtained methods preserve their convergence orders four and six, respectively, without calculating any derivatives. Finally numerical tests confirm the theoretical results and allow us to compare these variants with the corresponding methods that make use of derivatives and with classical Newton's method.

Key words: Central approximation, Steffensen's method, derivative free method, convergence order

1 Introduction

In the last years, a lot of papers have developed the idea of removing derivatives from the iteration function in order to avoid defining new functions as the first or second derivative, and calculate iterates only by using the function that describes the problem, obviously trying to preserve the convergence order. In this sense, in the literature of nonlinear equations can be frequently found the expression "derivative free", referring in most cases to the second derivative (see [3, 4, 5]). The interest of these methods is to be applied on nonlinear equations $f(x) = 0$, when there are many problems in order to obtain and evaluate the derivatives involved.

 ISBN 13: 978-84-613-5510-5

There are different methods for computing a zero of a nonlinear equation $f(x) = 0$, the most known of these methods is the classical Newton's method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, \ldots, \tag{1}$$

that, under certain conditions, has quadratic convergence.

Newton's method has been modified in a number of ways to avoid the use of derivatives without affecting the order of convergence. For example, replacing in (1) the first derivative by the forward approximation

$$f'(x_n) \approx \frac{f(x_n + f(x_n)) - f(x_n)}{f(x_n)},$$

Newton's method becomes

$$x_{n+1} = x_n - \frac{f(x_n)^2}{f(x_n + f(x_n)) - f(x_n)},$$

which is called Steffensen's method [6]. This method has still quadratic convergence, in spite of being derivative free.

When an iterative method is free from first derivative, authors refer to it as a "Steffensen-like method". Some of these methods use forward divided differences for approximating the derivatives. For example, in [7] Pankaj Jain present a family of Steffensen's methods for solving nonlinear equations, with second and third order of convergence. Other Steffensen-like method and its higher-order variants are presented in [8] and a modified forward difference approximation is used in [9] in order to obtain a third-order Steffensen's method. Amat et al. in [10] considered a class of the generalized Steffensen iterations procedures for solving nonlinear equations on Banach spaces without derivatives.

If we try to use the same strategy, forward-difference approximation, with the fourth order Ostrowski's method [11]

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} \frac{f(y_n) - f(x_n)}{2f(y_n) - f(x_n)}, \end{aligned} \tag{2}$$

the order of convergence of the new method goes down to three. For this reason, we have used central differences to replace the first derivative. Classical Newton method has been modified in this sense in the paper of M. Dehghan and M. Hajarian [12].

Using central approximation in (2), we obtain a variant of Ostrowski's method that preserves the convergence order four and is derivative free. In the same way, using central approximation to substitute the derivative in the sixth order method proposed by M. Grau et al. in [13] as an improvement to Ostrowski root-finding method, which

 ISBN 13: 978-84-613-5510-5

iteration is:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ \mu_n &= \frac{y_n - x_n}{2f(y_n) - f(x_n)}, \\ z_n &= y_n - \mu_n f(y_n), \\ x_{n+1} &= z_n - \mu_n f(z_n), \end{aligned} \tag{3}$$

we obtain a new method that preserves the sixth convergence order and is derivative free.

The rest of this paper is organized as follows. In Section 2, we describe our free from derivatives methods as a variants of Ostrowski's method and the improved Ostrowski's method, respectively. In Section 3, we establish the convergence order of these methods. Finally, in Section 4 different numerical tests confirm the theoretical results and allow us to compare these variants with the corresponding methods that make use of derivatives and with Newton's method.

2 Description of the methods

In [12], Dehghan et al. approximate the derivative by a central differences quotient

$$f'(x_n) \simeq \frac{f(x_n + f(x_n)) - f(x_n - f(x_n))}{2f(x_n)},$$

obtaining a variant of Steffensen's method that is of second order of convergence and derivative free,

$$x_{n+1} = x_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))}. \tag{4}$$

By using this approximation in the fourth order Ostrowski's method (2), we obtain a new method free from derivatives, that we call modified Ostrowski's method free from derivatives (ODF):

$$y_n = x_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))}, \tag{5}$$

$$x_{n+1} = x_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))} \frac{f(y_n) - f(x_n)}{2f(y_n) - f(x_n)}. \tag{6}$$

In [13], Grau et al. propose an improvement of Ostrowski's method (3) and prove that it has sixth order of convergence. By approximating the derivative by central difference quotients we obtain a new method free from derivatives, that we call improved Ostrowski's method free from derivatives (IODF):

$$y_n = x_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))}, \tag{7}$$

 ISBN 13: 978-84-613-5510-5

$$z_n = y_n - \frac{y_n - x_n}{2f(y_n) - f(x_n)} f(y_n), \tag{8}$$

$$x_{n+1} = z_n - \frac{y_n - x_n}{2f(y_n) - f(x_n)} f(z_n). \tag{9}$$

3 Convergence of the methods

In this section we analyze the order of convergence of the methods described previously.

Theorem 1 *Let $\alpha \in I$ be a simple zero of a sufficiently differentiable function $f : I \subseteq \mathbb{R} \longrightarrow \mathbb{R}$ in an open interval I. If x_0 is sufficiently close to α, then the modified Ostrowski's method free from derivatives defined by (5) and (6) has order of convergence four.*

Proof: Let $e_n = x_n - \alpha$. The Taylor series of $f(x_n)$ about α is:

$$f(x_n) = c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + O(e_n^5), \tag{10}$$

where $c_k = \dfrac{f^{(k)}(\alpha)}{k!}, k = 1, 2, \ldots$

Computing the Taylor series of $f(x_n + f(x_n))$ and substituting $f(x_n)$ by (10) we have

$$\begin{aligned} f(x_n + f(x_n)) = \\ = c_1(1+c_1)e_n + (c_1c_2 + (1+c_1)^2c_2)e_n^2 + (2(1+c_1)c_2^2 + c_1c_3 + (1+c_1)^3c_3)e_n^3 + \\ +(3(1+c_1)^2c_2c_3 + c_2(c_2^2 + 2(1+c_1)c_3) + c_1c_4 + (1+c_1)^4c_4)e_n^4 + O(e_n^5). \end{aligned} \tag{11}$$

Analogously, the Taylor series of $f(x_n - f(x_n))$ is:

$$\begin{aligned} f(x_n - f(x_n)) = \\ = (1-c_1)c_1e_n + ((1-c_1)^2c_2 - c_1c_2)e_n^2 + (-2(1-c_1)c_2^2 + (1-c_1)^3c_3 - c_1c_3)e_n^3 + \\ +(-3(1-c_1)^2c_2c_3 + c_2(c_2^2 - 2(1-c_1)c_3) + (1-c_1)^4c_4 - c_1c_4)e_n^4 + O(e_n^5). \end{aligned} \tag{12}$$

Then, the quotient in (5) is:

$$\begin{aligned} \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))} = e_n - \frac{c_2e_n^2}{c_1} + \frac{(2c_2^2 - c_1(2+c_1^2)c_3)e_n^3}{c_1^2} + \\ + \left(-\frac{4c_2^3}{c_1^3 + c_2c_3} + \frac{7c_2}{c_3}c_1^2 - \frac{3c_4}{c_1 - 4c_1c_4}\right)e_n^4 + O(e_n^5). \end{aligned} \tag{13}$$

We obtain $y_n - \alpha$ taking into account (13)

$$\begin{aligned} y_n - \alpha &= e_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))} = \\ &= \frac{c_2e_n^2}{c_1} - \frac{(2c_2^2 - c_1(2+c_1^2)c_3)e_n^3}{c_1^2} + \\ &+ \left(\frac{4c_2^3}{c_1^3 - c_2c_3} - \frac{7c_2c_3}{c_1^2} + \frac{3c_4}{c_1 + 4c_1c_4}\right)e_n^4 + O(e_n^5). \end{aligned} \tag{14}$$

 ISBN 13: 978-84-613-5510-5

Now, substituting (14) in the Taylor series of $f(y_n)$, we have

$$\begin{aligned} f(y_n) &= c_2 e_n^2 - \frac{(2c_2^2 - c_1(2+c_1^2)c_3)e_n^3}{c_1} + \\ &+ \left(\frac{c_2^3}{c_1^2} + c_1\left(\frac{4c_2^3}{c_1^3 - c_2c_3} - \frac{7c_2c_3}{c_1^2} + \frac{3c_4}{c_1 + 4c_1c_4}\right)\right) e_n^4 + O(e_n^5). \end{aligned} \quad (15)$$

From (10) and (15) we obtain

$$\begin{aligned} f(y_n) - f(x_n) = -c_1 e_n + \left(-c_3 - \frac{2c_2^2 - c_1(2+c_1^2)c_3}{c_1}\right) e_n^3 + \\ + \left(\frac{c_2^3}{c_1^2 - c_4 + c_1}\left(\frac{4c_2^3}{c_1^3 - c_2c_3} - \frac{7c_2c_3}{c_1^2} + \frac{3c_4}{c_1 + 4c_1c_4}\right)\right) e_n^4 + O(e_n^5). \end{aligned} \quad (16)$$

and

$$\begin{aligned} 2f(y_n) - f(x_n) = -c_1 e_n + c_2 e_n^2 + \left(-c_3 - \frac{2(2c_2^2 - c_1(2+c_1^2)c_3)}{c_1}\right) e_n^3 + \\ + \left(-c_4 + 2\left(\frac{c_2^3}{c_1^2} + c_1\left(\frac{4c_2^3}{c_1^3} - c_2c_3 - \frac{7c_2c_3}{c_1^2} + \frac{3c_4}{c_1} + 4c_1c_4\right)\right)\right) e_n^4 + O(e_n^5). \end{aligned} \quad (17)$$

Taking into account (13), (16) and (17), we finally obtain

$$e_{n+1} = -c_2\left(-\frac{c_2^2}{c_1^3} + c_3 + \frac{c_3}{c_1^2}\right) e_n^4 + O(e_n^5). \quad (18)$$

This proves that the method is of fourth order. □

Theorem 2 *Let $\alpha \in I$ be a simple zero of sufficiently differentiable function $f : I \subseteq \mathbb{R} \longrightarrow \mathbb{R}$ in an open interval I. If x_0 is sufficiently close to α, then the improved Ostrowsky's method free from derivatives defined by (7) -(9) has order of convergence six.*

Proof: Let $e_n = x_n - \alpha$. The Taylor series of $f(x_n)$ about α is:

$$f(x_n) = c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + c_6 e_n^6 + O(e_n^7), \quad (19)$$

where $c_k = \dfrac{f^{(k)}(\alpha)}{k!}, k = 1, 2, \ldots$
Computing the Taylor series of $f(x_n + f(x_n))$ and substituting $f(x_n)$ by (19) we have

$$\begin{aligned} f(x_n + f(x_n)) &= c_1(1+c_1)e_n + \left(c_1 + (1+c_1)^2\right) c_2 e_n^2 + \\ &+ \left(2(1+c_1)c_2^2 + c_1c_3 + (1+c_1)^3 c_3\right) e_n^3 + \\ &+ \left(3(1+c_1)^2 c_2c_3 + c_2\left(c_2^2 + 2(1+c_1)c_3\right) + \right. \\ &+ \left. c_1c_4 + (1+c_1)^4 c_4\right) e_n^4 + \\ &+ \left(3(1+c_1)c_3\left(c_2^2 + c_3 + c_1c_3\right) + 4(1+c_1)^3 c_2c_4 + \right. \end{aligned}$$

 ISBN 13: 978-84-613-5510-5

$$
\begin{aligned}
&+ 2c_2(c_2c_3 + c_4 + c_1c_4) + c_1c_5 + (1+c_1)^5c_5\big)\, e_n^5 + \\
&+ \big(2(1+c_1)^2\left(3c_2^2 + 2(1+c_1)c_3\right)c_4 + \\
&+ c_3\left(c_2^3 + 6(1+c_1)c_2c_3 + 3(1+c_1)^2c_4\right) + \\
&+ 5(1+c_1)^4c_2c_5 + c_2\left(c_3^2 + 2(c_2c_4 + c_5 + c_1c_5)\right) + \\
&+ c_1c_6 + (1+c_1)^6c_6\big)\, e_n^6 + O(e_n^7). \qquad (20)
\end{aligned}
$$

The Taylor series of $f(x_n - f(x_n))$ is:

$$
\begin{aligned}
f(x_n - f(x_n)) \;=\;& -(-1+c_1)c_1e_n + \left(1 - 3c_1 + c_1^2\right)c_2e_n^2 + \\
&+ \big(2(-1+c_1)c_2^2 - (-1+c_1)^3c_3 - c_1c_3\big)\, e_n^3 + \\
&+ \left(-3(-1+c_1)^2c_2c_3 + c_2\left(c_2^2 + 2(-1+c_1)c_3\right) + \right. \\
&+ \left.(-1+c_1)^4c_4 - c_1c_4\right) e_n^4 + \\
&+ \big(-3(-1+c_1)c_3\left(c_2^2 + (-1+c_1)c_3\right) + 4(-1+c_1)^3c_2c_4 + \\
&+ 2c_2(c_2c_3 + (-1+c_1)c_4) - (-1+c_1)^5c_5 - c_1c_5\big)\, e_n^5 + \\
&+ \big(-c_2^3c_3 + 2\left(4 - 6c_1 + 3c_1^2\right)c_2^2c_4 + \\
&+ (-1+c_1)^2(-7+4c_1)c_3c_4 + c_2\left((7-6c_1)c_3^2 + \right. \\
&+ \left.\left(-7 + 22c_1 - 30c_1^2 + 20c_1^3 - 5c_1^4\right)c_5\right) + \\
&+ \left(1 - 7c_1 + 15c_1^2 - 20c_1^3 + 15c_1^4 - 6c_1^5 + c_1^6\right)c_6\big)\, e_n^6 + O(e_n^7). \qquad (21)
\end{aligned}
$$

Substituting (20) and (21) in (7), gives us

$$
\begin{aligned}
y_n - \alpha \;=\;& x_n - \alpha - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))} = \\
=\;& e_n - \frac{2f(x_n)^2}{f(x_n + f(x_n)) - f(x_n - f(x_n))} = \\
=\;& \frac{c_2e_n^2}{c_1} - \frac{\left(2c_2^2 - c_1\left(2 + c_1^2\right)c_3\right)e_n^3}{c_1^2} + \\
&+ \left(\frac{4c_2^3}{c_1^3} - c_2c_3 - \frac{7c_2c_3}{c_1^2} + \frac{3c_4}{c_1} + 4c_1c_4\right)e_n^4 - \\
&- \frac{1}{c_1^4}\left(8c_2^4 - c_1\left(20 + 3c_1^2\right)c_2^2c_3 + 2c_1^2\left(5 + 2c_1^2\right)c_2c_4 + \right. \\
&+ \left. c_1^2\left(\left(6 + 3c_1^2 + c_1^4\right)c_3^2 - c_1\left(4 + 10c_1^2 + c_1^4\right)c_5\right)\right)e_n^5 - \\
&- \frac{1}{c_1^5}\left(-16c_2^5 + c_1\left(52 + 7c_1^2\right)c_2^3c_3 - 4c_1^2\left(7 + 3c_1^2\right)c_2^2c_4 - \right. \\
&- c_1^2c_2\left(\left(33 + 12c_1^2 + c_1^4\right)c_3^2 + c_1\left(-13 - 10c_1^2 + c_1^4\right)c_5\right) + \\
&+ c_1^3\left(\left(17 + 17c_1^2 + 8c_1^4\right)c_3c_4 - \right. \\
&- \left.\left. c_1\left(5 + 20c_1^2 + 6c_1^4\right)c_6\right)\right)e_n^6 + O(e_n^7). \qquad (22)
\end{aligned}
$$

 ISBN 13: 978-84-613-5510-5

Now, substituting (22) in the Taylor series of $f(y_n)$ we have

$$\begin{aligned} f(y_n) &= c_2 e_n^2 + \left(-\frac{2c_2^2}{c_1} + 2c_3 + c_1^2 c_3\right) e_n^3 + \\ &+ \left(\frac{5c_2^3}{c_1^2} - \frac{7c_2 c_3}{c_1} - c_1 c_2 c_3 + 3c_4 + 4c_1^2 c_4\right) e_n^4 + \\ &+ \frac{1}{c_1^3}\left(-12c_2^4 + c_1\left(24 + 5c_1^2\right) c_2^2 c_3 - 2c_1^2\left(5 + 2c_1^2\right) c_2 c_4 + \right. \\ &+ c_1^2\left(-\left(6 + 3c_1^2 + c_1^4\right) c_3^2 + c_1\left(4 + 10c_1^2 + c_1^4\right) c_5\right)\Big) e_n^5 + \\ &+ \frac{1}{c_1^4}\left(28c_2^5 - c_1\left(73 + 13c_1^2\right) c_2^3 c_3 + 2c_1^2\left(17 + 10c_1^2\right) c_2^2 c_4 + \right. \\ &+ c_1^2 c_2\left(\left(37 + 16c_1^2 + 2c_1^4\right) c_3^2 + c_1\left(-13 - 10c_1^2 + c_1^4\right) c_5\right) + \\ &+ c_1^3\left(-\left(17 + 17c_1^2 + 8c_1^4\right) c_3 c_4 + \right. \\ &+ \left. c_1\left(5 + 20c_1^2 + 6c_1^4\right) c_6\right)\Big) e_n^6 + O(e_n^7). \end{aligned} \tag{23}$$

Using (19), (22) and (23) into (8), gives

$$\begin{aligned} z_n - \alpha &= y_n - \mu_n f(y_n) = \frac{c_2\left(c_2^2 - c_1\left(1 + c_1^2\right) c_3\right) e_n^4}{c_1^3} - \\ &- \frac{\left(4c_2^4 - 2c_1\left(4 + c_1^2\right) c_2^2 c_3 + c_1^2\left(2 + 3c_1^2 + c_1^4\right) c_3^2 + 2c_1^2\left(1 + 2c_1^2\right) c_2 c_4\right) e_n^5}{c_1^4} - \\ &- \frac{1}{c_1^5}\left(-10c_2^5 + 2c_1\left(15 + 2c_1^2\right) c_2^3 c_3 - 4c_1^2\left(3 + 2c_1^2\right) c_2^2 c_4 + c_1^3\left(7 + 17c_1^2 + 8c_1^4\right) c_3 c_4 + \right. \\ &+ \left. c_1^2 c_2\left(\left(-18 - 8c_1^2 + c_1^4\right) c_3^2 + c_1\left(3 + 10c_1^2 + c_1^4\right) c_5\right)\right) e_n^6 + O(e_n^7) \end{aligned} \tag{24}$$

and substituting (24) in the Taylor series of $f(z_n)$ we have

$$\begin{aligned} f(z_n) &= \frac{c_2\left(c_2^2 - c_1\left(1 + c_1^2\right) c_3\right) e_n^4}{c_1^2} - \\ &- \frac{\left(4c_2^4 - 2c_1\left(4 + c_1^2\right) c_2^2 c_3 + c_1^2\left(2 + 3c_1^2 + c_1^4\right) c_3^2 + 2c_1^2\left(1 + 2c_1^2\right) c_2 c_4\right) e_n^5}{c_1^3} - \\ &- \frac{1}{c_1^4}\left(-10c_2^5 + 2c_1\left(15 + 2c_1^2\right) c_2^3 c_3 - 4c_1^2\left(3 + 2c_1^2\right) c_2^2 c_4 + c_1^3\left(7 + 17c_1^2 + 8c_1^4\right) c_3 c_4 + \right. \\ &+ \left. c_1^2 c_2\left(\left(-18 - 8c_1^2 + c_1^4\right) c_3^2 + c_1\left(3 + 10c_1^2 + c_1^4\right) c_5\right)\right) e_n^6 + O(e_n^7). \end{aligned} \tag{25}$$

Taking into account (24) and (25), we finally obtain

$$\begin{aligned} e_{n+1} &= z_n - \alpha - \mu_n f(z_n) = \\ &\frac{\left(-2c_2^2 + c_1\left(1 + c_1^2\right) c_3\right)\left(-c_2^3 + c_1\left(1 + c_1^2\right) c_2 c_3\right)}{c_1^5} e_n^6 + O(e_n^7). \end{aligned} \tag{26}$$

This proves that the method is of sixth order. □

 ISBN 13: 978-84-613-5510-5

4 Numerical results

In this section we check the effectiveness of the new methods ODF and $IODF$ applied to the solution of several nonlinear equations. We use equations (a) to (j) to compare the obtained methods with their counterparts that make use of derivatives, that is, Ostrowski's method (OM) and improved Ostrowski's method (IOM) and the classical Newton's method (NM).

(a) $f(x) = \sin^2 x - x^2 + 1$, $\alpha = 1.404492$,

(b) $f(x) = x^2 - e^x - 3x + 2$, $\alpha = 0.257530$,

(c) $f(x) = \cos x - x$, $\alpha = 0.739085$,

(d) $f(x) = (x-1)^3 - 1$, $\alpha = 2$,

(e) $f(x) = x^3 - 10$, $\alpha = 2.154435$,

(f) $f(x) = \cos(x) - xe^x + x^2$, $\alpha = 0.639154$,

(g) $f(x) = e^x - 1.5 - \arctan(x)$, $\alpha = 0.767653$,

(h) $f(x) = x^3 + 4x^2 - 10$, $\alpha = 1.365230$,

(i) $f(x) = 8x - \cos(x) - 2x^2$, $\alpha = 0.128077$,

(j) $f(x) = \arctan(x)$, $\alpha = 0$,

Numerical computations have been carried out using variable precision arithmetics with 256 digits in MATLAB 7.1. The stopping criterion used is $|x_{k+1} - x_k| + |f(x_k)| < 10^{-100}$, therefore, we check that the iterates succession converge to an approximation to the solution of the nonlinear equation. For every method, we count the number of iterations needed to reach the wished tolerance and estimate the computational order of convergence p, according to (see [14])

$$\frac{\ln(|x_{k+1} - x_k| \,/\, |x_k - x_{k-1}|)}{\ln(|x_k - x_{k-1}| \,/\, |x_{k-1} - x_{k-2}|)}. \tag{27}$$

The value of p that appears in Table 1 is the last coordinate of vector p when the variation between its values is small. A comparison between methods using derivatives and derivative free methods can be established. The behavior of the new methods is similar to the classical ones of the same order of convergence, as theoretical results show. It can be observed that new methods need more iterations than their partenaires, in some cases, but when the initial estimation is not good and methods using derivatives diverge, derivative free methods ODF and IODF converge quickly.

 ISBN 13: 978-84-613-5510-5

$f(x)$	x_0	**Iterations**					**p**				
		NM	OM	IOM	ODF	IODF	NM	OM	IOM	ODF	IODF
a)	1	9	5	5	5	5	2.00	4.00	6.00	4.00	6.00
b)	0.7	7	5	4	5	6	2.00	4.00	6.00	4.00	5.99
c)	1	8	5	4	5	5	2.00	4.00	6.00	3.80	6.00
d)	1.5	11	6	5	6	6	2.00	4.00	6.00	4.00	6.00
e)	2	8	5	4	5	6	2.00	4.00	6.00	4.00	5.99
f)	1	9	5	4	6	NC	2.00	4.00	6.00	4.00	-
g)	1	9	5	4	5	5	2.00	4.00	6.00	4.00	6.00
h)	1.5	8	5	4	6	6	2.00	4.00	6.00	4.00	6.01
i)	1	9	5	4	5	6	2.00	4.00	6.00	4.00	5.99
j)	1	8	5	5	5	5	3.00	5.00	7.00	5.00	7.00
j)	2.5	NC	NC	5	8	6	-	-	7.00	5.00	7.00

Table 1: Numerical results for nonlinear equations from (a) to (j)

5 Conclusions

We have used central quotient difference approximations for the first derivative in Ostrowski's method, that has order of convergence four and in a improved version of Ostrowski's method with sixth order of convergence, obtaining two new iterative methods for nonlinear equations free from derivatives and we have proven that they preserve their convergence order. The theoretical results have been checked with some numerical examples, comparing our algorithms with a modified Newton's method free from derivative and with the corresponding methods that make use of derivatives.

References

[1] C. W. Misner, K. S. Thorne and J. A. Wheeler, *Gravitation*, Freeman, San Francisco, 1970.

[2] E. Witten, *Supersymmetry and Morse theory*, J. Diff. Geom. **17** (1982) 661–692.

[3] Z. Xiaojian, *Modified ChebyshevHalley methods free from second derivative*, Applied Mathematics and Computation **203** (2008) 824-827.

[4] C. Chun, *Some second derivative free variants of ChebyshevHalley methods* Applied Mathematics and Computation **191** (2007) 410-414.

[5] C. Chun, Y. Ham, *Some second-derivative-free variants of super-Halley method with fourth-order convergence*, Applied Mathematics and Computation **195** (2008) 537-541.

[6] D. Kincaid, W. Cheney, *Numerical Analysis*, Second Edition, 1996.

 ISBN 13: 978-84-613-5510-5

[7] P. Jain, *Steffensen type methods for solving nonlinear equations*, Applied Mathematics and Computation **194** (2007) 527-533.

[8] Q. Zheng, J. Wang,P. Zhao, L. Zhang, *A Steffensen-like methods and its higher-order variants*, Applied Mathematics and Computation **214** (2009) 10-16.

[9] J.R. Sharma, *A composite third order Newton-Steffensen method for solving nonlinear equations*, Applied Mathematics and Computation **169** (2005) 242-246.

[10] S. Amat, S. Busquier, J.M. Gutiérrez, *Geometric constructions of iterative functions to solve nonlinear equations*, Journal of Computational and Applied Mathematics **157** (2003) 197-205.

[11] A.M. Ostrowski, *Solutions of Equations and Systems of Equations*, Academic Press, New York, 1960.

[12] M. Dehghan, M. Hajarian, *An Some derivative free quadratic and cubic convergence iterative formulas for solving nonlinear equations*, Journal of Computational and Applied Mathematics **29** (2010) 19-30.

[13] M. Grau, J.L. Díaz-Barrero, *An improvement to Ostrowski root-finding method*, Applied Mathematics and Computation **173** (2006) 450-456.

[14] A. Cordero, J.R. Torregrosa, *Variants of Newton's method using fifth order quadrature formulas*, Applied Mathematics and Computation **190** (2007) 686-698.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Reed–Solomon and Reed–Muller group codes

Elena Couselo[1], Santos González[1], Victor Markov[2], Consuelo Martínez[1] and Alexandr Nechaev[2]

[1] *Department of Mathematics, University of Oviedo*

[2] *Center of new information technologies, Moscow State University*

emails: couselo@uniovi.es, santos@uniovi.es, markov@mech.math.msu.su, cmartinez@uniovi.es, nechaev@cnit.msu.ru

Abstract

This work has mainly methodical purposes. Although the ideal presentations of the codes that we consider here appeared in [1, 2, 3] long ago, we give different presentation of these ideals. We also show that Reed–Muller codes are connected in some sense with Reed–Solomon codes by means of the trace function.

Key words: Reed–Solomon codes, Reed–Muller codes, group rings, trace function

1 Reed-Solomon codes as group codes

Let p be a prime, $Q = \mathbb{F}_{p^l}$ a field of $q = p^l > 2$ elements and 1 its unit element. Let $(H, \cdot)$ be a p-elementary abelian group of order q. The identity element of H (that is also identity element of the group ring QH) will be denoted by e.

The following representation of Reed-Solomon codes as group codes plays a key role in our approach.

For a given isomorphism of abelian groups $\varphi : (H, \cdot) \to (Q, +)$ we consider the following elements

$$u_s = \sum_{h \in H} \varphi(h)^s h \in QH, \quad s = 0, \ldots, q-2. \tag{1.1}$$

Theorem 1.1 *For every i, $1 \le i \le q-1$ the subspace*

$$\mathcal{R}_i = Qu_0 + \ldots + Qu_{i-1} \le {}_QQH \tag{1.2}$$

is a Reed-Solomon $[q,\ i,\ q+1-i]_q$-MDS code and an ideal in QH. In particular

$$\mathcal{R}_{q-1} = \Delta(QH). \tag{1.3}$$

 ISBN 13: 978-84-613-5510-5

It is well known that the code dual to a Reed–Solomon $[q,\ i,\ q+1-i]_q$-code is itself a Reed–Solomon $[q,\ q-i,\ i+1]_q$-code. A similar relation remains valid in ring theoretic terms.

Theorem 1.2 *For every $i \in \overline{1, q-1}$ the equality*

$$\mathrm{Ann}_S(\mathcal{R}_i) = \mathcal{R}_{q-i} \tag{1.4}$$

holds.

2 Basic Reed–Muller codes

Let now P be a subfield of order π in Q, $q = \pi^m$. For any $i \in \overline{0, q-1}$ let $w_\pi(i)$ be the π-*weight* of i, i.e.

$$w_\pi(i) = i_0(\pi) + i_1(\pi) + ... + i_{m-1}(\pi), \tag{2.1}$$

where

$$i = i_0(\pi) + i_1(\pi)\pi + ... + i_{m-1}(\pi)\pi^{m-1}, \quad i_0(\pi), ..., i_{m-1}(\pi) \in \overline{0, \pi-1} \tag{2.2}$$

is a π-adic decomposition of i. Keeping the notation of (1.1) define the *basic Reed–Muller code of order k* as

$$\mathcal{M}_\pi(m,k) = \sum_{i \in \overline{0,q-1},\ w_\pi(i) \le k} Qu_i. \tag{2.3}$$

Then $\mathcal{M}_\pi(m,k)$ is a linear code of dimension $M_\pi(m,k)$ over Q, where

$$M_\pi(m,k) = \sum_{r=0}^{k} \left\{ {m \atop r} \right\}_\pi, \tag{2.4}$$

$$\left\{ {m \atop r} \right\}_\pi = \sum_{j \le 0} (-1)^j \binom{m}{j} \binom{m+k-\pi j-1}{r-\pi j}. \tag{2.5}$$

Theorem 2.1 *For every $k \in \overline{0, (\pi-1)m}$ the code $\mathcal{M}_\pi(m,k)$ is an ideal in QH.*

Note that if $k = (\pi-1)m$ then $M_\pi(m,k) = \pi^m = q$ and $\mathcal{M}_\pi(m,k) = QH$, so this case is trivial.

3 Extended Reed–Muller codes

Let $k < (\pi-1)m$ and take a primitive element ϑ of the field Q. Consider a polynomial

$$G_k(x) = \prod_{i \in \overline{0,q-1},\ w_\pi(i) \le k} (x - \vartheta^i). \tag{3.1}$$

 ISBN 13: 978-84-613-5510-5

Then $G_k(x)$ is a polynomial over P and its degree is $M_\pi(m,k)$. Let $L_P(G_k(x))$ be the set of all linear recurring sequences (LRS) over P with characteristic polynomial $G_k(x)$. It is well known [4], that the set $L_P^{\overline{0,q-2}}(G_k(x))$ of all initial segments

$$u[\,\overline{0,q-2}\,] = (u(0),...,u(q-2))$$

of sequences $u \in L_P(G_k(x))$ is the *cyclic Reed–Muller* $[q-1, M_\pi(m,k), d_\pi(m,k)]_\pi$-code. The distance of this code is defined as follows:

$$d_\pi(m,k) = (\rho+1)\pi^{\varkappa} - 1,$$

where $\varkappa$ and ρ are respectively the integer ratio and remainder of $m(\pi-1)-k$ modulo $\pi-1$: $m(\pi-1)-k = \varkappa(\pi-1)+\rho$, $0 \le \rho < \pi-1$. Adding parity check to this code gives a $[q, M_\pi(m,k), d_\pi(m,k)+1]_\pi$-code called the *extended Reed–Muller code*.

We show how this code is presented as an ideal in the group ring PH.

Let $\mathrm{tr} = \mathrm{tr}_P^Q$ be the trace function from the field Q to the field P and $\mathrm{Tr} = \mathrm{Tr}_{PH}^{QH}$ its natural extension to group rings.

Theorem 3.1 *The image*

$$\mathcal{RM}_\pi(m,k) = \mathrm{Tr}(\mathcal{M}_\pi(m,k)) \lhd PH$$

of the ideal $\mathcal{M}(m,k) \lhd QH$ *is the extended Reed–Muller* $[q, M_\pi(m,k), d_\pi(m,k)+1]_\pi$*-code over the field* P*.*

Note again that if $k = (\pi-1)m$ then $\mathcal{RM}_\pi(m,k) = PH$ is a trivial extended Reed–Muller $[q,q,1]_\pi$-code.

Acknowledgements

This work was partially supported by grant MTM2007-67884-C04-01, by the President of RF grant NSh-8.2010.10 and RFBR grant 08-01-00693-a. V.Markov and A.Nechaev thank also University of Oviedo for hospitality.

References

[1] P. Charpin, *Les codes de Reed-Solomon en tant qu'ideáux d'une algèbre modulaire*, C R. Acad. Sci. Paris. **294** (1982), 597–600.

[2] P. Charpin, *Une description des codes de Reed–Solomon dans une algèbre modulaire*, C R. Acad. Sci. Paris. **299** (1984), 779–782.

[3] P. Landrock, O. Manz, *Classical codes as ideals in group algebras*, Designs, Codes and Cryptography **2** (1992) 273-285.

[4] F. J. MacWilliams, N. J. A. Sloane, *The theory of Error-Correcting Codes*, Elsevier Science Publishers, B.V., 1988 (North Holland Mathematical Library, Vol. 16).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Heterogeneous-type social networks: a multi-level mathematical model

Regino Criado[1], Julio Flores[1], Alejandro J. García del Amo[1] and Miguel Romance[1]

[1] *Department of Applied Mathematics, University Rey Juan Carlos, Madrid (Spain)*

emails: regino.criado@urjc.es, julio.flores@urjc.es, alejandro.garciadelamo@urjc.es, miguel.romance@urjc.es

Abstract

The concept of multilevel network and some structural tools are presented in order to analyze heterogeneous-type social networks and to show that this model fits perfectly with several real-life heterogeneous-type complex systems, including social systems and public transportation networks.

Key words: Complex networks, multi-scaled networks, hyper-networks, structural properties.

1 An overview of multi-scaled complex networks

The study of structural properties of complex networks is an attractive and fascinating branch of research in applied mathematics, sociology (social networks, acquaintances or collaborations between individuals), science (metabolic and protein networks, neural networks, genetic regulatory networks, protein folding) and engineering (phone call networks, computers in telecommunication networks, Internet, the World Wide Web) (see, for example, [1], [2], [5] or [12] and the references therein).

The wide range of systems in the real world which can be modeled by complex networks share behavioral and structural properties, and they can be studied by using non-linear mathematical models and computer modeling approaches (see, for example, [5], [10] and [13]).

Social networks analysis is used in the social and behavioral sciences, as well as in economics, marketing, and industrial engineering ([13]), but some questions related to the structure of social networks have been not understood properly. Starting from the fact that a social network can be understood as a set of people or groups of people with some pattern of contacts or interactions between them ([11],[13]), a first and naive approach to social networks such as *Facebook* or *Linkedin* networks can give us the

 ISBN 13: 978-84-613-5510-5

impression that all the connections or social relationships between the members of those networks take place at the same level. But the real situation is far from this. The real relationships amongst the members of a social network take place inside of different groups.

In this note we will analyze in a non exhaustive way how to combine these different levels into a multilevel mathematical model. For example, if we want to model how a rumour is spread within a social network, it is necessary to have in mind that, on one hand, different groups are linked only through some of their members and, on the other hand, two people who know the same person don't have necessarily to know each other.

From a schematic point of view, a complex network is a mathematical object $G = (V, E)$ composed by a set of nodes or vertices $V = \{v_1 \ldots, v_n\}$ that are pairwise joined by links or edges $\{\ell_1, \ldots, \ell_m\}$. We consider the adjacency matrix $A(G) = (a_{ij})$ determined by the conditions

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Hyper-graphs appeared as the natural extensions of graphs (see, for example [4] and [7]). They are used in many applications to represent different concepts that graphs cannot do. For example, ordinary graphs in Chemistry do not adequately describe chemical compounds of nonclassical structure ([9]). A substantial drawback of the structure theory is the lack of a convenient representation for molecules with delocalized polycentric bonds. By using hyper-graphs, the defects of the structure representation are eliminated.

Let $X = \{v_1, v_2, \ldots, v_n\}$ be a finite set. A *hyper-graph* on X ([4], [7]) is a family $H = (E_1, E_2, \ldots, E_m)$ of subsets of X such that:

(i) $E_i \neq \emptyset \quad (i = 1, 2, \ldots, m)$.

(ii) $\bigcup_{i=1}^{m} E_i = X$.

A *simple hyper-graph* is a hyper-graph $H = (E_1, E_2, \ldots, E_m)$ such that

$$E_i \subseteq E_j \Rightarrow i = j.$$

The elements $v_1, v_2, \ldots, v_n$ of X are called *vertices*, or *nodes* and the sets $E_1, E_2, \ldots, E_m$ are the *hyper-edges* or the *hyperlinks* of the hyper-graph. We say that a vertex v_i is *incident* to edge E_j if $v_i \in E_j$.

Two vertices are *adjacent* if there is a hyper-edge E_i that contains both of these vertices. The *degree of vertex* v_i is the cardinality of the set of all hyperedges incident to vertex v_i and is denoted by $d_H(v_i)$. The *degree of edge* E_j is the cardinality of the set of all vertices incident to the edge E_j. A simple graph is a simple hyper-graph each of whose edges has cardinality 2; a multigraph is a hyper-graph in which each edge has

 ISBN 13: 978-84-613-5510-5

cardinality ≤ 2. A hyper-graph in which all vertices have the same degree is said to be *regular*. The maximum degree of the hyper-graph H will be denoted by

$$\Delta(H) = \max_{v \in X} d_H(v).$$

We can define a finite hyper-graph by its incidence matrix. The *incidence matrix* $B = (b_{ij})$ is defined by

$$b_{ij} = \begin{cases} 1 & \text{if } v_i \in E_j, \\ 0 & \text{otherwise.} \end{cases}$$

with columns representing the edges $E_1, E_2, \ldots, E_m$ and rows representing the vertices $v_1, v_2, \ldots, v_n$. The *adjacency matrix*, $A(H) = (a_{ij})$, of the hyper-graph H is a square symmetric matrix whose entries a_{ij} are the number of hyper-edges that contain both vertices v_i and v_j, that is

$$a_{ij} = \begin{cases} 0 & \text{if } i = j \\ \left|\{E_k \in E : \{v_i, v_j\} \subset E_k\}\right|, & \text{if } i \neq i. \end{cases}$$

If we denote by D the diagonal matrix whose entries are the degrees of the vertices, then $A(H) = B(H) \cdot B(H)^T - D(H)$. $A(H)$ can be regarded as the adjacency matrix of a multigraph. We have to notice that the adjacency matrix of a hyper-graph does not inform about the hyper-edges, hence it is not as useful as the incidence matrix.

But, as we can see, the concept of hyper-graph does not fit for modeling different levels in social networks, since this model is only sensitive in a global way to the different social groups, i.e., for each node it only takes into a ccount which social groups it belongs, and not the actual relationships between the nodes or members belonging to the same group.

In order to avoid this drawback, the concept of hyper-structure was introduced in [6] since it represents special properties that cannot be regarded only in terms of graphs or hyper-graphs. If $G = (X, E)$ is a graph with n vertices and m edges, a hyper-graph H for this graph G is a family $H = (E_1, E_2, \ldots, E_k)$ of subsets of X. Then, a *hyper-structure* $S = (X, E, H)$ is a triple formed by the vertex set X, the edge set E and the hyper-edge set H.

Note that not every pair of vertices in the same hyper-edge has to be joined by and edge or a link. So, we can have two or more vertices belonging to the same hyper-edge but with no links (in terms of graph structure) between them. This makes the difference with the hyper-network approach of [8], where every vertex belonging to a hyper-edge is linked to all the vertices of that hyper-edge. But this approach is not adequated for modelling social networks either, because, in fact, not only the members (or nodes) belongs to a specific social group but also the links between them are part of a specific social group. If we give colours to our represention, we have to color not only the nodes (the members of the group) but also the edges (the links between them).

2 Mathematical Model and structural analysis

As we pointed out in the previous section, a sort of naive approach to heterogeneous-type complex systems could suggest that hyper-networks and hyper-structures fit per-

 ISBN 13: 978-84-613-5510-5

fectly to these real-life systems. The key-point that makes these mathematical model not to be the best solution for heterogeneous-type systems has to do with the fact that either hyper-networks and hyper-structures are node-based models, while many real systems combine a node-based point of view with a link-based perspective. For example, if we have a look again at an heterogeneous-type social network, when we consider a relationship between two members of one social group(or several), we have to take into account not only the social groups that hold the members, but also the social group that holds the relationship itself, i.e., if, for example, there is a relationship between two people that share the same working group and the same sport group, we have to highlight if the relationship is due to share the same group of work or it has sport nature. This fact is not particular of heterogeneous-type social networks and a similar situation occurs, for example, in public transportation systems, where a link between two stations belonging to several transport lines can occur as a part of different lines.

In order to avoid this node-based nature of hyper-networks and hyper-structures, we propose to introduce the following concept that combines the node-based with the link-based perspective:

Definition 2.1 *Let $G = (X, E)$ be a (simple, directed or un-directed) network. A* **multilevel network** *is a triple $M = (X, E, \mathcal{S})$, where $\mathcal{S} = \{S_1, \ldots, S_k\}$ is a family of subgraphs of G.*

The network G is the **projection network** *of M and each subgraph $S_j \in \mathcal{S}$ is called a* **slice** *of the multilevel network M.*

This mathematical model perfectly suits heterogeneous-type social systems as well as other heterogeneous-type complex systems, since each social group can be understood as a *slice graph* in a *multilevel network* and therefore we simultaneously take into account the nature of the links (i.e. relationships) and the nodes involved.

It is easy to check that this new mathematical object extends both, the classic complex network model and also the hyper-network model [4]. Let us point it out very briefly. On the one hand, if $G = (X, E)$ is a network, then it can be understood as a multilevel network by considering $M = (X, E, \mathcal{S})$, where $\mathcal{S} = \{G\}$.

On the other hand if $\mathcal{H} = (X, H)$ is an hyper-network (i.e. X is a finite set of nodes and $H = \{H_1, \ldots, H_k\}$ is a family of finite subsets of X, each of them called and *hyper-link* of $\mathcal{H}$), then it can be seen as the multilevel set $M = (X, E_{\mathcal{H}}, \mathcal{S}_{\mathcal{H}})$, given by

$$\mathcal{S}_{\mathcal{H}} = \{K_{H_1}, \ldots, K_{H_k}\},$$

where K_{H_j} is the complete network obtained by linking every pair of nodes of H_j and

$$E_{\mathcal{H}} = \bigcup_{j=1}^{k} K_{H_j}.$$

By using similar argument we can show that every hyper-structure [6] can be understood as a particular multilevel network, by considering one slice network for each hyper-link in the hyper-structure and each slice graph being a set of isolated nodes.

 ISBN 13: 978-84-613-5510-5

Once we have introduced this new and novel mathematical object, we have to give suitable structural parameters to analyze it. We can give natural extensions of many of the usual tools of the complex networks' analysis, such as the clustering coefficient, an adjacency matrix/tensor, a natural network representation as a tripartite network or a geodesic structure, among many others. For example, if we want to introduce metric tools in a multilevel network $M = (X, E, \mathcal{S})$, we first have to give the notion of *path* and *length*. A *path* P in $M = (X, E, \mathcal{S})$ is a set of the form $P = \{(\ell_1, \cdots, \ell_q), (S_1, \cdots, S_q)\}$ such that

(i) $(\ell_1, \cdots, \ell_q)$ is a sequence of links $\ell_1, \cdots, \ell_q \in E$,

(ii) $(S_1, \cdots, S_q)$ is a sequence of slice graphs $S_1, \cdots, S_q \in \mathcal{S}$,

(iii) For every $1 \leq j \leq q$, we have that $\ell_j \in S_j$, i.e. ℓ_j is an edge in the slice graph S_j.

By using this concept we can introduce a metric structure in a multilevel graph $M = (X, E, \mathcal{S})$ as follows.

Definition 2.2 *Let $M = (X, E, \mathcal{S})$ be a multilevel network, $\beta \geq 0$ fixed and $P = \{(\ell_1, \cdots, \ell_q), (S_1, \cdots, S_q)\}$ be a path in M. The* **length** *of P is the nonnegative value*

$$L(P) = q + \beta \sum_{j=2}^{q} \Delta(j),$$

where

$$\Delta(j) = \begin{cases} 1 & \text{if } S_j \neq S_{j-1}, \\ 0 & \text{otherwise.} \end{cases}$$

The **distance** *in M between two nodes $v_1, v_2 \in X$ is the minimal length among all possible paths in M from v_1 to v_2.*

If we take $\beta = 0$, the previous definition gives the natural metric in the projection graph (under some constraints), while if $\beta \neq 0$, we introduce new metrics that take into account, not only the global structure of the projection network, but also the interplay between the slice networks, that helps to model the multi-scale nature of real-life social networks.

In this work we introduce this type of parameters and many other structural tools for analyzing multilevel networks and we present some relationships between them and the corresponding parameters of the projection and slice networks. The analytical results obtained will support the validity of this model in real life heterogeneous-type complex systems, including social systems and public transport networks.

Acknowledgements

The authors of this work have been partially supported by the Spanish Government Project MTM2009-13848.

 ISBN 13: 978-84-613-5510-5

References

[1] Albert R., Barabási A. L., *Statistical mechanics of complex networks*, Rev. Mod. Phys. **74** (2002), 47–97.

[2] Barabási A. L., *Linked: The new science of networks*, Perseus Publishing, Cambridge, Massachusetts, 2002.

[3] Barabási A. L., Albert R., *Emergence of scaling in random networks*, Science **286**, 509–512.

[4] Berge C., *Hyper-graphs. Combinatorics of Finite Sets*, North-Holland, 1989.

[5] Boccaletti S., Latora V., Moreno Y., Chavez M. and Hwang D. U., *Complex Networks: Structure and dynamics*, Physics Reports **424** (2006), 175–308.

[6] Criado R., Romance R., Vela-Perez M., *Hyperstructures, a new approach to complex systems*, Int. J.Bif.Chaos **20** (2010), 877–883.

[7] Duchet P., *Hypergraphs*, *Handbook of Combinatorics* Ed. R. Graham, M. Grötschel and L. Lovász, 1995.

[8] Estrada E., Rodríguez-Velázquez J.A., *Subgraph centrality and clustering in complex hyper-networks*, Physica A (2006).

[9] Konstantinova E.V., Skorobogatov V. A., *Application of hyper-graph theory in chemistry*, Discrete Mathematics **235** (2001), 365–383.

[10] Newman, M.E.J., *The structure and function of complex networks*, SIAM Review **45** (2003), 167–256.

[11] Scott, J., *Social Networks Analysis: A Handbook*, 2nd ed., Sage, London, 2000.

[12] Strogatz, S.H., *Exploring complex networks*, Nature **410** (2001), 268–276.

[13] Wasserman, S. and Faust, K., *Social Networks Analysis*, Cambridge University Press, Cambridge, 1994.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Detecting Interest Points in Images by Analyzing Centrality Measures of Complex Networks

Regino Criado[1], Miguel Romance[1] and Ángel Sánchez[2]

[1] *Department of Applied Mathematics, University Rey Juan Carlos, Madrid (Spain)*

[2] *Department of Computing Sciences, University Rey Juan Carlos, Madrid (Spain)*

emails: regino.criado@urjc.es, miguel.romance@urjc.es,
angel.sanchez@urjc.es

Abstract

The theory and tools of Complex Networks have been few applied to Image Analysis and Computer Vision problems. This paper presents a new application for detecting interest points in digital images. We associate a spatial and weighted complex network to each image and propose two different methods for locating these feature points based on both local and global (spectral) centrality measures of the corresponding network.

Key words: Interest Points, Feature Detection, Image Analysis, Computer Vision, Geometrical Networks

1 Introduction

Feature detection is an essential stage in many Image Analysis and Computer Vision systems [2]. Some of the most low-level features to be detected in an image are the specific positions of some distinguishable points like corners. Interest points are a set of pixels in an image which are characterized by a mathematically well-founded definition [6]. These keypoints (usually, the corners which appear at the intersection of two or more image edges) present some interesting properties [7]: in particular, they have a clearly defined position in the image space, they are rich in terms of information content, and they are also stable on local and global changes in the image domain. These point variations are mainly due to image perspective transformations (i.e. scale changes, image rotations or translations) or due to illumination changes. Interest points are commonly used as local features in many image applications like content-based image retrieval or object recognition. In particular, the corresponding feature points in overlapping images can be matched among them using stereo vision techniques for

 ISBN 13: 978-84-613-5510-5

3D image reconstruction. Moreover, these feature points can also be good indicators of object boundaries and occlusion events in image sequences.

Some of the most known interest point detectors are: Moravec algorithm, Harris and Stephens algorithm, multi-scale Harris operator, SUSAN detector, genetic-programming algorithms, and affine-adapted interest point operators, among others. [11] Moravec algorithm (1980) was one of the first proposed algorithms and it defines the corner strength of a point as the smallest sum of squared differences (SSD) between the point patch and its neighbors patches (horizontal, vertical and on the two diagonals). The Harris and Stephens detector computes the locally averaged moment matrix using the image gradients, and then combines the eigenvalues of the moment matrix to compute each corner "strength". Multi-scale Harris detector works at different scales to produce a more robust detector which responds to interest points of varying sizes in the image domain. The SUSAN operator (acronym for *Smallest Univalue Segment Assimilating Nucleus*) is highly robust to noise and it finds corners based on the fraction of pixels that are similar to the center pixel within a small circular region. Some authors [12] have introduced genetic programming (GP) methods to automatically synthesize image operators aimed to find the interest points in an image. These GP operators use fitness functions which measure the stability of the operators through the repeatability rate, and also promote the uniform dispersion of detected points. Finally, detector which add robustness to perspective transformations has also been proposed [7]. These affine invariant interest points can be obtained through an affine shape adaptation process where the shape of a smoothing kernel is iteratively warped to match the local image structure around the interest point. Schmid et al [8] have proposed different techniques to compare the interest point detectors.

The purpose of this work is to introduce a novel approach to computing the interest points of an image by using complex network analysis. We associate a weighted geometrical and fast-computable complex network to each image that gives some valuable information about the location of the interest points and we can rank the regions of an image according to its interest in the whole image. The use of complex networks with a spatial structure are usual in several real-world applications [1], but this work presents a new use in the realms of Computer Vision. Since the classical mathematical definition of the interest points are mainly of local nature, we use local measures of the associated network and we discuss the use of other tools and properties of the weighted geometrical network.

2 Analyzing images through complex networks

The relevance and complexity of problems stated in Computer Vision area have motivated the use of different approaches coming from a wide range of scientific areas, including partial differential equations [9], wavelets [10] or physic-based models [13]. In this work, we propose a mathematical model based on complex networks that can help to give alternative solutions to some problems that come from Computer Vision.

The use of tools and techniques of complex network's analysis in problems dealing

 ISBN 13: 978-84-613-5510-5

with Computer Vision is an appealing scientific topic that have been stated in the last years [4] and that it is far from being well understood. The basic philosophy is to associate a complex network $G = (X, E)$ to each image I in such a way that we can analyze some properties of I from the structural and dynamical properties of G (see, for example [4]). One of the first mathematical models related to this idea was introduced in [3]. If I is an gray-level image of $N \times N$ pixels, we can associate to it a weighted network $G = (X, E)$ of $|X|{=}N^2$ nodes such that each node correspond to each pixel of I and the weight of each link $(i, j) \in E$ is:

$$w(i, j) = \|\overrightarrow{f_i} - \overrightarrow{f_j}\|_2,$$

where $\|\cdot\|_2$ denotes the Euclidean norm and $\overrightarrow{f_i} \in \mathbb{R}^m$ is a *feature vector* that describes some local *visual* properties about the respective image pixel [3]. Using the L-expansion of such network G, in [3] it was obtained an image characterization method and an image segmentation algorithm that showed the link between visual properties of an image I and local structural properties of G. Other examples of complex networks associated to an image can be found in [5], where a visual saliency detector method related to a Markov chain on other associated network G was proposed.

The main disadvantages of the associated networks introduced in [3, 5] deals with the computational complexity on such networks when the number of pixels is big. This inconvenience comes from two different facts. On the one hand the number of nodes of the associated network G is the same that the original image I, which makes computations on G to be quite slow and on the other hand, the weighted network is always a complete weighted graph, which implies inefficient computations (in time and memory). As a computationally efficient alternative, we propose considering a complex spatial network G with less nodes and much less links (actually it is a sparse network) associated to each image I.

We start with an image I of $N \times N$ pixels, such that for each of them $p \in I$ we have its intensity value $f(p) \in [0, K]$. We compute an watershed-based segmentation $R = R(I) = \{r_1, \ldots, r_k\}$ of I and we choose a set of pixels $X(R) = \{p_1, \ldots p_k\} \subseteq I$ such that for every $1 \leq j \leq k$, $p_j \in r_j$. There are several methods for spoting these pixels from the segmentation R (for example, by choosing the centroid of each region, at random, and many others), but the results obtained are similar since all the pixels in a given region have similar intensity. By using these pixels $X(R) = \{p_1, \ldots p_k\}$ as nodes we construct a weighted, sparse and spatial network $G(I) = (X(R), E)$ by defining each link weight $w(p_i, p_j)$ as follows

$$w(p_i, p_j) = \begin{cases} |f(p_i) - f(p_j)| & \text{if } r_i \text{ and } r_j \text{ are adjacent regions in } R(I), \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Note that in this case the weighted associated network is sparse and its number of nodes $k \ll N$, which makes that the computations on such networks be much more efficient than those previously stated in the associated networks. It is easy to check that the networks introduced in [3] or [5] can be also defined by using this model, simply by considering the appropriate *feature vector* describing some *visual* properties of each region r_j.

 ISBN 13: 978-84-613-5510-5

3 Interest points and centrality measures: local vs. global approaches

The main goal of associating a (weighted) complex network to each image is to analyze some visual properties of the image from the structural and dynamical properties of the corresponding network. In this section we spot the interest points of an image I by using some structural properties of its associated network $G(I)$. The heuristics behind the proposed methods deal with the fact that the interest points are related to points with a high gradient values compared to their surrounding pixels. As we pointed out in the introduction, many of the classic algorithms for the detection of interest points (such as the Moravec, or the Harris and Stephens algorithms) are based on this idea, and therefore we should try to translate them into structural properties of the associated network. Under the perspective of discrete mathematics, having in mind that the associated network $G(I)$ is a weighted network related to the difference of intensity between adjacent regions of the image, then points with high intensity gradient are related to points of high centrality in the associated network. Hence, we can spot the interest points of an image by computing the centrality of the corresponding nodes in the associated network.

There are many different centrality measures of the nodes of a network (see, for example [1]), each one of different nature and with different applications. The range of centrality measures goes from the local level (i.e. only taking into account the neighbors of each node) to the global scale (i.e. considering the whole structure of the network). Therefore, we propose two different methods according to these two scales, that are the classical level in the analysis of complex networks. In the local level, we can give an interest point detector based on the strength of each node, and therefore the interest points are related to nodes with high strength centrality. Let us remind that the *strength* of a node $i \in X$ in a network $G = (X, E)$ is the value

$$s(i) = \sum_{(i,j)\in E} w(i,j),$$

where $w(i,j)$ is the weight of the link $(i,j) \in E$. After some normalization, this value allow us to rank the nodes of the network by a local criterium that helps to locate the interest points.

If we also consider the global scale of the associated network $G(I)$, we can also get an alternative method to detect the interest points, by computing the Bonacich centrality of $G(I)$ (which is one of the most relevant example of global-scale centrality). This centrality measure is related to the dominant positive eigenvector of the adjacency matrix of $G(I)$ (see, for example [1]). After some normalization, the Bonacich centrality of each node help us to give a global ranking of the nodes in order to spot the interest points of a considered digital image.

Fig. 1 shows the results produced by our two proposed local and global centrality approaches on the same Cameraman image at a 256×256 spatial resolution. The global centrality method (i.e. highest-eigenvalue method) was applied by filtering the best 30% of the points with highest interest producing 9 interest points located at the

 ISBN 13: 978-84-613-5510-5

head region. The local centrality method (i.e. strenght-of-vertex method) was applied by filtering the best 40% of the points with highest interest produced 49 interest points uniformly distributed along the strongest edge regions in the image.

Figure 1: Visual comparison of our interest point detectors based on global centrality (on the left) with the interest point detector obtained by using the local centrality (on the right).

Acknowledgements

This work has been partially supported by the Spanish Government Projects TIN2008-06890-C02-02 and MTM2009-13848.

References

[1] Boccaletti S., Latora V., Moreno Y., Chavez M. and Hwang D. U., *Complex Networks: Structure and dynamics*, Physics Reports **424** (2006) 175–308.

[2] D.A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Prentice-Hall, 2003.

[3] L. da F. Costa, *Complex Networks, Simple Vision*, physics/0403346 (2004).

[4] L. da F. Costa, *Complex networks: new concepts and tools for real-time imaging and vision*, physics/0606060 (2006).

 ISBN 13: 978-84-613-5510-5

[5] L. da F. Costa, *Visual Saliency and Attention as Random Walks on Complex Networks*, physics/0603025 (2007).

[6] D. Lowe, *Distinctive Image features from Scale-Invariant Keypoints*, International Journal of Computer Vision **60**(2) (2004) 91–110.

[7] K. Mikolajczyk and C. Schmid, *Scale and affine invariant interest point detectors*, International Journal of Computer Vision **60**(1) (2004) 63–86.

[8] C. Schmid, R. Mohr and C. Bauckhage, *Evaluation of interest point detectors*, International Journal of Computer Vision, **37**(2) (2000), 151-172.

[9] J.A. Sethian, *Level-set methods and fast marching methods*, Cambridge University Press, 1999.

[10] J.L.Starck, F. Murtagh and A. Bijaoui, *Image Processing and data analysis*, Cambridge University Press, 1998.

[11] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2010.

[12] L. Trujillo and G. Olague, *Automated design of image operators that detect interest points*, Evolutionary Computation **16**(4) (2008), 483-507.

[13] L.B. Wolff, S.A. Shafer and G.E. Healey, *Physic based vision: principles and practice*, A.K.Peters Ed., 1992.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Hybrid MPI/PThreads Parallel implementations for 3D reconstruction in Electron Tomography

M. Laura da Silva[1], Javier Roca-Piera[1], José Antonio Martínez[1] and José Jesús Fernández[2]

[1] *Department of Computer Architecture, University of Almería*

[2] *National Center for Biotechnology, CSIC*

emails: mlauradsh@ual.es, jroca@ual.es, jmartine@ual.es, jjfdez@ual.es

Abstract

Electron tomography combines the acquisition of projection images using the electronic microscope and techniques of tomographic reconstruction to allow the structure determination of complex biological specimens. This kind of applications requires an extensive use of computational resources and considerable processing time because 3D reconstructions of high resolution are demanded. The new tendency of high performance computing heads for hierarchical computational systems, where several shared memory nodes with multi-core CPUs are connected. In this work, we propose a hybrid parallel implementation for tomographic reconstruction of cellular specimens. Our results show that the balanced and adaptative algorithm allows an ideal speedup factor when large datasets are used.

Key words: hybrid parallel computing, heterogeneous clusters

1 Introduction

The study of 3D structure of cellular specimens is essential for understanding the cellular role played by the specimen in the environment where it is located [9]. The electron microscope allows us to tilt the specimen around one or more axes and to take views from different directions collecting the projection images in digital format. The technique which makes possible to determine the 3D structure of biological samples from two-dimensional projection images obtained by electron microscope, is known as electron tomography (ET) [6]. Weighted back-projection (WBP) is the standard 3D reconstruction algorithm in ET. Furthermore, because of the resolution needs, ET of complex biological specimens requires large projection images. So, ET requires an extensive use of computational resources and considerable processing time to allow the 3D structure of cellular specimens [11]. High performace computing (HPC) has

 ISBN 13: 978-84-613-5510-5

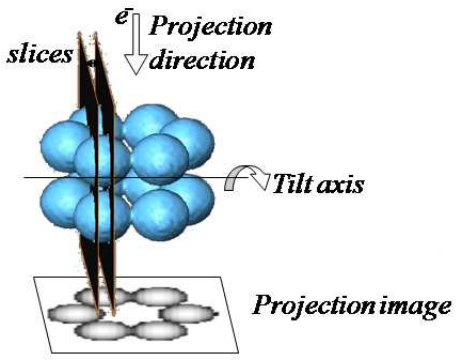


Figure 1: Acquisition of a 2D projection image while the object is tilted around an axis.

been widely investigated for many years as a means to address large-scale and grand-challenge applications. In the particular, in the field of ET, HPC allows determination of 3D structure of large volumes in reasonable computation time [4, 5, 7, 8].

On the other hand, Moore's law estimated that the number of transistors that can be placed on an integrated circuit would double approximately every two years. Given the actual physical limitation of this prediction, new types of architectures begin to appear getting less computing time. Nowadays, supercomputers are based on hierarchical computer systems which consist on several shared memory multicore nodes interconnected. Therefore the parallelism of this new generation of computer systems must be exploited at two levels: one level of parallelism distributed among the interconnected nodes and a second level of parallelism shared within the node itself [10]. In this paper, we propose a hybrid parallel implementation that exploits the parallelism of the heterogeneous architectures for 3D reconstruction of cellular specimens. Message passing libraries (OpenMPI) for communications between distributed nodes and POSIX-Thread for parallel processing within each node have been used. The results show that the balanced distribution of workload and the optimal choice of processors determine the goodness in the execution times obtained.

2 Electron Tomography

Tomography refers to the cross-sectional imaging of an object viewed from different angles. The electron tomography (ET) consists on the three dimensional (3D) reconstruction of a object from the projected two-dimensional slices which were obtained through the electron microscope. The biological specimen is placed inside the electron microscope, it is tilted over a limited range and electron beams will cross the specimen resulting a projection image with the same object area (see Fig. 1). The specimen is tilted typically from -70° to $+70^\circ$, at small tilt increments (1°–2°). These projection images will be acquired using the so-called single-axis tilt geometry and they will be recorded for each tilt angle via usually in CCD cameras. In the field of ET these projection images are known as sinograms.

The most common reconstruction methods in ET are Weighted BackProjection (WBP)

 ISBN 13: 978-84-613-5510-5

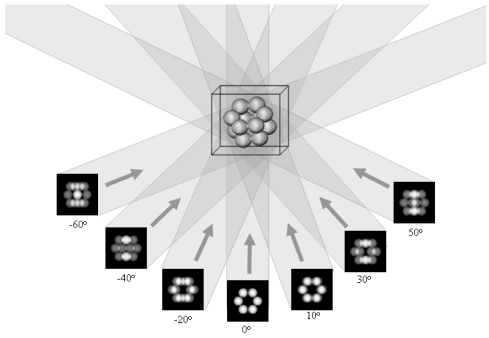


Figure 2: 3D reconstruction from projections using the WBP method.

and iterative reconstruction. Specifically, our work looks at first method, that is, WBP. Under the assumption that projection images represent the amount of mass density encountered by imaging rays, this method simply distributes the known specimen mass present in projection images evenly over the reconstruction volume. When this process is repeated for a series of projection images recorded at different tilt angles, backprojection rays from the different images intersect and reinforce each other at the points where mass is found in the original structure (see Fig. 2).

3 New tendencies of High Performance Computing and ET

Parallel computing has been widely investigated for many years as a means to provide high-performance computational facilities for large-scale and grand-challenge applications [12]. HPC addresses the computational requirements of different applications by means of the use of parallel computing on supercomputers or networks of workstations, sophisticated code optimization techniques, intelligent use of the hierarchical systems in the computers and awareness of communication latencies. In ET, the reconstruction files are usually large and, as a consequence, the processing time needed is considerable [1]. Parallelization strategies with data decomposition provide solutions to this kind of problem [4, 5, 7, 8].

The single-axis tilt geometry in ET involves the application of a computational model widely used in parallel computing known as SPMD (single-program multiple-data). In this model, all nodes of the parallel system run the same program for different data subdomains. In ET, the SPMD model consists on the decomposition of the volume in subsets of 2D slices which will be distributed among different nodes. This computational model led us to implement different strategies based on MPI parallel master-slave paradigm to study the tradeoff between distributed load and the number of nodes that perform the processing in distributed systems [3].

On the other hand, Moore's law estimated that the number of transistors that can be placed on an integrated circuit would double approximately every two years. Given the actual physical limitation of this prediction, new types of architectures begin to appear getting less computing

 ISBN 13: 978-84-613-5510-5

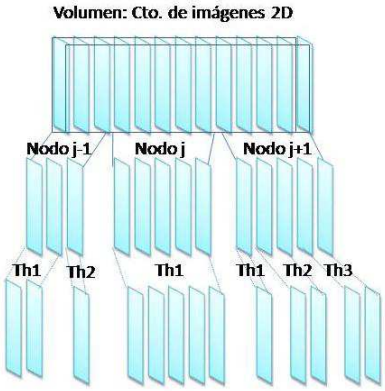


Figure 3: Decomposition of the global 3D problem into multiple, independent reconstruction problems of slabs (i.e. subsets) of slices that are assigned to different nodes in the parallel computer. Each column represent a 2D slice orthogonal to the tilt axis of the volume to be reconstructed.

time. In fact, the new architectures are based on a hierarchical computer system consisting of a distributed memory system where each node is a shared memory system with several cores and different architectural features. The SPMD parallel computation model and the new architectures lead us now to study the data parallelism at two levels: one level of parallelism distributed among the interconnected nodes and a second level of parallelism shared within the node itself. Therefore in this case, the SPMD model assumes that the different data subdomains will be distributed among the nodes and they will be again distributed within each node assigning the same workload to each core (each thread) of the shared memory system. We can observe this fact in Fig. 3.

3.1 Hybrid approaches for 3D reconstruction of cellular specimens

In this paper, we propose a parallel algorithm with centralized load balancing for the 3D reconstruction of cellular specimens, BAHPTomo (Balancing Adaptative algorithm for Heterogeneous Parallel systems in Tomography).

The proposed algorithm has two steps. During the first step, the program evaluates the performance of each node in the distributed system. To this end, node 0 sends the same sinogram to each node using MPI and each node creates a thread to perform the processing of the sinogram. Finally, node 0 gathers the time spent by each node in the processing of the sinogram. It can be observed in Fig. 4, in the first diagrams of step 0.

In the second step, node 0 does a final distribution of workload among nodes. Node 0 decides what is the best choice of cores for each node and what is the optimal workload distribution among nodes and cores. The data subsets are sent from node 0 to each node. Each one receives the new workload and it creates as threads as cores; it can be seen in Fig. 4 in the first diagram of step 1. Each thread runs the same reconstruction algorithm for different data

 ISBN 13: 978-84-613-5510-5

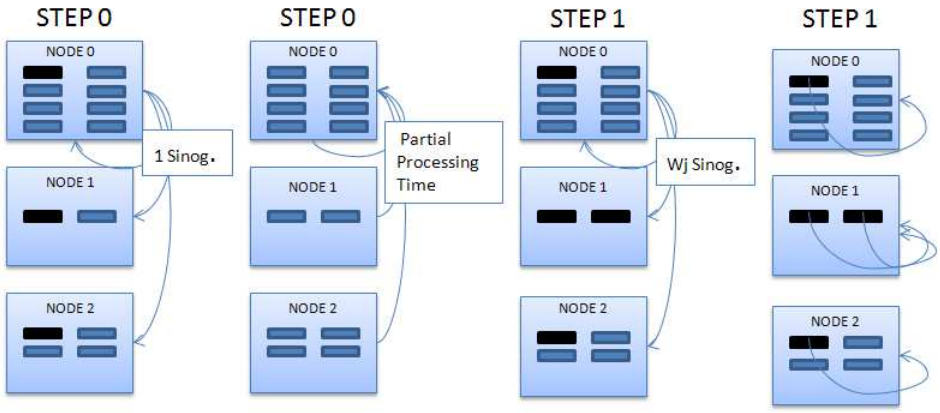


Figure 4: Diagrams of the different steps that BAHPTomo follows.

subset (it was explained in Fig. 3). Thus, each node will have a different subset of reconstructed images (see the last diagram of Fig. 4). These slices or reconstructed images will form the 3D reconstruction of the biological specimen.

In order to evaluate the algorithm eficency described above, other reconstruction algorithm was implemented. This algorithm has been called HPTomo (Hybrid Parallel algoritm for Tomography). It does not perform the testing step and it takes an equal number of processors in each node. The criterion for the workload distribution is according to the number of processors on each node.

3.1.1 Static Load Balancing

The different characteristics of each node of the heterogeneous system should be considered to do a balanced distribution of workload [2]. The approach explained above (BAHPTomo) takes into account the one hand the time spent in processing a sinogram on each node and on the other hand, the number of processors available on each node. This load balancing strategy will be formalized mathematically below.

We consider a heterogeneous computing system composed of N nodes, where each node is a shared memory system consisting on p_j processors with $j = 0, \ldots, N-1$. TS is the total number of sinograms to be distributed between the different nodes and W_j is partial work assigned to each node. The apportionment of workload W_j proposed in this article can be mathematically expressed as follows:

$$k_j = \frac{p_j * t_{min}}{t_j} \tag{1}$$

$$W_j = \frac{k_j * TS}{\sum_{j=0}^{N-1} k_j} \tag{2}$$

, where t_{min} is the time spent by the fastest node for the processing of a sinogram, and t_j is

 ISBN 13: 978-84-613-5510-5

the time spent by each node for processing of the same sinogram.

3.1.2 Optimal choice of processors

The heterogeneous systems have several nodes with different computational performances as we already have explained previously. These nodes can be composed of the same or different number of processors. If we decide to do experiments, we will have to scale the number of processors to study the performance of each one. Then, we will have to decide how many processors choose in each node for obtaining the best performance.

Different processor combinations have been tested and we have concluded that to choose the larger number of processors in the fastest node is the optimal choice. This conclussion has been taken into account in our algorithm (BAHPTomo) so that not only the application adapts to the architecture, also the architectures adapts to the application. The algorithm applied to choose the best processors is next explained.

Algorithm 1 Choice of processors

```
While total > 0 do
     i <- i + 1
     posMin <- posMinTime(tParc)
     if MAXTHRDSMAXTHRDS[posMin] >= total - (numprocs-i) do
          MAXTHRDS[posMin] <- total - (numprocs-i)
     end if
     total <- total - MAXTHRDS[posMin]
end while
```

4 Results

Datasets based on a synthetic mitochondrion phantom [4] have been used for the evaluation of the hybrid implementations. These datasets consisted of 180 projection images taken at a tilt range $[-90^{\circ}, +89^{\circ}]$ at interval of 1° and they had different sizes: 128, 256, 512 and 1024. The dataset referred to as 128 had 128 sinograms of 180 1D projections of 128×128 pixels to yield a reconstruction of $128 \times 128 \times 128$ voxels, and so forth. The number of processors has been increased following a geometric progression at the rate of 2, $P \in \{2, \ldots, 32\}$, and up to reach eventually the total number of processors among the three nodes, that is, 56 processors. Each experiment was evaluated three times and the average times for the reading, processing, writing, comunications, testing, balancing and total time were computed.

4.1 Preliminary study of our heterogeneous cluster

The hybrid approaches were implemented in C, using MPI and POSIX-Thread libraries to exploit the parallelism on node and core levels. Our models were evaluated in a heterogeneous cluster, which has three nodes with different architectural features. The first node consists on

 ISBN 13: 978-84-613-5510-5

Table 1: Processing time of 1 sinogram in each node.

Node	Sin. 128x180	Sin. 256x180	Sin. 512x180	Sin. 1024x180
N0: Opteron	0,0745709	0,287073	1,12996	4,56124
N1: Xeon	0,0477531	0,179863	0,700783	2,80184
N2: Itanium	0,193778	0,738971	2,88673	11,5836

8 processors Opteron Quad Core, it has 64 GB of RAM and the memory access is NUMA. The second node has 2 processors Intel Xeon Quad Core and it has 16 GB of RAM. Finally, the third node consists on 8 processors Intel Itanium Dual Core, it has 64 GB of RAM and the memory access is NUMA.

A preliminary study of our heterogeneous cluster has been performed. The processing time of the same sinogram in each node has been measured. We can notice in Table 1 that N2 is 1,56 times faster than N1, N2 is 4 times faster than N3 and N1 is 2,6 faster than N3. Several tests have been done concluding that the best distribution of processors is obtained when more processors of the fastest node are chosen. The BAHPTomo algorithms take into account this event.

4.2 Speed-up for heterogeneous clusters

New indicators for the measurement of the performance must be used in heterogeneous environments. We will consider the heterogeneous speed-up suggested in [10], that is, $HS = T_1/T_N$, where T_1 is the execution time in the fastest node, T_N is the execution time using N nodes. Following the same notation that Eq. 2, the ideal value of HS will be:

$$HSIdeal = \sum_{j=0}^{P-1} \frac{t_{min}}{t_j} = \sum_{j=0}^{N-1} \frac{t_{min} * p_j}{t_j} \tag{3}$$

, where P is the total number of processors.

We can observe in Fig. 5 that BAHPTomo achieves the ideal speedup when the number of processors and the size of datasets are increased. However, we can see too, if we works with a dataset of 128 and the algorithm is runned in 32 or 56 processors, the speed-up of BAHPtomo decreases. This inflection point occurs because each node has few sinograms to process and then the load balancing does not mean a great advantage between the algorithms. In fact, we can sense that the performance of BAHPtomo converges from 56 processors on.

On the other hand, we can see in Fig. 5 that penalties for testing and communication times are not significant because the speedup of BAHPTomo is very well suited to ideal speedup. This fact does not affect to the curvature change shown in Fig. 5 with 128 sinograms and 32 or 56 processors, because although there are more communications, they are lighter. Finally, we can observe in Fig. 5 that BAHPtomo algorithm gets an ideal speedup when large datasets are used, where we can obtain a speedup almost of 30 with 1024 sinograms and 56 processors.

 ISBN 13: 978-84-613-5510-5

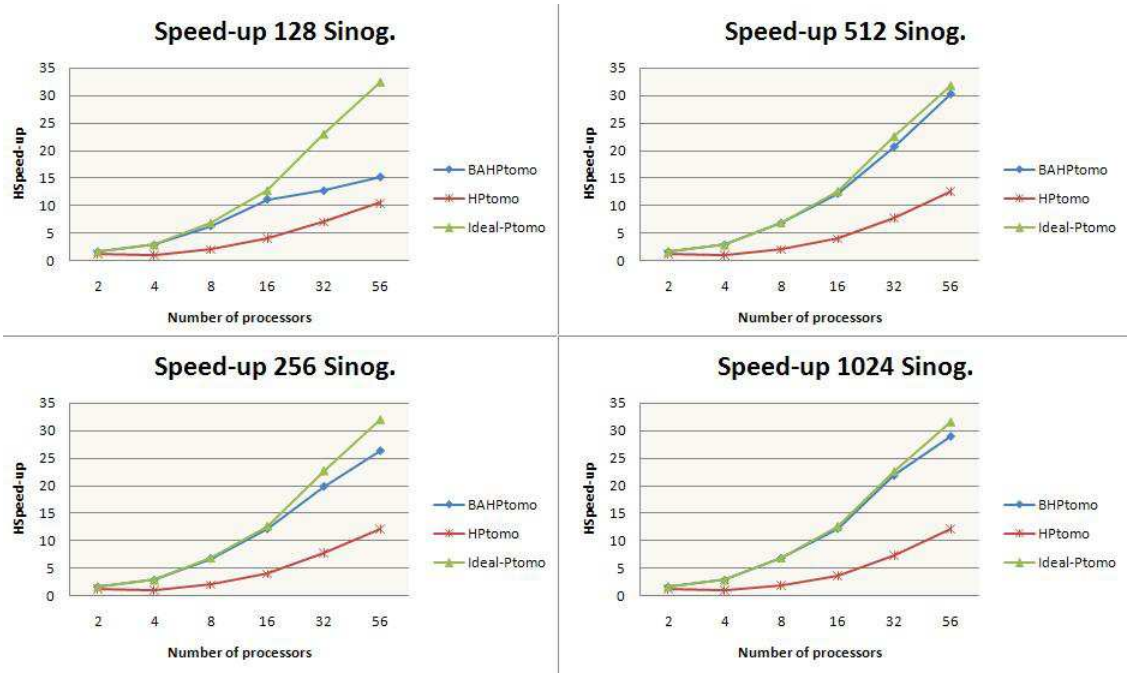


Figure 5: Speedup for datasets of 128, 256, 512 and 1024 sinograms.

5 CONCLUSIONS

In this work, the computational requirements to allow the 3D reconstruction of cellular specimens through ET have been shown. The new tendencies of high performance computing lead us to implement hybrid algorithms in order to exploit the parallelism at node and core levels. So, a hybrid C algorithm have been implemented using MPI and POSIXThread libraries. Static centralized load balancing and optimal choice of processors are taken into account in this algorithm (BAHPtomo) and a balancing method has been proposed. BAHPTomo has been evaluated in a heterogeneous cluster and it was compared with another algorithm (HPTomo) which does not take into account the different features. The results have shown that BAHPtomo algorithm gets an ideal speedup when large datasets are used. In fact, the penalties for testing time are offset by optimal load distribution. Our results demonstrate that to use our suggested balancing method has been crucial to achieve a speedup nearly at 30. So, we can conclude that it is very important to use a good load balancer to obtain the best performance in heterogeneous environments.

Acknowledgements

This work was supported by the Ministry of Science and Innovation (TIN2005-00447, TIN2008-01117), la Junta de Andalucía (P06-TIC-01426), y partially financed by European Regional Development Fund (ERDF).

 ISBN 13: 978-84-613-5510-5

References

[1] J.A. ÁLVAREZ, J. ROCA-PIERA, J.J. FERNÁNDEZ, *From structured to object oriented programming in parallel algorithms for 3D image reconstruction*, In Proceedings of the 8th Workshop on Parallel/High-Performance Object-Oriented Scientific Computing (2009) DOI: 10.1145/1595655.1595656

[2] J.A. ÁLVAREZ, J. ROCA-PIERA, J.J. FERNÁNDEZ, *A load balancing framework in multithreaded tomographic reconstruction*, Parallel Computing: Architectures, Algorithms and Applications. (NIC) 165–172

[3] M.L. DA SILVA, J. ROCA-PIERA, J.J. FERNÁNDEZ, *Evaluation of Master-Slave Approaches for 3D Reconstruction in Electron Tomography*, Lecture Notes in Computer Science **5518** (2009) 227–231

[4] J.J. FERNÁNDEZ, A.F. LAWRENCE, J. ROCA, I. GARCÍA, M.H. ELLISMAN, J.M. CARAZO, *High performance electron tomography of complex biological specimens*, J. Struct. Biol. **138** (2002) 6–20

[5] J.J. FERNÁNDEZ, J.M. CARAZO, I. GARCÍA, *Three-dimensional reconstruction of cellular structures by electron microscope tomography and parallel computing*, J. Paral. Distrib. Computing **64** (2004) 285–300

[6] J.J. FERNÁNDEZ, C.O.S. SORZANO, R. MARABINI, J.M. CARAZO, *Image processing and 3D reconstruction in electron microscopy*, IEEE Signal Process. Mag. **23(3)** (2006) 84–94

[7] J.J. FERNÁNDEZ, D. GORDON, R. GORDON, *Efficient parallel implementation of iterative reconstruction algorithms for electron tomography*, J. Paral. Distrib. Computing **68** (2008) 626–640

[8] J.J. FERNÁNDEZ, *High performance computing in structural determination by electron cryomicroscopy*, J. Struct. Biol. **165** (2008) 1–6

[9] V. LUCIC, F. FOERSTER, W. BAUMEISTER, *Structural studies by electron tomography: From cells to molecules*, Annual Review of Biochemistry **74** (2005) 833–865

[10] J.A. MARTNEZ, E.M. GARZN, A. PLAZA, I. GARCA, *ADITHE: An approach to optimise iterative computation on heterogeneous multiprocessors*, J. Supercomput. DOI 10.1007/s11227-009-0350-1 (2009)

[11] G.A. PERKINS, C.W. RENKEN, J.Y. SONG, T.G. FREY, S.J. YOUNG, S. LAMONT, M.E MARTONE, S. LINDSEY, M.H. ELLISMAN, *Electron tomography of large, multicomponent biological structures*, J. Struct. Biol. **120** (1997) 219–227

[12] B. WILKINSON, M. ALLEN, *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*, Prentice Hall (2004)

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A Low Cost Virtual 3D Human Interface Device using an Optical Flow Algorithm and GPUs

Rafael del Riego[1], José Otero[1] and José Ranilla[1]

[1] *Department of Computer Science, University of Oviedo (Spain)*

emails: rafarfn@gmail.com, jotero@uniovi.es, ranilla@uniovi.es

Abstract

Modern personal computers, including laptops, notebooks and perhaps smartphones, often have a low-resolution camera and a powerful graphic card. In this paper we present a system that uses these resources (camera and GPU) to build a low cost virtual 3D Human Interface Device. To do this, we apply an optical flow algorithm which is characterized by its high degree of parallelization. The experimental results confirm the performance of our system.

Key words: Virtual 3D-HID, GPU, Optical Flow Algorithms.

1 Introduction

Today, personal computers, laptops, notebooks and smartphones have an integrated low-resolution camera (high-resolution in the case of smartphones). Besides, most of them have a graphic processing unit (GPU), a specialized processor that offloads 3D/2D graphics rendering from the microprocessor.

A new computing parading is to use a GPU as a stream processor. This concept turns the massive floating-point computational power of a modern graphics accelerators into general-purpose computing power. In certain applications, this allow us to increase the performance in several orders of magnitude compared to a conventional CPU.

Recently, NVIDIA[1] began releasing cards supporting an API extension to the C programming language CUDA (Compute Unified Device Architecture), which allows specified functions from a normal C program to run on the GPU's stream processors. This makes C programs capable of taking advantage of a GPU's ability to operate on large matrices in parallel while still making use of the CPU when appropriate.

Being aware of these capabilities (CUDA compatible GPUs and a low-resolution camera), in this work we present a system that uses these resources to build a Low Cost Virtual 3D Human Interface Device (3D-HID). Users can interact with the environment

[1] http://www.nvidia.com

 ISBN 13: 978-84-613-5510-5

(real 3D world) by simply moving the camera (in the case of lightweight devices such as smartphones) or moving objects (e.g. hand) in the vicinity of the camera (e.g. laptops). In order to do this, we use an optical flow algorithm, which is characterized by its high degree of parallelization.

In order to point out the aim of this paper, we briefly review some aspects that will be considered. Thus, in section 2 we explain the optical flow algorithms. Section 3 is devoted to the built system. The experimental results are showed in section 4 and finally, section 5 summarizes our conclusions.

2 Optical Flow

Optical flow is the 2D vector field projection of the 3D velocities of object points. In Figure 1 a pair of frames of a classic test sequence is shown, with the true optical flow overimposed. As can be seen, the motion of the objects in the scene is well represented by the optical flow.

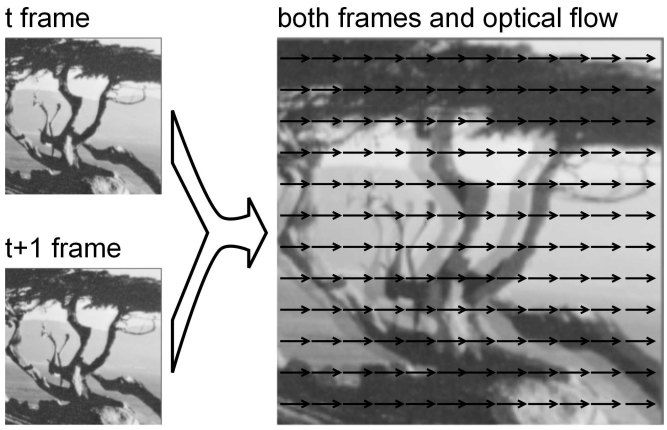


Figure 1: Optical Flow example, from two frames (left) the motion of the scene is measured for each pixel. The resulting vector field (red arrows) is shown in the right side of the figure. Both frames are overimposed.

In the literature, optical flow algorithms are classified in: correlation based techniques, frequency based techniques and gradient based techniques.

Correlation based techniques or block matching algorithms [1] try to maximize a measure of similarity between patches (taken from two consecutive frames) centered in a given pixel. The displacement that maximizes the selected measure divided by the time interval within the acquisition of the frames is the velocity of the pixel (see Figure 2).

Frequency based techniques use a set of tuned spatiotemporal filters to search for the velocity of a pixel [3].

Gradient based techniques use the well known Optical Flow Constraint (OFC) shown in equation 1 in order to compute the optical flow [4].

 ISBN 13: 978-84-613-5510-5

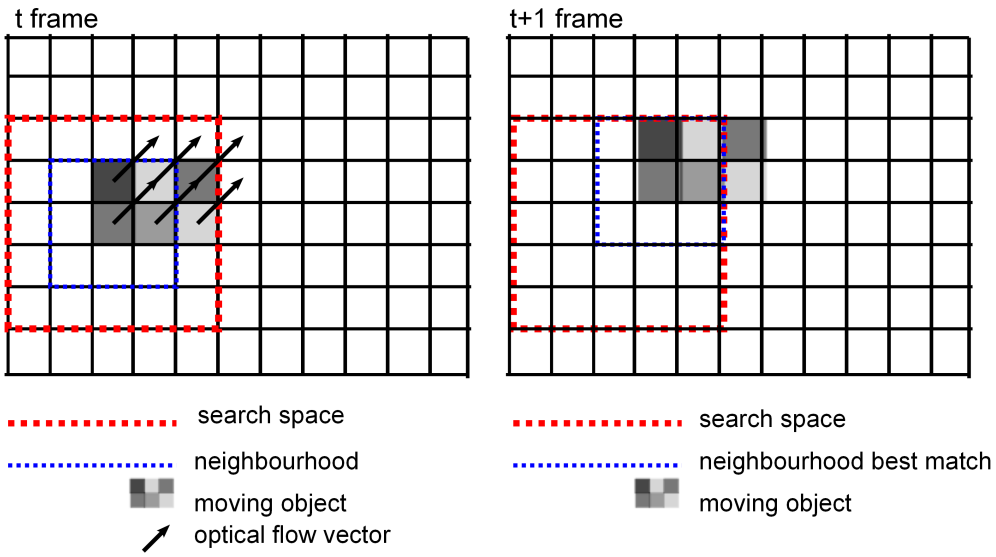


Figure 2: Optical Flow computation using a block matching algorithm. The neighborhood in the first frame (blue dotted square) is found in the second frame in different position. This displacement defines the Optical Flow vector for each pixel.

$$-\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} = \frac{\partial f}{\partial x}u + \frac{\partial f}{\partial y}v = \nabla(f)\cdot\overrightarrow{c} \tag{1}$$

Equation 1 makes the assumption that intensity changes in a sequence of images are only due to the movement of the objects in the scene: a single pixel will have constant brightness in the different positions that it takes during the sequence. Unfortunately, the Aperture Problem (see [5]) states that there is no way to recover the complete optical flow vector using only local (one pixel) information.

In Figure 3 a synthetic example is shown. As can be seen, OFC holds for the selected pixel $(4,1)$, $(-1,-1)$ velocity verifies the obtained equation: replacing $\frac{\partial f}{\partial x}$ by 1, $\frac{\partial f}{\partial y}$ by 2 and $\frac{\partial f}{\partial t}$ by 3, $u+2v=-3$ is obtained. Unfortunately, a suitable value that verifies the OFC can be found for one of the components substituting the other by an arbitrary value.

Some authors try to solve the aperture problem with the incorporation of some kind of global information, involving a process of regularization [4]. Some researchers perform a clustering of the OFCs themselves in order to find the most reliable one. Once obtained, the corresponding normal flow to that OFC is obtained [6]. Another alternative is to analyze the measurements in the space of the velocities that is, performing an estimation of the velocity with the results of many systems of OFC equations. Each system of equations is obtained from one pair of pixels in order to estimate the velocity. In this way, the analysis is performed directly in the domain of the data that we want to recover, that is, the u, v space [7, 8, 9].

All the previously approaches are computationally expensive. For example, for a

 ISBN 13: 978-84-613-5510-5

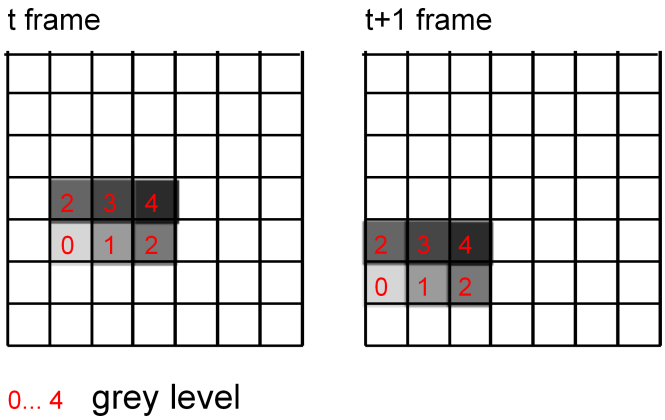


Figure 3: Optical Flow computation using OFC. As can be seen OFC holds for pixel (4,1): $\frac{\partial f}{\partial x} = 1$, $\frac{\partial f}{\partial y} = 2$ and $\frac{\partial f}{\partial t} = 3$, then OFC is $u + 2v = -3$, that holds for $(-1, -1)$.

$n \times m$ pixels image, a search space of $p \times q$ pixels and a neighborhood of $s \times t$ pixels, the number of floating point operations in the case of BMA is $n \times m \times p \times q \times s \times t$. Similar numbers are obtained for gradient based approaches. Because of this, an implementation using a GPU and CUDA speeds up the computation process.

3 The System

System architecture is divided in four main blocks: Video Input, Optical Flow Algorithm, Motion Estimator and Control (see figure 4).

Each subsystem comprises several basic computing units (called *basic-units*). Thus, the execution of several *basic-units* of different images is concurrent (like works pipelined CPUs). This solution is adequate even for single core CPUs; simply the degree of concurrence is smaller.

The communications between *basic-units* use circular buffers. This leads to an increasing of memory consumption but allows more efficient asynchronous execution of the *basic-units*.

Because of the low image resolution, only a smoothing with a small Gaussian kernel is needed in order to decrease the acquisition noise. This benefits the parallel implementation of the whole system because the separability of the filtering.

We will use optical flow field estimation in order to measure/detect the motion in the image sequence acquired with the cameras. The goal is to use the translations in X, Y and Z along with rotation in Z as input signals to the proposed virtual interface.

In order to discriminate the predominant motion in the scene, we use the following operators:

- X and Y translations are measured as the average optical flow in the image: $(X, Y)_T = \frac{\sum_{i=0}^{i=n} \sum_{j=0}^{j=m} (F_x, F_y)_{i,j}}{nm}$, where $(X, Y)_T$ is the traslation vector, $(F_x, F_y)_{i,j}$

 ISBN 13: 978-84-613-5510-5

is the Optical Flow Vector for pixel (i, j), n, m are the number of rows and columns in the image.

- Z translation is measured with the divergence of the optical flow averaged across the image: $Z_T = \frac{\sum_{i=0}^{i=n} \sum_{j=0}^{j=m} \bigtriangledown \cdot (F_x, F_y)_{i,j}}{nm}$ The previous expression is evaluated and averaged for each optical flow value across the whole image.
- Z rotation is measured with the rotational of the optical flow averaged across the image: $Z_R = \frac{\sum_{i=0}^{i=n} \sum_{j=0}^{j=m} \bigtriangledown \times (F_x, F_y)_{i,j}}{nm}$

We implemented two algorithms using CUDA, the proposal in [7] and an hierarchical implementation of Lucas-Kanade algorithm available with OpenCV library [10]. Finally, a bottleneck in Otero et. al. algorithm [7] leads us to choose Lukas-Kanade algorithm [10]. The output of this algorithm is evaluated with the previous operators. The highest output defines the predominant motion in the scene.

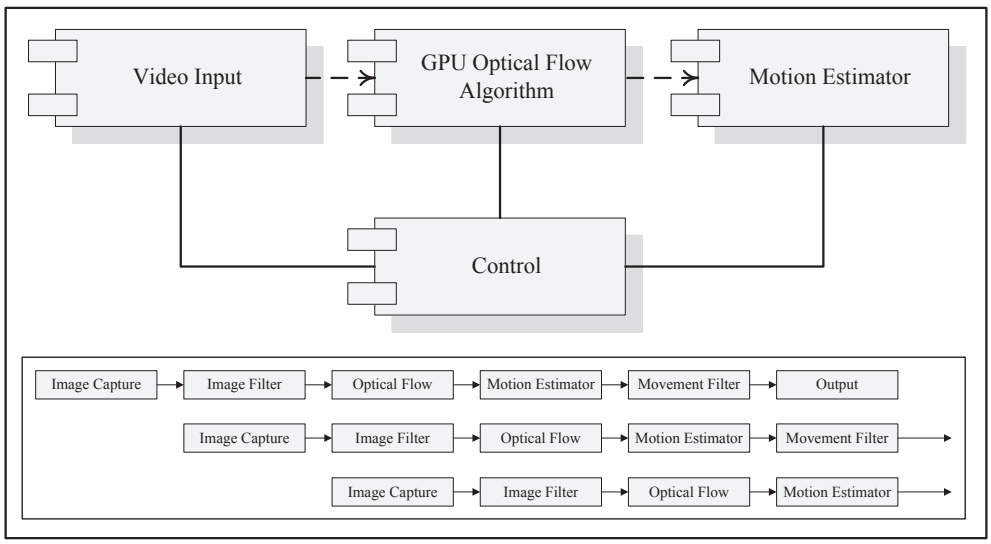


Figure 4: System Architecture

4 Experiments

The hardware setup comprises a laptop and two cameras with different resolution, the usual integrated in the screen frame and an off the shelf usb camera, with the following technical specs:

- CPU AMD Athlon 64 3000+ (1.8 GHz) AM2.
- GPU nVidia GeForce 9500 GT.
- RAM 1024 MB Dual Channel 800 MHz.

 ISBN 13: 978-84-613-5510-5

- Web cam 1 Logitech Quickcam E2500.
- Web cam 2 Logitech Webcam C200.

The system detects easily the motion in different axis, only when velocity module falls below a threshold some errors may appear, mainly due to image acquisition and illumination issues (flickering or low illumination).

Low illumination leads to noise in the image and to the false detection of small movements due to the noisy pixels that appear and disappear. We decided to filter the movements that are below a threshold in order to minimize that error.

During the experiments we found that X and Y translations are easily detected but Z translation is easily detected as X or Y motion, because small displacements in X or Y directions (by the user) lead to relatively high values compared with the divergence of the optical flow field.

When the algorithm was running in the CPU we obtained 17 frames per second, with the performance of other tasks being degraded.

Using CUDA implementations the frame rate increases to 30 frames per second, the hardware limit of the cameras. Standard GPU tasks are not degraded and the CPU can be fully dedicated to other tasks. Thus, the user experience is real-time alike.

Another kind of movement useful as input signal in the virtual interface is rotation in Z axis. The amount of motion cannot be accurately measured but if we use a threshold it can be used to simulate a click.

Summarizing, X and Y translations are correctly detected and measured. Z translations cannot be accurately measured and it is not useful as input signal. Z rotation cannot be accurately measured but a suitable threshold can be used and then, it serves as binary signal.

5 Conclusions

In this work we have showed how to build a Virtual 3D Human Interface Device by using standard resources of the current computers: the integrated camera (or Web cam) and the graphic processing unit (GPU). The system applies optical flow techniques and uses CUDA (Compute Unified Device Architecture, nVidia) to exploit the capabilities of the GPU as stream processor.

X and Y translations are correctly detected and measured by the system. Z translations are detected but not accurately measured and Z rotation cannot be accurately measured but a suitable threshold can be used and then serves as binary signal.

Summarizing, the presented solution is simple, the frame rate is now limited by the resolution of the camera (30 frames per second *vs* 17 in the case of CPU's based solutions), users experience is real-time alike, computer's performance is not being degraded and powerful/additional hardware it is not necessary. Therefore, we have built an efficient and low cost 3D interface.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

We would like to acknowledge the help received from the Spanish Office of Science and FEDER, for their support of the projects TIN2007-61273 and TIN2008-06681-C06-04.

References

[1] P. Anandan, *A computational framework and an algorithm for the measurement of visual motion*, International Journal of Computer Vision **2** (1989) 283–310.

[2] J. L. Barron, D. J. Fleet and S. S. Beauchemin, Performance of optical flow techniques, International Journal of Computer Vision **12(1)** (1994) 43-77.

[3] D. J. Heeger, *Optical flow using spatiotemporal filters*, International Journal of Computer Vision (1988) 279-302.

[4] Berthold K. P. Horn and Brian G. Schunk, *Determining Optical Flow*, In Computer Vision Principles, MIT, Artificial Intelligence Laboratory, 481-497, 1980.

[5] David W. Murray and Bernard F. Buxton, *Experiments in the Machine Interpretation of Visual Motion*, MIT Press, 1990.

[6] P. Nesi, A. del Bimbo and D. Ben Tzvi, *Robust Algorithm for Optical Flow Estimation*, Journal on Computer Vision, Graphics and Image Processing: Image Understanding **61(2)** (1995) 59-68.

[7] J. Otero, A. Otero and L. Sánchez, *Mode based hierarchical optical flow estimation*, MG&V **10(4)** (2001) 489-501.

[8] José Otero, Adolfo Otero and Luciano Sánchez, *3D motion estimation of bubbles of gas in fluid glass, using an optical flow gradient technique extended to a third dimension*, Mach. Vis. Appl. **14(3)** (2003) 185-191.

[9] Brian G. Schunck, *Image Flow Segmentation and Estimation by Constraint Line and Clustering*, IEEE Transactions on Pattern Analysis and Machine Intelligence (1989) 1010-1027.

[10] Jean-Yves Bouguet, *Pyramidal implementation of the lucas kanade feature tracker*, OpenCV Documentation, 2000.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Assessing the disclosure risk by fuzzy sets and cardinalities

Irene Díaz[1], Luis J. Rodríguez-Muniz[2] and Luigi Troiano[3]

[1] *Department of Computer Science, University of Oviedo*

[2] *Department of Statistics and O.R., University of Oviedo*

[3] *Department of Engineering, University of Sannio*

emails: sirene@uniovi.es, luisj@uniovi.es, troiano@unisannio.it

Abstract

Key words: Privacy, k-anonymity, l-diversity, t-closeness, cardinalities

Introduction

As gaining access to statistical micro-data (i.e. those data are not summarized by some statistics) is becoming a common tool for researchers, issues regarding disclosure of sensitive information about individuals and organizations arise. While releasing provides researchers with useful information, the risk is to involuntary disclose information that should kept reserved.

Micro-data are generally organized in tables whose attributes can (i) lead to identities, such as address, name, social security number, and (ii) release sensitive information, such as diseases and income, such those regarding census, medical issues, finance and others. In particular those attributes that are directly linked to identity are know as *identifiers*, whilst other attributes related at some extent to identity potentially able to identify an individual are known as *quasi-identifiers*.

This is functional to make distinction between *identity disclosure* and *attribute disclosure* in released table. The first occurs when an individual is linked to a particular record, whilst the latter occurs when new information regarding some individuals is revealed.

Objective of statistical disclosure control (SDC) is to limit the risk of releasing sensitive information to an acceptable level. This goal is achieved by anonymization of data, obtained by removing explicit references to identity and by replacing the other attribute values related to identity with values less specific but semantically consistent. This leads to group records with the same quasi-identifier values into *equivalence classes*. In an equivalence class, individuals are made indistinguishable.

 ISBN 13: 978-84-613-5510-5

Disclosure risk metrics

In literature, different metrics able to quantify the disclosure risk with respect to an anonymized table have been proposed.

Samarati and Sweeney [7, 8] define k-anonymity as the property that each record is indistinguishable with at least $k-1$ other records with respect to the quasi-identifier, that is requiring that each equivalence class contains at least k records. Although k-anonymity is able to quantify the risk of identity disclosure, it is not able to assess the risk of attribute disclosure.

Machanavajjhala et al. [5] propose l-diversity as means to overcome k-anonymity limitations. l-diversity requires that the distribution of a sensitive attribute in each equivalence class has at least l values. In particular, authors consider three different declinations of l-diversity, known as (i) *distinct diversity*, entailing that at least l distinct values of sensitive data occur in each equivalence class, (ii) *entropy diversity*, requiting that entropy of sensitive values distribution is greater or equal than $log(n)$, (iii) *recursive diversity*, ensuring that the most frequent value does not appear too frequently, and the less frequent values do not appear too rarely. Although improving the definition of k-anonymity (i.e. it is stronger definition of privacy), this metric does not asses the risk of *Skewness* and *Similarity* attacks. Indeed l-diversity does not face the risk of attribute disclosure when the distribution of sensitive data is skewed, as belonging to an equivalence class would make individuals more vulnerable to be associated to sensitive data than considering the overall distribution. In addition, l-diversity does not take into account that data although different can be similar. This is especially the case of numeric values.

Li, Li and Venkatasubramanian [4] attempt to solve leaks of k-anonymity and l-diversity by proposing a definition of privacy based on distance between values and known as t-closeness. In particular, this metric requires that the distribution of a sensitive attribute in any equivalence class is close (i.e. below the threshold t) to the distribution of the attribute in the overall table. The distance between distribution is measured as Earth Mover distance (EMD), that is the minimal amount of work needed to transform one distribution to another by moving distribution mass between each other. Although this approach introduces the concept of similarity between values and distribution, definition is based on distance instead of number of values, so that it is not possible to link it to k-anonimity and l-diversity. In addition, EMD does not fit well categorical data as based on total ordering of values.

Contribution

In this paper we assume a different perspective, related to the granularity of information. Similarly to t-closeness, relations that are not visible when observed on punctual data, become more evident when we generalize data. For instance, let us consider Table 1.

Table 2 and Table 3 respectively propose 3-diversity and 0.3-closeness anomyzation schemes. In particular the second is EMD is 0.167 for Salary and 0.278 for Disease.

 ISBN 13: 978-84-613-5510-5

Table 1: Original Salary/Disease Data

	ZIP Code	Age	Salary	Disease
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Table 2: Anonymization induced by l-diversity

	ZIP Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

Table 3: Anonymization induced by t-closeness

	ZIP Code	Age	Salary	Disease
1	4767*	≤ 40	3K	gastric ulcer
3	4767*	≤ 40	5K	stomach cancer
8	4767*	≤ 40	9K	pneumonia
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
2	4760*	≤ 40	7K	gastritis
7	4760*	≤ 40	9K	bronchitis
9	4760*	≤ 40	10K	stomach cancer

 ISBN 13: 978-84-613-5510-5

Although Table 2 provides an anonymized version of Table 1 which satisfies the distinct and entropy 3-diversity, there are some threats to privacy. For example, let us suppose to know that Bob is in the 20s and that he lives in the area 47678. Then we deduce that his salary is relatively low, i.e. in between 3k and 5k. In addition we can infer he suffers of some stomach related disease. Both information can be deduced due to similarity of information.

At this end, fuzzy set theory provides a natural framework to analyze data generalization and to identify threats to privacy. Since generalization is about grouping elements in classes, and membership cannot be sharply defined, a class of elements can be regarded as fuzzy set. Privacy is preserved and disclosure protected, if the anonymization scheme chosen is able to mix sensitive data in such a way to make them indistinguishable at different level of generalization.

Therefore, a key point in defining privacy in our approach is about counting elements in a fuzzy set. In literature, several definitions of cardinality for finite fuzzy sets have been proposed [9]. They can be roughly divided in two categories: the scalar cardinalities [2, 10], which associate to each fuzzy set a quantity (natural or real number) and the fuzzy cardinalities [6, 3, 1], which associate to each fuzzy set a function over the natural numbers with values in the unit interval $[0, 1]$. Both can be appropriately employed in a definition of privacy by fuzzy sets. Fuzzy sets can be assumed in isolation or as member of partitions. With respect to a fuzzy set, to a collection of them or to a partition, the aim is to check if the number of elements for each equivalence class, is similarly distributed along the whole dataset, and if elements are enough sparse so that identification of individuals does not lead to associate sensitive data to them.

As an example let us consider for Salary a partition made of triangular fuzzy sets $Low \equiv (-\infty; 2; 5)$, $Average \equiv (2; 5; 8)$ and $High \equiv (5; 8; +\infty)$. With respect to anonymization scheme outlined in Table 2, we get

Table 4: Cardinalities w.r.t. Table 2

	ZIP Code	Age	Salary	Low	Average	High
1	476**	2*	3K	0.67	0.33	0
2	476**	2*	4K	0.33	0.67	0
3	476**	2*	5K	0	1	0
4	4790*	≥ 40	6K	0	0.67	0.33
5	4790*	≥ 40	11K	0	0	1
6	4790*	≥ 40	8K	0	0	1
7	476**	3*	7K	0	0.33	0.67
8	476**	3*	9K	0	0	1
9	476**	3*	10K	0	0	1

If we assume σ-count as simple definition of cardinality, we get that $|Low| = 1$, $|Average| = 3$, $|High| = 5$, whose entropy is 0.937 ($\log 3 = 1.099$). If we restrict attention to the first three records, we have $|Low|_{1,2,3} = 1$, $|Average|_{1,2,3} = 2$ and $|High|_{1,2,3} = 0$, with entropy 0.637. Differently, if we assume anonymization outlined in Table 3, we have

 ISBN 13: 978-84-613-5510-5

Table 5: Cardinalities w.r.t. Table 3

	ZIP Code	Age	Salary	Low	Average	High
1	4767*	≤ 40	3K	0.67	0.33	0
3	4767*	≤ 40	5K	0	1	0
8	4767*	≤ 40	9K	0	0	1
4	4790*	≥ 40	6K	0	0.67	0.33
5	4790*	≥ 40	11K	0	0	1
6	4790*	≥ 40	8K	0	0	1
2	4760*	≤ 40	4K	0.33	0.67	0
7	4760*	≤ 40	7K	0	0	1
9	4760*	≤ 40	10K	0	0	1

Obviously the overall cardinalities do not change as still $|Low| = 1$, $|Average| = 3$, $|High| = 5$. But $|Low|_{1,3,8} = 0.67$, $|Average|_{1,3,8} = 1.33$ and $|High|_{1,3,8} = 1$, whose entropy is 1.062. Higher entropy entails better dissimulation of data, thus stronger privacy preservation. Example above, shows how t-closeness is related to cardinalities. This relationship stands in a more general way.

As this is required at any level of generalization, a further step consists in checking if there exists at least one fuzzy set or partition able to violate the condition above.

This contribution aims at proposing a theoretical framework for privacy based on fuzzy sets and cardinalities, and investigating properties and relationship to other privacy definitions. Several examples and experiments prove this approach is feasible, leading to a natural definition of privacy able to include k-anonymity and l-diversity as special cases. Although similar to t-closeness in facing similarity and skewness attacks, and in being oriented to information gain, this approach differs as it is directly based on notion of classes instead of assuming distance as means of similarity.

Acknowledgement

Authors acknowledge financial support by Grant MTM2008-01519 from Ministry of Science and Innovation and Grant TIN2007-61273 from Ministry of Education and Science, Government of Spain

References

[1] J. Casasnovas and J. Torrens. An axiomatic approach to fuzzy cardinalities of finite fuzzy sets. *Fuzzy Sets and Systems*, 133(2):193–209, 2003.

[2] D. Dubois and H. Prade. Fuzzy cardinality and the modeling of imprecise quantification. *Fuzzy Sets and Systems*, 16(3):199 – 230, 1985.

[3] L.-C. Jang and D. Ralescu. Cardinality concepts for type-two fuzzy sets. *Fuzzy Sets Syst.*, 118(3):479–487, 2001.

 ISBN 13: 978-84-613-5510-5

[4] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE*, pages 106–115. IEEE, 2007.

[5] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. -diversity: Privacy beyond -anonymity. *TKDD*, 1(1), 2007.

[6] D. Ralescu. Cardinality, quantifiers, and the aggregation of fuzzy criteria. *Fuzzy Sets Syst.*, 69(3):355–365, 1995.

[7] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. on Knowl. and Data Eng.*, 13(6):1010–1027, 2001.

[8] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):571–588, 2002.

[9] M. Wygralak. Questions of cardinality of finite fuzzy sets. *Fuzzy Sets Syst.*, 102(2):185–210, 1999.

[10] M. Wygralak. An axiomatic approach to scalar cardinalities of fuzzy sets. *Fuzzy Sets Syst.*, 110(2):175–179, 2000.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Analysis and numerical simulation of an induction-conduction model arising in steel heat treating

J. M. Díaz Moreno[1], C. García Vázquez[1], M. T. González Montesinos[2] and F. Ortegón Gallego[1]

[1] *Departamento de Matemáticas, Universidad de Cádiz*

[2] *Departamento de Matemática Aplicada I, Universidad de Sevilla*

emails: josemanuel.diaz@uca.es, concepcion.garcia@uca.es, mategon@us.es, francisco.ortegon@uca.es

Abstract

The goal of steel heat treating is to create a hard enough part over certain critical surfaces or volumes of the workpiece and at the same time keeping its ductibility properties all over the rest of the workpiece.

We consider a mathematical model for the description of the heating-cooling industrial process of a steel workpiece. This model consists of a nonlinear coupled partial differential system of equations involving the electric potential, the magnetic vector potential, the temperature, together with a system of ordinary differential equations for the steel phase fractions. Due to the different time scales related to the electric potential and the magnetic vector potential versus the temperature, we introduce the harmonic regime, leading to a new system of nonlinear PDEs. Finally, we have carried out some 2D numerical simulations of this heating-cooling industrial process.

Key words: Steel hardening, phase fractions, nonlinear parabolic-elliptic equations, Sobolev spaces, finite elements method.

MSC 2000: 35A15, 35G30, 35J57, 35K05, 35K55, 35Q61, 35Q80, 46E35.

1 Introduction

This work deals with the mathematical analysis and numerical simulations of a model governed in terms of a nonlinear system of partial differential equations/ordinary differential equations describing the industrial process of steel hardening, including phase transitions. This subject has been extensively studied during the last years ([3, 5, 6, 7, 9]). A complete model, including thermomechanical effects can be seen, for instance, in [9]. Here our main concerned is the description of the temperature, dropping out

 ISBN 13: 978-84-613-5510-5

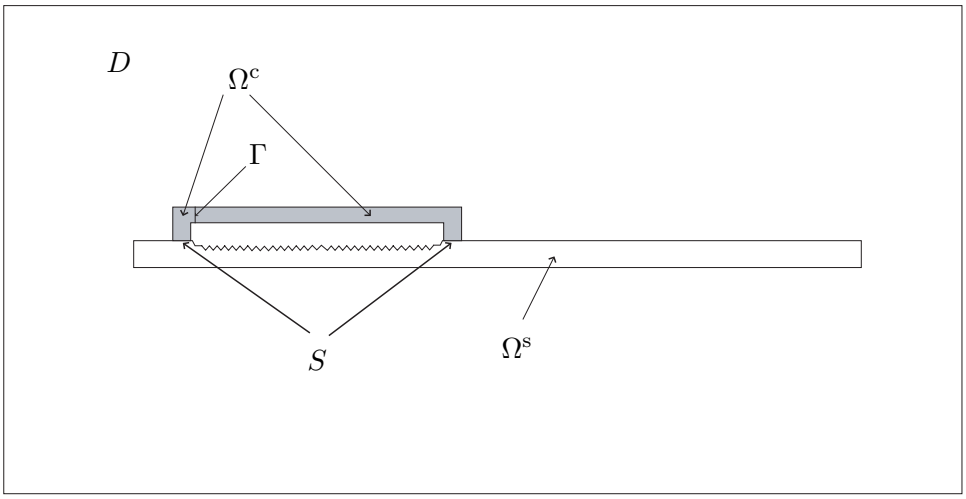


Figure 1: Domains D, $\Omega = \Omega^{\rm s} \cup \Omega^{\rm c} \cup S$ and the interface $\Gamma \subset \Omega^{\rm c}$. The inductor $\Omega^{\rm c}$ is made of copper. The workpiece contains a toothed part to be hardened by means of the heating-cooling process described below. It is made of a hypoeutectoid steel.

mechanical effects. The mathematical model is governed by a coupled nonlinear system of PDEs/ODEs, namely

$$\left.\begin{aligned} \nabla \cdot (\sigma(\theta)\nabla\phi) &= 0 \text{ in } \Omega_T = \Omega \times (0,T), \\ \sigma_0(\theta)\boldsymbol{\mathcal{A}}_t + \nabla \times (\tfrac{1}{\mu}\nabla \times \boldsymbol{\mathcal{A}}) - \delta\nabla(\nabla \cdot \boldsymbol{\mathcal{A}}) &= -\sigma_0(\theta)\nabla\phi \text{ in } D_T = D \times (0,T), \\ \boldsymbol{\mathcal{A}}(0) &= \boldsymbol{\mathcal{A}}_0 \text{ in } \Omega, \\ z_t &= F(\theta, z) \text{ in } \Omega^{\rm s}_T = \Omega^{\rm s} \times (0,T), \\ z(0) &= z_0 \text{ in } \Omega^{\rm s}, \\ \rho c_\epsilon \theta_t - \nabla \cdot (\kappa(\theta)\nabla\theta) &= \sigma_0(\theta)|\boldsymbol{\mathcal{A}}_t + \nabla\phi|^2 + \rho L z_t + G \text{ in } \Omega_T, \\ \theta(0) &= \theta_0 \text{ in } \Omega. \end{aligned}\right\} \quad (1)$$

where $\Omega, D \subset \mathbb{R}^N$, $N = 2$ or 3, are bounded, connected and Lipschitz-continuous open sets such that $\bar{\Omega} \subset D$, $\Omega = \Omega^{\rm c} \cup \Omega^{\rm s} \cup S$ is the set of conductors, $\Omega^{\rm c}$ the inductor (usually made of copper), $\Omega^{\rm s}$ the steel workpiece, $\Omega^{\rm c}$ and $\Omega^{\rm s}$ being open sets, and $S = \bar{\Omega}^{\rm c} \cap \bar{\Omega}^{\rm s}$ is the surface contact between $\Omega^{\rm c}$ and $\Omega^{\rm s}$, $\Omega^{\rm c} \cap \Omega^{\rm s} = \varnothing$ (see Figure 1); T stands for the final time of observation; ϕ the electrical potential; $\boldsymbol{\mathcal{A}}$ the magnetic vector potential; G a given external source coming for the mechanical deformation (here assumed to be known); θ the temperature; $z = (z_1, z_2)$, z_1 and z_2 are the phase fractions ([1,2,6]) of austenite and martensite, respectively; $F = (F_1, F_2)$ gives the phase fractions model; $\kappa(\theta)$ is the thermal conductivity; $\sigma(\theta)$ the electrical conductivity (by $\sigma(\theta)$ we mean the function $(x,t) \mapsto \sigma(x, \theta(x,t))$, and also for $\kappa(\theta)$, etc.); $\sigma_0(x,s) = \sigma(x,s)$ if $x \in \bar{\Omega}$, $\sigma_0(x,s) = 0$ elsewhere; $\mu = \mu(x)$ is the magnetic permeability; ρ the density; $L =$

 ISBN 13: 978-84-613-5510-5

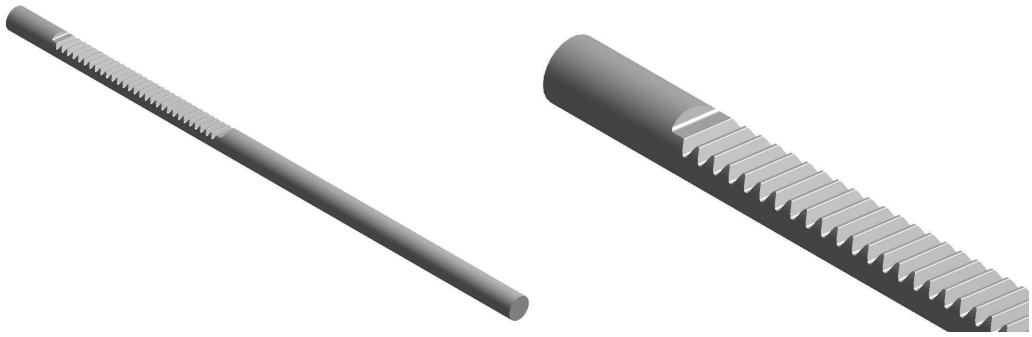


Figure 2: Car steering rack.

(L_1, L_2) is the latent heat; c_ϵ is the specific heat capacity at constant strain; $\delta > 0$ is a small constant. System (1) is supplied with suitable boundary conditions.

The induction-conduction model (1) describes the heating process of a steel workpiece. Once the desired high level of temperature is reached at certain critical parts along the workpiece, the supplied electric current is switched off and the workpiece is then quenched in order to cool it down rapidly. The goal is to produce martensite (hard and brittle steel phase transition) in these critical parts, keeping the rest ductile. Usually, these parts correspond to particular structural components whose surface is going to be highly stressed during its mechanical lifetime. This is the case of a car steering rack (see Figure 2).

In [9], it is assumed the Coulomb gauge condition for the magnetic vector potential, namely, $\nabla \cdot \boldsymbol{\mathcal{A}} = 0$. In our analysis, we do not impose this condition since this makes appear an undesired pressure gradient in the equation for $\boldsymbol{\mathcal{A}}$. In its turn, we include a penalty term in this equation of the form $-\delta\nabla(\nabla \cdot \boldsymbol{\mathcal{A}})$.

2 The mathematical description for the heating-cooling process

We split the time interval $[0, T]$ into two intervals: $[0, T] = [0, T_{\rm h}) \cup [T_h, T_{\rm c}]$, $T_{\rm c} > T_{\rm h} > 0$. The first one $[0, T_{\rm h})$ corresponds to the heating process. All along this time interval, a high frequency electric current is supplied through the conductor which in its turn induces a magnetic field. The combined effect of both conduction and induction gives rise to a production term in the energy balance equation, namely $b(\theta)|\boldsymbol{\mathcal{A}}_t + \nabla\phi|^2$. This is Joule's heating. At the instant $t = T_{\rm h}$, the current is switched off and during the time interval $[T_h, T_{\rm c}]$ the workpiece is cooled down by means of aqua-quenching.

 ISBN 13: 978-84-613-5510-5

The heating model

The current passing through the set of conductors $\Omega = \Omega^{\rm c} \cup \Omega^{\rm s}$ is modeled with the aid of an auxiliary smooth surface $\Gamma \subset \Omega^{\rm c}$ cutting the inductor $\Omega^{\rm c}$ into two parts, each one of them having a surface contact over the boundary of the workpiece $\Omega^{\rm s}$ (see Figure 1). For the sake of simplicity, we will assume that $\rho c_\epsilon = 1$. The heating model reads as follows

$$\nabla \cdot (\sigma(\theta)\nabla\phi) = 0 \text{ in } \Omega_{T_{\rm h}} = \Omega \times (0, T_{\rm h}), \tag{2}$$

$$\frac{\partial \phi}{\partial n} = 0 \text{ on } \partial\Omega \times (0, T_{\rm h}), \tag{3}$$

$$\left[\sigma(\theta)\frac{\partial \phi}{\partial n}\right]_\Gamma = j_S \text{ on } \Gamma \times (0, T_{\rm h}), \tag{4}$$

$$\sigma_0(\theta)\boldsymbol{\mathcal{A}}_t + \nabla \times \left(\frac{1}{\mu}\nabla \times \boldsymbol{\mathcal{A}}\right) - \delta\nabla(\nabla \cdot \boldsymbol{\mathcal{A}}) = -\sigma_0(\theta)\nabla\phi \text{ in } D \times (0, T_{\rm h}), \tag{5}$$

$$\boldsymbol{\mathcal{A}} = 0 \text{ on } \partial D \times (0, T_{\rm h}), \tag{6}$$

$$\boldsymbol{\mathcal{A}}(0) = \boldsymbol{\mathcal{A}}_0 \text{ in } \Omega, \tag{7}$$

$$z_t = F(\theta, z) \text{ in } \Omega^{\rm s} \times (0, T_{\rm h}), \tag{8}$$

$$z(0) = z_0 \text{ in } \Omega^{\rm s}, \tag{9}$$

$$\theta_t - \nabla \cdot (\kappa(\theta)\nabla\theta) = \sigma_0(\theta)|\boldsymbol{\mathcal{A}}_t + \nabla\phi|^2 + \rho L z_t + G \text{ in } \Omega_{T_{\rm h}}, \tag{10}$$

$$\frac{\partial \theta}{\partial n} = 0 \text{ on } \partial\Omega \times (0, T_{\rm h}), \tag{11}$$

$$\theta(0) = \theta_0 \text{ in } \Omega. \tag{12}$$

In (4) $[\,\cdot\,]_\Gamma$ stands for the jump across the inner surface Γ. The function j_S represents the external source current density. The domain D containing the set of conductors is taken big enough so that the magnetic vector potential $\boldsymbol{\mathcal{A}}$ vanishes on its boundary ∂D. Since z is only defined in $\Omega^{\rm s}$, the term $\rho L z_t$ appearing in (10), and in (15) below, is assumed to be zero outside $\Omega^{\rm s}$.

The cooling model

Once the heating process ends, aqua-quenching begins. This situation is modeled via the Robin boundary condition given in (16).

We put $z_{T_{\rm h}} = z(T_{\rm h})$, that is, $z_{T_{\rm h}}$ is the phase fraction distribution at the final heating instant $T_{\rm h}$ obtained from (8). In the same way, we define $\theta_{T_{\rm h}} = \theta(T_{\rm h})$. Obviously, these functions will be taken as the initial phase fraction distribution and temperature,

 ISBN 13: 978-84-613-5510-5

respectively, in the cooling model.

$$z_t = F(\theta, z) \text{ in } \Omega^{\mathrm{s}} \times (T_{\mathrm{h}}, T_{\mathrm{c}}), \tag{13}$$

$$z(T_{\mathrm{h}}) = z_{T_{\mathrm{h}}} \text{ in } \Omega^{\mathrm{s}}, \tag{14}$$

$$\theta_t - \nabla \cdot (\kappa(\theta)\nabla\theta) = \rho L z_t + G \text{ in } \Omega \times (T_{\mathrm{h}}, T_{\mathrm{c}}), \tag{15}$$

$$-\kappa(\theta)\frac{\partial\theta}{\partial n} = \beta(\theta - \theta_{\mathrm{e}}) \text{ on } \partial\Omega \times (T_{\mathrm{h}}, T_{\mathrm{c}}), \tag{16}$$

$$\theta(T_{\mathrm{h}}) = \theta_{T_{\mathrm{h}}} \text{ in } \Omega. \tag{17}$$

In (16), the constant value θ_{e} stands for the temperature of the spray water quenching the workpiece during the cooling time interval $[T_{\mathrm{h}}, T_{\mathrm{c}}]$. Also, the function β is a heat transfer coefficient and is given by

$$\beta(x,t) = \begin{cases} 0 & \text{on } \partial\Omega \cap \partial\Omega^{\mathrm{c}}, \\ \beta_0(t) & \text{on } \partial\Omega \cap \partial\Omega^{\mathrm{s}}. \end{cases}$$

where $\beta_0(t) > 0$ (usually taken to be constant).

3 The harmonic regime

We focus our attention on the heating induction-conduction process. For this reason and from now on, we will just write T instead of T_{h}.

Electromagnetic fields generated by high frequency currents are sinusoidal in time. Consequently, both the electric potential, ϕ, and the magnetic potential field, $\boldsymbol{\mathcal{A}}$, take the form ([1, 2, 12, 13]) $\mathcal{M}(x,t) = \mathrm{Re}\left[e^{i\omega t}M(x)\right]$, where F is a complex-valued function or vector field, and $\omega = 2\pi f$ is the angular frequency, f being the electric current frequency. In general, M also depends on t, but at a time scale much greater than $1/\omega$. In this way, we may introduce the complex-valued fields φ, $\boldsymbol{A}$ and $\boldsymbol{j}$ as

$$\phi = \mathrm{Re}[e^{i\omega t}\varphi(x,t)], \quad \boldsymbol{\mathcal{A}} = \mathrm{Re}[e^{i\omega t}\boldsymbol{A}(x,t)], \quad j_S = \mathrm{Re}[e^{i\omega t}\boldsymbol{j}(x)]. \tag{18}$$

As a far as the numerical simulation of a system like (2)-(12) is concerned, the introduction of the new variables φ and $\boldsymbol{A}$ is quite convenient since the time scale describing the evolution of both φ and $\boldsymbol{A}$ is much smaller than that of the temperature θ. In the case of steel heat treating, f is about 80 KHz.

When we rewrite the original system (2)-(12) in terms of the new complex-valued variables, φ and $\boldsymbol{A}$, neglecting the term $\boldsymbol{A}_t$, we obtain the so-called harmonic regime. Furthermore, in the energy equation, the expression $|\boldsymbol{\mathcal{A}}_t + \nabla\phi|^2$ is substituted by its mean value measured over a time period $[t, t+\omega]$:

$$\frac{1}{\omega}\int_t^{t+\omega} |\boldsymbol{\mathcal{A}}_t + \nabla\phi|^2 \simeq \frac{1}{2}|i\omega\boldsymbol{A} + \nabla\varphi|^2.$$

 ISBN 13: 978-84-613-5510-5

In this way, the effective Joule's heating takes the form $\frac{1}{2}\sigma(\theta)|i\omega\boldsymbol{A}+\nabla\varphi|^2$. The equations in the harmonic regime are the following.

$$\nabla\cdot(\sigma(\theta)\nabla\varphi)=0 \text{ in } \Omega_T, \tag{19}$$

$$\frac{\partial\varphi}{\partial n}=0 \text{ on } \partial\Omega\times(0,T), \tag{20}$$

$$\left[\sigma(\theta)\frac{\partial\varphi}{\partial\nu}\right]_\Gamma=\boldsymbol{j} \text{ on } \Gamma\times(0,T), \tag{21}$$

$$i\omega\sigma_0(\theta)\boldsymbol{A}+\nabla\times\left(\frac{1}{\mu}\nabla\times\boldsymbol{A}\right)-\delta\nabla(\nabla\cdot\boldsymbol{A})=-\sigma_0(\theta)\nabla\varphi \text{ in } D_T, \tag{22}$$

$$\boldsymbol{A}=0 \text{ on } \partial D\times(0,T), \tag{23}$$

$$z_t=F(\theta,z) \text{ in } \Omega^{\mathrm{s}}\times(0,T), \tag{24}$$

$$z(0)=z_0 \text{ in } \Omega^{\mathrm{s}}, \tag{25}$$

$$\theta_t-\nabla\cdot(\kappa(\theta)\nabla\theta)=\frac{1}{2}\sigma(\theta)|i\omega\boldsymbol{A}+\nabla\varphi|^2+\rho L z_t+G \text{ in } \Omega_T, \tag{26}$$

$$\frac{\partial\theta}{\partial n}=0 \text{ on } \partial\Omega\times(0,T), \tag{27}$$

$$\theta(\cdot,0)=\theta_0 \text{ in } \Omega, \tag{28}$$

Remark A similar simpler stationary model involving only the unknowns $\boldsymbol{A}$ and θ and with non-homogeneous Dirichlet boundary conditions is studied in [4].

4 An existence result

We consider the system (19)-(28) describing the heating process by conduction-induction in the harmonic regime. Besides the assumptions on data already mentioned along the Introduction, we will considered the following hypotheses.

(H.1) $\sigma,\kappa:\Omega\times\mathbb{R}\mapsto\mathbb{R}$ are Carathéodory functions and there exist some constant values $\sigma_1,\sigma_2,\kappa_1,\kappa_2\in\mathbb{R}$ such that $0<\sigma_1\le\sigma(x,s)\le\sigma_2$, $0<\kappa_1\le\kappa(x,s)\le\kappa_2$, almost everywhere $x\in\Omega$ and for all $s\in\mathbb{R}$.

(H.2) $\boldsymbol{j}\in L^2(0,T;H^{-1/2}(\Gamma))$ and $\langle\boldsymbol{j}(t),1\rangle_\Gamma=0$, almost everywhere $t\in(0,T_h)$. Here, $\langle\cdot,\cdot\rangle_\Gamma$ stands for the duality pair between $H^{-1/2}(\Gamma)$ and $H^{1/2}(\Gamma)$.

(H.3) $\mu\in L^\infty(D)$ and there exists a constant value μ_* such that $0<\mu_*\le\mu$ in D.

(H.4) $F\in L^\infty(\mathbb{R}\times\mathbb{R}^2)\cap C(\mathbb{R}\times\mathbb{R}^2)$ and there exists a constant L_F such that

$$|F(s,s_1)-F(s,s_2)|\le L_F|s_1-s_2|, \text{ for all } s\in\mathbb{R} \text{ and for all } s_1,s_2\in\mathbb{R}^2.$$

(H.5) $z_0=(z_{01},z_{02})\in L^\infty(\Omega^{\mathrm{s}})$.

(H.6) $\rho L,G\in L^1(\Omega^{\mathrm{s}}\times(0,T))$.

 ISBN 13: 978-84-613-5510-5

(H.7) $\theta_0 \in L^1(\Omega)$.

Remark In the situation described here, we are just considering the evolution of two phase fractions which correspond to austenite and martensite. Of course, we may consider a more general setting which includes other phase fractions like bainite, pearlite and ferrite or a mixing of them all (see [7]).

Remark In practice, the magnetic permeability is of the form

$$\mu(x) = \mu_1 \chi_{\Omega^s} + \mu_2 \chi_{\Omega^c} + \mu_3 \chi_{D\setminus\Omega},$$

where $\mu_i > 0$, $1 \leq i \leq 3$, are constant values such that tales que $\mu_2 < \mu_3 \ll \mu_1$.

Variational formulation

The variational formulation corresponding to the system (19)-(28) allows us to give the concept of a solution $(\varphi, \boldsymbol{A}, z, \theta)$ to this system.

We denote by $H^1(\Omega) = \{v \in L^2(\Omega) \,/\, \nabla v \in (L^2(\Omega))^N\}$, $N = 2$ or 3, the complex-valued usual Sobolev space, the derivatives of v taken in the sense of distributions. We also use the complex-valued Sobolev space $H_0^1(D) = \{v \in H^1(D) \,/\, v = 0 \text{ on } \partial D\}$. Then we put $\boldsymbol{H}_0^1(D) = (H_0^1(D))^N$. All these spaces are Hilbert spaces provided with their respective inner products.

The quotient space $H^1(\Omega)/\mathbb{C}$ is a Hilbert space provided with the inner product

$$(\dot{u}, \dot{v}) = \int_\Omega \nabla u \nabla \bar{v},$$

where u, respectively v, is any element in the class of $\dot{u}$, respectively $\dot{v}$, and $\bar{v}$ stands for the conjugate of v.

For $1 \leq p \leq \infty$, we also consider the Banach (real) space $W^{1,p}(\Omega)$ provided with their standard norm, and $(W^{1,p}(\Omega))'$ its dual (topological and algebraic) space.

If X is a Banach space, we put $L^p(X) = L^p(0,T;X)$ and $W^{1,p}(X) = W^{1,p}(0,T;X)$, that is

$$W^{1,p}(X) = \{v \in L^p(X) \,/\, v_t \in L^p(X)\},$$

the derivative v_t taken in the sense of distributions in $(0,T)$. Both, $L^p(X)$ and $W^{1,p}(X)$ are Banach spaces. Remember that

$$W^{1,p}(X) \subset C([0,T];X)$$

with continuous embedding.

 ISBN 13: 978-84-613-5510-5

Definition 1 *We say that $(\varphi, \boldsymbol{A}, z, \theta)$ is a weak solution to the system (19)-(28) if the following conditions hold*

$$\varphi \in L^2(H^1(\Omega)/\mathbb{C}), \tag{29}$$

$$\boldsymbol{A} \in L^2(\boldsymbol{H}_0^1(D)), \tag{30}$$

$$z \in W^{1,\infty}(L^\infty(\Omega^{\mathrm{s}})), \tag{31}$$

$$\theta \in L^p(W^{1,p}(\Omega)) \cap C([0,T];(W^{1,p'}(\Omega))') \text{ for all } p \in \left[1, \tfrac{N+2}{N+1}\right), \tfrac{1}{p} + \tfrac{1}{p'} = 1, \tag{32}$$

$$\theta(\cdot, 0) = \theta_0 \text{ in } \Omega, \tag{33}$$

$$\int_0^T \int_\Omega \sigma(\theta)\nabla\varphi \cdot \nabla\bar{\psi} + \int_0^T \langle \boldsymbol{j}, \bar{\psi}\rangle_\Gamma = 0, \text{ for all } \psi \in L^2(H^1(\Omega)/\mathbb{C}), \tag{34}$$

$$\begin{aligned} i\omega \int_0^T \int_\Omega \sigma(\theta)\boldsymbol{A}\cdot\bar{\boldsymbol{v}} + \int_0^T \int_D \frac{1}{\mu}\nabla\times\boldsymbol{A}\cdot\nabla\times\bar{\boldsymbol{v}} + \delta\int_0^T\int_D \nabla\cdot\boldsymbol{A}\,\nabla\cdot\bar{\boldsymbol{v}} \\ + \int_0^T\int_\Omega \sigma(\theta)\nabla\varphi\cdot\bar{\boldsymbol{v}} = 0, \text{ for all } \boldsymbol{v} \in L^2(\boldsymbol{H}_0^1(D)) \end{aligned} \tag{35}$$

$$z = z_0 + \int_0^t F(\theta, z), \text{ for all } t \in [0,T] \tag{36}$$

$$\begin{aligned} -\int_0^T\int_\Omega \theta\zeta_t + \int_0^T\int_\Omega \kappa(\theta)\nabla\theta\nabla\zeta \\ = \int_0^T\int_\Omega \left(\frac{1}{2}\sigma(\theta)|i\omega\boldsymbol{A} + \nabla\varphi|^2 + \rho L z_t + G\right)\zeta, \\ \text{for all } \zeta \in \mathcal{C}^1(\bar{\Omega}\times[0,T]) \text{ such that } \zeta(\cdot,0) = \zeta(\cdot,T) = 0 \text{ in } \Omega. \end{aligned} \tag{37}$$

Remark As long as $N \le 3$, Sobolev embedding implies that $L^1(\Omega) \subset (W^{1,q}(\Omega))'$ for all $q > 3$. On the other hand, since $p < 5/4 \le (N+2)/(N+1)$ we have $p' > 5$; in particular, $L^1(\Omega) \subset (W^{1,p'}(\Omega))'$ for all $p \in [1, 5/4)$. Consequently, according to (H.7) and the regularity $\theta \in C([0,T];(W^{1,p'}(\Omega))')$ stated in (32), the initial condition (33) makes sense at least in the space $(W^{1,p'}(\Omega))'$. Under a more restrictive assumption on the thermal conductivity κ (see (H.8) below), it can be shown that $\theta \in C([0,T];L^1(\Omega))$. Thus, the initial condition (33) also makes sense in $L^1(\Omega)$.

The main result

An existence result of a weak solution $(\varphi, \boldsymbol{A}, z, \theta)$ to the system (19)-(28) is given below. To this end, we also consider the following hypothesis on the thermal conductivity κ.

(H.8) There exist $\varepsilon_0 > 0$ and $L_0 > 0$ such that for all $\varepsilon \in (0, \varepsilon_0]$ one has

$$|\kappa(x, s_1) - \kappa(x, s_2)| \le L_0|s_1 - s_2|,$$

almost everywhere $x \in \Omega$ and for all $s_1, s_2 \in \mathbb{R}$ such that $|s_1 - s_2| < \varepsilon$.

 ISBN 13: 978-84-613-5510-5

Theorem **1** *Assume the assumptions (H.1)-(H.7). Then there exists a weak solution to the system (19)–(28) in the sense of Definition 1.*

Moreover, if the thermal conductivity κ verifies (H.8), then $\theta \in C([0,T];L^1(\Omega))$ and it also satisfies the variational formulation

$$-\int_0^T\int_\Omega \theta\zeta_t + \int_\Omega \theta(x,T)\zeta(x,T) - \int_\Omega \theta_0(x)\zeta(x,0) + \int_0^T\int_\Omega \kappa(\theta)\nabla\theta\nabla\zeta$$
$$= \int_0^T\int_\Omega \left(\frac{1}{2}\sigma(\theta)|i\omega\boldsymbol{A}+\nabla\varphi|^2 + \rho L z_t + G\right)\zeta, \text{ for all } \zeta \in \mathcal{C}^1(\bar{\Omega}\times[0,T]).$$

The proof of this result will be developed in a forthcoming paper ([8]).

5 Numerical simulation

We have carried out some numerical simulations for the approximation of the solution to the system (19)-(28). We want to describe the hardening treatment of a car steering rack during the heating-cooling process. The goal is to produce martensite along the tooth line together with a thin layer in its neighborhood inside the steel workpiece.

Figure 1 shows the open sets D, $\Omega = \Omega^{\rm s}\cup\Omega^{\rm c}\cup S$ and the inferface Γ which intervene in the setting of the problem. The inductor $\Omega^{\rm c}$ is made of copper. The workpiece contains a toothed part to be hardened by means of the heating-cooling process described above. It is made of a hypoeutectoid steel. The open set $D\setminus\bar{\Omega}$ is air. The magnetic permeability μ in (22) is then given by

$$\mu(x) = \begin{cases} \mu_0 & \text{if } x\in D\setminus\bar{\Omega},\\ 0.99995\mu_0 & \text{if } x\in\Omega^{\rm c},\\ 2.24\times10^3\mu_0 & \text{if } x\in\Omega^{\rm s},\end{cases}$$

where $\mu_0 = 4\pi\times10^{-7}$ (N/A^2) is the magnetic constant (vacuum permeability).

The martensite phase can only derive from the austenite phase. Thus we need to transform first the critical part to be hardened (the tooth line) into austenite. For our hypoeutectoid steel, austenite only exists in a temperature range close to the interval $[1050, 1670]$ (in °K). During the first stage, the workpiece is heated up by conduction and induction (Joule's heating) which renders the tooth line to the desired temperature. In order to transform the austenite into martensite, we must cool it down at a very high rate. This second stage is accomplished by spraying water over the workpiece. This latter process is called aquaquenching.

In this simulation, the final time of the heating process is $T_{\rm h} = 5.5$ seconds and the cooling process extends also for 5.5 seconds, that is $T_{\rm c} = 11$.

We have used the finite elements method for the space approximation and a Crank-Nicolson scheme for the time discretization. Figures 3 and 4 show the triangulation of D in our numerical simulations. We have used P_2-Lagrange approximation for φ, $\boldsymbol{A}$ and θ and P_1 for z.

In Figure 5 we can see the temperature distribution of the rack along the tooth line at the final stage of the heating process. The initial temperature is $\theta_0 = 300$°K.

 ISBN 13: 978-84-613-5510-5

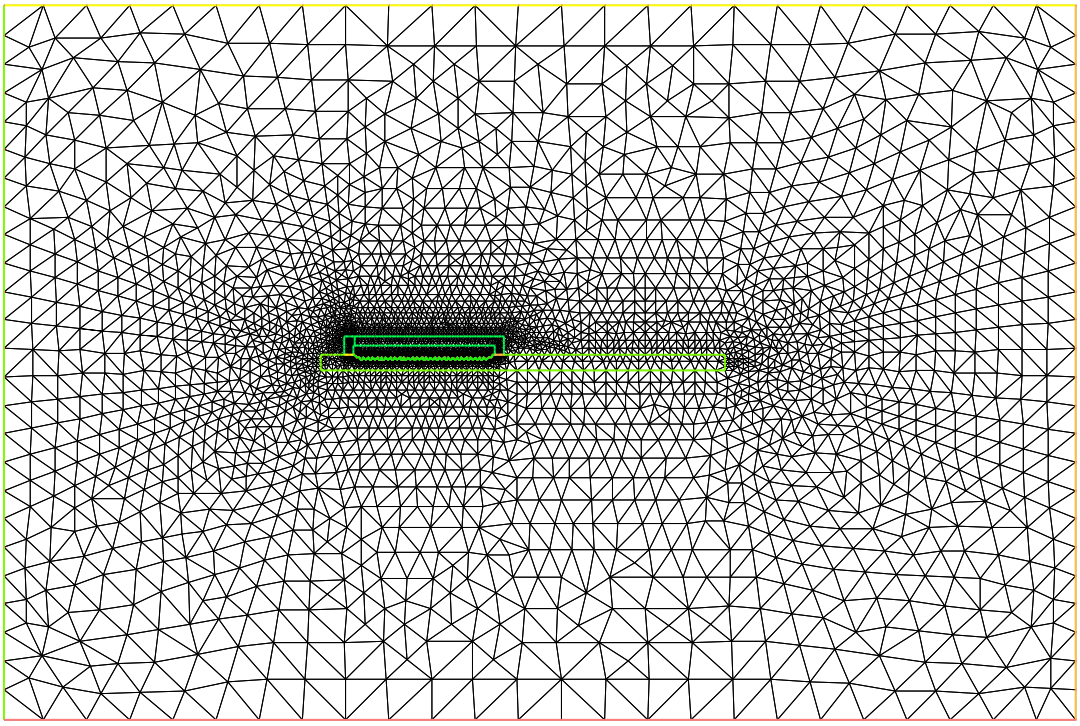


Figure 3: Domain triangulation. The triangulation of D contains 61790 triangles and 30946 vertices.

At $t = 5.5$ the heating process ends and the computed temperature shows that the temperature along the rack tooth line lies in the interval $[1050, 1670]$ (°K).

Figure 6 shows the austenization along the tooth line at the end of the heating process $T = 5.5$ seconds.

Figure 7 shows the final distribution of martensite from austenite along the rack tooth line through the cooling stage $t = 11$ seconds. We have good agreement versus the experimental results obtained in the industrial process.

Acknowledgements

This research was partially supported by Ministerio de Educación y Ciencia under grant MTM2006-04436 with the participation of FEDER, and Consejería de Educación y Ciencia de la Junta de Andalucía, research group FQM–315.

 ISBN 13: 978-84-613-5510-5

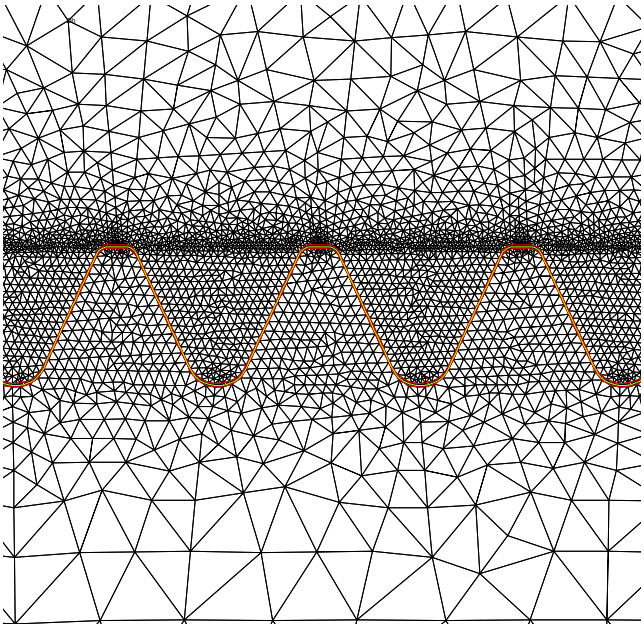


Figure 4: Domain triangulation. Elements density near three teeth.

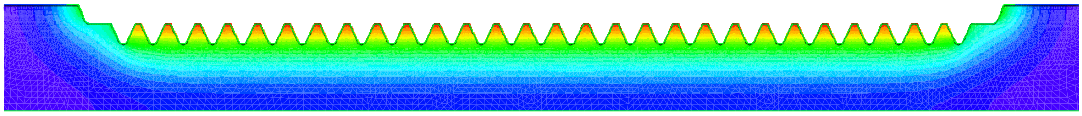


Figure 5: Temperature at the final stage of the heating process $t = 5.5$ seconds near the tooth line.

References

[1] A. Bermúdez, J. Bullón, F. Pena and P. Salgado, *A numerical method for transient simulation of metallurgical compound electrodes*, Finite Elem. Anal. Des., **39**, 283-299, 2003.

[2] A. Bermúdez, D. Gómez, M. C. Muñiz and P. Salgado, *Transient numerical simulation of a thermoelectrical problem in cylindrical induction heating furnaces*, Adv. Comput. Math., **26**, 39-62, 2007.

[3] K. Chełminski, D. Hömberg and D. Kern, *On a thermomechanical model of phase transitions in steel*, WIAS preprint, **1125**, Berlin 2007.

[4] S. Clain and R. Touzani, *A Two-dimensional Stationary Induction Heating Problem*, Mathematical Methods in the Applied Sciences, **20**, 759-766, 1997.

[5] J. M. Díaz Moreno, C. García Vázquez, M. T. González Montesinos and F. Ortegón Gallego, *Un modelo para la descripción de las transiciones*

 ISBN 13: 978-84-613-5510-5

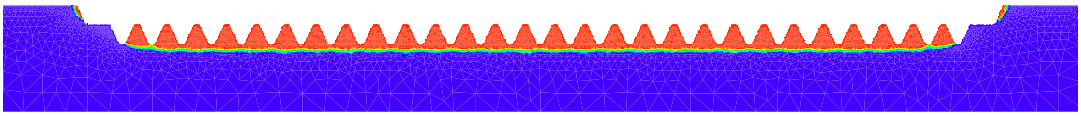


Figure 6: Heating process. Austenite at $t = 5.5$ along the rack tooth line.

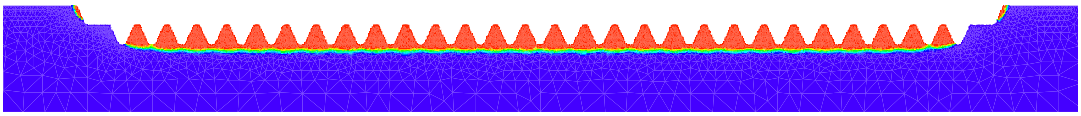


Figure 7: Cooling process. Martensite transformation at the final stage of the cooling process $t = 11$ seconds.

de fases en una barra de acero, Actas XX Congreso de Ecuaciones Diferenciales y Aplicaciones/X Congreso de Matemática Aplicada. Sevilla, 24-28 septiembre 2007

[6] J. M. Díaz Moreno, C. García Vázquez, M. T. González Montesinos and F. Ortegón Gallego, *Numerical simulation of a Induction-Conduction Model Arising in Steel Hardening model arising in steel hardening*, Lecture Notes in Engineering and Computer Science, World Congress on Engineering 2009, Volume II, July 2009, 1251–1255.

[7] J. Fuhrmann, D. Hömberg and M. Uhle, *Numerical simulation of induction hardening of steel,* COMPEL, **18**, No. 3, 482–493, 1999.

[8] M. T. González Montesinos and F. Ortegón Gallego. *To appear.*

[9] D. Hömberg, *A mathematical model for induction hardening including mechanical effects*, Nonlinear Analysis: Real World Applications, **5**, 55–90, 2004.

[10] D. Hömberg and W. Weiss, *PID control of laser surface hardening of steel*, IEEE Transactions on Control Systems Technology, **14**, No. 5, 896–904, 2006.

[11] J. B. Leblond and J. Devaux, *A new kinetic model for anisothermal metallurgical transformations in steels including effect of austenite grain size*, Acta metall., **32**, No. 1, 137–146, 1984.

[12] F. J. Pena Brage, *Contribución al modelado matemático de algunos problemas en la metalurgia del silicio*, Ph. thesis, Universidade de Santiago de Compostela, 2003.

[13] H. M. Yin, *Regularity of weak solution to Maxwell's equations and applications to microwave heating*, J. Differential Equations, **200**, 137-161, 2004.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Logic-based Functional Dependencies Programming

Manuel Enciso[1], Angel Mora[1], Pablo Cordero[1], Gabriel Aguilera[1] and I. P. de Guzmán[1]

[1] *E.T.S.I. Informática., Universidad de Málaga. Spain*

emails: enciso@lcc.uma.es, amora@ctima.uma.es, cordero@uma.es, gabri@ctima.uma.es, guzman@ctima.uma.es

Abstract

Although classical logic was studied by logicians in depth, no efficient applications were presented until the Robinson resolution rule was introduced. Thus, the resolution rule may be considered as the first step in logic programming. Besides that, in artificial intelligence and relational databases, functional dependency (FD) is a useful notion to describe data knowledge. In literature, there exists several logics to specify and manipulate FDs. Nevertheless, their inference systems are suitable to illustrate FD semantics but they may not be used as a formal base to develop automated deduction methods. In this work, we use an axiomatic system, denoted $\mathcal{S}^e_{FDS}$, which is based on new FD inference rule, named *Simplification Rule*, which plays the same role as the resolution rule. We introduce the *Deduction Theorem for Functional Dependencies* that has theoretical relevance, and is the key for solving the FD implication problem using $\mathcal{S}^e_{FDS}$. Moreover, we present the paradigm *functional dependencies programming*. This paradigm uses an *inference engine* as an automated deduction method, based on three rules of simplification. An extension of classical FD language with empty attributes ($\top$ atom) is introduced in order to specify *facts* and *goals*. The set of FDs plays the role of rules in classical logic programming languages. Finally, we apply this paradigm to solve the FD implication problem with a novel and efficient algorithm.

Key words: Logic, Implication, Functional dependencies

1 Introduction

Classical logic was conceived (and successfully used) as a formal framework suitable for specification and metatheoretical development. Thus, Logic has gone down in history as the formalism selected by mathematicians and philosophers to formally explain their theories. When computer science was born, classical logic (with its different deduction methods) was moved from a theoretical field to a practical one, where it reveals its power.

 ISBN 13: 978-84-613-5510-5

From then, logic was not considered to be only a formal tool and it became itself a subject of research.

One of the most important milestones in the history of logic was the introduction of the *resolution method*, due to Robinson [11]. He presents a new rule, called the *resolution rule*, which allows the definition of a new automated deduction method.

Before the resolution rule was introduced, *semantic tableaux* [7] were considered to be a good method to build logical inferences, but only at a theoretical level. In fact, tableaux was considered to be a method to systematize countermodel searches, but the method was not applied in practice until the 80s, when computers acquired better perfomance.

Robinson's work opens a new door, not only to the use of logic, but also to the use of computers. Up until that moment, software was developed solely with the Von Newman style in mind. Since Robinson introduced the Resolution Rule, the field of programming was enlarged and the area of *logic programming* was born. Logic programming paradigm got the support of a great number of researchers and, in a few years, it matured and was considered as an important subject in computer science.

In the relational database community, researchers focus not only on the data itself, they are also interested in the constraints that these data must fulfill. A very important role in database constraint is played by *dependencies*. As J. Paredaens et al. say in [10]: "they are constraints in the description of a database in order to ensure that the instances we might obtain are meaningful".

There exists a wide range of dependencies. Some of them were investigated in the past (Functional Dependencies, Multivalued Dependencies, Join Dependencies, Inclusion Dependencies, etc.) and others are being studied today (Nested Functional Dependencies, Generalized Data Dependencies, XML Functional Dependencies, Fuzzy Dependencies, etc.).

At the same time, logic has been used properly as a specification tool and for metatheoretical development in the area of database dependencies. Moreover, each dependency definition is usually followed by its corresponding logic. These different dependency logics provide several formal languages to specify different kinds of database constraints but none of them has been used successfully in automated deduction. The reason is that their corresponding inference systems were created to explain dependency semantics more than to design an automated deduction system.

In this work we concentrate on functional dependencies (FD), the most popular database dependencies. They add semantics to a database schema and are useful for studying various problems such as database design, query optimization and how dependencies are carried out a view. They were introduced by E.F. Codd [1] in 1970. FDs may be viewed as a relationship among some attributes of a table. Thus, if the FD $A_1, \ldots, A_n \rightarrow B_1, \ldots, B_n$ holds in a database $\mathcal{D}$, then two tuples of $\mathcal{D}$ that agree on $A_1, \ldots, A_n$ agree on $B_1, \ldots, B_n$.

There exist several equivalent FD Logics [3, 6, 8, 10, 13] but all their inference

[1] E.F. Codd died in April 2003. We are in debt to him for his revolutionary ideas about data storage and management and we particularly appreciate his tireless fight at the beginning of the 70s', when academic and business organizations had no faith in his Relational Model.

 ISBN 13: 978-84-613-5510-5

systems are strongly based on the transitivity paradigm. These characteristics avoid the construction of efficient deduction methods directly based on these inference systems and the most successful approaches come from indirect ways (graph theory, matrix operators, etc).

As semantics tableaux in the case of classical logic, all these inference systems allow us to guide the search for new FD inferred from a given set of FDs, but they do not allow us to search for new dependencies automatically. In the same role that the Robinson rule plays in the application of classical logic, we will use the Simplication Rule presented in [4] to make FD logic become a useful tool for computer science.

The Simplification rule was presented in [4] and in [9] we use it to design a preprocessing transformation which efficiently reduces database redundancy.

Unfortunately, this transformation of sets of FDs is not complete, and it cannot be used to solve the implication problem: *to answer the question if an FD can be deduced from a given set of FDs.*

In this work, we illustrate how the *Simplification Rule* can be considered to be the triggering event at the beginning of *Functional Dependencies Programming.* Thus, we extend the language of FD Logic to allow empty left hand side formulae and we use this new well formed formulae as *goals* to be satisfied in a given FD theory. The Simplification rule is used to build a novel Simplification algorithm directly based on the inference system. This work opens the door to the management of FD constraints in relational databases in an efficient and intelligent way.

This work is organized as follows: In section 2 we show how previous FD logics reason about FDs and the limitations of these logics to manipulate automatically a set of FDs. We introduce the problem that we solve in this paper, named "*the FD implication problem*". Section 3 shows the novel Simplification rules and introduces the FD logic with Simplifications. In section 4 we propose a new automated deduction method to solve the FD implication problem and finally we establish several conclusions and future works in section 5.

2 Reasoning about FDs

In literature, there exists a set of equivalent FD logics [3, 6, 8, 10, 13]. These classical FD logics may be considered as formal tools to formally explain how to deduce a FD from a given set of FDs. With no loss of generality, we select FD Paredaens Logic [10] to illustrate how these inference systems work:

Definition 1 (The $\mathcal{L}_{FD}$ language) *Let Ω be an infinite numerable set of atoms and let $\mapsto$ be a binary connective, we define the language $\mathcal{L}_{FD} = \{X \mapsto Y \mid X, Y \in 2^{\Omega}$ and $X \neq \varnothing\}$.*

Notation 1 *Let X, Y be a set of atoms. In the following, XY denotes the union $X \cup Y$; $X \subseteq Y$ denotes the set inclusion relation; $Y - X$ denotes the difference (elements in Y that are not in X) and $\top$ denotes the empty set.*

 ISBN 13: 978-84-613-5510-5

Definition 2 (The $\mathbf{L}_{Par}$ logic) $\mathbf{L}_{Par}$ *is the logic given by the pair* $(\mathcal{L}_{FD}, \mathcal{S}_{Par})$ *where* $\mathcal{S}_{Par}$ *is an axiomatic system with one axiom scheme and two inference rules:*

$\lfloor Ax_{Par} \rfloor$: $\vdash_{\mathcal{S}_{Par}} X \mapsto Y$ *if* $Y \subseteq X$ **Axiom**

$\lfloor Trans \rfloor$ $X \mapsto Y \ Y \mapsto Z \vdash_{\mathcal{S}_{Par}} X \mapsto Z$ **Transitivity Rule**

$\lfloor Augm \rfloor$ $X \mapsto Y \ \vdash_{\mathcal{S}_{Par}} X \mapsto XY$ **Augmentation Rule**

In $\mathcal{S}_{Par}$ we have the following derived rules (these rules appear in [10] together with other derived rules):

$\lfloor Comp \rfloor$ $X \mapsto Y, \ W \mapsto Z \vdash_{\mathcal{S}_{Par}} XW \mapsto YZ$ **Composition Rule**

$\lfloor Frag \rfloor$ $X \mapsto YZ \vdash_{\mathcal{S}_{Par}} X \mapsto Y$ **Fragmentation Rule**

$\lfloor gAugm \rfloor$ $X \mapsto Y \vdash_{\mathcal{S}_{Par}} U \mapsto V$, where $X \subseteq U$ and $V \subseteq XY$

Generalized Augmentation Rule

$\lfloor gTrans \rfloor$ $X \mapsto Y, Z \mapsto U \vdash_{\mathcal{S}_{Par}} V \mapsto W$, where $Z \subseteq XY$; $X \subseteq V$;$W \subseteq UV$

Generalized Transitivity Rule

Unfortunately, $\mathcal{S}_{Par}$ and all the other classical FD axiomatic systems are not suitable tools to develop automated deduction techniques, because all of them are generated by Armstrong Axioms [2], a set of propositions introduced in 1974 to explain FD semantics. This is a well-known problem in other deduction methods, like *tableaux*-like methods, whose rules are closed to connective semantics.

The main problem concerning FD deduction is the implication problem, which can be enunciated as follows:

Let Γ *to be a set of FDs and* γ *a FD. Is it possible to affirm that* $\Gamma \vdash \gamma$*?*

In the following example we apply $\mathcal{S}_{Par}$ to solve an implication problem.

Exmaple 1 *Let* Γ *be the following set of FDs*

$$\{ad \mapsto c, b \mapsto eh, be \mapsto c, bc \mapsto d, c \mapsto a, cd \mapsto b, ce \mapsto af, cf \mapsto bdh\}$$

We try to prove that $\Gamma \vdash bd \mapsto ah$*. We apply* $\mathcal{S}_{Par}$ *to obtain the following sequence of equivalent sets of FDs.*

$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h\}$, *using* $\lfloor Frag \rfloor b \mapsto eh \vdash_{\mathcal{S}_{Par}} b \mapsto e, b \mapsto h$
$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h, bd \mapsto h\}$, *using* $\lfloor gAugm \rfloor : b \mapsto h \vdash_{\mathcal{S}_{Par}} bd \mapsto h$
$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h, bd \mapsto h, b \mapsto c\}$, *using* $\lfloor gTrans \rfloor : b \mapsto e, be \mapsto c \vdash_{\mathcal{S}_{Par}} b \mapsto c$
$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h, bd \mapsto h, b \mapsto c, bd \mapsto c\}$, *using* $\lfloor gAugm \rfloor : b \mapsto c \vdash_{\mathcal{S}_{Par}} bd \mapsto c$
$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h, bd \mapsto h, b \mapsto c, bd \mapsto c, bd \mapsto a\}$, *using* $\lfloor Trans \rfloor : bd \mapsto c, c \mapsto a \vdash_{\mathcal{S}_{Par}} bd \mapsto a$
$\Gamma \bigcup$ $\{b \mapsto e, b \mapsto h, bd \mapsto h, b \mapsto c, bd \mapsto c, bd \mapsto a, bd \mapsto ah\}$, *using* $\lfloor Comp \rfloor : bd \mapsto a, bd \mapsto h \vdash_{\mathcal{S}_{Par}} bd \mapsto ah$

So, we have that $bd \mapsto ah$ *is deduced from* Γ.

 ISBN 13: 978-84-613-5510-5

How can $X \mapsto Y$ be deduced from Γ? The classical methods consist of adding derived FDs to Γ in each step until $X \mapsto Y$ is obtained. This method has two disadvantages:

- The rules are not applied in a systematic way, so it is very difficult to select the rule that will be applied in each step. This is not an inference system suitable for automation [2].

- An additional problem arises when the target FD is not obtained from Γ. In this case, two undistinguishable causes exist: the FD can not be obtained from Γ or the selection of the rules has not been done properly. The theoretical approaches of testing every applicable rule in each step is not a real solution, since it is an exponential method that consumes a huge amount of time.

The use of this method in computers is not viable even with a set of FDs whose size is not very great [1].

We are looking for an efficient method to solve the implication problem. Instead of that, in literature a closure operator for attributes is used. Thus, if we have to prove if $X \mapsto Y$ is a consequence of Γ, we compute X^+ [3] and we test if Y is a subset of X^+. In literature there are several algorithms to compute the closure of a set of attributes in linear time (see [3, 5] for further details). These works ensure that the implication problem can be solved in polynomial time.

3 A novel rule in a new logic: The Simplification rule and $\mathbf{SL}_{FD}$

In [4] we formally introduce a new database redundancy notion and, for the first time, lattice theory is used as a formal framework for functional dependencies. In the cited work, we also present a new logic $\mathbf{SL}_{FD}$, that incorporates two novel Simplification rules [4], which removes redundancy from a given set of FDs.

Definition 3 *The* $\mathbf{SL}_{FD}$ *logic is the pair* $(\mathcal{L}_{FD}, \mathcal{S}_{FDS})$ *where* $\mathcal{L}_{FD}$ *is the language shown in Definition 1 and the axiomatic system* $\mathcal{S}_{FDS}$ *has one axiom scheme:*

Ax_{FDS} : $\vdash_{\mathcal{S}_{FDS}} X \mapsto Y$, *where* $Y \subseteq X \neq \varnothing$.

And the inference rules are the following:

Fragmentation rule: $\lfloor Frag \rfloor$: $X \mapsto Y \vdash_{\mathcal{S}_{FDS}} X \mapsto Y'$, *where* $Y' \subseteq Y$

Composition rule: $\lfloor Comp \rfloor$: $X \mapsto Y$, $U \mapsto V \vdash_{\mathcal{S}_{FDS}} XU \mapsto YV$

Simplification rule: $\lfloor Subst \rfloor$: $X \mapsto Y$, $U \mapsto V \vdash_{\mathcal{S}_{FDS}} (U\text{-}Y) \mapsto (V\text{-}Y)$, *where* $X \subseteq U$ *and* $X \cap Y = \varnothing$

[2] A human must have a high level of expertise to find the sequence of rules in a reasonable period of time. Moreover, if the reader tries to solve the above example, the derivation could be different from the one presented here.

[3] The closure of X in Γ.

[4] The rules that we introduce may be considered as transformations of *equivalence* [9].

 ISBN 13: 978-84-613-5510-5

In [9] we proved that $\mathbf{SL}_{FD}$ axiomatic system is equivalent to other well known FD axiomatic systems [3, 6, 8, 10] and thus, all Paredaens derived rules are derived rules in $\mathbf{SL}_{FD}$.

Besides that, in [4] we introduce a new derived rule:

Right Simplification rule

$\lfloor rSust \rfloor$: $\quad X \mapsto Y, U \mapsto V \vdash U \mapsto (V\text{-}Y), \quad$ if $X \subseteq UV, X \cap Y = \varnothing$

The definition of $\mathbf{SL}_{FD}$ has made it possible that, for the first time, interesting problems in database area can be solved using logic-based automated deduction methods.

The first step in this direction was the use of Simplification rules included in $\mathbf{SL}_{FD}$ [4], as a new tool for *reasoning* about FDs. In [9] we used Prolog for implementing a new pre-processing transformation that prunes a set of FDs and in [1] we used Maude for comparing FD logic by Paredaens and $\mathbf{SL}_{FD}$. Mainly, in these previous works the rules of $\mathbf{SL}_{FD}$ are applied in order to reduce the size of a set of FDS progressively.

4 A new automated deduction method

In this section, we extend the language of FD Logic to allow empty left hand side formulae and we use this new well formed formulae as *goals* to be satisfied in a given FD theory. The Simplification rule is used to build a novel Simplification method to solve the implication problem directly based on the inference system.

4.1 An extension of $\mathbf{SL}_{FD}$: The $\mathbf{SL}^{e}_{FD}$ logic

We remark that $\mathcal{L}_{FD}$ includes the following formula schema: $X \mapsto \top$. In [12] the author considers this FD schema to solve some problems, concerning the management of FDs in a given algorithm but $X \mapsto \top$ does not appear in any FD logic in literature. In our logic, the symbol $\top$ is a central element in $\mathbf{SL}_{FD}$ and it guides the utilization of the logic to solve several problems.

However, $\mathcal{L}_{FD}$ does not allow the use of the FD $\top \mapsto X$. We extend $\mathbf{SL}_{FD}$ to represent and manipulate this FD.

Definition 4 (The $\mathcal{L}^{e}_{FD}$ language) *Let Ω be an infinite numerable set of atoms and let $\mapsto$ be a binary connective, we define the language $\mathcal{L}^{e}_{FD} = \{X \mapsto Y \mid X, Y \in 2^{\Omega}\}$ That is, $\mathcal{L}^{e}_{FD} = \mathcal{L}_{FD} \cup \{\top \mapsto V \mid V \in 2^{\Omega}\}$.*

Definition 5 (The $\mathbf{SL}^{e}_{FD}$ logic) *The $\mathbf{SL}^{e}_{FD}$ logic is the pair $(\mathcal{L}^{e}_{FD}, \mathcal{S}^{e}_{FDS})$ where $\mathcal{L}^{e}_{FD}$ is the language shown in Definition 4 and the axiomatic system $\mathcal{S}^{e}_{FDS}$ has one axiom scheme: $Ax^{e}_{FDS}: \ \vdash_{\mathcal{S}^{e}_{FDS}} X \mapsto Y$, where $Y \subseteq X$. And the inference rules are $\lfloor Frag \rfloor$, $\lfloor Comp \rfloor$ and $\lfloor Subst \rfloor$.*

Obviously, from the above definition we directly obtain that: for all $\Gamma \subseteq \mathcal{L}_{FD}$ and all $X \mapsto Y \in \mathcal{L}_{FD}$, if $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$ then $\Gamma \vdash_{\mathcal{S}^{e}_{FDS}} X \mapsto Y$.

Now we are interested in the benefits of the extension that we have just introduced.

 ISBN 13: 978-84-613-5510-5

Lemma 1 *Let* $\Gamma \subseteq \mathcal{L}^e_{FD}$*, there exists* $\Gamma' \subseteq \mathcal{L}_{FD}$ *and* $X \in 2^\Omega$ *such that* $\Gamma \equiv_{\mathcal{S}^e_{FDS}} \Gamma' \cup \{\top \mapsto X\}$ *i.e,* $\Gamma \vdash_{\mathcal{S}^e_{FDS}} \Gamma' \cup \{\top \mapsto X\}$ *and* $\Gamma' \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} \Gamma$

Proof 1 *It is an immediate consequence of the composition and fragmentation rules.*

The following theorem is the key to solving the FD implication problem using the $\mathbf{SL}_{FD}$ logic.

Theorem 2 (Deduction Theorem for Functional Dependencies) *Given* $\Gamma \subseteq \mathcal{L}_{FD}$*, we have the following equivalence:*

1. *For all* $X, U, V \in 2^\Omega$, $\Gamma \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} U \mapsto V$ *if and only if* $\Gamma \vdash_{\mathcal{S}_{FDS}} UX \mapsto V$*. And, in particular:*

2. *For all* $X, Y \in 2^\Omega$*. the following equivalence is stated:* $\Gamma \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} \top \mapsto Y$ *if and only if* $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$

Proof 2 *First, if* $\Gamma \vdash_{\mathcal{S}_{FDS}} UX \mapsto V$ *then we have that* $\Gamma \vdash_{\mathcal{S}^e_{FDS}} UX \mapsto V$ *and the following sequence proves that* $\Gamma \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} U \mapsto V$*.*

1. $\top \mapsto X$ ..... *by hypothesis*
2. $UX \mapsto V$ ..... *by hypothesis*
3. $U - X \mapsto V - X$ ..... *by 1., 2. and* $\lfloor Subst \rfloor$
4. $U - X \mapsto VX$ ..... *by 1., 3. and* $\lfloor Comp \rfloor$
5. $U - X \mapsto V$ ..... *by 4. and* $\lfloor Frag \rfloor$
6. $U \mapsto \top$ ..... *by* Ax^e_{FDS}
7. $U \mapsto V$ ..... *by 5., 6. and* $\lfloor Comp \rfloor$

Conversely, we prove that, if $\Gamma \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} U \mapsto V$*, then* $\Gamma \vdash_{\mathcal{S}_{FDS}} UX \mapsto V$*. Let us consider the following sets:*

- Σ *is the inductive set freely generated by* Γ *and the axioms set in* $\mathcal{S}_{FDS}$ *via the constructors given the inference rules in* $\mathcal{S}_{FDS}$*. That is,*

$$\Sigma = \{Y \mapsto Z \mid \Gamma \vdash_{\mathcal{S}_{FDS}} Y \mapsto Z\}$$

- Σ^e_X *is the inductive set freely generated by* $\Gamma \cup \{\top \mapsto X\}$ *and the axioms set in* $\mathcal{S}^e_{FDS}$ *via the constructors given the inference rules in* $\mathcal{S}^e_{FDS}$*. That is,*

$$\Sigma^e_X = \{Y \mapsto Z \mid \Gamma \cup \{\top \mapsto X\} \vdash_{\mathcal{S}^e_{FDS}} Y \mapsto Z\}$$

 ISBN 13: 978-84-613-5510-5

We prove, by induction, that

$$U{\mapsto}V \in \Sigma^e_X \qquad \textit{implies that} \qquad UX{\mapsto}V \in \Sigma$$

1. *If* $U{\mapsto}V \in \Gamma$, *then* $U{\mapsto}V \in \Sigma$ *and, since* Σ *is closed for the inference rules in* $\mathcal{S}_{FDS}$ *and* $UX{\mapsto}V$ *is obtained from* $U{\mapsto}V$ *by* $\lfloor gAugm \rfloor$, *we have that* $UX{\mapsto}V \in \Sigma$.

2. *From* $\top{\mapsto}X$ *we obtain* $X{\mapsto}X$ *that belongs to* Σ *because it is an axiom in* $\mathcal{S}_{FDS}$.

3. *If* $U{\mapsto}V$ *is an axiom in* $\mathcal{S}^e_{FDS}$, *then* $UX{\mapsto}V \in \Sigma$ *because it is an axiom in* $\mathcal{S}_{FDS}$.

4. *If* $U{\mapsto}V \in \Sigma^e_X$ *because it is obtained applying* $\lfloor Frag \rfloor$, *then there exist* $V' \supseteq V$ *such that* $U{\mapsto}V' \in \Sigma^e_X$ *and, by* $\lfloor Frag \rfloor$, $U{\mapsto}V \in \Sigma^e_X$.

   By induction hypothesis, we have that $UX{\mapsto}V' \in \Sigma$ *and applying* $\lfloor Frag \rfloor$ *we have that* $UX{\mapsto}V \in \Sigma$.

5. *If* $U{\mapsto}V \in \Sigma^e_X$ *because it is obtained applying* $\lfloor Comp \rfloor$, *then there exists* $U_1{\mapsto}V_1$, $U_2{\mapsto}V_2 \in \Sigma^e_X$ *such that* $U_1U_2 = U$ *and* $V_1V_2 = V$ *and, by induction hypothesis, we have that* $U_1X{\mapsto}V_1$, $U_1X{\mapsto}V_2 \in \Sigma$. *Finally, we obtain that* $UX{\mapsto}V \in \Sigma$ *applying* $\lfloor Comp \rfloor$.

6. *If* $U{\mapsto}V \in \Sigma^e_X$ *because it is obtained applying* $\lfloor Subst \rfloor$ *then there exists* $U_1{\mapsto}V_1$, $U_2{\mapsto}V_2 \in \Sigma^e_X$ *such that* $U_1 \subseteq U_2$, $U_2 - V_1 = U$ *and* $V_2 - V_1 = V$. *Moreover, by induction hypothesis, we have that* $U_1X{\mapsto}V_1, U_2X{\mapsto}V_2 \in \Sigma$. *Finally, the following sequence proves that* $UX{\mapsto}V \in \Sigma$.

   1. $U_1X{\mapsto}V_1$ ..... *by hypothesis*
   2. $U_2X{\mapsto}V_2$ ..... *by hypothesis*
   3. $U_1X{\mapsto}V_1 - X$ ..... *by 1. and* $\lfloor Frag \rfloor$
   4. $U_2X - (V_1 - X){\mapsto}V_2 - (V_1 - X)$ ..... *by 3., 2. and* $\lfloor Subst \rfloor$
   5. $U_2X - (V_1 - X){\mapsto}V$ ..... *by 4. and* $\lfloor Frag \rfloor$[5]
   6. $UX{\mapsto}\top$ ..... *by* Ax^e_{FDS}
   7. $UX{\mapsto}V$ ..... *by 5., 6. and* $\lfloor Comp \rfloor$[6]

In the next section the above theorem is used for designing a new methodology able to manage FDs as a logic programming language.

4.2 Rules of simplification for solving the FD implication problem

In this section, a novel technique for applying in a systematic way the system $\mathbf{SL}^e_{FD}$is introduced. With this aim, three rewriting rules of simplification are defined using the symbol $\rightsquigarrow$ where $\Gamma \rightsquigarrow \Gamma'$ means that all the elements in Γ must be replaced by all the elements in Γ'.

[5]Note that $V = V_2 - V_1 \subseteq V_2 - (V_1 - X)$
[6]Note that $U_2X - (V_1 - X) \subseteq U_2X \subseteq UX$

 ISBN 13: 978-84-613-5510-5

Definition 6 *Given* $X, U, V \in 2^{\Omega}$.

SC *Simplification: If* $U \subseteq X$ *then* $\{\top \mapsto X, U \mapsto V\} \rightsquigarrow \{\top \mapsto XV\}$

SA *Simplification: If* $V \subseteq X$ *then* $\{\top \mapsto X, U \mapsto V\} \rightsquigarrow \{\top \mapsto X\}$

S *Simplification:* $\{\top \mapsto X, U \mapsto V\} \rightsquigarrow \{\top \mapsto X, U - X \mapsto V - X\}$

Lemma 2 *Let* Γ *and* Γ' *be two sets of FDs. If* Γ' *is obtained from* Γ *applying the rewriting rules of simplification introduced in definition 6 then* $\Gamma \equiv_{\mathcal{S}^e_{FDS}} \Gamma'$

Proof 3

SC *Simplification:* $\{\top \mapsto X, U \mapsto V\} \overset{1}{\equiv}_{\mathcal{S}^e_{FDS}} \{\top \mapsto X, U - X \mapsto V - X\} \overset{2}{\equiv}_{\mathcal{S}^e_{FDS}} \{\top \mapsto XV\}$ Simplification *rule is applied in 1 and, since* $U \subseteq X$, Composition *rule is applied in 2.*

SA *Simplification:* $\{\top \mapsto X, U \mapsto V\} \overset{3}{\equiv}_{\mathcal{S}^e_{FDS}} \{\top \mapsto X, U - X \mapsto V - X\} \overset{4}{\equiv}_{\mathcal{S}^e_{FDS}} \{\top \mapsto X\}$ Simplification *rule is applied in 3 and, since* $V \subseteq X$, *the* Axiom *is applied in 4.*

S *Simplification:* $\{\top \mapsto X, U \mapsto V\} \overset{5}{\equiv}_{\mathcal{S}^e_{FDS}} \{\top \mapsto X, U - X \mapsto V - X\}$ Simplification *rule is applied in 5.*

Theorem 3 *Given* $\Gamma \subseteq \mathcal{L}_{FD}$ *and* $X \mapsto Y \in \mathcal{L}_{FD}$. *If* Γ' *is obtained from* $\Gamma \cup \{\top \mapsto X\}$ *applying the rewriting rules of simplification introduced in definition 6 while these rules can be applied then there exists a unique* $\top \mapsto Z \in \Gamma'$ *with* $X \subseteq Z$ *and*

$$\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y \quad \text{if and only if} \quad Y \subseteq Z$$

Proof 4 *First, there exist* $\top \mapsto Z \in \Gamma'$ *with* $X \subseteq Z$ *because we apply the rewriting rules to* $\Gamma \cup \{\top \mapsto X\}$ *and, when these rules modify* X, X *increases.*

The uniqueness of $\top \mapsto Z$ *is ensured by* **SC** *rule.*

$Y \subseteq Z$ *implies* $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$ *is obtained using Theorem 2, Lemma 2 and the fragmentation rule.*

Conversely, the following steps prove that $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$ *implies that* $Y \subseteq Z$. *Let* Γ'' *be* $\Gamma' - \{\top \mapsto Z\}$*:*

1. *Since* $\top \mapsto Z$ *is unique,* $\Gamma'' \subseteq \mathcal{L}_{FD}$.
2. *If* $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$ *then Theorem 2 ensures that* $\Gamma' \vdash_{\mathcal{S}^e_{FDS}} \top \mapsto Y$ *and, from* $\Gamma' = \{\top \mapsto Z\} \cup \Gamma''$, *(1) and Theorem 2,* $\Gamma'' \vdash_{\mathcal{S}_{FDS}} Z \mapsto Y$ *is obtained.*
3. *If* $U \mapsto V \in \Gamma''$ *then* $U \cap Z = \varnothing$ *and* $V \cap Z = \varnothing$, *since otherwise* **S** *rule of simplification could be applied.*
4. *If* $\Gamma'' \vdash_{\mathcal{S}_{FDS}} Z \mapsto Y$ *then* $Y \subseteq Z$ *because, due to (3),* $Z \mapsto Y$ *must be an axiom.*

 ISBN 13: 978-84-613-5510-5

$\top \mapsto X$	*Simp.Rule*	$ad \mapsto c$	$b \mapsto e$	$be \mapsto cg$	$bc \mapsto g$	$c \mapsto a$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto bd}$	S	$\underline{ad \mapsto c}$	$b \mapsto e$	$be \mapsto cg$	$bc \mapsto g$	$c \mapsto a$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto bd}$	SC	$a \mapsto c$	$\underline{b \mapsto e}$	$be \mapsto cg$	$bc \mapsto g$	$c \mapsto a$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto bde}$	SC	$a \mapsto c$		$\underline{be \mapsto cg}$	$bc \mapsto g$	$c \mapsto a$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto bcdeg}$	SA	$a \mapsto c$			$\underline{bc \mapsto g}$	$c \mapsto a$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto bcdeg}$	SC	$a \mapsto c$				$\underline{c \mapsto a}$	$cd \mapsto b$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto abcdeg}$	SA	$a \mapsto c$					$\underline{cd \mapsto b}$	$cf \mapsto bh$	$cg \mapsto af$
$\underline{\top \mapsto abcdeg}$	S	$a \mapsto c$						$\underline{cf \mapsto bh}$	$cg \mapsto af$
$\underline{\top \mapsto abcdeg}$	SC	$a \mapsto c$						$f \mapsto h$	$\underline{cg \mapsto af}$
$\underline{\top \mapsto abcdefg}$	SA	$\underline{a \mapsto c}$						$f \mapsto h$	
$\underline{\top \mapsto abcdefg}$	SC							$\underline{f \mapsto h}$	
$\underline{\top \mapsto abcdefgh}$									

Figure 1: Table of the example 1

The above theorem states the method to determinate if $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$. The solution arose from adding the goal $\top \mapsto X$ to Γ, rendering an initial Γ'. Then, rewriting rules of simplification are applied to Γ' obtaining $\{\top \mapsto Z\} \cup \Gamma''$. Finally, $\Gamma \vdash_{\mathcal{S}_{FDS}} X \mapsto Y$ if and only if $Y \subseteq Z$.

Below, we solve the implication problem presented in Example 1 using our new methodology:

Exmaple 2 *Let* $\Gamma = \{ad \mapsto c, b \mapsto eh, be \mapsto c, bc \mapsto d, c \mapsto a, cd \mapsto b, ce \mapsto af, cf \mapsto bdh\}$ *be this set of FDs.*

In order to know whether $\Gamma \vdash bd \mapsto ah$*, firstly we initialize* $\Gamma' = \Gamma \cup \{\top \mapsto bd\}$ *rendering:* $\Gamma' = \{\top \mapsto bd, ad \mapsto c, b \mapsto eh, be \mapsto c, bc \mapsto d, c \mapsto a, cd \mapsto b, ce \mapsto af, cf \mapsto bdh\}$

The table in figure 1 shows step by step how the rewriting rules of simplification are applied. Note that the underscore points the FD that is being reduced. The second column shows the applied rule:

Since $ah \subseteq abcdefgh$ and by Theorem 3, the following deduction is obtained: $\Gamma \models bd \mapsto ah$

A novel algorithm for solving the implication problem using rules of simplification defined below is shown in figure *2*. The algorithm simply adds $\top \mapsto X$ and, in an exhaustive way applies the rules of simplification based on the theoretical study (Theorem 2).

Since every step adds at least one attribute, in the worst case, the "Closure" loop is repeated at most $\mid \mathcal{A} \mid$ times. The "Simplify" loop is repeated at most $\mid \Gamma \mid$ times. Consequently, the complexity of the algorithm is $O(\mid \mathcal{A} \mid\mid \Gamma \mid)$. We emphasize the following characteristics of the algorithm :

- The algorithm has the same complexity as the previous algorithms [5, 10] cited in literature, namely linear with regard to the input.

 ISBN 13: 978-84-613-5510-5

$$\texttt{Implies?}(\Gamma, X \to Y) = \begin{cases} Yes, & \text{if } Y \subseteq \texttt{Closure}(X, \texttt{nil}, \Gamma, 1); \\ No, & \text{otherwise.} \end{cases}$$

$$\texttt{Closure}(X, \Gamma_1, \Gamma_2, b) = \begin{cases} X, & \text{if } \Gamma_2 = \texttt{nil or } b = 0; \\ \texttt{Closure}(\texttt{Simplify}(X, \Gamma_1, \Gamma_2, 0)), & \text{otherwise.} \end{cases}$$

$$\texttt{Simplify}(X, \Gamma_1, \texttt{nil}, b) = (X, \texttt{nil}, \Gamma_1, b)$$

$$\texttt{Simplify}(X, \Gamma_1, U \to V :: \Gamma_2, b) = \\ = \begin{cases} \texttt{Simplify}(X, \Gamma_1, \Gamma_2, b), & \text{if } V \subseteq X; \\ \texttt{Simplify}(XV, \Gamma_1, \Gamma_2, 1), & \text{if } U \subseteq X \text{ and } V \not\subseteq X; \\ \texttt{Simplify}(X, U\text{-}X \to V\text{-}X :: \Gamma_1, \Gamma_2, b), & \text{otherwise.} \end{cases}$$

Figure 2: Algorithm to solve the implication problem

- Contrary to these previous algorithms, our algorithm has a solid base, since it uses the $\mathbf{SL}_{FD}$ logic . Consequently proofs and explanations are given automatically by the algorithm applying directly the logic $\mathbf{SL}_{FD}$. Namely, the trace shown in column *Simp.Rule* reflects the rules of $\mathbf{SL}_{FD}$ logic that must be applied to prove the implication and the order in which the rules need to be applied.

5 Conclusions

As the Resolution Rule is considered to be the first step of Logic Programming, Simplification Rules are the key to open the door to a new area: functional dependencies programming . We have illustrated the difficulties of directly using other previous FD logics to face up to the implication problem.

As in a logic programming language we present an engine inference for solving the implication problem, and how rules, facts and goals are established. We propose a novel paradigm: *the functional dependencies programming*. This paradigm uses three rules of simplification based on theorem 2 and an inference engine. The set of FDs in the $\mathcal{L}_{FD}$ language plays the role of rules in a usual logic programming language as Prolog. We have defined $\mathbf{SL}^e_{FD}$ logic as extension of $\mathbf{SL}_{FD}$ logic in order to specify *goals* in $\mathcal{L}^e_{FD}$ language. The fact X (a set of attributes) is specified adding $\top \mapsto X$ to the set of FDs. And the goal is another set of attributes Y. Finally, the inference engine, based on the theoretical study, is applied automatically to the extended set of FDs and $\top \mapsto Z$ is obtained. The goal Y is achieved if $Y \subseteq Z$.

None of the classical FD logics can solve the implication problem efficiently without using indirect methods. The algorithm we have proposed in this paper has the same complexity as typical indirect methods but using directly a novel logic. Thus, we can reason and we are ready to offer explanations. So, this new algorithm is more appropriate to be used in an artificial intelligence environment.

 ISBN 13: 978-84-613-5510-5

References

[1] Gabriel Aguilera, Pablo Cordero, Manuel Enciso, Angel Mora, and Inmaculada P. de Guzmn. A non-explosive treatment of functional dependencies using rewriting logic. *XVII Brazilian Symposium on Artificial Intelligence - SBIA'04. In Lecture Notes in Artificial Intelligence, Springer-Verlag*, 2004.

[2] William W. Armstrong. Dependency structures of data base relationships. *Proc. IFIP Congress. North Holland, Amsterdam*, pages 580–583, 1974.

[3] Paolo Atzeni and Valeria De Antonellis. Relational Database Theory. *The Benjamin/Cummings Publishing Company Inc.*, 1993.

[4] Pablo Cordero, Manuel Enciso, Inmaculada P. de Guzmn, and Angel Mora. Slfd logic: Elimination of data redundancy in knowledge representation. *Lecture Notes in Artificial Intelligence 2527, Springer-Verlag*, pages 141–150, 2002.

[5] Jim Diederich and Jack Milton. New methods and fast algorithms for database normalization. *ACM Transactions on Database Systems*, 13 (3):339–365, 1988.

[6] Ronald Fagin. Functional dependencies in a relational database and propositional logic. *IBM. Journal of research and development*, 21 (6):534–544, 1977.

[7] G. Gentzen. Investigation into logical deduction. *M.S. Zabo, ed. The Collected Papers of Gerhard Gentzen. North Holland*, 1935.

[8] Toshihide Ibaraki, Alexander Kogan, and Kazuhisa Makino. Functional dependencies in horn theories. *Artificial Intelligence*, 108 1-2:1–30, 1999.

[9] Angel Mora, Manuel Enciso, Pablo Cordero, and Inmaculada P. de Guzmn. The functional dependence implication problem: optimality and minimality. an efficient preprocessing transformation based on the substitution paradigm. *Lecture Notes in Artificial Intelligence, Springer-Verlag*, 3040, 2004.

[10] Jan Paredaens, Paul De Bra, Marc Gyssens, and Dirk Van Van Gucht. The structure of the relational database model. *EATCS Monographs on Theoretical Computer Science*, 1989.

[11] J. A. Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12.1, 1965.

[12] Solveig Torgersen. Automatic design of relational databases. *Ph. D. Thesis. TR 89-1038*, 1989.

[13] Jeffrey D. Ullman. Database and knowledge-base systems. *Computer Science Press*, 1988.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Optimal Cooling Strategies in Polymer Crystallization

Ramón Escobedo[1] and Luis A. Fernández[2]

[1] *Departamento de Matemática Aplicada y Ciencias de la Computación, Universidad de Cantabria, Av. de Los Castros s/n, Santander 39005, Spain*

[2] *Departamento de Matemáticas, Estadística y Computación, Universidad de Cantabria, Av. de Los Castros s/n, Santander 39005, Spain*

emails: escobedo@unican.es, lafernandez@unican.es

Abstract

An optimal control problem for cooling strategies in polymer crystallization processes described by a deterministic model is solved in the framework of a free boundary problem. The strategy of cooling both sides of a one dimensional sample is introduced for the first time in this model, and is shown to be well approximated by the sum of the solutions of two one-phase Stefan problems, even for arbitrary applied temperature profiles. This result is then used to show that cooling both sides is always more effective in polymer production than injecting the same amount of cold through only one side. The optimal cooling strategy, focused in avoiding low temperatures and in shortening cooling times, is derived, and consists in applying the same constant temperature at both sides. Explicit expressions of the optimal controls in terms of the parameters of the material are also obtained.

Key words: optimal control, Stefan problem, polymer crystallization
MSC 2000: 49J20, 35R35, 35K55, 65M06, 80A22

1 Introduction

Optimization of cooling strategies is a fundamental part of modeling polymerization processes. A recent model of polymer crystallization [2, 3] is being studied to derive the optimal cooling strategy in terms of the industrial main interests, focused in reducing the duration of the cooling process while avoiding excessively low temperatures.

The model consists of two non-linear partial differential equations for the degree of crystallinity $y(x,t)$, defined as the mean volume fraction of the space occupied by crystals, and the temperature field $T(x,t)$, coupled by means of the rate functions of nucleation and growth $b_N(T)$ and $b_G(T)$, the function of starting of nucleation $\kappa(y) = (1-y)^2$, and the function of aggregation and saturation of nuclei $\beta(y) = y(1-y)$:

$$y_t(x,t) = \beta(y(x,t))b_G(T(x,t)) + v_0\kappa(y(x,t))b_N(T(x,t)), \tag{1}$$

$$T_t(x,t) = \sigma T_{xx}(x,t) + a_G\beta(y(x,t))b_G(T(x,t)), \tag{2}$$

 ISBN 13: 978-84-613-5510-5

for $(x,t) \in Q_\tau = (0,L) \times (0,\tau)$, where L is the length of the sample and τ is the time at which the cooling process is stopped.

Equations (1)–(2) are solved with the following boundary and initial conditions:

$$T(0,t) = u_0(t), \quad T(L,t) = u_L(t), \qquad t \in (0,\tau), \tag{3}$$

$$y(x,0) = 0, \qquad T(x,0) = T_0, \qquad x \in (0,L). \tag{4}$$

The nucleation and growth rate functions are such that $b_G(T)/G = b_N(T)/N = \theta(T)$, where

$$\theta(T) \stackrel{def}{=} \begin{cases} \exp(-\eta T) & \text{if } T < T_f, \\ 0 & \text{if } T \geq T_f. \end{cases} \tag{5}$$

The parameters $G, v_0, N, \sigma, a_G, \eta$ and T_f are positive real constants denoting the growth factor, the initial mass, the nucleation factor, the heat diffusion coefficient, the non-isothermal factor, the nucleation and growth exponent and the critical phase transition temperature (from liquid to solid), respectively. Typical values and more details of the model can be found in Refs. [2], [3] and [4].

Condition (3) means that the injection of cold is applied at both sides of the sample, $x_0 = 0$ and $x_L = L$; we call this case a *double cooling* strategy. Previous strategies used in this model have only considered to cool one side of the sample (*single cooling*), using a thermally insulated boundary at the other side (e.g. $T_x(L,t) = 0$); see Refs. [3, 4, 5].

In the single cooling case, a crystallization front is formed close to the cooling side and moves towards the interior of the sample until the other side is reached. The front separates the liquid ($y = 0$) and the solid ($y = 1$) phases, and is not a travelling wave; instead, it is a *band of crystallization* which exhibits an oscillating advance with variable shape and velocity strongly dependent on the parameters of the material [3].

Under some conditions, the crystallization band can be identified with a thin interface where the nucleation and growth processes are confined and take place at the freezing temperature T_f [5]. Then, a free boundary problem (FBP) framework can be used to describe the polymerization process by means of a one-phase Stefan problem [1]. Before this framework was established, numerical simulations were recently used to derive both the optimal applied temperature $\overline{u}_0$ and the cooling process duration $\overline{\tau}$ giving rise to the optimal single cooling strategy [4].

In the present paper this FBP framework is used to characterize the solution of the double cooling problem (1)–(4) by means of two Stefan problems, allowing us to show that double cooling is always more effective than single cooling (injecting the same amount of cold), and to derive explicit expressions of the optimal controls $\overline{u}_0(t)$, $\overline{u}_L(t)$ and $\overline{\tau}$ giving rise to the optimal cooling strategy, expressions written in terms of the parameters of the material.

 ISBN 13: 978-84-613-5510-5

2 Stefan problems describing polymerization processes

The FBP framework for single cooling strategies consists in identifying a free boundary $h(t)$ with the instantaneous amount of crystallized polymer $P(t)$ defined by

$$P(t) \stackrel{def}{=} \int_0^L y(x,t)\,dx. \tag{6}$$

The free boundary $h(t)$ allows to consider the crystallinity as a step function in the whole sample, $y(x,t) = 1$ in $[0, h(t)]$ and $y(x,t) = 0$ in $[h(t), L]$, so that $P(t) = h(t)$. At the interface, the temperature is assumed to be precisely T_f, i.e. $T(h(t), t) = T_f$.

These assumptions allow us to derive a Stefan condition and the corresponding Stefan problem; details of the derivation and the solution of the Stefan problem for different applied temperature profiles will be presented elsewhere [5]. Here it suffices to say that the Stefan condition provides us with an explicit expression of the ratio of the latent heat $\mathcal{L}_\delta$ to the specific heat c in terms of the parameters of the material, $\mathcal{L}_\delta/c = a_G K_\delta$, where $K_\delta = [1+\delta(\ln\delta - 1)]/(1-\delta)^2$, $\delta = v_0 N/G$, and that the solution of the Stefan problem for arbitrary applied temperature profiles is given by the so-called *pseudo-steady state* (PSS) approximation, valid in the limit $Ste \ll 1$, where the Stefan number Ste is the ratio of the sensible heat $c\Delta T = c \max_t\{T_f - u(t)\}$ to the latent heat [1]:

$$Ste \stackrel{def}{=} \frac{c\Delta T}{\mathcal{L}_\delta}. \tag{7}$$

2.1 Stefan problems for double cooling strategies

When both sides of the sample are cooled, two crystallization bands emerge and move towards each other until they merge somewhere in the interior of the sample.

We claim that a double cooling process can be seen as the sum of two single cooling processes, and therefore can be approximated by means of two Stefan problems for two free boundaries $h_0(t)$ and $h_L(t)$: for $i = 0, L$,

$$\frac{\partial T_i}{\partial t}(x,t) = \sigma \frac{\partial^2 T_i}{\partial x^2}(x,t), \quad x \in [0, h_i(t)),\ t > 0, \tag{8}$$

$$T_i(x,t) = T_f, \quad x \in (h_i(t), +\infty),\ t > 0, \tag{9}$$

$$T_i(0,t) = u_i(t), \quad t > 0, \tag{10}$$

$$T_i(h_i(t), t) = T_f, \quad t > 0, \tag{11}$$

$$\frac{\mathcal{L}_\delta}{c}\, h_i'(t) = \sigma \frac{\partial T_i}{\partial x}(h_i(t), t), \quad t > 0. \tag{12}$$

The PSS solution of these Stefan problems are, for $i = 0, L$ (see Refs. [1, 5]),

$$h_i^{\mathrm{PSS}}(t) = \sqrt{\frac{2\sigma c}{\mathcal{L}_\delta} Q_i(t)}, \tag{13}$$

$$T_i^{\mathrm{PSS}}(x,t) = \begin{cases} u_i(t) + \dfrac{T_f - u_i(t)}{h_i^{\mathrm{PSS}}(t)}\, x & \text{if } x \le h_i^{\mathrm{PSS}}(t), \\ T_f & \text{if } h_i^{\mathrm{PSS}}(t) \le x, \end{cases} \tag{14}$$

 ISBN 13: 978-84-613-5510-5

where $Q_i(t)$ is the total amount of cold injected into the sample along the time interval $[0,t]$ through the boundary x_i:

$$Q_i(t) \stackrel{def}{=} \int_0^t \Big(T_f - u_i(s)\Big)\, ds, \quad \text{for } i = 0, L. \tag{15}$$

Then, the temperature and crystallinity profiles of the double cooling process can be approximated by the following functions: (see Fig. 1 and error estimates)

$$T^{\rm PSS}(x,t) = T_0^{\rm PSS}(x,t) + T_L^{\rm PSS}(L-x,t) - T_f, \tag{16}$$

$$y^{\rm PSS}(x,t) = \begin{cases} 0 & \text{if } x \in [h_0^{\rm PSS}(t), L - h_L^{\rm PSS}(t)], \\ 1 & \text{if not.} \end{cases} \tag{17}$$

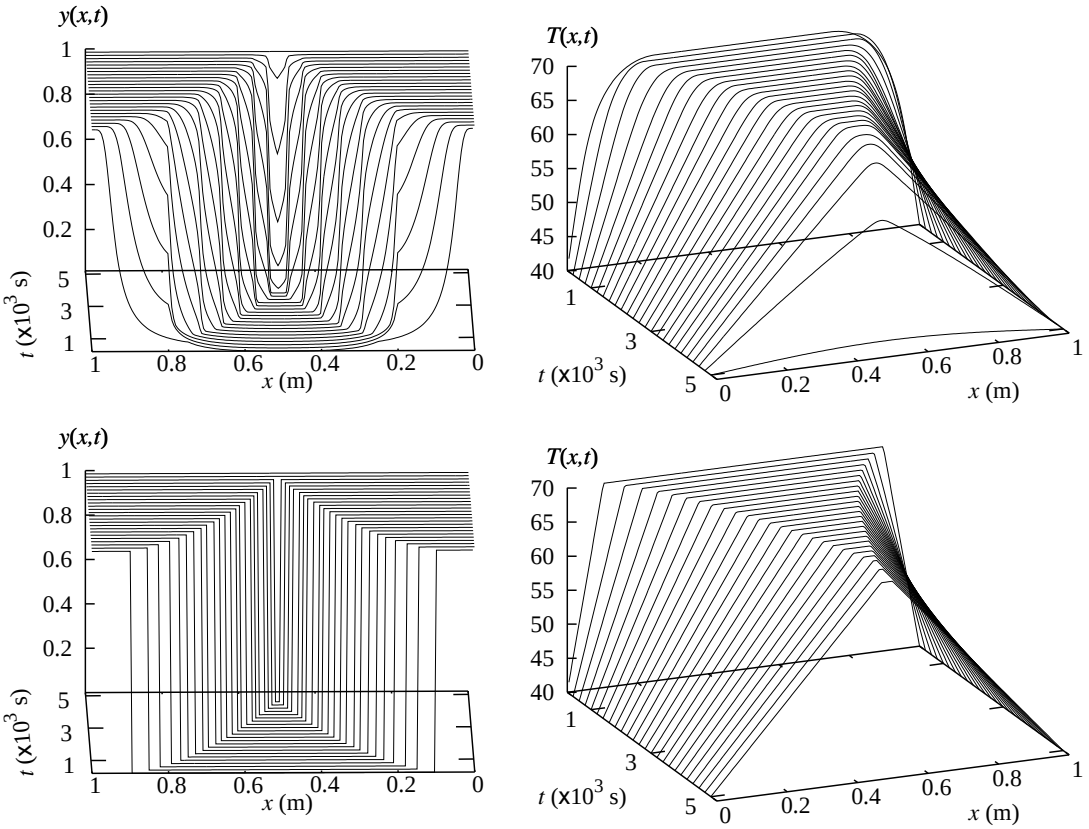


Figure 1: Upper row: Crystallinity (left) and temperature field (right) obtained by direct simulation of the polymerization problem (1)–(4). Lower row: same as above but obtained by using (16)–(17) with the solution (13)–(14) of the Stefan problems (8)–(12). Parameter values are $u_0 = u_L = 40$ °C, $T_f = 70$ °C, $\sigma = 0.002$ m^{2}s^{-1}, $a_G = 2500$ °C, $G = 5$ s^{-1}, $N = 20$ s^{-1}, $v_0 = 0.01$, $\eta = 0.1$, $L = 1$ m and $T_0 = 100$ °C. Resulting values are $\delta = 0.04$, $K_\delta = 0.902$ and $Ste = 0.013$. Note the symmetry with respect to $x = 0.5$.

An excellent agreement is also obtained for different applied temperatures profiles, as shown in Fig. 2, where we have depicted the time evolution of the free boundaries

 ISBN 13: 978-84-613-5510-5

$h_0(t)$ and $h_L(t)$ together with their sum and the magnitude thus approximated, $P(t)$, for the case described in Fig. 1, a case where the applied temperature is variable in time, and a case of an asymmetric double cooling. Error estimates are obtained later.

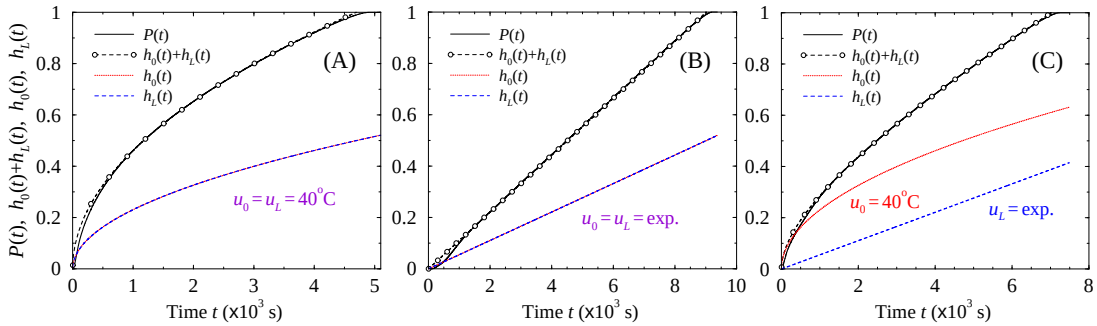


Figure 2: Total amount of polymer $P(t)$ (solid line) compared with the sum of the two free boundaries $h_0(t) + h_L(t)$ (solid line with circles); also depicted are $h_0(t)$ and $h_L(t)$ (doted and dashed lines respectively). (A) $u_0 = u_L = 40$ °C, (B) $u_0 = u_L = T_f + \mathcal{L}_\delta(1 - e^{\gamma^2\sigma t})/c$. (C) $u_0 = 40$ °C, $u_L = T_f + \mathcal{L}_\delta(1 - e^{\gamma^2\sigma t})/c$ (asymmetric cooling). We have used $\gamma = 2.76 \times 10^{-2}$ m^{-1}.

2.2 Amount of crystallized polymer in double cooling strategies

According to the results obtained in the previous section, it turns out that the amount of crystallized polymer $P(t)$ can be accurately approximated by

$$P(t) = h_0^{\mathrm{PSS}}(t) + h_L^{\mathrm{PSS}}(t). \tag{18}$$

Therefore, a given amount of cold $Q(t) = Q_0(t) + Q_L(t)$ injected into the sample with a double cooling strategy will produce an amount of polymer given by

$$\begin{aligned} P(t) &= \sqrt{\frac{2\sigma c}{\mathcal{L}_\delta} Q_0(t)} + \sqrt{\frac{2\sigma c}{\mathcal{L}_\delta} Q_L(t)} && (19)\\ &\geq \sqrt{\frac{2\sigma c}{\mathcal{L}_\delta}\Big[Q_0(t) + Q_L(t)\Big]} && (20)\\ &\geq \sqrt{\frac{2\sigma c}{\mathcal{L}_\delta} Q(t)} \stackrel{def}{=} \overline{P}(t), && (21) \end{aligned}$$

where $\overline{P}(t)$ is the amount of crystallized polymer produced by injecting an amount of cold $Q(t)$ by cooling only one side of the sample, showing that double cooling always produces a greater or equal amount of crystallized polymer than single cooling.

Moreover, the maximal production of crystallized polymer achievable by injecting a given amount of cold $Q(t)$ is reached when a double cooling strategy with $Q_0(t) = Q_L(t) = Q(t)/2$ is used. In this case,

$$P(t) = \sqrt{2}\,\bar{P}(t). \tag{22}$$

 ISBN 13: 978-84-613-5510-5

2.3 Crystallization time in double cooling strategies

The total crystallization is reached when $P(t_{\text{cryst}}) = L$. Thus, the PSS approximation provides us with the following equation to estimate the crystallization time:

$$\left(\sqrt{Q_0(t^{\text{PSS}}_{\text{cryst}})} + \sqrt{Q_L(t^{\text{PSS}}_{\text{cryst}})}\right)^2 = \frac{\mathcal{L}_\delta}{2\sigma c} L^2. \tag{23}$$

When the same constant temperature u is applied at both sides, expression (23) yields

$$t^{\text{PSS}}_{\text{cryst}} = \frac{L^2}{8\sigma Ste}, \tag{24}$$

where $Ste = c(T_f - u)/\mathcal{L}_\delta$, showing that the time needed for complete crystallization when cooling both sides at a given constant applied temperature is a quarter of the time needed when cooling only one side with the same temperature, and that the double cooling requires only one half of the amount of cold required by the simple cooling.

2.4 Errors estimates

To check how accurate our approximation is, the following error estimates, introduced in Ref. [5] to test the FBP framework, are used:

$$\xi(t) = P(t) - \left(h_0^{\text{PSS}}(t) + h_L^{\text{PSS}}(t)\right), \tag{25}$$

$$\varepsilon(x,t) = T^{\text{NUM}}(x,t) - T^{\text{PSS}}(x,t), \tag{26}$$

where T^{NUM} denotes the temperature obtained by solving the polymerization problem (1)–(4) numerically. They are measured with the normalized L_1 and L_2 norms,

$$\xi_{L_2} = \frac{1}{(t_2 - t_1)L}\left(\int_{t_1}^{t_2} \xi^2(t)\, dt\right)^{1/2}, \tag{27}$$

$$\varepsilon_T = \frac{1}{(t_2 - t_1)T_f L}\int_{t_1}^{t_2}\left(\int_0^L \varepsilon^2(x,t)\, dx\right)^{1/2} dt, \tag{28}$$

where t_1 and t_2 correspond to the short transient times at the beginning and the end of the polymerization process, during which the crystallization band is not formed [5].

Moreover, $t^{\text{PSS}}_{\text{cryst}}$, the solution of (23), can be also compared with the crystallization time $t^{\text{NUM}}_{\text{cryst}}$, obtained numerically by solving $P(t^{\text{NUM}}_{\text{cryst}}) = L$, by using the relative error

$$\epsilon = \frac{\left|t^{\text{NUM}}_{\text{cryst}} - t^{\text{PSS}}_{\text{cryst}}\right|}{t^{\text{NUM}}_{\text{cryst}}}. \tag{29}$$

Error estimates ξ_{L_2}, ε_T and ϵ are calculated for the three cooling strategies depicted in Figs. 1 and 2, and are reported in Table 1.

The results show that error estimates are of the same order, but twice the value, than those obtained when describing single cooling strategies under the FBP framework

 ISBN 13: 978-84-613-5510-5

	u_0 (°C)	u_L (°C)	ξ_{L_2} (10^{-5})	ε_T (10^{-3})	ϵ (10^{-2})	$t_{\text{cryst}}^{\text{NUM}}$ (10^3 s)	t_1 (10^3 s)	t_2 (10^3 s)	% –
A	40	40	8.12	3.73	8.1	5.11	0.23	4.55	84.6
B	exp.	exp.	2.6	0.18	3.5	9.37	1.1	8.9	84.7
C	40	exp.	6.98	2.16	6.2	7.49	0.22	6.66	85.9

Table 1: Error estimates and crystallization times for the cases depicted in Fig. 2.

sketched above [5], thus confirming that the double cooling polymerization process is accurately described by the sum of the two Stefan problems. A priori, this seems quite surprising, due to the nonlinear character of the polymerization problem.

Note that we are intentionally not using the exact solution of the Stefan problems (available when the applied temperature is constant or exponentially decreasing [1]) because our goal consists in proving that the PSS approximation is accurate enough to describe the double cooling polymerization process.

3 The optimal control problem

The optimal control problem of the single cooling case was solved numerically in Ref. [4]. The FBP framework allows us to rewrite the solution in terms of the parameters of the material, and to derive the corresponding optimal controls for the double cooling case.

3.1 Single cooling

Expression (8) from Ref. [4] shows the relation between the amount of crystallized polymer $P(t)$ and the amount of injected cold $Q(t)$, obtained numerically:

$$P(t) \approx \alpha\sqrt{Q(t)}, \tag{30}$$

where α is a positive real constant. Expression (13) provides us with its analytical expression, $\alpha = \sqrt{2\sigma c/\mathcal{L}_\delta}$, and consequently with the analytical expression of the optimal controls $\bar{u}$ and $\bar{\tau}$, see [4]:

$$\text{if } \sigma_2 > \sigma_1 T_f^2: \qquad \overline{u}(t) \equiv 0, \quad \overline{\tau} = \frac{\mathcal{L}_\delta}{2\sigma c T_f} L^2, \tag{31}$$

$$\text{if } \sigma_2 \leq \sigma_1 T_f^2: \qquad \overline{u}(t) \equiv T_f - \sqrt{\frac{\sigma_2}{\sigma_1}}, \quad \overline{\tau} = \sqrt{\frac{\sigma_1}{\sigma_2}}\frac{\mathcal{L}_\delta}{2\sigma c} L^2, \tag{32}$$

where the control problem and the set of admissible controls U_{ad} were defined as follows,

$$\begin{cases} Min \;\; J(u,\tau) = \sigma_1 \displaystyle\int_0^\tau \Big(T_f - u(t)\Big)^2 dt + \sigma_2 \tau, \\ (u,\tau) \in U_{ad} = \big\{(u,\tau) \in L^2(0,\tau) \times [0,+\infty) : u(t) \in [0,T_f],\ \text{a.e.}, P(\tau) = L\big\}, \end{cases} \tag{33}$$

and σ_1 and σ_2 are non-negative weights fixed to balance the contribution of each term.

 ISBN 13: 978-84-613-5510-5

3.2 Double cooling

For the optimal control of double cooling strategies, the control parameters are the applied temperatures $\vec{u}(t) = (u_0(t), u_L(t)) \in (L^2(0,\tau))^2$ and the duration of the cooling process $\tau \in [0, +\infty)$. Following the same argument used in [4], a cost functional $J(\vec{u},\tau)$ promoting a short duration of the cooling process and avoiding excessively low applied temperatures can be written as

$$(CP)\begin{cases} Min \;\; J(\vec{u},\tau) = \sigma_1 \left[\int_0^\tau \Big(T_f - u_0(t)\Big)^2 + \Big(T_f - u_L(t)\Big)^2 dt\right] + \sigma_2\tau, \\ (\vec{u},\tau) \in U_{ad}, \end{cases} \tag{34}$$

$$\text{where } U_{ad} = \Big\{(\vec{u},\tau) \in (L^2(0,\tau))^2 \times [0,+\infty) : \vec{u}(t) \in [0,T_f]^2, \text{ a.e., } P(\tau) = L\Big\}.$$

As in the single cooling case, the rate σ_2/σ_1 is a measure of the relative cost of the two opposite contributions to the cost functional: to avoid low temperatures (σ_2/σ_1 small) and to shorten the cooling time (σ_2/σ_1 large). The cost of double cooling is considered symmetric, that is, cooling at $x_0 = 0$ has the same cost than cooling at $x_L = L$.

Optimal controls. Let $(\vec{u},\tau)$ be an admissible control; then, complete crystallization is reached at $t = \tau$ and expression (23) means

$$\Big(\sqrt{Q_0(\tau)} + \sqrt{Q_L(\tau)}\Big)^2 = \frac{\mathcal{L}_\delta}{2\sigma c} L^2. \tag{35}$$

Therefore, (CP) can be reformulated as the following optimization problem:

$$(OP)\begin{cases} Min \;\; J(\vec{u},\tau) = \sigma_1 \left[\int_0^\tau \Big(T_f - u_0(t)\Big)^2 + \Big(T_f - u_L(t)\Big)^2 dt\right] + \sigma_2\tau, \\ (\vec{u},\tau) \in V_{ad}, \end{cases} \tag{36}$$

$$\text{where } V_{ad} = \Big\{(\vec{u},\tau) \in (L^2(0,\tau))^2 \times [0,+\infty) : \vec{u}(t) \in [0,T_f]^2 \text{ a.e., and (35) holds}\Big\}.$$

The admissibility condition $(\vec{u},\tau) \in V_{ad}$ implies that there exists a lower bound for τ, corresponding to $u_0 = u_L = 0$ °C:

$$\tau \geq \hat{\tau} \stackrel{def}{=} \frac{\mathcal{L}_\delta}{8\sigma c T_f} L^2. \tag{37}$$

Theorem 1 *Assume that $\sigma_1 \in [0,+\infty)$ and $\sigma_2 \in (0,+\infty)$. Then,*

a) If $\sigma_2 > 2\sigma_1 T_f^2$, the unique solution of the optimization problem (OP) is given by

$$\overline{u}_0(t) = \overline{u}_L(t) \equiv 0, \qquad \overline{\tau} = \frac{\mathcal{L}_\delta}{8\sigma c T_f} L^2. \tag{38}$$

b) If $\sigma_2 \leq 2\sigma_1 T_f^2$, the unique solution of the optimization problem (OP) is given by

$$\overline{u}_0(t) = \overline{u}_L(t) \equiv T_f - \frac{1}{2}\sqrt{\frac{2\sigma_2}{\sigma_1}}, \qquad \overline{\tau} = \sqrt{\frac{\sigma_1}{2\sigma_2}}\left(\frac{\mathcal{L}_\delta}{4\sigma c} L^2\right). \tag{39}$$

 ISBN 13: 978-84-613-5510-5

Proof: Using that $2(a+b) \geq (\sqrt{a}+\sqrt{b}\,)^2$ in expression (35) yields

$$Q_0(\tau) + Q_L(\tau) \geq \frac{\mathcal{L}_\delta}{4\sigma c} L^2. \tag{40}$$

Thanks to Hölder's inequality (for $i = 0, L$)

$$\sqrt{\tau}\left(\int_0^\tau \Big(T_f - u_i(t)\Big)^2 dt\right)^{1/2} \geq \int_0^\tau \Big(T_f - u_i(t)\Big)\, dt, \tag{41}$$

we get, $\forall (\vec{u}, \tau) \in V_{ad}$:

$$\left(\int_0^\tau \Big(T_f - u_0(t)\Big)^2 dt\right)^{1/2} + \left(\int_0^\tau \Big(T_f - u_L(t)\Big)^2 dt\right)^{1/2} \geq \frac{\mathcal{L}_\delta}{4\sqrt{\tau}\sigma c} L^2. \tag{42}$$

Using again $2(a+b) \geq (\sqrt{a}+\sqrt{b}\,)^2$, we have

$$\int_0^\tau \Big(T_f - u_0(t)\Big)^2 dt + \int_0^\tau \Big(T_f - u_L(t)\Big)^2 dt \geq \frac{1}{2\tau}\left(\frac{\mathcal{L}_\delta}{4\sigma c} L^2\right)^2. \tag{43}$$

Consequently,

$$J(\vec{u}, \tau) \geq \psi(\tau), \quad \forall (\vec{u}, \tau) \in V_{ad}, \tag{44}$$

where we have used the auxiliary real function

$$\psi(\tau) = \frac{\sigma_1}{2\tau}\left(\frac{\mathcal{L}_\delta}{4\sigma c} L^2\right)^2 + \sigma_2 \tau, \quad \tau \in [\hat{\tau}, +\infty). \tag{45}$$

Elementary calculus shows that $\psi(\tau)$ has a unique global minimum at $\overline{\tau}$ in $[\hat{\tau}, +\infty)$, where

$$\overline{\tau} = \begin{cases} \hat{\tau} & \text{if } \sigma_2 > 2\sigma_1 T_f^2, \\ \sqrt{\dfrac{\sigma_1}{2\sigma_2}}\left(\dfrac{\mathcal{L}_\delta}{4\sigma c} L^2\right) & \text{if } \sigma_2 \leq 2\sigma_1 T_f^2. \end{cases} \tag{46}$$

Then, it is quite easy to verify that if $\sigma_2 \leq 2\sigma_1 T_f^2$,

$$J(\vec{u}, \tau) \geq \psi(\tau) \geq \psi(\overline{\tau}) = J\left(T_f - \frac{1}{2}\sqrt{\frac{2\sigma_2}{\sigma_1}}, T_f - \frac{1}{2}\sqrt{\frac{2\sigma_2}{\sigma_1}}, \overline{\tau}\right), \quad \forall (\vec{u}, \tau) \in V_{ad}, \tag{47}$$

meanwhile, if $\sigma_2 > 2\sigma_1 T_f^2$,

$$J(\vec{u}, \tau) \geq \psi(\tau) \geq \psi(\hat{\tau}) = J(0, 0, \hat{\tau}), \quad \forall (\vec{u}, \tau) \in V_{ad}. \tag{48}$$

To prove the uniqueness, let us assume that $(\vec{u}^\star, \tau^\star) \in V_{ad}$ is another solution of (OP), i.e. $J(\vec{u}^\star, \tau^\star) = J(\vec{\overline{u}}, \overline{\tau})$. In any case it is easy to deduce from previous estimations that $\psi(\tau^\star) = \psi(\overline{\tau})$. Since ψ is strictly convex, we get that $\tau^\star = \overline{\tau}$.

 ISBN 13: 978-84-613-5510-5

We conclude that $\vec{u}^\star(t) = \vec{\overline{u}}(t)$, a.e. $t \in (0, \overline{\tau})$, by seeing that

$$\begin{aligned}
&\int_0^{\overline{\tau}} \Big(u_0^\star(t) - \overline{u}_0\Big)^2 dt + \int_0^{\overline{\tau}} \Big(u_L^\star(t) - \overline{u}_L\Big)^2 dt \\
&\quad = \int_0^{\overline{\tau}} \Big((u_0^\star(t) - T_f) + (T_f - \overline{u}_0)\Big)^2 dt + \int_0^{\overline{\tau}} \Big((u_L^\star(t) - T_f) + (T_f - \overline{u}_L)\Big)^2 dt \\
&\quad = \int_0^{\overline{\tau}} \Big(u_0^\star(t) - T_f\Big)^2 dt + \int_0^{\overline{\tau}} (T_f - \overline{u}_0)^2 dt - 2(T_f - \overline{u}_0) \int_0^{\overline{\tau}} \Big(T_f - u_0^\star(t)\Big)\, dt \\
&\quad + \int_0^{\overline{\tau}} \Big(u_L^\star(t) - T_f\Big)^2 dt + \int_0^{\overline{\tau}} (T_f - \overline{u}_L)^2 dt - 2(T_f - \overline{u}_L) \int_0^{\overline{\tau}} \Big(T_f - u_L^\star(t)\Big)\, dt \\
&\quad = 2\int_0^{\overline{\tau}} (T_f - \overline{u}_0)^2 dt - 2(T_f - \overline{u}_0) \int_0^{\overline{\tau}} \Big(T_f - u_0^\star(t)\Big)\, dt + 2\int_0^{\overline{\tau}} (T_f - \overline{u}_L)^2 dt \\
&\quad - 2(T_f - \overline{u}_L) \int_0^{\overline{\tau}} \Big(T_f - u_L^\star(t)\Big)\, dt \;\leq\; 0,
\end{aligned}$$

where we have used the inequality (40) for $(\vec{u}^\star, \overline{\tau})$ and the equality

$$\int_0^{\overline{\tau}} \Big(T_f - u_0^\star(t)\Big)^2 dt + \int_0^{\overline{\tau}} \Big(T_f - u_L^\star(t)\Big)^2 dt = \int_0^{\overline{\tau}} (T_f - \overline{u}_0)^2 dt + \int_0^{\overline{\tau}} (T_f - \overline{u}_L)^2 dt. \quad (49)$$

Noticeably, the remarks written in Ref. [4] about the choice of the ratio σ_2/σ_1 for the single cooling are also in order in the double cooling case.

4 Conclusion

We have analyzed a recent polymer crystallization model for a new kind of cooling strategy, consisting in cooling the sample at both sides. By means of a free boundary problem framework whose main features have been presented here, we have shown that the double cooling crystallization process can be approximated by the sum of two Stefan problems. Explicit expressions of the solution have been derived and errors estimates have revealed a quite high accuracy, both in reproducing the behaviour of the crystallization front and the distribution of the temperature field. Also, important magnitudes such as the crystallization time and the amount of crystallized polymer with respect to the single cooling case have been derived explicitly.

The characterization of the double cooling crystallization process by means of two Stefan problems has then be used to find the optimal cooling strategy when both sides can be cooled. The solution of the control problem is obtained explicitly in terms of the parameters of the material, and shows that the optimal strategy consists in first, using a symmetric strategy, that is, applying the same cooling temperature at both sides of the sample, and second, using a constant temperature, thus recovering the result obtained for the single cooling case.

The free boundary problem framework has therefore shown to be quite effective in the analysis of cooling strategies in polymerization processes and could be used in higher dimension problems in future works.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science and Innovation under grant No. $MTM2008-04206$ and a "Ramón y Cajal" contract.

References

[1] V. Alexiades and A. D. Solomon, *Mathematical modeling of melting and freezing processes*, (Hemisphere Publ. Co., Washington DC, 1993).

[2] V. Capasso, H. Engl and J. Periaux, Eds., *Computational Mathematics Driven by Industrial Problems*, Springer Berlin / Heidelberg, 2000. V. Capasso, *Mathematical models for polymer crystallization processes*, pp 39–67.

[3] R. Escobedo and V. Capasso, *Moving bands and moving boundaries with decreasing speed in polymer crystallization*, Math. Mod. Meth. in Appl. Sci. (M3AS) **15**, No. 3 (2005) 325–341.

[4] R. Escobedo and L. A. Fernández, *Optimal control of chemical birth and growth processes in a deterministic model*, J. Math. Chem. (online first) (2009) DOI: 10.1007/s10910-009-9638-x

[5] R. Escobedo and L. A. Fernández, *A classical one-phase Stefan problem for describing a polymer crystallization process*, (2010, in preparation).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A Rakhmanov-like theorem for orthogonal polynomials on Jordan arcs in the complex plane

C. Escribano[1], M. A. Sastre[1], A. Giraldo[1] and E. Torrano[1]

[1] *Departamento de Matemática Aplicada, Facultad de Informática, Universidad Politécnica de Madrid*

emails: cescribano@fi.upm.es, masastre@fi.upm.es, agiraldo@fi.upm.es, emilio@fi.upm.es

Abstract

Rakhmanov's theorem establishes a result about the asymptotic behavior of the elements of the Jacobi matrix associated with a measure μ which is defined on the interval $\mathcal{I} = [-1, 1]$ with $\mu' > 0$ almost everywhere on $\mathcal{I}$. In this work we give a weak version of this theorem, for a measure with support on a connected finite union of Jordan arcs on the complex plane, in terms of the Hessenberg matrix, the natural generalization of the tridiagonal Jacobi matrix to the complex plane.

Key words: Hessenberg matrix, regular measures, Riemann map.

1 Introduction

In this paper, we consider regular Borel measures μ defined on subsets of the complex plane which are Jordan arcs, or connected finite union of Jordan arcs, and we show how the support of μ is determined by the entries of the Hessenberg matrix D associated with μ. The Hessenberg matrix is the natural generalization of the tridiagonal Jacobi matrix to the complex plane and, in the particular case of measures with support the unit circle $\mathbb{T}$, the Hessenberg matrix is a Toeplitz matrix.

Our result represents a broader, although weaker, extension of Rakhmanov's theorem to $\mathbb{C}$. In the real case, Rakhmanov's theorem [15, 16] states that, if the support of a Borel measure is $[-1, 1]$ and $\mu' > 0$ almost everywhere in $[-1, 1]$, then $a_n \to \frac{1}{2}$ and $b_n \to 0$, where a_n are the sequences of elements in the subdiagonal and superdiagonal, and b_n are the sequences of elements in the diagonal, in the Jacobi matrix J associated with μ. Moreover, if the support of μ is the interval $[-2a + b, b + 2a]$, then the above limits are, respectively, $a_n \to a$ y $b_n \to b$. Conversely, if we know that $\mu' > 0$ and that the support of μ is a compact connected set of $\mathbb{R}$, knowing the limits of the diagonals

 ISBN 13: 978-84-613-5510-5

of J we could obtain the support of μ, i.e., if $a_n \to a$ y $b_n \to b$ then the support is $[-2a+b, b+2a]$.

Generalizations of Rakhmanov's theorem to orthogonal polynomials and to orthogonal matrix polynomials on the unit circle has been given in [13] and [22]. The case of orthogonal polynomials in an arc of circumference has been studied in [2]).

There exist some previous results relating the properties of D and the support of μ. For example, if the Hessenberg matrix D defines a subnormal operator [12] in ℓ^2, then the closure of the convex hull of its numerical range agrees with the convex hull of its spectrum. On the other hand, the spectrum of the matrix D contains the spectrum of its minimal normal extension $N = \text{men}(D)$ which is precisely the support of the measure [6].

In this work we show that, in the case of regular measures μ whose support is a Jordan arc or a connected union of Jordan arcs in the complex plane $\mathbb{C}$, the limits of the values at the diagonals of the Hessenberg matrix D of μ, supposing those limits exist, determine the terms of the coefficients of the series expansion of the Riemann map $\phi(z)$ (see [20]) which applies conformally the exterior of the unit disk in the exterior of the support of the measure. As a consequence, the support of μ can be determined just knowing the limits of the values at the diagonals of its Hessenberg matrix D.

For general information on the theory of orthogonal polynomials, we recommend the books [4, 20] by T. S. Chihara and G. Szegö, respectively, and the survey [11] by Golinskii and Totik.

2 Main result

Let $\mu(z)$ be a regular positive Borel measure with compact support Ω in the complex plane. Let $\mathcal{P}$ be the space of polynomials. The associated inner product is given by the expression

$$\langle Q(z), R(z)\rangle_\mu = \int_{\text{supp}(\mu)} Q(z)\overline{R(z)}d\mu(z),$$

for $R, S \in \mathcal{P}$. Then there exists a unique orthonormal polynomials sequence (ONPS) $\{P_n(z)\}_{n=0}^\infty$ associated to the measure μ (see [4], [8] or [20]).

In the space $\mathcal{P}^2(\mu)$, closure of the polynomials space $\mathcal{P}$ in $L^2_\mu(\Omega)$, we consider the multiplication by z operator. Let $D = (d_{jk})_{j,k=0}^\infty$ be the infinite upper Hessenberg matrix of this operator in the basis of ONPS $\{P_n(z)\}_{n=0}^\infty$, hence

$$zP_n(z) = \sum_{k=0}^{n+1} d_{k,n}P_k(z), \quad n \geq 0, \tag{1}$$

with $P_0(z) = 1$ when $c_{00} = 1$.

It is a well known fact that the monic polynomials are the characteristic polynomials of the finite sections of D.

In order to state our main result, we will need that the measure μ is regular with support a connected finite union of Jordan arcs, and we will also need to consider an auxiliar Toeplitz matrix. We next recall the definitions of all these notions.

 ISBN 13: 978-84-613-5510-5

A Jordan arc in $\mathbb{C}$ is any subset of $\mathbb{C}$ homeomorphic to the closed interval $[0,1]$ on the real line.

A measure μ is regular if $\lim_{n\to\infty} \frac{1}{\sqrt[n]{\gamma_n}} = \mathrm{cap}(\mathrm{supp}(\mu))$, the capacity of the support of μ, where the γ_n are the conductor coefficients of the orthonormal polynomials, i.e., $P_n(z) = \gamma_n z^n + \ldots$.

An infinite matrix $T = (a_{i,j})_{i,j=0}^{\infty}$ is a Toeplitz matrix if each descending diagonal from left to right is constant, i.e, there exists $(a_i)_{i\in\mathbb{Z}}$ such that $a_{i,j} = a_{i-j}$, for every $i, j \in \mathbb{N}\cup\{0\}$. Given a Toeplitz matrix T, the Laurent series whose coefficients are the entries a_i defines a function known as the symbol of T.

We are now in a position to state and prove the main result of the paper.

Theorem 1. *Let $D = (d_{ij})_{i,j=1}^{\infty}$ be a Hessenberg matrix associated with a measure μ with compact support on the complex plane. Assume that:*

1. *The measure μ is regular with support $\mathrm{supp}(\mu)$ a Jordan arc or a connected finite union of Jordan arcs Γ such that $\mathbb{C}\setminus\Gamma$ is a simply connected set of the Riemann sphere $\mathbb{C}_\infty$.*

2. *There exists a Hessenberg-Toeplitz matrix T such that $D - T$ defines a compact operator in ℓ^2 with its rows in ℓ^1.*

Then, the symbol of T is the Riemann function $\phi : \mathbb{C}_\infty \setminus \overline{\mathbb{D}} \to \mathbb{C}_\infty \setminus \Gamma$

Proof. Since $\mathrm{supp}(\mu) = \Gamma$ is a compact set and $\mathbb{C}_\infty \setminus\Gamma$ is connected, we can apply Merguelyan's theorem [9, p.97] which asserts that every continuous function in Γ can be uniformly approximated by polynomials. Since the set of continuous functions with compact support is dense in $L^2_\mu(\Gamma)$, then $L^2_\mu(\Gamma) = P^2_\mu(\Gamma)$. Therefore, D defines a normal operator in ℓ^2, hence $\sigma(D) = \Gamma$ [5, 21]. Since

$$\sigma(D) \setminus \sigma_{ess}(D) = \{\lambda \mid \lambda \text{ isolated eigenvalue if finite multiplicity}\},$$

where $\sigma_{ess}(D)$ is the essential spectrum of D (see, for example, [6] for its definition), and the support is connected, then it has not isolated points, and $\Gamma = \sigma(D) = \sigma_{ess}(D)$.

Consider now $K = D - T$ which, by hypothesis is a compact operator. Then all its diagonals converge to 0 [1] and hence the limits

$$\lim_n d_{n-k,n} = d_{-k}, \quad k = -1, 0, 1, 2, \ldots$$

exist, and the matrix T is

$$T = \begin{pmatrix} d_0 & d_{-1} & d_{-2} & \ldots \\ d_1 & d_0 & d_{-1} & \ldots \\ 0 & d_1 & d_0 & \ldots \\ 0 & 0 & d_1 & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

 ISBN 13: 978-84-613-5510-5

Since the essential spectrum is invariant via compact perturbations [5], we have that $\sigma_{ess}(D) = \sigma_{ess}(T)$. Moreover, T is bounded in ℓ^2 and hence the rows and columns of T are in ℓ^2. Therefore, $(d_1, d_0, d_{-1}, d_{-2}, \ldots) \in \ell^2$.

The elements $d_{n,n-1}$ of the subdiagonal of the matrix D agree with the quotients γ_{n-1}/γ_n. Since $\lim_{n\to\infty} d_{n+1,n} = d_1$, then

$$d_1 = \lim_{n\to\infty} \frac{\gamma_{n-1}}{\gamma_n} = \lim_{n\to\infty} \frac{1}{\sqrt[n]{\gamma_n}}.$$

On the other hand, since μ is regular, then [19, p.100]

$$\lim_{n\to\infty} \frac{1}{\sqrt[n]{\gamma_n}} = \mathrm{cap}(\mathrm{supp}(\mu)).$$

Therefore, $d_1 = \mathrm{cap}(\mathrm{supp}(\mu))$.

Consider now the Laurent series

$$d(z) = d_1 z + d_0 + \frac{d_{-1}}{z} + \frac{d_{-2}}{z^2} + \cdots$$

We see now that the fact that $(d_1, d_0, d_{-1}, \ldots) \in \ell^2$ implies that $d(z)$ is analytic for every z such that $1 < |z| < \infty$.

If $|z| > 1$, then $\dfrac{1}{|z|} < 1$ and

$$\sum_{k=-1}^{\infty} |d_{-k} z^{-k}| \leq \sqrt{\sum_{k=-1}^{\infty} |d_{-k}|^2} \sqrt{\sum_{k=-1}^{\infty} |z^{-k}|^2} < +\infty.$$

Therefore $d(z)$ converges absolutely for every $1 < |z| < \infty$. To see that $d(z)$ is analytic we have just to show that $d'(z)$ exists for every $|z| > 1$. But

$$d\,'(z) = d_1 - \sum_{k=1}^{\infty} k \frac{d_{-k}}{z^{k+1}}$$

where

$$\sum_{k=1}^{\infty} k |\frac{d_{-k}}{z^{k+1}}| \leq \sqrt{\sum_{k=1}^{\infty} |d_{-k}|^2} \sqrt{\sum_{k=1}^{\infty} \frac{k^2}{|z|^{2k+2}}} < +\infty$$

if $|z| > 1$. Hence $d'(z)$ exists for every $|z| > 1$.

Since $(d_1, d_0, d_{-1}, \ldots) \in \ell^1$, then $(d|_{\mathbb{T}})(z)$ is continuous (where $\mathbb{T}$ is the unit circle) and [3, p.10]

$$\Gamma = \sigma_{ess}(T) = d(\mathbb{T}) = \{d_1 w + d_0 + \frac{d_{-1}}{w} + \frac{d_{-2}}{w^2} + \ldots \mid w \in \mathbb{T}\}.$$

We can now apply Theorem 1.1 in [14] to conclude that

$$d : \mathbb{C}_\infty \setminus \mathrm{cl}(\mathbb{D}) \to \mathbb{C}_\infty \setminus \Gamma,$$

 ISBN 13: 978-84-613-5510-5

(where $\mathbb{D}$ the unit disk) is an univalent map and, being also analytic, is conformal in $\mathbb{C}_\infty \setminus \mathrm{cl}(\mathbb{D})$.

Consider now the Riemann map

$$\phi(z) = c_1 w + c_0 + \frac{c_{-1}}{w} + \frac{c_2}{w^2} + \dots$$

in $\mathbb{C}_\infty \setminus \Gamma$ which is the unique conformal map which applies the exterior of the unit disk in the exterior of $\Gamma = \mathrm{supp}(\mu)$, which preserves the point at infinity and the direction therein, and which also satisfies $\mathrm{cap}(\Gamma) = c_1$ [20]. The map d satisfies that $d_1 = \mathrm{cap}(\Gamma)$. Moreover, since $d'(\infty) = \phi'(\infty) = d_1 = c_1$, then $d(z)$ preserves the point at infinity and the direction therein. Therefore $d = \phi$. □

3 Examples

As an illustration of the previous theorem we consider the following examples.

Example 1. Consider Γ the segment $[-1, 1]$ in $\mathbb{C}$. The Riemann map ϕ which applies the exterior of the unit disk in the exterior of Γ is

$$\phi(z) = \frac{1}{2}\left(z + \frac{1}{z}\right).$$

By Rakhmanov's theorem, if μ is a Borel measure is $[-1, 1]$ and $\mu' > 0$ almost everywhere in $[-1, 1]$, then $a_n \to \frac{1}{2}$ and $b_n \to 0$, where a_n are the sequences of elements in the subdiagonal and superdiagonal, and b_n are the sequences of elements in the diagonal, in the Jacobi matrix J associated with μ. Note that these are the coefficients of the Riemann map ϕ. Although Theorem 1 does not guarantee the existence of the limits of the diagonals of the Jacobi matrix in any case, in the case that those limits exist, they must agree with the coefficients of μ, even if μ is not absolutely continuous.

Example 2. Let Γ be a cross-like set, and μ the uniform measure on γ. The Riemann map is

$$\phi(z) = \frac{\sqrt{a^2(z^2+1)^2 + b^2(z^2-1)^2}}{2z},$$

where a and b are the length of the horizontal and vertical semi-axis, respectively. In the particular case of $a = b$,

$$\phi(z) = \frac{a\sqrt{2}}{2z}\sqrt{z^4+1}.$$

The series expansion of ϕ is

$$\phi(z) = \frac{\sqrt{a^2+b^2}}{2}\, z + \frac{-2\,b^2 + 2\,a^2}{4\,\sqrt{a^2+b^2}}\,\frac{1}{z} + \frac{\sqrt{a^2+b^2}\left(\frac{1}{2} - \frac{(-2\,b^2+2\,a^2)^2}{8\,(a^2+b^2)^2}\right)}{2\,z^3} + \mathrm{O}\left(\frac{1}{z^5}\right),$$

 ISBN 13: 978-84-613-5510-5

where the first coefficient $\dfrac{\sqrt{a^2+b^2}}{2}$ agrees with the capacity of the support. If $a = b$, the series expansion is

$$\phi(z) = \frac{a\sqrt{2}}{2}\, z + \frac{a\sqrt{2}}{4}\frac{1}{z^3} + \mathrm{O}\left(\frac{1}{z^5}\right).$$

The image under ϕ of the unit circle is shown in Figure 1, where we have included on the right the same result with an interpolation with less steps to give a better insight of the Riemann map.

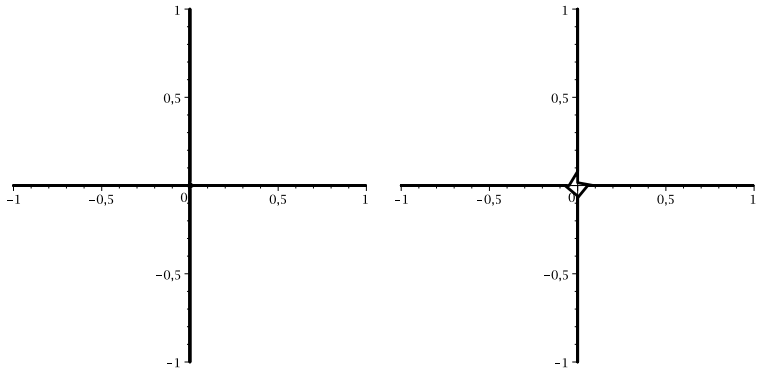


Figure 1: $\phi(\mathbb{T})$ for a cross-like set

There are many instances, however, when the Hessenberg matrix can not computed completely, but only finite sections of it, and it is not possible to compute the limits of the diagonals of D. In this case, it is still possible to compute approximations of the support of the measure μ obtained computing the image of the unit circle under suitable approximations of the Riemann map. Specifically, since the coefficients of the Riemann map are the limits of the elements in each of the diagonals of the Hessenberg matrix, we may consider, as approximations of the Riemann map ϕ, the functions

$$\phi_k(z) = d_{k,k-1}z + d_{k,k} + \sum_{i=1}^{k-1} \frac{d_{k-i,k}}{z^i},$$

where $D = (d_{i,j})$ is the Hessenberg matrix of μ [7].

The result of approximating $\mathrm{supp}(\mu)$ using this method, for $k = 30$, $k = 40$ and $k = 50$, respectively, is shown in Figure 1.

 ISBN 13: 978-84-613-5510-5

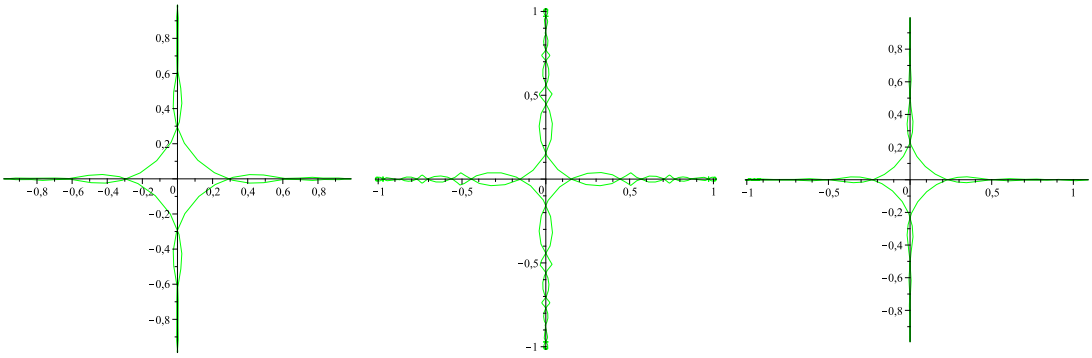


Figure 2: $\phi_k(\mathbb{T})$ for $k = 30$, $k = 40$ and $k = 50$, respectively

Example 3. Consider now Γ an arc of circumference. In this case [10] (see also [17, 18]), there exists a measure for which the diagonals of the Hessenberg matrix stabilize from the second element on. The monic orthogonal polynomials associated to this measure satisfy $\Phi_0(0) = 1$ and $\Phi_n(0) = \dfrac{1}{a}$ $(a > 1)$, if $n \geq 1$, and the corresponding Hessenberg matrix it the following unitary matrix:

$$D = \begin{pmatrix} -\frac{1}{a} & -\frac{(a^2-1)^{1/2}}{a^2} & -\frac{(a^2-1)^{2/2}}{a^3} & -\frac{(a^2-1)^{3/2}}{a^4} & -\frac{(a^2-1)^{4/2}}{a^5} & \cdots \\ \frac{(a^2-1)^{1/2}}{a} & -\frac{1}{a^2} & -\frac{(a^2-1)^{1/2}}{a^3} & -\frac{(a^2-1)^{2/2}}{a^4} & -\frac{(a^2-1)^{3/2}}{a^5} & \cdots \\ 0 & \frac{(a^2-1)^{1/2}}{a} & -\frac{1}{a^2} & -\frac{(a^2-1)^{1/2}}{a^3} & -\frac{(a^2-1)^{2/2}}{a^4} & \cdots \\ 0 & 0 & \frac{(a^2-1)^{1/2}}{a} & -\frac{1}{a^2} & \frac{(a^2-1)^{1/2}}{a^3} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Hence we know the limits of the diagonals, and we can obtain the sum of these limits. It is easy to check that $D - T$ is compact and that the rows of T are in ℓ^1, and hence the expression of the Riemann map is

$$\begin{aligned} \phi(z) &= \frac{z\left(a - \sqrt{a^2-1}\, z\right)}{\sqrt{a^2-1} - az} \\ &= \frac{\sqrt{a^2-1}}{a} z - \frac{1}{a^2} - \frac{\sqrt{a^2-1}}{a^3 z} - O\left(\frac{1}{z^2}\right), \end{aligned}$$

and we can compute the image under ϕ of the unit circle. The result is shown in Figure 1, where we have included on the right the same result with an interpolation with less steps to give a better insight of the Riemann map.

 ISBN 13: 978-84-613-5510-5

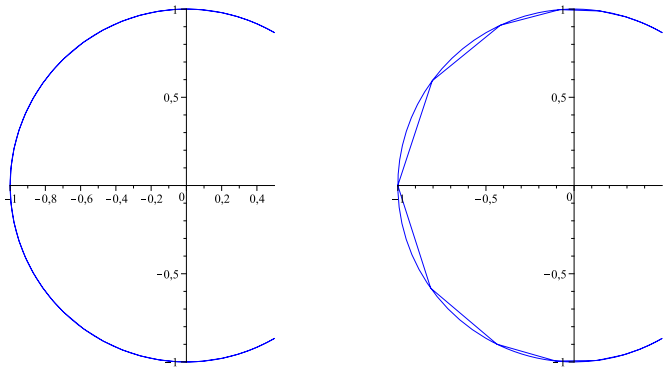


Figure 3: $\phi(\mathbb{T})$ for an arc of circumference

In the following figure we compute several approximations of the support of μ, for the particular case $a = 2$, using the above method, for $k = 10$, $k = 20$ and $k = 30$, respectively.

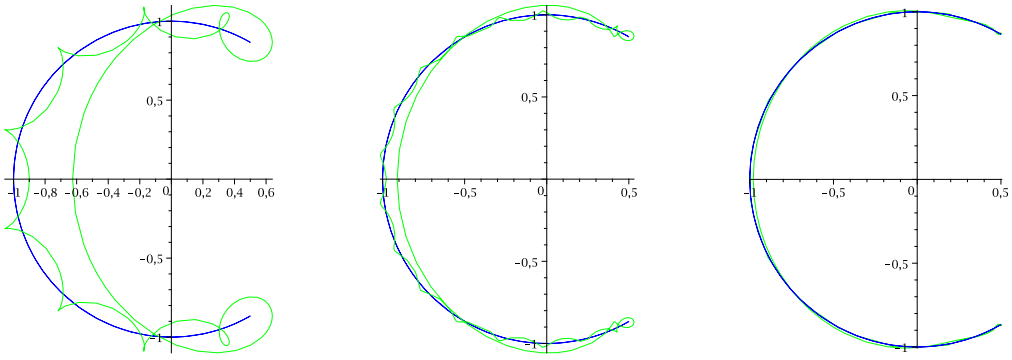


Figure 4: $\phi_k(\mathbb{T})$ for $k = 10$, $k = 20$ and $k = 30$, respectively

Example 4. In the following example we take Γ as the half part of a drop-like set of parametric equation

$$z(t) = \frac{(e^{it})^2}{1 + 2e^{it}}, t \in [0, \pi].$$

and μ the uniform measure on γ. In the following figure we show several approximations of the support of μ using this method, for $k = 5$, $k = 8$ and $k = 11$, respectively.

 ISBN 13: 978-84-613-5510-5

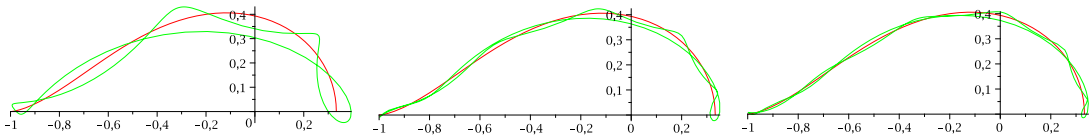


Figure 5: $\phi_k(\mathbb{T})$ for $k = 5$, $k = 8$ and $k = 11$, respectively

Example 5. For the last example we take Γ as the spiral with parametric equation

$$z(t) = t\frac{e^{it}}{6}, t \in [0, 2\pi]$$

and we consider μ the uniform measure on γ. In the following figure we show several approximations of the support of μ using this method, for $k = 1$ and $k = 12$, respectively.

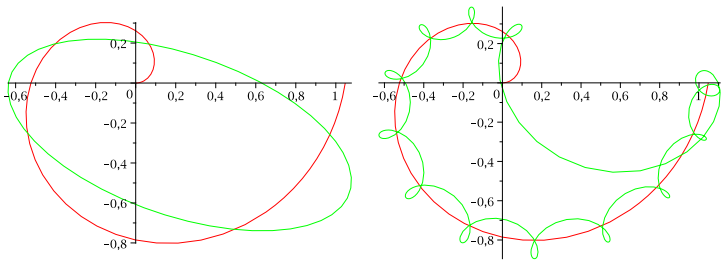


Figure 6: $\phi_k(\mathbb{T})$ for $k = 1$ and $k = 12$, respectively

Acknowledgements

The authors have been supported by Comunidad Autónoma de Madrid and Universidad Politécnica de Madrid (UPM-CAM Q061010133).

References

[1] N. I. Akhiezer and I. M. Glazman, *Theory of linear operators in Hilbert space*, Vol.I and II, Pitman, London, 1981.

 ISBN 13: 978-84-613-5510-5

[2] M. Bello and G. López, *Ratio and relative asymptotics of polynomials orthogonal on an arc of the unit circle*, J. Approx. Theory, 92 (1998) 216-244.

[3] A. Böttcher and S. M. Grudsky, *Spectral properties of banded Toeplitz matrices*, Siam, Philadelphia, 2005.

[4] T. S. Chihara, *An introduction to orthogonal polynomials*, Gordon and Breach, New York, 1978.

[5] J. B. Conway, *A course in functional analysis*, Graduate Texts in Mathematics, Springer-Verlag, New York, 1985.

[6] J. B. Conway, *The theory of subnormal operators*, Mathematical Surveys and Monographs, vol. **36**, AMS, Providence, Rhode Island, 1985.

[7] C. Escribano, A. Giraldo. M. A. Sastre y E. Torrano, *Approximation of Riemann maps for Jordan arcs*, Preprint.

[8] G. Freud, *Orthogonal polynomials*, Consultants Bureau, New York, 1961.

[9] D. Gaier, *Lectures on complex approximations*, Birkhäuser, Boston, 1985.

[10] L. Golinskii, P. Nevai and W. Van Assche, *Perturbation of orthogonal polynomials on an arc of the unit circle*, J. Approx. Theory **83** (3) (1995) 392–422.

[11] L. Golinskii and V. Totik, *Orthogonal polynomials: from Jacobi to Simon*, in *Spectral Theory and Mathematical Physics: A Festschrift in Honor of Barry Simon's 60th Birthday*, P. Deift, F. Gesztesy, P. Perry, and W. Schlag (eds.), Proceedings of Symposia in Pure Mathematics, **76**, Amer. Math. Soc., Providence, RI, 2007, pp. 821-874.

[12] P. R. Halmos *Ten problems in Hilbert space*. Bull. Amer. Math. Soc. **76** (5) (1970) 887–933.

[13] A. Máté, P. Nevai and V. Totik, *Asymptotics for the ratio of leading coefficients of orthonormal polynomials on the unit circle*, Constr. Approx. **1** (1985) 63-69.

[14] C. Pommerenke, *Univalent functions*, Vandenhoeck and Ruprecht in Göttingen, Studia Mathematica, 1975.

[15] E. A. Rakhmanov, On the asymptotics of the ratio of orthogonal polynomials, Math. USSR Sb. **32** (1977) 199–213.

[16] E. A Rakhmanov, On the asymptotics of the ratio of orthogonal polynomials. II, Math. USSR Sb. **47** (1983) 105–117.

[17] B. Simon, *Orthogonal polynomials on the unit circle, Part1: Classical Theory*, AMS Colloquium Publications, American Mathematical Society, Providence, RI, 2005.

 ISBN 13: 978-84-613-5510-5

[18] B. Simon, *Orthogonal polynomials on the unit circle, Part 2: Spectral Theory*, AMS Colloquium Publications, American Mathematical Society, Providence, RI, 2005

[19] H. Stahl and V. Totik, *General Orthogonal Polynomials*, Cambridge University Press, 1992.

[20] G. Szegö, *Orthogonal polynomials*, American Mathematical Society, Coloquium Publications, Vol. 32, first ed. 1939, fourth ed. 1975.

[21] V. Tomeo, *La subnormalidad de la matriz de Hessenberg asociada a los P.O. ortogonales en el caso hermitiano*, Tesis Doctoral, Madrid, 2004.

[22] W. V. Van Asche, *Rakhmanovs theorem for orthogonal matrix polynomials on the unit circle*, J. Approx. Th. **146** (2007) 227–242.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Positive Quadrature and Hyperinterpolation on the Rotation Group

Frank Filbir[1] **and Dominik Schmid**[1]

[1] *Institute of Biomathematics and Biometry,*
Helmholtz Zentrum München,
German Research Center for Environmental Health

emails: filbir@helmholtz-muenchen.de, dominik.schmid@helmholtz-muenchen.de

Abstract

We consider scattered data approximation problems on the rotation group $SO(3)$. More precisely, we provide sufficient conditions on the distribution of the scattered sampling points to guarantee the existence of positive quadrature formulas with respect to these points on the rotation group. These quadrature weights, in turn, are then used to define hyperinterpolation schemes that are exact for polynomials on $SO(3)$ up to a given degree.

Key words: rotation group, scattered data, quadrature rules, Wigner D-functions
MSC 2000: 41A30, 42C10, 42C15, 43A75, 65D32

1 Introduction

A typical problem in science is the development of a theoretical model for a hidden process from observational data. More precisely, we are given a set of measurements

$$\mathcal{D} = \{(\mathsf{x}_j, y_j) \in \mathbb{X} \times \mathbb{C} : j = 0, \dots, M-1\},$$

where we assume that the sampling points $\mathcal{X} = \{\mathsf{x}_j \in \mathbb{X} : j = 0, \dots, M-1\}$ are a finite subset of a metric space $(\mathbb{X}, d)$. We suppose that there exists an unknown function $f : \mathbb{X} \to \mathbb{C}$ that generated the observed data. This unknown function f is considered as a model for the underlying process and one is asked to construct an approximant ψ_f to f from the given data $\mathcal{D}$. In view of applications, we can hardly expect that the sampling nodes x_j are equally spaced or lie on a particular grid in $\mathbb{X}$. Hence, we are interested in interpolation respectively approximation procedures that are meshless, i.e. the methods should be applicable to arbitrary data sets $\mathcal{D}$ without any specific structure.

 ISBN 13: 978-84-613-5510-5

The case where the underlying structures are Euclidean spaces $\mathbb{R}^d$ or Euclidean spheres $\mathbb{S}^d$ has been studied in great detail over the last decade and there exists an enormous amount of research papers dealing with scattered data approximation problems in these classical settings. We refer to [7, 23] and all the references therein.

However, in various applications we are confronted with the situation that the underlying set is a compact or locally compact possibly non-Abelian group and one is asked to propose suitable approximation procedures on these structures. Such problems arise in biochemistry, crystallography and robotics to name only a few. The monograph [2] provides a great collection of scattered data approximation problems where different matrix groups are involved. In view of applications, the rotation group $SO(3)$ is, without doubt, one of the most important groups in this regard.

A typical approximation problem on $SO(3)$ that has a number of applications, for instance, in molecular biology [4] and 3-D shape-matching [14] is the correlation of two functions defined on the Euclidean sphere $\mathbb{S}^2$. More precisely, given two functions $f, g \in L^2(\mathbb{S}^2)$ and the knowledge that g is the rotated version of f, one is asked to determine $\boldsymbol{x} \in SO(3)$ such that $f(\xi) = g(\boldsymbol{x}^{-1}\xi)$ for all $\xi \in \mathbb{S}^2$. By the Cauchy-Schwarz inequality this can be accomplied by considering the associated correlation function

$$F : SO(3) \to \mathbb{C}, \qquad F(\boldsymbol{x}) := \int_{\mathbb{S}^2} f(\xi)\overline{g(\boldsymbol{x}^{-1}\xi)}\mathrm{d}\xi$$

and finding the $\boldsymbol{x} \in SO(3)$ that maximizes F.

Of course, the correlation of two functions defined on the Euclidean sphere $\mathbb{S}^2$ is not the only approximation problem in science and engineering where the rotation group plays an important role. There exist a wealth of scattered data approximation problems on $SO(3)$, cf. [1, 3, 19, 24]. Such problems have attracted significant recent attention from the mathematical community, which has investigated approximation via Wigner D-functions [9, 10, 15, 17, 20], and kernels [5, 11, 12]. In this article, we focus on approximation procedures that use finite expansions into Wigner D-functions, subsequently called polynomials on $SO(3)$, which constitute an orthogonal basis of the $L^2(SO(3))$. More precisely, we provide sufficient conditions on the distribution of the sampling points to guarantee the existence of positive quadrature formulas that are exact for polynomials on $SO(3)$ up to a given degree. Of course, such quadrature formulas can be applied in various ways and fields. As one important application, we utilize the quadrature weights in order to establish so-called hyperinterpolation schemes that are a powerful tool for the approximation of scattered data on the rotation group $SO(3)$.

The article is organized as follows. In the next section we give the necessary background on analysis and sampling on $SO(3)$ to keep the paper self-contained. We also present a preliminary result that is of central importance for our considerations - Marcinkiewicz-Zygmund inequalities from scattered data on $SO(3)$. Finally, in Section 3 we derive our main results.

 ISBN 13: 978-84-613-5510-5

2 Background

2.1 Analysis on the Rotation Group

Let $SO(3) := \{\boldsymbol{x} \in \mathbb{R}^{3\times 3} : \boldsymbol{x}^T\boldsymbol{x} = \boldsymbol{I}, \det \boldsymbol{x} = 1\}$ denote the non-Abelian compact group of proper rotations in the Euclidean space $\mathbb{R}^3$ and let μ be the normalized Haar measure on $SO(3)$, i.e. we have $\int_{SO(3)} \mathrm{d}\mu(\boldsymbol{x}) = 1$. The Hilbert space $L^2(SO(3))$ of all square integrable functions is determined by the scalar product with induced norm

$$\langle f, g\rangle := \int_{SO(3)} f(\boldsymbol{x})\overline{g(\boldsymbol{x})}\mathrm{d}\mu(\boldsymbol{x}), \quad \|f\|_2^2 := \int_{SO(3)} |f(\boldsymbol{x})|^2\mathrm{d}\mu(\boldsymbol{x}).$$

In order to get an orthogonal basis system for the $L^2(SO(3))$ we make use of some fundamental results from the representation theory of this non-Abelian compact group. Let $\{Y_k^l : l \in \mathbb{N}_0,\ k = -l, \dots, l\}$ denote the canonical orthonormal basis of spherical harmonics on the space of all square integrable functions on the unit sphere $\mathbb{S}^2 \subset \mathbb{R}^3$ and let $\mathcal{H}_l := \mathrm{span}\{Y_k^l : k = -l, \dots, l\}$. For given $l \in \mathbb{N}_0$ we assign each element $\boldsymbol{x} \in SO(3)$ the linear transformation $D^l(\boldsymbol{x}) : \mathcal{H}_l \to \mathcal{H}_l$ defined by

$$D^l(\boldsymbol{x})Y(\xi) := Y(\boldsymbol{x}^{-1}\xi), \quad Y \in \mathcal{H}_l,\ \xi \in \mathbb{S}^2.$$

Each $D^l, l \in \mathbb{N}_0$, can be written as a $(2l+1)\times(2l+1)$- matrix with matrix coefficients defined by the following system of linear equations

$$Y_k^l(\boldsymbol{x}^{-1}\xi) = \sum_{k'=-l}^{l} D_{k,k'}^l(\boldsymbol{x})Y_{k'}^l(\xi), \quad k = -l, \dots, l,\ \xi \in \mathbb{S}^2.$$

The functions $D_{k,k'}^l$ are usually called Wigner D-functions of degree l and orders k and k'. It is well-known that the $D^l, l \in \mathbb{N}_0$, form a complete set of unitary irreducible representations of the rotation group and due to the Peter-Weyl Theorem the matrix coefficients $D_{k,k'}^l$ form an orthogonal basis of the $L^2(SO(3))$, see [8]. Hence, every $f \in L^2(SO(3))$ can be expanded in a $SO(3)$ Fourier series

$$f = \sum_{l\in\mathbb{N}_0} \sum_{k,k'=-l}^{l} \sqrt{2l+1}\, \hat{f}_{k,k'}^l D_{k,k'}^l$$

with $SO(3)$ Fourier coefficients $\hat{f}_{k,k'}^l = \sqrt{2l+1}\cdot\langle f, D_{k,k'}^l\rangle$. We call functions with finite Fourier expansion polynomials on $SO(3)$ and we define the space of polynomials on $SO(3)$ with degree at most n by

$$\boldsymbol{\Pi}_n := \mathrm{span}\left\{D_{k,k'}^l : l = 0, \dots, n;\ k, k' = -l, \dots, l\right\}.$$

The spaces $\boldsymbol{\Pi}_n$, indeed, admit a polynomial behavior, i.e. for $P_1 \in \boldsymbol{\Pi}_{n_1}$ and $P_2 \in \boldsymbol{\Pi}_{n_2}$ we have, cf. [21, Eq. (3.54)],

$$P_1 \cdot P_2 \in \boldsymbol{\Pi}_{n_1+n_2}.$$

 ISBN 13: 978-84-613-5510-5

2.2 Sampling data and Marcinkiewicz-Zygmund Inequalities

There are various ways to parameterize the rotation group. In this article the parameterization via the projective space is of significant importance to us. This parameterization yields a translation invariant metric on $SO(3)$ which enables us to quantify different sampling sets on the rotation group.

Let $\mathcal{K}_\pi$ be the closed ball in $\mathbb{R}^3$ of radius π centered at the origin and identify antipodal points on its surface. This is the three dimensional projective space. An element $\boldsymbol{x} \in SO(3)$ is identified with a point in the projective space $\mathcal{K}_\pi$ by $\boldsymbol{x} \to \omega \cdot r$ where r, satisfying $\boldsymbol{x}r = r$ and $\|r\| = 1$, is the rotation axis and ω, which can be chosen in $[0, \pi]$, is the rotation angle of $\boldsymbol{x}$.

Then it is easy to see that

$$d(\boldsymbol{x}, \boldsymbol{y}) := \omega(\boldsymbol{y}^{-1}\boldsymbol{x})$$

defines a translation invariant metric on $SO(3)$. As already mentioned earlier, in view of applications, we are mainly interested in situations where the sampling points

$$\mathcal{X} := \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$$

are scattered on $SO(3)$, i.e. the sampling points are not located on a particular grid on $SO(3)$. In order to compensate for clusters in the sampling set $\mathcal{X}$, it is reasonable to weight the sampling nodes $\boldsymbol{x}_j \in \mathcal{X}$. To this end, we introduce for a given sampling set $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$ an associated partition

$$\mathcal{R} := \{\Omega_j \subset SO(3) : j = 0, \dots, M-1\}$$

of $SO(3)$, i.e. $\mathcal{R}$ is a collection of M closed regions $\Omega_j \subset SO(3)$, having no common interior points and covering the whole rotation group, i.e. $\bigcup_{j=0}^{M-1} \Omega_j = SO(3)$. Moreover, we require that $\boldsymbol{x}_j$ is an interior point of Ω_j for all $j = 0, \dots, M-1$. With respect to the partition $\mathcal{R}$ we now define the corresponding weights by

$$\boldsymbol{w} := (w_0, \dots, w_{M-1})^T \in \mathbb{R}^M, \quad w_j := \int_{\Omega_j} \mathrm{d}\mu(\boldsymbol{x}) = \mu(\Omega_j).$$

Finally, the partition norm $\|\mathcal{R}\|$ of the partition $\mathcal{R}$ is given by

$$\|\mathcal{R}\| := \max_{j=0,\dots,M-1} \operatorname{diam} \Omega_j := \max_{j=0,\dots,M-1} \max_{\boldsymbol{x},\boldsymbol{y} \in \Omega_j} d(\boldsymbol{x}, \boldsymbol{y}).$$

Remark 2.1 *Given a sampling set $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$ of scattered points on $SO(3)$, there are many possibilities to construct an associated partition $\mathcal{R}$. In many situations, the so-called Voronoi partition $\mathcal{R}^V$, which is determined by*

$$\Omega_j^V := \{\boldsymbol{y} \in SO(3) : d(\boldsymbol{x}_j, \boldsymbol{y}) = \min_{k=0,\dots,M-1} d(\boldsymbol{x}_k, \boldsymbol{y})\}, \quad j = 0, \dots, M-1,$$

is a reasonable choice. Using the Voronoi partition corresponding to a given sampling set $\mathcal{X} \subset SO(3)$, we easily obtain $h_{\mathcal{X}} \le \|\mathcal{R}^V\| \le 2h_{\mathcal{X}}$, where $h_{\mathcal{X}} := \max_{\boldsymbol{y}} \min_j d(\boldsymbol{y}, \boldsymbol{x}_j)$

 ISBN 13: 978-84-613-5510-5

is the mesh norm of the sampling set $\mathcal{X}$. This relation, in turn, makes it possible by using the Voronoi partition to state all the following results that depend on the partition norm of the underlying partition in terms of the mesh norm $h_{\mathcal{X}}$ of the given sampling set.

Next we define the discrete spaces $\ell^p_{\boldsymbol{w}}(SO(3))$, $1 \leq p \leq \infty$, corresponding to the sampling set $\mathcal{X}$ with associated partition $\mathcal{R}$ in the usual manner with norm

$$\|f\|_{\boldsymbol{w},p} := \begin{cases} \left(\sum_{j=0}^{M-1} w_j |f(\boldsymbol{x}_j)|^p \right)^{1/p} & 1 \leq p < \infty, \\ \sup_{j=0,\dots,M-1} |f(\boldsymbol{x}_j)| & p = \infty. \end{cases}$$

In order to develop our results in the next section, it is of fundamental importance to have norm equivalences between the continuous L^p-norm of a polynomial P on $SO(3)$ and the weighted discrete norm $\|\cdot\|_{\boldsymbol{w},p}$ of the vector of samples of P at the scattered sampling points $\boldsymbol{x}_j$. In other words, we want to get bounds on the norm of the corresponding sampling operator and its inverse on the space $\mathbf{\Pi}_n$. In the mathematical community this kind of norm equivalences are usually called Marcinkiewicz-Zygmund inequalities. Recently in [20, 21] such inequalities from scattered data were shown for polynomials on the rotation group. We are particularly interested in the L^1- and L^∞-Marcinkiewicz-Zygmund inequality, cf. [20, Theorem 4.4] and [21, Theorem 4.22].

Theorem 2.2 *Let $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$ be a sampling set and $\mathcal{R} = \{\Omega_j \subset SO(3) : j = 0, \dots, M-1\}$ be an associated partition. If $n \in \mathbb{N}_0$ with $n \leq \frac{1}{\|\mathcal{R}\|}$, then we have for any polynomial $P \in \mathbf{\Pi}_n$*

$$\big(1 - n\|\mathcal{R}\|\big) \cdot \|P\|_\infty \leq \|P\|_{\boldsymbol{w},\infty} \leq \big(1 + n\|\mathcal{R}\|\big) \cdot \|P\|_\infty. \tag{1}$$

Furthermore, if $n \in \mathbb{N}_0$ with $n \leq \frac{1}{462\|\mathcal{R}\|}$, then we have for any polynomial $P \in \mathbf{\Pi}_n$

$$\big(1 - 462n\|\mathcal{R}\|\big) \cdot \|P\|_1 \leq \|P\|_{\boldsymbol{w},1} \leq \big(1 + 462n\|\mathcal{R}\|\big) \cdot \|P\|_1. \tag{2}$$

3 Positive Quadrature Formulas and Hyperinterpolation

In this section we present the main results of the present paper. In order to show the existence of positive quadrature formulas on the rotation group, we pick up a basic idea from [16] where spherical Marcinkiewicz-Zygmund inequalities have been used in order to prove a similar result on Euclidean spheres. These quadrature weights, in turn, can then be used to define hyperinterpolation schemes that are exact for polynomials on $SO(3)$ up to a given degree.

For the sake of clarity we left out the technical proofs of the results of this section. However, the interested reader can find all these details in [21, Chapter 4]. First, we need to introduce the notion of a norm generating set, which was also central to the arguments in [13] and [16].

 ISBN 13: 978-84-613-5510-5

Definition 3.1 *Let V be a finite-dimensional vector space with norm $\|\cdot\|_V$, and let $\mathcal{Z} \subset V^*$ be a finite set. We say that $\mathcal{Z}$ is a norm generating set for V if the mapping $T_{\mathcal{Z}} : V \to \mathbb{R}^{|\mathcal{Z}|}$ defined by $T_{\mathcal{Z}}(v) = (z(v))_{z\in\mathcal{Z}}$ is injective.*

Let $W := T_{\mathcal{Z}}(V)$ be the range of $T_{\mathcal{Z}}$. We remark that if $\mathcal{Z}$ is a norm generating set for V, we have by the injectivity of $T_{\mathcal{Z}}$ that $T_{\mathcal{Z}}^{-1} : W \to V$ exists. Now using the Hahn-Banach Theorem and the Krein-Rutman Theorem we get the following important very general result, see [16, Proposition 4.1].

Proposition 3.2 *Let $\mathcal{Z}$ be a norm generating set for V, with $T_{\mathcal{Z}}$ defined as in Definition* 3.1. *Let $\mathbb{R}^{|\mathcal{Z}|}$ have a norm $\|\cdot\|_{\mathbb{R}^{|\mathcal{Z}|}}$, with $\|\cdot\|_{\mathbb{R}^{|\mathcal{Z}|^*}}$ being its dual norm on $\mathbb{R}^{|\mathcal{Z}|^*}$. Furthermore, we equip $W \subset \mathbb{R}^{|\mathcal{Z}|}$ with the induced norm and let $\|T_{\mathcal{Z}}^{-1}\| := \|T_{\mathcal{Z}}^{-1}\|_{W\to V}$. By $\mathcal{C}_+$ we denote the positive cone of $\mathbb{R}^{|\mathcal{Z}|}$, i.e. all $(r_z) \in \mathbb{R}^{|\mathcal{Z}|}$ for which $r_z \geq 0$ for all $z \in \mathcal{Z}$.*
If $y \in V^*$ with $\|y\|_{V^*} \leq C$, then there exist real numbers $\{a_z\}_{z\in\mathcal{Z}}$ depending only on y such that for every $v \in V$,

$$y(v) = \sum_{z\in\mathcal{Z}} a_z z(v). \tag{3}$$

Moreover, if W contains an interior point $w_0 \in \mathcal{C}_+$ and if $y(T_{\mathcal{Z}}^{-1}w) \geq 0$ whenever $w \in W \cap \mathcal{C}_+$, then we may choose each $a_z \geq 0$ in (3).

In order to prove the existence of positive quadrature formulas that are exact for all polynomials on $SO(3)$ up to a given degree n we want to apply Proposition 3.2 to the linear functional $y \in \mathbf{\Pi}_n^*$ defined by

$$y(P) = \int_{SO(3)} P(\boldsymbol{x})\mathrm{d}\mu(\boldsymbol{x}).$$

That means have to guarantee, on the one hand, that the sampling operator

$$\mathcal{S}(P) := \big((P(\boldsymbol{x}_1), \ldots, P(\boldsymbol{x}_{M-1})\big)$$

is injective on $\mathbf{\Pi}_n$ and, on the other hand, that the cone conditions are satisfied. But these statements are now easy consequences of the L^∞- and the L^1-Marcinkiewicz-Zygmund inequality. Firstly, the L^∞-Marcinkiewicz-Zygmund (1) ensures that the sampling operator is injective on $\mathbf{\Pi}_n$ whenever $n < \frac{1}{\|\mathcal{R}\|}$. Secondly, we can utilize the L^1-Marcinkiewicz-Zygmund inequality (2) to show that the cone condition are satisfied whenever $n \leq \frac{1}{924\|\mathcal{R}\|}$. Altogether, we get the following theorem which gives a sufficient condition for the existence of positive quadrature formulas on the rotation group $SO(3)$.

Theorem 3.3 *Let $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \ldots, M-1\}$ be a sampling set and $\mathcal{R} = \{\Omega_j \subset SO(3) : j = 0, \ldots, M-1\}$ an associated partition of $SO(3)$. If $n \leq \frac{1}{924\|\mathcal{R}\|}$, then there exist nonnegative real numbers $\{\alpha_j\}_{j=0,\ldots,M-1}$ such that for every $P \in \mathbf{\Pi}_n$ we have*

$$\int_{SO(3)} P(\boldsymbol{x})\mathrm{d}\mu(\boldsymbol{x}) = \sum_{j=0}^{M-1} \alpha_j P(\boldsymbol{x}_j). \tag{4}$$

 ISBN 13: 978-84-613-5510-5

If we have $\frac{1}{924\|\mathcal{R}\|} \leq n < \frac{1}{\|\mathcal{R}\|}$, *we can still find real coefficients* α_j *such that* (4) *holds. However, in this case these coefficients are no longer guaranteed to be nonnegative.*

Remark 3.4 *We point out that essentially we need* $\|\mathcal{R}\| \sim n^{-1}$ *for the quadrature formula to hold. On the other hand, we can find sampling sets and associated partitions of cardinality* M *such that* $M^{-1/3} \sim \|\mathcal{R}\|$. *Hence, for such partitions we have* $M \sim n^3$ *which is the order of the dimension of the space that we are exactly integrating.*

Next, let us use Theorem 3.3 to define so-called hyperinterpolation schemes on $SO(3)$. Hyperinterpolation of multivariate continuous functions on compact subsets or manifolds in $\mathbb{R}^d$ was originally introduced by Sloan in the seminal paper [22] and subsequently studied by several authors, see the monograph [18] and the references therein. In [6], the authors utilized spherical quadrature formulas in order to define hyperinterpolation schemes on Euclidean spheres. In what follows, we seek to establish similar schemes on the rotation group $SO(3)$. More precisely, we want to discretize a "nice" linear convolution operator

$$\mathcal{V}_n f(\boldsymbol{x}) := \int_{SO(3)} f(\boldsymbol{y}) v_n(\boldsymbol{y}^{-1}\boldsymbol{x}) d\mu(\boldsymbol{y}), \tag{5}$$

in order to get an approximation procedure that shares some approximation properties with its continuous counterpart and can be computed extremely fast. To this end, let us suppose that for $n \in \mathbb{N}_0$ we are given a convolution kernel $v_n : SO(3) \to \mathbb{R}$ with the following properties

(i) $v_n \in \mathbf{\Pi}_{m\cdot n}$ for some $m \in \mathbb{N}$,

(ii) $P(\boldsymbol{x}) = \int_{SO(3)} P(\boldsymbol{y}) v_n(\boldsymbol{y}^{-1}\boldsymbol{x}) d\mu(\boldsymbol{y})$ for all $P \in \mathbf{\Pi}_n$,

(iii) $\sup_{n\in\mathbb{N}_0} \|v_n\|_1 \leq c$ for some $c > 0$.

A family of convolution kernels possessing properties (i)-(iii), indeed, are good for approximation purposes on the rotation group. For example, let us consider the error of best polynomial approximation on $SO(3)$ given by

$$\mathbb{E}_n(f)_p := \inf_{P\in\mathbf{\Pi}_n} \|f - P\|_p. \tag{6}$$

Then the associated linear operator $\mathcal{V}_n$ (5) provides a "near best" polynomial approximation in the following sense.

Proposition 3.5 *For all* $n \in \mathbb{N}_0$ *and any* $1 \leq p \leq \infty$ *we have*

$$\mathbb{E}_{m\cdot n}(f)_p \leq \|f - \mathcal{V}_n f\|_p \leq (1 + c)\mathbb{E}_n(f)_p$$

for all $f \in L^p(SO(3))$.

 ISBN 13: 978-84-613-5510-5

Remark 3.6 *We would like to point out that in [20] a family of convolution kernels* $\{v_n\}_{n\in\mathbb{N}_0}$ *on the rotation group* $SO(3)$ *possessing properties (i)-(iii) with constants* $m = 2$ *and* $c = \sqrt{27}$ *has been explicitly constructed.*

In the next step we want to discretize the "nice" convolution operator $\mathcal{V}_n$ using the positive quadrature weights from Theorem 3.3. To this end, let $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$ be a sampling set on the rotation group with associated partition $\mathcal{R} = \{\Omega_j \subset SO(3) : j = 0, \dots, M-1\}$. Furthermore, let $n \in \mathbb{N}_0$ be such that

$$(1+m)n \leq \frac{1}{924\|\mathcal{R}\|}.$$

Then Theorem 3.3 guarantees the existence of nonnegative real numbers β_j such that

$$\int_{SO(3)} P(\boldsymbol{x})\mathrm{d}\mu(\boldsymbol{x}) = \sum_{j=0}^{M-1} \beta_j P(\boldsymbol{x}_j)$$

for all polynomials $P \in \mathbf{\Pi}_{(1+m)n}$. By means of the weights β_j, we define the discrete operator $\widetilde{\mathcal{V}}_n : C(SO(3)) \to C(SO(3))$ given by

$$\widetilde{\mathcal{V}}_n f(\boldsymbol{x}) := \sum_{j=0}^{M-1} \beta_j f(\boldsymbol{x}_j) v_n(\boldsymbol{x}_j^{-1}\boldsymbol{x}).$$

The discrete operator $\widetilde{\mathcal{V}}_n$ shares some nice approximation properties with its continuous counterpart $\mathcal{V}_n$. Let us collect these results in the following theorem. Again, $\mathbb{E}_n(f)_\infty$ denotes the error of best polynomial approximation as defined in (6).

Theorem 3.7 *Let* $\mathcal{X} = \{\boldsymbol{x}_j \in SO(3) : j = 0, \dots, M-1\}$ *be a sampling set and* $\mathcal{R} = \{\Omega_j \subset SO(3) : j = 0, \dots, M-1\}$ *be an associated partition. If* $n \in \mathbb{N}_0$ *with* $(1+m)n \leq \frac{1}{924\|\mathcal{R}\|}$, *then there exist nonnegative real numbers* $\beta_j, j = 0, \dots, M-1$, *such that the operator* $\widetilde{\mathcal{V}}_n : C(SO(3)) \to C(SO(3))$ *defined by*

$$\widetilde{\mathcal{V}}_n f(\boldsymbol{x}) = \sum_{j=0}^{M-1} \beta_j f(\boldsymbol{x}_j) v_n(\boldsymbol{x}_j^{-1}\boldsymbol{x})$$

satisfies the following statements.

(i) $\widetilde{\mathcal{V}}_n P = P$ *for all* $P \in \mathbf{\Pi}_n$,

(ii) $\|\widetilde{\mathcal{V}}_n f\|_\infty \leq \Lambda \|f\|_{\boldsymbol{w},\infty} \leq \Lambda \|f\|_\infty$ *for all* $f \in C(SO(3))$ *and*

(iii) $\mathbb{E}_{m\cdot n}(f)_\infty \leq \|f - \widetilde{\mathcal{V}}_n f\|_\infty \leq (1+\Lambda)\mathbb{E}_n(f)_\infty$, *where* $\Lambda = \max\limits_{\boldsymbol{x}\in SO(3)} \sum\limits_{j=0}^{M-1} \beta_j |v_n(\boldsymbol{x}_j^{-1}\boldsymbol{x})|$.

Moreover, we obtain an upper bound on Λ *as follows. It is*

(iv) $\Lambda \leq \frac{3}{2}\sqrt{27}\,\sigma$, *where* $\sigma = \max\left\{\frac{\beta_j}{w_j} : j = 0, \dots, M-1\right\}$.

 ISBN 13: 978-84-613-5510-5

By means of the developed theory in this section, we were able to define so-called hyperinterpolation schemes on $SO(3)$, which are exact for polynomials up to a given degree n. Once we have computed the quadrature weights β_j, which can be done, for example, by convex optimization [10], we only have to evaluate the values $v_n(\boldsymbol{x}_j^{-1}\boldsymbol{x})$ in order to get the approximant $\widetilde{\mathcal{V}}_n f$ at the point $\boldsymbol{x} \in SO(3)$. If we consider the concrete example of the convolution operator constructed in [20], we even have a closed form expression for the kernel v_n and therefore this can be done extremely fast.

Acknowledgements

This work has been partially supported by Deutsche Forschungsgemeinschaft Grant FI 883/3-1 and PO 711/9-1.

References

[1] K. G. van den Boogaart, R. Hielscher, J. Prestin and H. Schaeben, *Kernel-based methods for inversion of the Radon transform on SO(3) and their applications to texture analysis*. J. Comput. Appl. Math. **199** (2007) 122–140.

[2] G. S. Chirikjian and A. B. Kyatkin, *Engineering Applications of Noncommutative Harmonic Analysis*, CRC Press, Boca Raton, 2000.

[3] J. E. Castrillon-Candas and V. Siddavanahalli, C. Bajaj, *Nonequispaced Fourier transforms for protein-protein docking*, ICES Report 05–44, University of Texas at Austin, 2005.

[4] R. A. Crowther, *The fast rotation function*, In: M. G. Rossman (ed.), The Molecular Replacement Method, Gordon and Breach, New York, pages 173–178, 1972.

[5] F. Filbir and D. Schmid, *Stability results for approximation by positive definite functions on SO*(3), J. Approx. Theory **153** (2008) 170–183.

[6] F. Filbir and W. Themistoclakis, *Polynomial approximation on the sphere using scattered data*, Math. Nachr. **281** (2008) 650–668.

[7] W. Freeden, T. Gervens and M. Schreiner, *Constructive Approximation on The Sphere. With Applications to Geomathematics*, Clarendon Press, Oxford, 2000.

[8] I. M. Gelfand, R. A. Minlos and Z. Ya. Shapiro, *Representations of the Rotation and Lorentz Groups and Their Applications*, Pergamon Press, Oxford, 1963.

[9] M. Gräf and S. Kunis, *Stability results for scattered data interpolation on the rotation group*, Electron. Trans. Numer. Anal. **31** (2008) 30–39.

 ISBN 13: 978-84-613-5510-5

[10] M. Gräf and D. Potts, *Sampling sets and quadrature formulae on the rotation group*, Numer. Funct. Anal. Optim. **30** (2009) 665 – 688.

[11] T. Gutzmer, *Interpolation by positive definite functions on locally compact groups with application to* SO(3), Results Math. **29** (1996) 69–77.

[12] T. Hangelbroek and D. Schmid, *Surface spline approximation on $SO(3)$*, submitted, arXiv:0911.1836v1.

[13] K. Jetter, J. Stöckler and J. D. Ward, *Error estimates for scattered data interpolation on spheres*, Math. Comp. **68** (1999) 733–747.

[14] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz, *Rotation invariant spherical harmonic representation of 3D shape descriptors*, In: L. Kobbelt, P. Schröder, H. Hoppe (eds.), Eurographics Symposium in Geometry Processing, pages 167–175, 2003.

[15] P. J. Kostelec and D. N. Rockmore, *FFTs on the rotation group*, J. Fourier Anal. Appl. **14** (2008) 145–179.

[16] H. N. Mhaskar, F. J. Narcowich and J. D. Ward, *Spherical Marcinkiewicz-Zygmund inequalities and positive quadrature*, Math. Comp. **70** (2001) 1113–1130.

[17] D. Potts, J. Prestin and A. Vollrath, *A Fast algorithm for nonequispaced Fourier transforms on the rotation group*, Numer. Algorithms **52** (2009) 355 – 384.

[18] M. Reimer *Multivariate Polynomial Approximation*, Birkhäuser Verlag, Basel, 2003.

[19] H. Schaeben, R. Hielscher, J. J. Fundenberger, D. Potts and J. Prestin, *Orientation density function-controlled pole probability density function measurements: automated adaptive control of texture goniometers*, J. Appl. Cryst. **40** (2007) 570–579.

[20] D. Schmid, *Marcinkiewicz-Zygmund inequalities and polynomial approximation from scattered data on $SO(3)$*, Numer. Funct. Anal. Optim. **29** (2008) 855–882.

[21] D. Schmid, *Scattered Data Approximation on the Rotation Group and Generalizations*, PhD Thesis, Technische Universität München, 2009.

[22] I. H. Sloan, *Polynomial interpolation and hyperinterpolation over general regions*, J. Approx. Theory **83** (1995) 238–254.

[23] H. Wendland, *Scattered Data Approximation*, Cambridge University Press, Cambridge, 2005.

[24] A. Yershova and S. M. LaValle, *Deterministic sampling methods for spheres and $SO(3)$*, In: Proceedings. ICRA 04. IEEE International Conference on Robotics and Automation, Volume 4, New Orleans, pages 3974–3980, 2004.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Pseudospectral methods for stochastic partial differential equations with additive white noise

Rafael Gallego[1]

[1] *Department of Mathematics, university of Oviedo*

emails: rgallego@uniovi.es

Abstract

Commonly used finite-difference numerical schemes show some deficiencies in the integration of certain types of stochastic partial differential equations with additive white noise. In this work efficient spectral schemes to integrate these equations are discussed. They all are based on the discretization of the system in Fourier space. The nonlinear terms are treated using a pseudospectral approach so as to speed up the computations without a significant loss of accuracy. The proposed schemes are applied to solve, both in one and two spatial dimensions, two paradigmatic models arising in the context of nonequilibrium dynamics of growing interfaces: the continuum Kardar-Parisi-Zhang and Lai-Das Sarma-Villain equations.

Key words: pseudospectral methods, stochastic partial differential equations, growth models, stochastic PDEs, Fourier transform

1 Introduction

Numerical integration is often a direct and convenient way to study the behavior of partial differential equations (PDEs). In the particular case of stochastic partial differential equations (SPDEs), finite-difference methods have been traditionally used. Spectral methods, on the other hand, are frequently used to integrate PDEs arising in Fluid Dynamics but only recently they have been considered for SPDEs [7, 9]. Spectral schemes are in general more reliable and accurate that finite-difference based algorithms since the latters may give rise to numerical artifacts due to discretization effects. These artifacts may lead to a misleading interpretation of the results of numerical simulations. Moreover, it has been proved that the results of numerical simulations can depend on the particular discretization used to evaluate the partial derivatives [16]. This unwanted feature is not likely to be present with spectral methods inasmuch as derivatives are computed in Fourier space using the values of the field at all points. On the other hand, spectral methods may be rather time-consuming if complex nonlinear terms are

 ISBN 13: 978-84-613-5510-5

involved. To speed up the computation, a pseudospectral approach is commonly used to deal with the nonlinear terms.

The problem of kinetic surface roughening has attracted much attention in the last years owing to its many important applications as molecular beam epitaxy, fluid flow in porous media, fracture cracks, etc. Theoretical approaches make use of both discrete atomistic simulations and stochastic continuum equations for the evolution of the coarse-grained surface height $h(\mathbf{x}, t)$. Growth models are often classified into universality classes according to the values of certain critical exponents that characterize the growth process and that do not depend on the microscopic details of the system under study.

Universality classes are generically represented by SPDEs. The Kardar-Parisi-Zhang (KPZ) [11] and the Lai-Das Sarma-Villain equations (LDV) [15, 23, 24] are examples of continuum nonlinear growth models. Both equations can be derived based on physical and symmetry principles. The KPZ equation is a Langevin type equation that was first introduced to give a hydrodynamic description of ballistic deposition growth far from equilibrium. If h is the local height on a d-dimensional substrate, the KPZ equation reads:

$$\partial_t h(x,t) = \nabla^2 h + g(\nabla h)^2 + \eta(x,t), \tag{1}$$

where g is the so-called nonlinear coupling parameter. The stochastic term $\eta(x,t)$ represents the influx of atoms on the surface. It is a Gaussian noise with mean zero and uncorrelated in space and time, that is,

$$\langle \eta(x,t) \rangle = 0, \qquad \langle \eta(x,t)\eta(x',t') \rangle = \delta^d(x-x')\delta(t-t').$$

The LDV equation describes the long-wavelength fluctuations of crystal growth from atom beams in the absence of diffusion bias:

$$\partial_t h(x,t) = -\nabla^4 h + g\nabla^2(\nabla h)^2 + \eta(x,t). \tag{2}$$

In this work four spectral schemes to integrate SPDEs with additive white noise are presented. They all are based on the discretization of the system in Fourier space. They are applied to the numerical integration of Eqs. (1) and (2) in both one (1D) and two (2D) spatial dimensions. As far as I know no results of the numerical integration of the 2D LDV equation have been appeared yet in the literature.

2 Numerical schemes

In this section four numerical schemes to integrate SPDEs with additive white noise are presented. They all are spectral methods based on the discretization of the system in momentum space.

The numerical schemes to be considered are valid to integrate SPDEs of the form:

$$\partial_t h(x,t) = \mathcal{L}[h](x,t) + \Phi[h](x,t) + \xi(x,t), \tag{3}$$

 ISBN 13: 978-84-613-5510-5

where $\mathcal{L}[h]$ is a linear functional of the field h, $\Phi[h]$ is another functional comprising the nonlinear terms and $\xi(x,t)$ is a white noise in space and time:

$$\langle \xi(x,t) \rangle = 0, \qquad \langle \xi(x,t)\xi(x',t') \rangle = 2D\delta(x-x')\delta(t-t').$$

The explicit expressions of the functionals $\mathcal{L}[h]$ and $\Phi[h]$ for equations (1) and (2) are:

$$\text{KPZ} \begin{cases} \mathcal{L}[h](x,t) = \nabla^2 h \\ \Phi[h](x,t) = g(\nabla h)^2 \end{cases} \qquad \text{LDV} \begin{cases} \mathcal{L}[h](x,t) = -\nabla^4 h \\ \Phi[h](x,t) = g\nabla^2(\nabla h)^2 \end{cases}$$

Without loss of generality, a d-dimensional lattice of lateral size L with uniform spacing Δx in each direction is considered and the field h is assumed to satisfy periodic boundary conditions in the multidimensional interval $I = [0,L]^d$. The positions of the nodes in the lattice are given by $x_j = \Delta x(j_1, j_2, \ldots, j_d)$, $0 \leqslant j_i \leqslant N-1$, $1 \leqslant i \leqslant d$, where $N = (\Delta x)^{-1}L$ is the lattice size in each direction.

In order to construct a spectral method, we represent the field $h(x,t)$ as a truncated Fourier series:

$$h_N(x,t) = \sum_{k \in \Gamma_N} \tilde{h}_k(t) e^{iqx}, \quad q = \frac{2\pi}{L}k,$$

where $\Gamma_N = \{(k_1, k_2, \ldots, k_d) \diagup -N/2 \leqslant k_i \leqslant N/2 - 1, 1 \leqslant i \leqslant d\}$ and the $\tilde{h}_k(t)$'s are the Fourier coefficients of h.

The noise term is also replaced by its expansion ξ_N in Fourier modes. In the limit $N \to \infty$, the usual Fourier series is recovered. Finally, Eq. (3) is written in Fourier space as follows:

$$\frac{\mathrm{d}\tilde{h}_k(t)}{\mathrm{d}t} = \omega_k \tilde{h}_k(t) + \tilde{\Phi}_k(t) + \tilde{\xi}_k(t), \quad k \in \Gamma_N. \tag{4}$$

The quantity ω_k is the so-called linear dispersion relation, and comes from the Fourier transform of the linear part of the equation. It is $\omega_k = -q^2$ for the KPZ model and $\omega_k = -q^4$ for the LDV model. On the other hand, the $\tilde{\Phi}_k(t)$'s are the Fourier modes of the nonlinear terms.

As discussed in [7], Eq. (4) is difficult to treat numerically. The reason being that the Fourier coefficients are difficult and expensive to compute, in special the $\tilde{\Phi}_k$'s associated to the nonlinear terms. The process of going from real space to Fourier space and vice versa becomes a stiffly and a time-consuming task. Due to these reasons, we approach the coefficients of the Fourier series with those of the *discrete* Fourier series that will be denoted by $\hat{h}_k$.

Then instead of integrating the set of equations given by (4), we consider the same set of equations with the Fourier coefficients replaced by their discrete version, so that

$$\frac{\mathrm{d}\hat{h}_k(t)}{\mathrm{d}t} = \omega_k \hat{h}_k(t) + \hat{\Phi}_k(t) + \hat{\xi}_k(t). \tag{5}$$

 ISBN 13: 978-84-613-5510-5

The correlations of the variables $\hat{\xi}_k(t)$ are given by:

$$\langle \hat{\xi}_k(t)\hat{\xi}_{k'}(t')\rangle = 2DL^{-d}\delta(t-t')\delta_{k,-k'}.$$

In order to integrate (5) it is useful to make the following change of variables based on the solution of the linear equation:

$$\hat{h}_k(t) = e^{\omega_k t}\hat{z}_k(t) + \hat{R}_k(t), \tag{6}$$

where

$$\hat{R}_k(t) = e^{\omega_k t}\int_0^t \mathrm{d}s\, e^{-\omega_k s}\hat{\xi}_k(s). \tag{7}$$

Then, the variables $\hat{z}_k$ satisfy the following set of ordinary differential equations:

$$\frac{\mathrm{d}\hat{z}_k(t)}{\mathrm{d}t} = \hat{\Phi}_k(t)e^{-\omega_k t}. \tag{8}$$

2.1 Numerical scheme E1

A simple spectral algorithm is obtained by applying to (5) a one step Euler's method. Special care must be put when dealing with the noise term. We first compute the mean and variance of the stochastic variable $\hat{\psi}_k(t) = \int_t^{t+\Delta t}\mathrm{d}s\,\hat{\xi}_k(s)$, where Δt is the time step used in the integration.

$$\langle\hat{\psi}_k(t)\rangle = 0, \qquad \langle\hat{\xi}_k(s)\hat{\xi}_k(s')\rangle = \frac{2D}{L^d}\Delta t.$$

Then, we obtain the following numerical integration scheme which will be referred to as the E1 numerical scheme:

$$\hat{h}_k(t+\Delta t) = \Delta t[\omega_k\hat{h}_k(t) + \hat{\Phi}_k(t)] + \sqrt{\frac{2D\Delta t}{(\Delta x)^d}}\,\hat{v}_k(t), \quad \hat{v}_k(t) = \mathcal{F}[v_n(t)], \tag{9}$$

where $v_n(t)$ is a vector or random Gaussian numbers of mean 0 and variance 1.

Algorithm (9) has been used in [9] to integrate the KPZ equation in 1D and 2D. This quite simple (an easy to code) algorithm has shown much better performance than finite-difference algorithms traditionally used to integrate the KPZ equation.

2.2 Numerical scheme E2

Another more stable scheme than the previous one can be obtained by applying the one step Euler's method to (8):

$$\hat{z}_k(t+\Delta t) = \hat{z}_k(t) + \Delta t\,\hat{\Phi}_k(t)e^{-\omega_k t}.$$

The original variable $\hat{h}_k(t)$ is obtained by using the following relationship derived from Equations (6) and (7):

$$\hat{z}_k(t) = e^{-\omega_k t}[\hat{h}_k(t) - \hat{R}_k(t)]. \tag{10}$$

 ISBN 13: 978-84-613-5510-5

Then, we get

$$\hat{h}_k(t+\Delta t) = \hat{g}_k(t) + e^{\omega_k \Delta t}[\hat{h}_k(t) + \Delta t\, \hat{\Phi}_k(t)].$$

In the computer, the numbers $\hat{g}_k$'s are obtained as follows:

$$\hat{g}_k(t) = \sqrt{\frac{e^{2\omega_k \Delta t}-1}{\omega_k}\frac{D}{(\Delta x)^d}}\ \hat{v}_k(t), \tag{11}$$

where $\hat{v}_k(t)$ are the discrete Fourier coefficients of a vector $v_n(t)$ of random Gaussian numbers of mean 0 and variance 1.

To sum up, we have the following algorithm which will be referred to as the E2 method:

$$\hat{h}_k(t+\Delta t) = \hat{g}_k(t) + e^{\omega_k \Delta t}[\hat{h}_k(t) + \Delta t\, \hat{\Phi}_k(t)], \tag{12}$$

$$\hat{g}_k(t) = \sqrt{\frac{e^{2\omega_k \Delta t}-1}{\omega_k}\frac{D}{(\Delta x)^d}}\ \hat{v}_k(t), \quad \hat{v}_k(t) = \mathcal{F}[v_n(t)]. \tag{13}$$

The E2 scheme has been used in [7] to integrate the 1D KPZ and 1D LDV equations with the aim of comparing its performance to those of finite-difference schemes.

2.3 Numerical scheme PC2

The algorithm presented in this section is a stochastic variant of the so-called two-step method [20]. The starting point is the following formal solution of Eq. (5):

$$\hat{h}_k(t) = e^{\omega_k t}\left(\hat{h}_k(t_0)e^{-\omega_k t_0} + \int_{t_0}^{t} \mathrm{d}s\, \hat{\Phi}_k(s)e^{-\omega_k s} + \int_{t_0}^{t} \mathrm{d}s\, \hat{\xi}_k(s)e^{-\omega_k s}\right). \tag{14}$$

From (14), the following relationship is obtained:

$$\frac{\hat{h}_k(t+\Delta t)}{e^{\omega_k \Delta t}} - \frac{\hat{h}_k(t-\Delta t)}{e^{-\omega_k \Delta t}} = e^{\omega_k t}\int_{t-\Delta t}^{t+\Delta t} \mathrm{d}s\, \hat{\Phi}_k(s)e^{-\omega_k s} + e^{\omega_k t}\int_{t-\Delta t}^{t+\Delta t} \mathrm{d}s\, \hat{\xi}_k(s)e^{-\omega_k s}. \tag{15}$$

The Taylor expansion of $\hat{\Phi}_k(s)$ around $s = t$ for Δt small gives an expression for the first term of the right hand side of (15):

$$e^{\omega_k t}\int_{t-\Delta t}^{t+\Delta t} \mathrm{d}s\, \hat{\Phi}_k(s)e^{-\omega_k s} = \hat{\Phi}_k(t)\frac{e^{\omega_k \Delta t} - e^{-\omega_k \Delta t}}{\omega_k} + O\big((\Delta t)^3\big). \tag{16}$$

Now, taking (16) to (15), we get:

$$\hat{h}_k(t+\Delta t) = e^{2\omega_k \Delta t}\hat{h}_k(t-\Delta t) + \frac{e^{2\omega_k \Delta t}-1}{\omega_k}\hat{\Phi}_k(t) + \hat{\alpha}_k(t) + O\big((\Delta t)^3\big), \tag{17}$$

$$\hat{\alpha}_k(t) := e^{\omega_k (t+\Delta t)}\int_{t-\Delta t}^{t+\Delta t} \mathrm{d}s\, \hat{\xi}_k(s)e^{-\omega_k s}.$$

 ISBN 13: 978-84-613-5510-5

The numerical scheme (17) is often unstable so a *corrector* algorithm is commonly used. The correction is obtained in a similar way to that of Eq. (17). Starting from (14), we obtain

$$\hat{h}_k(t) - \frac{\hat{h}_k(t-\Delta t)}{e^{-\omega_k \Delta t}} = e^{\omega_k t} \int_{t-\Delta t}^{t} \mathrm{d}s\, \hat{\Phi}_k(s) e^{-\omega_k s},$$

from where we obtain the auxiliary Equation:

$$\hat{h}_k(t) = e^{\omega_k \Delta t} \hat{h}_k(t-\Delta t) + \frac{e^{\omega_k \Delta t} - 1}{\omega_k} \hat{\Phi}_k(t-\Delta t) + \hat{\beta}_k(t) + O\big((\Delta t)^2\big), \tag{18}$$

$$\hat{\beta}_k := e^{\omega_k t} \int_{t-\Delta t}^{t} \mathrm{d}s\, \hat{\xi}_k(s) e^{-\omega_k s}.$$

The stochastic variables $\hat{\alpha}_k(t)$ and $\hat{\beta}_k(t)$ can be cast in terms of the variables $\hat{g}_k(t)$:

$$\hat{\alpha}_k(t) = e^{\omega_k \Delta t} \hat{\beta}_k(t) + \hat{g}_k(t), \quad \hat{\beta}_k(t) = \hat{g}_k(t-\Delta t).$$

Summing up, the complete second-order predictor-corrector numerical scheme can be written as follows:

$$\text{Predictor:} \qquad \hat{h}_k(t) = e^{\omega_k \Delta t} \hat{h}_k(t-\Delta t) + \frac{e^{\omega_k \Delta t} - 1}{\omega_k} \hat{\Phi}_k(t-\Delta t) + \hat{\beta}_k(t),$$

$$\text{Corrector:} \quad \hat{h}_k(t+\Delta t) = e^{2\omega_k \Delta t} \hat{h}_k(t-\Delta t) + \frac{e^{2\omega_k \Delta t} - 1}{\omega_k} \hat{\Phi}_k(t) + \hat{\alpha}_k(t),$$

where

$$\hat{\alpha}_k(t) = e^{\omega_k \Delta t} \hat{\beta}_k(t) + \hat{g}_k(t), \quad \hat{\beta}_k(t) = \hat{g}_k(t-\Delta t),$$

$$\hat{g}_k(t) = \sqrt{\frac{e^{2\omega_k \Delta t} - 1}{\omega_k} \frac{D}{(\Delta x)^d}}\, \hat{v}_k(t), \quad \hat{v}_k(t) = \mathcal{F}[v_n(t)].$$

Here $v_n(t)$ is an array of Gaussian random numbers of mean zero and variance one. Note that time advances $2\Delta t$ in each step.

2.4 Numerical scheme PC4

The numerical scheme described in this section is a fourth-order predictor-corrector method based on the following algorithm to integrate a first order differential equation $y' = f(x, y)$:

$$\begin{aligned} \text{Predictor:} \quad & y_{i+1} = y_i + \frac{h}{24}(55 f_i - 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3}), \\ \text{Corrector:} \quad & y_{i+1} = y_i + \frac{h}{24}(9 f_{i+1} + 19 f_i - 5 f_{i-1} + f_{i-2}). \end{aligned} \tag{19}$$

The predictor is the four-step Adams-Bashforth method and the corrector is the three-step Adams-Moulton method (see for example [8, 21]).

 ISBN 13: 978-84-613-5510-5

Using the algorithm (19) applied to the resolution of (8), the following equations for the evolution of the $\hat{z}_k$'s are obtained:

Predictor formula:

$$\hat{z}_k(t+\Delta t) = \hat{z}_k(t) + \frac{\Delta t}{24} e^{-\omega_k t}[55\hat{\Phi}_k(t) - 59\hat{\Phi}_k(t-\Delta t)e^{\omega_k \Delta t} + \\ + 37\hat{\Phi}_k(t-2\Delta t)e^{2\omega_k \Delta t} - 9\hat{\Phi}_k(t-3\Delta t)e^{3\omega_k \Delta t}].$$

Corrector formula:

$$\hat{z}_k(t+\Delta t) = \hat{z}_k(t) + \frac{\Delta t}{24} e^{-\omega_k t}[9\hat{\Phi}_k(t+\Delta t)e^{-\omega_k \Delta t} + \\ + 19\hat{\Phi}_k(t) - 5\hat{\Phi}_k(t-\Delta t)e^{\omega_k \Delta t} + \hat{\Phi}_k(t-2\Delta t)e^{2\omega_k \Delta t}].$$

Now, coming back to the original variable $\hat{h}_k$ with (10), the final algorithm is obtained:

Predictor:

$$\hat{h}_k(t+\Delta t) = \hat{g}_k(t) + e^{\omega_k \Delta t}[\hat{h}_k(t) + \frac{\Delta t}{24}(55\hat{\Phi}_k(t) - 59\hat{\Phi}_k(t-\Delta t)e^{\omega_k \Delta t} + \\ + 37e^{2\omega_k \Delta t}\hat{\Phi}_k(t-2\Delta t) - 9e^{3\omega_k \Delta t}\hat{\Phi}_k(t-3\Delta t))].$$

Corrector:

$$\hat{h}_k(t+\Delta t) = \hat{g}_k(t) + \hat{h}_k(t)e^{\omega_k \Delta t} + \frac{\Delta t}{24}[9\hat{\Phi}_k(t+\Delta t) + \\ + 19\hat{\Phi}_k(t)e^{\omega_k \Delta t} - 5\hat{\Phi}_k(t-\Delta t)e^{2\omega_k \Delta t} + \hat{\Phi}_k(t-2\Delta t)e^{3\omega_k \Delta t}].$$

As before, the stochastic variables $\hat{g}_k(t)$ are computed in practice as:

$$\hat{g}_k(t) = \sqrt{\frac{e^{2\omega_k \Delta t} - 1}{\omega_k} \frac{D}{(\Delta x)^d}}\ \hat{v}_k(t), \quad \hat{v}_k(t) = \mathcal{F}[v_n(t)],$$

where $v_n(t)$ is a vector or random Gaussian numbers of mean 0 and variance 1.

It is important to notice that the previous algorithm needs to be initialized in order to find the values of the field at the first three time steps. Preferably, a stochastic algorithm of the same order should be used, but for simplicity a deterministic fourth-order Runge-Kutta algorithm (RK4) will be used. This can be seen as a slight modification of the initial condition. However, taking into account that we deal with random initial conditions, the impact on the solutions of starting with a determinist algorithm is negligible.

3 Numerical results

In this section we show some results obtained when applying the numerical schemes described in Section 2 to the KPZ and LDV models (Eqs. (1) and (2) respectively) in 1D and 2D. A lattice spacing of $\Delta x = 1$ will be used in all the cases as is customary in this kind of simulations. As for the time step, the value $\Delta t = 0.01$ has been taken

 ISBN 13: 978-84-613-5510-5

in most simulations although a small value needed to be considered in some cases so as to ensure convergence. The results of the simulations were averaged over a number of noise realizations, typically 100. For the Fourier transform, a freeware fast Fourier package has been used [6].

3.1 Critical exponents

The LDV and KPZ growth models exhibit scaling properties so that they can be divided into universality classes characterized by critical exponents [1, 13]. First let us consider the global width of the interface which, in a system of lateral size L, is given by

$$W(L,t) = \left\langle \overline{(h(x,t) - \overline{h})^2} \right\rangle^{1/2},$$

where the angular brackets mean an average over noise measures and the overbars mean an average over lattice sites. The global width scales according to the Family-Vicsek ansatz [5]:

$$W(L,t) = t^{\alpha/z} f(L/t^{1/z}), \qquad f(u) \sim \begin{cases} u^{\alpha} & \text{if } u \ll 1, \\ \text{constant} & \text{if } u \gg 1. \end{cases}$$

Parameter α is the so-called roughness exponent whereas z is the dynamic exponent. Both of them characterize the universality class of the model. The ratio $\beta = \alpha/z$ is the time exponent. For the LDV model the critical exponents can be calculated exactly in any dimension d by means of renormalization group techniques [3]: $\alpha = (4-d)/3$ and $z = (8-d)/3$ (and so $\beta = (4-d)/(8+d)$). For the 1D KPZ equation the critical exponents $\alpha = 1/2$ and $z = 3/2$ (so $\beta = 1/3$) can also be calculated exactly. Finally, for the 2D KPZ equation, the exponents obtained with field-theoretical methods [18] and a Flory-type approach [10, 12] are $\alpha = 0.4$ and $z = 1.6$ (so $\beta = 0.25$). For small times ($t \ll L^z$), a behaviour $W(L,t) \sim t^{\beta}$ is expected so exponent β can be computed by measuring the slope of the curve $\log W(L,t)$ vs. $\log t$. At long times ($t \gg L^z$) the system reaches a saturation regime where $W(L,t) \sim L^{\alpha}$. Since the value of α for the LDV equation is larger than that of the KPZ equation, the time it takes the system to get to saturation for a fixed L is larger for the LDV equation.

The global width for the KPZ and LDV equations in the linear regime are depicted in Figure 1 for systems of lateral sizes $L_{1\mathrm{D}} = 2048$ and $L_{2\mathrm{D}} = 512$. The time exponents derived from a linear fit of the curves of Figure 1 are given in Table 1. As can be seen, all the numerical schemes provide approximately the same exponents which are in very good agreement with their theoretical values. For the 2D KPZ equation, a time exponent $\beta \sim 0.24$ is found, a value previously reported in other works, using either finite-difference schemes [2, 19] or a pseudospectral method [9].

 ISBN 13: 978-84-613-5510-5

3.2 Global width at saturation for the 1D KPZ equation

For the 1D KPZ, the behaviour of the global width in the steady-state (i.e., the saturation regime) is known exactly [14, 22], namely:

$$W(L, t \to \infty) = \sqrt{\frac{1}{24}}\, L^{1/2} \approx 0.204 L^{1/2}.$$

The previous expression can be used to test the performance of the several numerical schemes. In the computation of $W(L, t \to \infty)$ system sizes $L_i = 2^i$, with $4 \leqslant i \leqslant 10$ were used and the data was averaged over 100 noise realizations. In the computation of each value of the width, several tens of points inside the steady-state region were taken. Then a linear fitting of the data $(\log L_i, \log W(L_i))$ was performed, giving parameters A and B such that $\log W = \log A + B \log L$. The values of A and B obtained in this way for the four numerical schemes are shown in Table 2. As can be seen, the values of A and B are very close to the theoretical values $A = 0.204$ and $B = 0.5$ in all cases. In [9] the authors show that a finite-difference based method with a one-step Euler scheme and standard gradient square discretization does not give an accurate result as compared with a pseudospectral method (i.e., the E1 method) for the global with in the steady-state. The so-called Lam and Shin discretization [17] for the gradient square $(\nabla h)^2$ gives better results than the standard discretization, but they both involve much more fluctuations than the pseudospectral method.

3.3 Multiscaling

In systems exhibiting anomalous scaling the height-difference correlation functions $G_m(r, t)$, are expected to follow the following power-law behavior:

$$G_m(r, t) \sim r^{\gamma_m}, \quad 1 \ll r \ll t^{1/z}.$$

If the quantities γ_m depend on m, the system is said to exhibit multiscaling behavior. Numerically, we can infer the existence multiscaling by plotting $G_m(r, t)$ for several values of m at the stationary state. If there were multiscaling the behavior of the correlation functions would depend on m. Both, the KPZ and LDV models are not expected to show multiscaling. In [4] the existence of multiscaling was suggested for the 1D LDV equation based on the instability of the system against the growth of isolated pillars with a height larger than a certain threshold. However, it was shown in [7] that this conclusion was misleading and an artifact of the numerical simulations which were based on a finite-difference scheme. With a direct computation using a pseudospectral method (namely, the E2 method), the results of [7] show that there are no multiscaling for the 1D LDV equation.

In Figure 2 we show the height difference correlation functions $G_m(r)$ for $1 \leqslant m \leqslant 5$ at saturation for the 1D KPZ equation integrated with the E1 numerical scheme. As expected, all the correlation functions scale in a similar way indicating that there is no trace of multiscaling. The same conclusion is reached for the other numerical schemes and for the 2D KPZ, 1D LDV and 2D LDV equations. On the other hand,

 ISBN 13: 978-84-613-5510-5

the behavior of the correlation functions $G_m(r,t)$ with r fixed as a function of time was also considered. In these cases no multiscaling was observed either.

3.4 Numerical stability

Finally the stability of the algorithms was tested by measuring the the maximum time up to which the system can be integrated before a numerical overflows occurs. The schemes PC2 and PC4 show as the most stable and the E1 scheme as the least stable. However taking into account that all the schemes provide approximately the same results, and that the PC2 and PC4 methods are much more time-consuming than the E1 and E2 schemes, the E2 is the one that performs better to integrate the KPZ and LDV equations.

4 Figures and tables

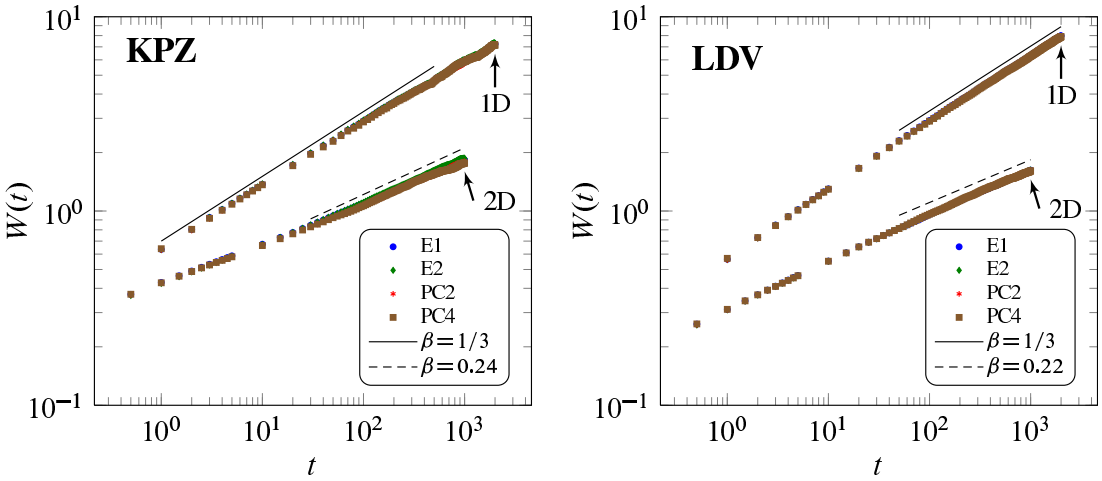


Figure 1. Global width for the KPZ and LDV equations in one and two spatial dimensions. The lateral sizes are $L_{1D} = 2048$ and $L_{2D} = 512$. The averages are taken over 100 noise measures. The kind of numerical method and the value of the parameter g for each curve are shown in the legends. Straight lines are shown as a guide for the eye.

 ISBN 13: 978-84-613-5510-5

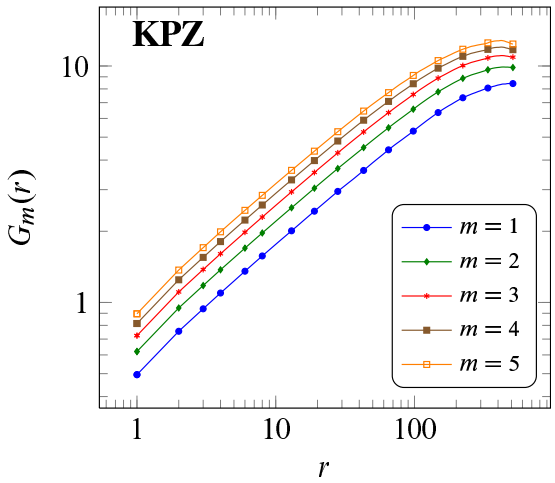


Figure 2. Height-difference correlation functions for the 1D KPZ equation in the saturation regime for $1 \leqslant m \leqslant 5$.

		Equations			
		1D KPZ	**1D LDV**	**2D KPZ**	**2D LDV**
Schemes	**E1**	0.331 ± 0.007	0.338 ± 0.005	0.230 ± 0.005	0.220 ± 0.002
	E2	0.333 ± 0.007	0.336 ± 0.006	0.235 ± 0.006	0.224 ± 0.002
	PC2	0.333 ± 0.007	0.334 ± 0.006	0.230 ± 0.006	0.224 ± 0.002
	PC4	0.326 ± 0.007	0.336 ± 0.006	0.230 ± 0.005	0.219 ± 0.002
	Theor.	$1/3$	$1/3$	0.25	0.20

Table 1. Time exponents obtained from a linear fit of the global width for the KPZ and LDV equations. The lateral sizes are $L_{1D} = 2048$ and $L_{2D} = 512$. The values of the nonlinear coupling parameter are $g_{KPZ} = 2$ and $g_{LDV} = 1$. In the last row the theoretical values of the exponents are given.

Parameter	E1	E2	PC2	PC4
A	0.195	0.195	0.200	0.194
B	0.500	0.501	0.495	0.501

Table 2. Results of the linear fitting of the global width in the steady-state for the 1D KPZ model integrated with the four numerical schemes described in the text. Parameters A and B correspond to a fitting of the form $\log W = \log A + B \log L$. The data have been averaged over 100 noise realizations and the global width values have been obtained by averaging several tens of data inside the steady-state regime.

Acknowledgements

This work is supported by the DGI of the Ministerio de Educacin y Ciencia (Spain) through Grant No. FIS2009-12964-C05-05.

 ISBN 13: 978-84-613-5510-5

References

[1] A.-L. Barabási, H. E. Stanley, Fractal Concepts in Surface Growth, Cambridge University Press, Cambridge, 1995.

[2] M. Beccaria, G. Curci, Numerical simulation of the Kardar-Parisi-Zhang equation, Phys. Rev. E 50 (1994) 4560.

[3] S. Das Sarma, R. Kotlyar, Dynamical renormalization group analysis of fourth-order conserved growth nonlinearities, Phys. Rev. E 50 (1994) R4275.

[4] C. Dasgupta, J. M. Kim, M. Dutta, S. Das Sarma, Instability, intermittency, and multiscaling in discrete growth models of kinetic roughening, Phys. Rev. E 55 (1997) 2235.

[5] F. Family, J. Vicsek, Scaling of the active zone in the eden process on percolation networks and the ballistic deposition model, J. Phys. A 18 (1985) L75.

[6] M. Frigo, S. G. Johnson., The fftw package, `http://www.fftw.org`.

[7] R. Gallego, M. Castro, J. M. López, Pseudospectral versus finite-difference schemes in the numerical integration of stochastic models of surface growth, Phys. Rev. E 76 (2007) 051121.

[8] C. W. Gear, Numerical initial value problems in ordinary differential equations, Prentice-Hall Inc., New Jersey, 1971.

[9] L. Giada, A. Giacometti, M. Rossi, Pseudospectral method for the kardar-parisi-zhang equation, Phys. Rev. E 65 (2002) 036134.

[10] H. G. E. Hentschel, F. Family, Scaling in open dissipative systems, Phys. Rev. Lett. 66 (1991) 1982.

[11] M. Kardar, G. Parisi, Y.-C. Zhang, Dynamic scaling of growing interfaces, Phys. Rev. Lett. 56 (1986) 889.

[12] J. M. Kim, J. M. Kosterlitz, Growth in a restricted solid-on-solid model, Phys. Rev. Lett. 62 (1989) 2289.

[13] J. Krug, Origins of scale invariance in growth processes, Avd. Phys. 46 (1997) 139.

[14] J. Krug, P. Meakin, T. Halpin-Healy, Amplitude universality for driven interfaces and directed polymers in random media, Phys. Rev. A 45 (1992) 638.

[15] Z.-W. Lai, S. Das Sarma, Kinetic growth with surface relaxation: Continuum versus atomistic models, Phys. Rev. Lett. 66 (1991) 2348.

[16] C.-H. Lam, F. G. Shin, Anomaly in numerical integrations of the kardar-parisi-zhang equation, Phys. Rev. E 57 (1998) 6506.

 ISBN 13: 978-84-613-5510-5

[17] C.-H. Lam, F. G. Shin, Improved discretization of the Kardar-Parisi-Zhang equation, Phys. Rev. E 58 (1998) 5592.

[18] M. Lässig, Quantized scaling of growing surfaces, Phys. Rev. Lett. 80 (1998) 2366.

[19] K. Moser, J. Kertsz, D. E. Wolf, Numerical solution of the kardar-parisi-zhang equation in one, two and three dimensions, Physica A 178 (1991) 215.

[20] D. Potter, Computational Physics, John Wiley and Sons, 1973.

[21] Shampine, L. F., Gordon, M. K., Computer Solution of Ordinary Differential Equations : The Initial Value Problem, Freeman & Co., 1975.

[22] K. Sneppen, J. Krug, M. H. Jensen, C. Jayaprakash, T. Bohr, Dynamic scaling and crossover analysis for the Kuramoto-Sivashinsky equation, Phys. Rev. A 46 (1992) R7351.

[23] J. Villain, Continuum models of crystal-growth from atomic-beams with and without desorption, Journal de Physique I 1 (1991) 19.

[24] D. E. Wolf, J. Villain, Growth with surface diffusion, Europhys. Lett. 13 (1990) 389.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Droplet and bubble pinch-off computations using Level Sets

Maria Garzon[1], Len Gray[2] and James Sethian[3]

[1] *Dept.of Applied Mathematics, University of Oviedo*

[2] *Computer Science and Mathematics Division, Oak Ridge National Laboratory, University of Tenessee*

[3] *Dept.of Mathematics, University of Berkeley, CA*

emails: maria@orion.ciencias.uniovi.es, ljg@ornl.org, sethian@math.berkeley.edu

Abstract

A two fluid potential flow model is used to analyze the pinching characteristics of an inviscid fluid immersed in another infinite inviscid fluid of different density. The system behavior is controlled by the two fluid relative densities $D = \rho_{\mathcal{E}}/\rho_{\mathcal{I}}$, $D = 0$ corresponds to droplets in air, while D=100 will lead to bubbles in water. The numerical method employed in this work combines the Level Set method for advancing the free surface position and the boundary condition, together with a 3-D axisymmetric boundary integral formulation to obtain fluid velocities. This approach provides a numerical methodology to analyze the pinch-off behavior up to and beyond the initial break up of the inner fluid. The algorithm is validated using the analytical solution for an oscillating sphere in a two fluid system and a series of numerical experiments, up to and beyond the initial break up of the inner fluid, have been carried out. The calculated scaling exponents match the known values at the droplet and bubble extremes ($D = 0$, $D = 100$), and the computed front profiles obtained are in good agreement with recent experimental findings.

Key words: Nonlinear potential flow, Level Set Method.

1 Introduction

In part due to many important technological applications [2], fluid break-up has been extensively studied by means of experimental [3, 16] theoretical, and computational analysis [4, 5].

Considering that the interior and exterior fluids are inviscid and incompressible and the movement is due to an irrotational velocity field, the potential flow assumptions

 ISBN 13: 978-84-613-5510-5

are valid up to the nanometers length scale. The mathematical model, written in non-dimensional form, is controlled by only one parameter, $D = \rho_{\mathcal{E}}/\rho_{\mathcal{I}}$, the relative density of the two fluids. $D = 0$ corresponds to the formation of droplets in air, while D=100 will lead to air bubbles in water.

The goal of the present work is to study numerically the evolution of this two inviscid fluid system, through pinch-off into satellite break-off for the two extreme D values ($D = 0$, $D = 100$). Previous numerical approximations of the same model equations [12, 13] have employed marker particle methods, and thus cannot go past pinch-off and are unable to predict post break up dynamics.

In a recent work [9], the collapse of a single inviscid fluid column ($D = 0$, Rayleigh-Taylor instability) has been modeled. The algorithm combines a Level Set method [15] for advancing the free surface and the free surface boundary condition, together with a boundary integral approach for the evaluation of surface velocities. This approach was successful in continuing the interface evolution beyond initial separation and through the subsequent satellite drop evolution, while accurately predicting the known scaling exponent, $\alpha_D = 2/3$, of the self similar power law $r \propto \tau^{\alpha_D}$. Here, r is the minimum neck radius and τ the remaining time until pinch-off.

We have extended the corresponding model equations and developed the numerical algorithm for the two fluid system case, which allow us to simulate the droplet and bubble pinch-off behavior and their post separation dynamics. The computed scaling exponents for $D = 0$ and $D = 100$ are in perfect agreement with known theoretical and experimental values [6, 12], and the front profiles before and after separation match the patterns seen in laboratory experiments [3, 16] both for water droplets and air bubbles.

2 The two fluid potential flow model

To briefly describe the equations consider a fluid of density $\rho_{\mathcal{I}}$ immersed in an (infinite) exterior fluid of density $\rho_{\mathcal{E}}$. The system is initially at rest and, in absence of gravity, the fluid movement is induced by surface tension forces.

Let $\Omega_k(t)$, $k = \mathcal{I}, \mathcal{E}$, be the 3D interior and exterior fluid domains respectively and $\Gamma_t(\mathbf{s}) = (x(\mathbf{s}, t), y(\mathbf{s}, t), z(\mathbf{s}, t))$ a parametrization of the free surface boundary at time t. Assuming potential flow, the fluid velocities $\mathbf{u}_k$ for each fluid domain $\Omega_k(t)$, $k = \mathcal{I}, \mathcal{E}$, are given in terms of a potential ϕ_k

$$\begin{aligned} \mathbf{u}_k &= \nabla\phi_k & (1)\\ \Delta\phi_k &= 0 & (2)\\ D_t\mathbf{R} &= \mathbf{u}_{\mathcal{I}} \text{ on } \Gamma_t(\mathbf{s}) \ . & (3) \end{aligned}$$

The last equation is the kinematic boundary condition (for the interior fluid), which states that the interface moves with the fluid velocity, with D_t denoting the total derivative following the fluid (interior) particles

$$D_t = \frac{\partial}{\partial t} + \mathbf{u}_{\mathcal{I}} \cdot \nabla \ . \quad (4)$$

 ISBN 13: 978-84-613-5510-5

On the free boundary between the two fluid domains, $\Gamma_t(\mathbf{s})$, the continuity of the normal velocity and normal stress tensor gives

$$\mathbf{n} \cdot \nabla \phi_{\mathcal{I}} = \mathbf{n} \cdot \nabla \phi_{\mathcal{E}},$$

$$\rho_{\mathcal{I}}(\frac{\partial \phi_{\mathcal{I}}}{\partial t} + \frac{1}{2}|\nabla \phi_{\mathcal{I}}|^2) - \rho_{\mathcal{E}}(\frac{\partial \phi_{\mathcal{E}}}{\partial t} + \frac{1}{2}|\nabla \phi_{\mathcal{E}}|^2) + \gamma\kappa = 0,$$

where $\mathbf{n}$ is the unit normal vector pointing from the interior to the exterior domain and $\kappa = 1/R_1 + 1/R_2$ is twice the mean curvature of the surface. Taking as characteristic time scale $t_0 = (\rho_{\mathcal{I}} R_0^3/\gamma)^{1/2}$ we get

$$(\frac{\partial \phi_{\mathcal{I}}}{\partial t} + \frac{1}{2}|\nabla \phi_{\mathcal{I}}|^2) - D(\frac{\partial \phi_{\mathcal{E}}}{\partial t} + \frac{1}{2}|\nabla \phi_{\mathcal{E}}|^2) + \kappa = 0,$$

where now all the quantities are non dimensional. Next adding and subtracting the needed terms we obtain

$$\frac{\partial \phi_{\mathcal{I}}}{\partial t} - D\frac{\partial \phi_{\mathcal{E}}}{\partial t} + \mathbf{u}_{\mathcal{I}} \cdot (\nabla \phi_{\mathcal{I}} - D\nabla \phi_{\mathcal{E}}) = \mathbf{u}_{\mathcal{I}} \cdot (\frac{1}{2}\nabla \phi_{\mathcal{I}} - D\nabla \phi_{\mathcal{E}}) + \frac{D}{2}\mathbf{u}_{\mathcal{E}} \cdot \nabla \phi_{\mathcal{E}} - \kappa,$$

and setting $\phi_d = \phi_{\mathcal{I}} - D\phi_{\mathcal{E}}$, we have

$$\frac{\partial \phi_d}{\partial t} + \mathbf{u}_{\mathcal{I}} \cdot \nabla \phi_d = f \ \text{ on } \ \Gamma_t(\mathbf{s}),$$

being

$$f = \mathbf{u}_{\mathcal{I}} \cdot (\frac{1}{2}\mathbf{u}_{\mathcal{I}} - D\mathbf{u}_{\mathcal{E}}) + \frac{D}{2}\mathbf{u}_{\mathcal{E}} \cdot \mathbf{u}_{\mathcal{E}} - \kappa$$

The model equations in 3D ($k = \mathcal{I}, \mathcal{E}$) are therefore,

$$\mathbf{u}_k = \nabla \phi_k \ \text{ in } \ \Omega_k(t) \tag{5}$$

$$\Delta \phi_k = 0 \ \text{ in } \ \Omega_k(t) \tag{6}$$

$$D_t \mathbf{R} = \mathbf{u}_{\mathcal{I}} \ \text{ on } \ \Gamma_t(\mathbf{s}) \tag{7}$$

$$D_t \phi_d = f \ \text{ on } \ \Gamma_t(\mathbf{s}) \tag{8}$$

This Lagrangian-Eulerian formulation is frequently numerically approximated using the so-called "front tracking method", which suffers difficulties when the free boundary changes topology. These problems are avoided using the level set formulation for both the kinematic and dynamic boundary conditions, (7) and (8).

The remaining boundary condition needed to simultaneously solve the two Laplace equations (6) is that the normal velocities in the two fluids are equal and opposite in sign.

3 The Level Set formulation

The Level Set method is a mathematical tool developed by Osher and Sethian [14] to follow interfaces which move with a given velocity field. The key idea is to view the moving front as the zero level set of one higher dimensional function called the level

 ISBN 13: 978-84-613-5510-5

set function. One main advantage of this approach comes when the moving boundary changes topology, and thus a simple connected domain splits into separated disconnected domains. Let Ω_D be a fictitious fixed 3D rectangular domain that contains the free boundary at any time t and $\Gamma_t(\mathbf{s})$ the set of points lying in the surface boundary at time t. This surface is defined through the zero level set of the scalar field $\Psi(x, y, z, t)$. An equation of motion for Ψ that ties the zero level set of Ψ to the evolving front comes from observing that the level set value of a particle on the front with path $\mathbf{R}(\mathbf{s}, t)$ must always be zero:

$$\Psi(\mathbf{R}(\mathbf{s}, t), t) = 0. \tag{9}$$

To embed the free surface boundary condition given by Eq. (8) into the level set framework, we define $G(x, y, z, t)$ on Ω_D such that

$$G(\mathbf{R}(\mathbf{s}, t), t) = \phi_d(x, y, z, t)|_{\Gamma_\mathbf{s}} = \phi_d(\mathbf{R}(\mathbf{s}, t), t). \tag{10}$$

Deriving (9) and (10) with respect to time, following the interior fluid particles characteristics, we have

$$\begin{aligned} \Psi_t + \mathbf{u}_\mathcal{I} \cdot \nabla\Psi &= 0, & (11)\\ G_t + \mathbf{u}_\mathcal{I} \cdot \nabla G = D_t\phi_d &= f, & (12) \end{aligned}$$

which holds on $\Gamma_t(\mathbf{s})$. Note that $G(x, y, z, t)$ is an auxiliary function that can be chosen arbitrarily, with the only restriction that it is equal to $\phi_d(x, y, z, t)$ on the free surface. The velocity $\mathbf{u}_\mathcal{I}$ and the right hand side of Eq. (12) are only defined on $\Gamma_t(\mathbf{s})$, and thus, in order to solve Eq. (11) and (12) over the domain Ω_D, these variables must be extended off the front. A detailed description of how to perform these extensions is given in [1]. The system of equations, written in a complete Eulerian framework, $(k = \mathcal{I}, \mathcal{E})$, is

$$\begin{aligned} \mathbf{u}_k &= \nabla\phi_k \text{ in } \Omega_k(t), & (13)\\ \Delta\phi_k &= 0 \text{ in } \Omega_k(t), & (14)\\ \Psi_t + \mathbf{u}_{\mathcal{I}_{\text{ext}}} \cdot \nabla\Psi &= 0 \text{ in } \Omega_D, & (15)\\ G_t + \mathbf{u}_{\mathcal{I}_{\text{ext}}} \cdot \nabla G &= f_{\text{ext}} \text{ in } \Omega_D. & (16) \end{aligned}$$

The subscript "ext" denotes the extension of f and $\mathbf{u}_\mathcal{I}$ onto Ω_D.

The free surface equations 7 and 8 have now been embedded into the higher dimension equations 15 and 16 and it can be shown that system (13)-(16) is equivalent to system (5)-(8) and in fact enriches the kinematics of the system, in the sense that it can incorporate topological changes of the free surface, and as well the evolution of the associated potential function within this boundary, see [7],[9].

Assuming symmetry around the z axis the previous system can be formulated in 2D by writing the equations in cylindrical coordinates. The equations in the (r, z) plane remain the same except for the laplacian that should be changed accordingly. In what follows $\Omega_k(t)$, $k = \mathcal{I}, \mathcal{E}$, will denote the 2D fluid domains in the (r, z) plane, $\Gamma_t(s)$ the free boundary between these fluid domains and Ω_D a 2D fixed domain that contains the free boundary for all times.

 ISBN 13: 978-84-613-5510-5

4 The Numerical Approximation

The numerical approximation of system (13)-(16) in the (r, z) plane can be described in two basic steps.

First, using a standard first order backward Euler explicit scheme to approximate time derivatives in the level set equations, the system to be solved for each time t_n and time step Δt, $k = \mathcal{I}, \mathcal{E}$, is:

$$\begin{aligned}
\mathbf{u}_k^n &= \nabla \phi_k^n \text{ in } \Omega_k(t_n) && (17)\\
\Delta \phi_k^n(r,z) &= 0 \text{ in } \Omega_k(t_n) && (18)\\
\frac{\Psi^{n+1}-\Psi^n}{\Delta t} &= -\mathbf{u}^{\mathbf{n}}{}_{\mathcal{I}_{\text{ext}}} \cdot \nabla \Psi^n \text{ in } \Omega_D && (19)\\
\frac{G^{n+1}-G^n}{\Delta t} &= -\mathbf{u}^{\mathbf{n}}{}_{\mathcal{I}_{\text{ext}}} \cdot \nabla G^n + f_{\text{ext}}^n \text{ in } \Omega_D. && (20)
\end{aligned}$$

The second main task is to solve Eqs. (18) for the free surface velocity, subject to the boundary condition $\phi_{\mathcal{I}}^n - D\phi_{\mathcal{E}}^n = G^n$. This is accomplished by solving the boundary integral equation corresponding to the Laplace equations in the corresponding axisymmetric geometry. With the computed velocity, the new position of the boundary is determined from the level set equation (19), and the potential ϕ_d on $\Gamma_{t_{n+1}}(s)$ will be obtained from Eq. (20). These procedures are described below.

4.1 Level Set numerical schemes

The fixed computational domain for equations (19) and (20), $\Omega_D = [0, L_1] \times [0, L_2]$, is chosen such that it contains the free boundary for all $t \in [0, T]$.

A rectangular mesh over the domain Ω_D defines a set of points $D_\Delta = \{(r_i, z_j) : r_i = i\Delta r, z_j = j\Delta z, i = 1, N, j = 1, M\}$, with N, M the number of mesh points in the r and z directions and Δr, Δz the corresponding mesh sizes. Let be $\mathbf{n} = (n_r, n_z)$ the unit normal vector to $\Gamma_{t_n}(s)$ and u, v the radial and axial inner fluid velocity components. The axisymmetric assumption implies $u = 0$ and $n_r = 0$ at Γ_z, and thus

$$\frac{\partial \Psi^n}{\partial r} = 0; \quad \frac{\partial G^n}{\partial r} = 0 \text{ at } \Gamma_z.$$

will be imposed for (19) and (20) Let be $G_{i,j}^n$ the numerical approximation of the fictitious potential $G(r_i, z_j, t_n)$. A first order upwind scheme approximation of Eq. (20) yields, for $i = 2, N-1$; $j = 2, M-1$,

$$\begin{aligned}
G_{i,j}^{n+1} &= G_{i,j}^n - \Delta t(\max(u_{i,j}^n, 0)D_{i,j}^{-r} + \min(u_{i,j}^n, 0)D_{i,j}^{+r}\\
&+ \max(v_{i,j}^n, 0)D_{i,j}^{-z} + \min(v_{i,j}^n, 0)D_{i,j}^{+z}) + \Delta t f_{i,j}^n,
\end{aligned}$$

where

$$\begin{aligned}
D_{i,j}^{-r} &= D_{i,j}^{-r}\left\{G_{i,j}^n\right\} = \frac{G_{i,j}^n - G_{i-1,j}^n}{\Delta r}\\
D_{i,j}^{+r} &= D_{i,j}^{+r}\left\{G_{i,j}^n\right\} = \frac{G_{i+1,j}^n - G_{i,j}^n}{\Delta r}
\end{aligned}$$

 ISBN 13: 978-84-613-5510-5

are the backward and forward finite difference approximations for the derivative in the radial direction (the same expressions hold for the corresponding z derivatives $D^{-z}_{i,j}$ and $D^{+z}_{i,j}$). The discrete boundary conditions are:

$$v_{1,j} = 0 \ \text{ for } \ j = 1, M$$

$$\frac{\partial G^n_{i,j}}{\partial r} \approx \frac{4G^n_{2,j} - 3G^n_{1,j} - G^n_{3,j}}{2\Delta r} \quad \text{for } \ (r_i, z_j) \ \in \ \Gamma_z$$

$$G^n_{i,1} = G^n_{i,2}; \ \ G^n_{i,M-1} = G^n_{i,M} \quad \text{for } \ i = 1, N.$$

$$G^n_{N,j} = G^n_{N-1,j}; \ \ G^n_{1,j} = G^n_{2,j} \quad \text{for } \ j = 1, M.$$

The same discrete equations, without source term, can be written for Ψ, Eq. (19).

Note that, for simplicity, we have written u, v, f instead of $u_{\text{ext}}, v_{\text{ext}}, f_{\text{ext}}$, and we describe a first order explicit scheme with a centered source term. Initial values of $G^0_{i,j}$ are obtained by extending $\phi(r, z, 0)|_{\Gamma_0(s)}$. However, at any time step n it is always possible to perform a new extension of $\phi^n(r, z, n\Delta t)$ and a reinitialization of the level set function. We remark here that if reinitialization is done too often, especially using poor reinitialization techniques, spurious mass loss/gain will occur. Thus, it is important to perform reinitialization both sparingly and accurately.

4.2 Boundary Integral Equations

Although the Laplace solver used in previous numerical work by other authors [12, 13] was also a boundary integral method, there are some significant differences in the numerical schemes. The earlier algorithms employed a vortex boundary integral formulation, based upon high order quintic polynomial collocation approximation. In this paper we use a direct 3D axisymmetric boundary integral formulation for the two fluid system. The boundary integral solution algorithm employs a linear element Galerkin approximation, incorporating non-standard Galerkin weight functions that simplify the treatment near the symmetry axis [10]. The detailed boundary integral approximation will be presented elsewhere.

5 Numerical Results

5.1 The analytical solution of an oscillating sphere

According to the linear theory and following [12] if a spherical drop is perturbed such that at $t = 0$, we set

$$\phi_{\mathcal{I}}(r, z, 0) = \phi_{\mathcal{E}}(r, z, 0) = 0,$$

$$z(s) = -\cos(s)(1 + \epsilon P_m(\cos(s))),$$

$$z(s) = -\sin(s)(1 + \epsilon P_m(\cos(s))),$$

 ISBN 13: 978-84-613-5510-5

for $0 \leq s \leq \pi$, with $\epsilon \ll 1$,and P_m the Legendre polynomial of order m, the drop will oscillate with frequency ω, given by

$$\omega^2 = \frac{m(m-1)(m+1)(m+2)}{Dm + (m+1)}$$

For these numerical experiments we choose $\epsilon = 0.05$, $\Omega_D = [-2,2] \times [-2,2]$, $\Delta r = \Delta z$ is the fixed mesh size and N_p the number of BEM nodes. Simulations were performed for $m = 2$ and $D = 0, 2, 10$ for the following sets of discretization parameters:
a) $\Delta r = \Delta z = 0.01, \quad \Delta t = 0.001, \quad N_p = 65$.
b) $\Delta r = \Delta z = 0.005, \quad \Delta t = 0.001, \quad N_p = 130$.

In figure 1 we show the time evolution of the radial coordinate r_0 at $z = 0$ for $D = 0, 2, 10$ using the finer grid. The arrow in the figure mark the computed oscillation period for each D value.

Table 1 shows the values of the exact and calculated oscillation period, T_e, T_c respectively, the relative error in the drop volume, e_V, and the relative error in the total energy of the two fluid system, e_E at $t = 2.5$. It can be concluded that first order convergence rate with respect to space is achieved.

case	D	T_e	T_c	e_V	e_E
a	0	2.2214	2.2250	1.0980e-03	7.6821e-04
a	2	3.3933	3.4000	7.3875e-04	4.5782e-04
a	10	6.1509	6.1600	4.1336e-04	2.5347e-04
b	0	2.2214	2.2250	5.5150e-04	3.8332e-04
b	2	3.3933	3.4000	3.8075e-04	2.0735e-04
b	10	6.1509	6.1600	2.2936e-04	6.3190e-05

Table 1: Comparison with analytical solution

5.2 Droplet and bubble break up simulations

A set of numerical experiments have been carried out for different D values, starting always with the same initial conditions described in [12]. We present here the results for $D = 0$ and $D = 100$, which correspond to droplet and bubble behavior. The fixed domain for the level set computations is set to $\Omega_D = [-2,2] \times [0,8]$, the time step is set variable following the same criteria as in [9] and the simulations have been performed with two different mesh sizes:
a) $\Delta r = \Delta z = 0.01, \quad N_p = 201$
b) $\Delta r = \Delta z = 0.005, \quad N_p = 301$
In table 2 the non-dimensional pinch-off time, t_p and the relative error in the volume of the inner fluid, E_V, are listed for both grids. Note that the non-dimensional pinch-off time increases with D, but when converting it to real time, multiplying by t_0 for each case, it actually diminishes as D increases, in accordance with the physical evidence

 ISBN 13: 978-84-613-5510-5

D	t_p (coarse)	t_p (fine)	E_V (coarse)	E_V (fine)
0	0.4551	0.4551	1.2036 E-03	1.4499 E-03
100	1.9300	1.8956	5.6643 E-04	1.83060 E-04

Table 2: Pinch-off time and relative error in volume

that bubbles breakup sooner than droplets. The relative error in drop volume is always less than 0.1%.

The computed front profiles are almost the same for the coarse and fine grids, and we show in figure 3 and 4 the time evolution up and beyond the pinch-off time for $D = 0$ and $D = 100$ respectively. For $D = 0$ the front profile overturns before first and second pinch-off event and a satellite drop is formed. At the opposite extreme, $D = 100$, the air bubble separation occurs exhibiting the characteristic symmetric cone shape and there in not satellite bubble formation.

Finally, the calculated scaling results for $D = 0$ and $D = 100$ are shown in Fig. 2, which plots $\log r_{\min}$ versus $\log \tau$. The linear fit yielded 0.67 and 0.56 for the computed power law exponents, in excellent agreement with known results [6, 12].

References

[1] D. Adalsteinsson and J.A. Sethian, *The Fast Construction of Extension Velocities in Level Set Methods*, J. Comp. Phys. **148** (1999) 2–22.

[2] O. A. Basaran, , AIChE Journal **185** (2002) 271–288.

[3] J. C. Burton and P. Taborek, *Bifurcation from Bubble to Droplet Behavior in Inviscid Pinch-off*, Phys. Rev. Lett., **101** (2008) 21452.

[4] J. Eggers, *Nonlinear dynamics and breakup of free-surface flows*, Rev. Mod. Phys., **48** 2002) 1842–1848.

[5] J. Eggers and E. Villermaux, , Rep. Prg. Phys., **71** (2008) 036601.

[6] J. Eggers M. A. Fontelos, D. Leppinen and J. H. Snoeijer, , Phys. Rev. lett, **98** (2007) 094502.

[7] M. Garzon, D. Adalsteinsson, L.J. Gray and J.A. Sethian, *A coupled level set-boundary integral method for moving boundaries simulations*, Interfaces and Free Boundaries, **7** (2005) 277–302.

[8] M. Garzon, J.A. Sethian, *Wave breaking over sloping beaches using a coupled boundary integral-level set method*, International Series of Numerical methods **154** (2006) 189–198.

 ISBN 13: 978-84-613-5510-5

[9] M. Garzon,L.J. Gray and J.A. Sethian, *Numerical simulation of non-viscous liquid pinch off using a coupled level set-boundary integral method*, J. Comp Phys. **228** (2009) 6079–6106.

[10] L.J. Gray, M. Garzon,V. Mantič and E. Graciani, *Galerkin boundary integral analysis for the axisymmetric Laplace equation*, Int. J. Numer. Meth. Engrg. **66** (2006) 2014–2034.

[11] L.J.Gray, M. Garzon, *On a Hermite boundary integral approximation*, Computers and Structures **83** (2005) 889–894.

[12] D. Leppinen and J. R. Lister, , Phys. of Fluids **15** (2003) 568–578.

[13] M. Nitsche and P. H. Steen, , J. Comp Phys. **200** (2004) 299–324.

[14] S. Osher and J.A. Sethian, *Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations*, Journal of Computational Physics **79** (1988) 12–49.

[15] J.A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press (1999).

[16] S.T. Thoroddsen, *Micro-dropslets and micro-bubbles. Imaging motion at small scales*, Nus. Engineering research news **22** (2007).

 ISBN 13: 978-84-613-5510-5

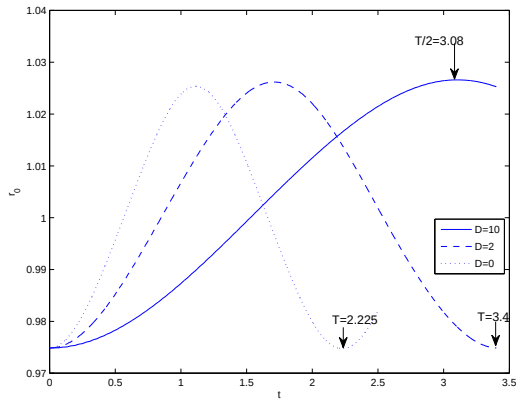


Figure 1: Computed oscillation periods

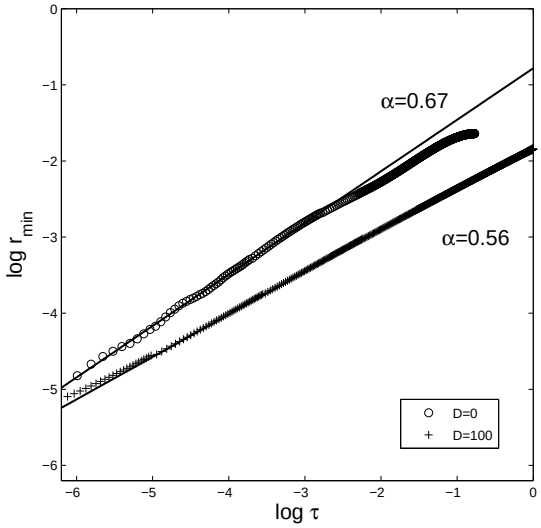


Figure 2: Scaling of $r_{\min}$ at pinch-off for $D = 0$ and $D = 100$.

 ISBN 13: 978-84-613-5510-5

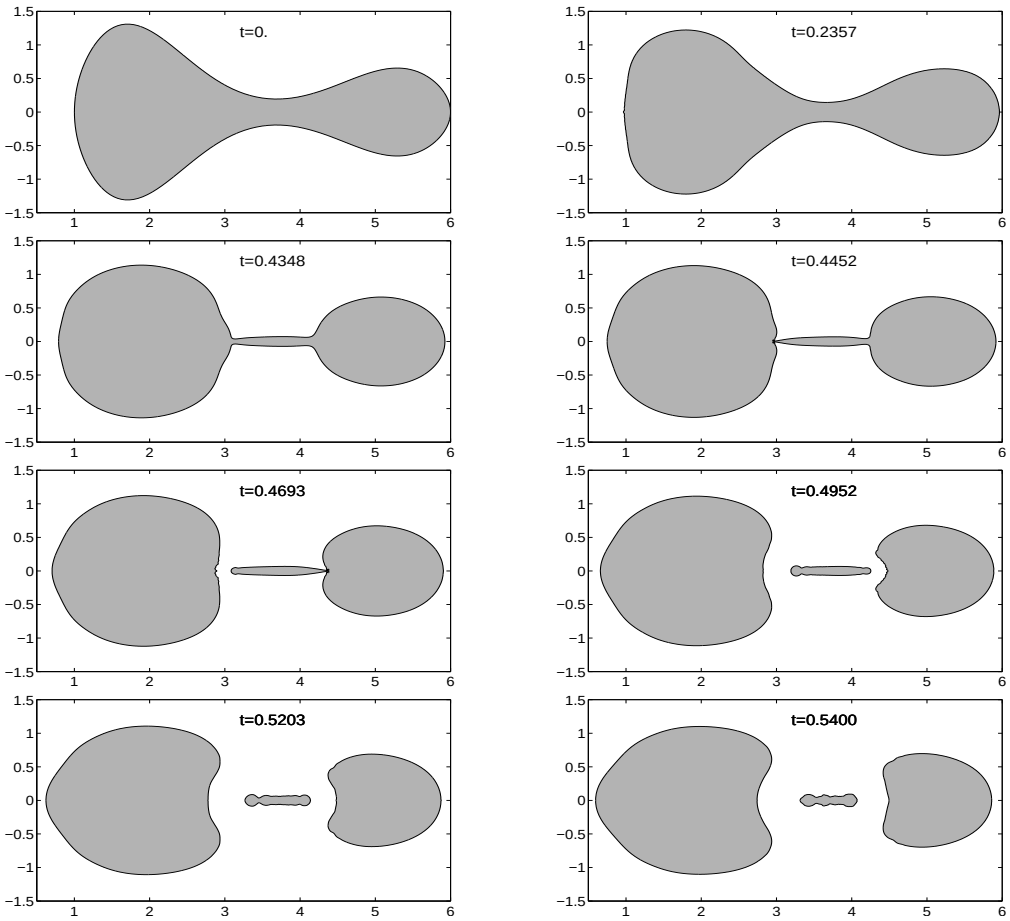


Figure 3: Front profiles at indicated times, $D = 0$

 ISBN 13: 978-84-613-5510-5

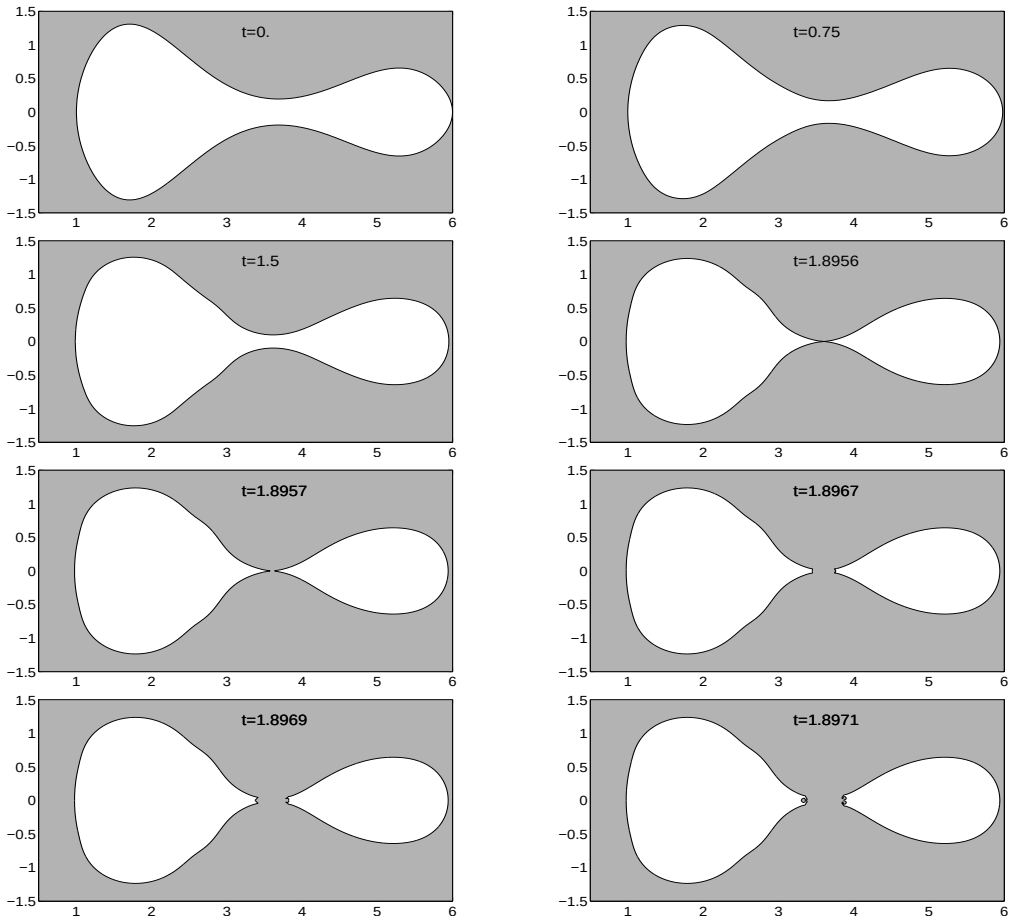


Figure 4: Front profiles at indicated times, $D = 100$

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Mathematical modelling of 1-safe Petri Nets as an special type of Discrete Dynamical Systems

Juan L. G. Guirao[a], Fernando L. Pelayo[b] and Jose C. Valverde[c]

[a] *Department of Applied Mathematics, Polytechnic University of Cartagena*

[b] *Department of Computing Systems, University of Castilla - La Mancha*

[c] *Department of Applied Mathematics, University of Castilla - La Mancha*

emails: Juan.Garcia@upct.es, FernandoL.Pelayo@uclm.es*,
Jose.Valverde@uclm.es

Abstract

This paper presents the second step in the line of applying the theory of discrete dynamical systems to the analysis of Concurrent Computing Systems. In order to do that, Petri Nets are properly modelled as discrete dynamical systems, so defining the corresponding phase space, which is endowed with an original quasi-pseudometric, and the evolution operator of the system. We conclude with a summary of several conclusions about the dynamics of this kind of dynamical systems and with an outlook of intended further research directions.

Key words: Formal Computing Science, Petri Net, Quasi-Pseudometric, Discrete Dynamical System.

1 Introduction

Petri Nets, PNs, have been the first Formal Model proposed for the specification of the behaviour of concurrent and distributed systems. The great success of PNs can be measured not only for the amount of practical applications of them, but also for the size and usability of the developments of the theoretical aspects, which range from the analysis of simple models of nets to the definition of extensions of PNs capturing almost all the features which deserve to be studied in these systems.

Software designers work happily with Process Algebras, PAs, as a consequence of the short distance between them and the pseudo-code or even the programming language they use, but PAs are not able, in general, to capture true concurrency, and even formal verification is a bit harder than it is in other formalisms like PNs.

PNs can easily describe the evolution of systems whose states have a distributed nature, in this line the state of a system is represented by a *marking*, which is a set

 ISBN 13: 978-84-613-5510-5

of *tokens* distributed among a set of *places*, therefore an state is essentially distributed and it makes easier the representation of *reachability, concurrency, mutual exclusion, non-determinism* and so on.

The theory of dynamical systems is one of the major mathematical disciplines closely intertwined with most of the areas of Science. Its core is the study of the global orbit structure of maps and flows, and the invariant properties under small perturbations. Its concepts and methods have stimulated research in many fields, having given rise to a vast new area of applied dynamics, also named nonlinear science.

In our case, we are concerning with mathematical modelling of Petri nets as symbolic (finite) dynamical systems. Symbolic dynamical systems is a class of particular importance not only for its own mathematical interest, but also for the theory of smooth dynamical systems, because in many respects symbolic dynamical systems serve as models for smooth ones. Moreover, in some cases, symbolic dynamical systems can be used to code some smooth systems.

The characteristic feature of dynamical theories is the emphasis on the asymptotic behaviour, that is, properties related with the changes of the states of the system as time goes to infinity.

The most general and accepted notion of a dynamical system includes the set of all possible *states* of the system and the *evolution law* in *time*

In this sense, we can state a formal definition of a dynamical system (see [1, 5, 11]), as follows:

Definition 1. *A dynamical system is a triple (X, τ, Φ), where X is a set, τ is a subset of $\mathbb{R}$ which is a monoid, and $\Phi : \tau \times X \to X$ is a function verifying:*

1. $\Phi(0, x) = x \quad \forall x \in X$, *i.e.,* $\Phi_0 = id_X$
2. $\Phi(t, \Phi(s, x)) = \Phi(t + s, x) \quad \forall t, s \in \tau, \forall x \in X$

The set X is called the *state space* (or *phase space*) of the system. Very often, the state space can be characterized by $\mathbb{R}^n$ or a submanifold in it. But, as we will show later, it could also be a finite set as $\mathcal{P}(\{0, 1\}^n)$, which is the one we use. This set is usually a topological space. Also, it is very common that the state space allows to relate each pair of states by means of a distance, making this set a metric space. In fact, it uses to be very interesting to have a complete or compact metric (state) space.

For our purposes, we consider a metric in [4] which provide a structure of complete metric space for the set $\mathcal{P}(\{0, 1\}^n)$. But, in order to describe a more accurate model of the reality in this paper we give a quasi-pseudometric that allow us to recognize when an state is contained inside another previous state in the evolution of the orbit and hence, when we have controlled the variation of the states. Quasi-metric spaces were firstly studied by Wilson [12].

We recall some definitions which will be used later on this paper, see also [8]. Let X be a set and let

$$d : X \times X \to [0, \infty)$$

 ISBN 13: 978-84-613-5510-5

be a function such that for all $x, y, z \in X$,

i) $d(x, x) = 0$,

ii) $d(x, y) \leq d(x, z) + d(z, y)$.

Then d is called a *quasi-pseudometric* on X. If d is a quasi-pseudometric on X, then its *conjugate* denoted by d' on X is such that $d'(x, y) = d(y, x)$ for all $x, y \in X$. Obviously d' is a quasi-pseudometric. Let $\overline{d} = \max\{d, d'\}$. Then $\overline{d}$ is also quasi-pseudometric on X.

Now if d is a quasi-pseudometric such that $d(x, y) + d(y, x) > 0$ for all $x \neq y$, then d is said to *separate points in* X. A quasi-pseudometric that separates points is said a *separating quasi-pseudometric*. When d is a separating quasi-pseudometric, then the topology τ_d on X induced by d is Hausdorff.

If a quasi-pseudometric d on X satisfies $d(x, y) = d(y, x)$ for all $x, y \in X$ in addition to *i)* and *ii)*, then d is called *pseudometric*. A pseudometric d that satisfies $d(x, y) = 0$ if and only if $x = y$ is a metric.

On the other hand, if a quasi-pseudometric d on X satisfies $d(x, y) = 0$ if and only if $x = y$ in addition to (i) and (ii), then d is called *quasi-metric*. A symmetric quasi-metric d is a metric.

Depending on the monoid τ, it is distinguished between continuous (time) dynamical systems, when $\tau = \mathbb{R}$ ($\mathbb{R}^+ \cup \{0\}$ or $\mathbb{R}^- \cup \{0\}$); and discrete (time) dynamical systems, provided that $\tau = \mathbb{Z}$ ($\mathbb{N} \cup \{0\}$ or $\mathbb{Z}^- \cup \{0\}$). In this work, we consider a particular case of discrete.

The function $\Phi : \tau \times X \to X$ is called the *evolution operator* and, generally, it is a continuous function in the state variable and if $\tau = \mathbb{R}$ ($\mathbb{R}^+ \cup \{0\}$ or $\mathbb{R}^- \cup \{0\}$), it is also continuous in the time variable. This continuity is supposed to be with respect to the metric in X.

This paper is organized as follows. The next section provides an overview on Petri nets and introduces some important concepts in order to consider the evolution by executing an arbitrary number of transitions simultaneously, what allows us to model Petri nets as discrete dynamical systems. Afterwards, in section 3, we set out the model, defining the corresponding phase space, which is endowed with an pioneer quasi-pseudometric, and the evolution operator of the system. Finally, we conclude with a summary of several conclusions about the dynamics of this kind of dynamical systems and with an outlook of intended further research directions.

2 Petri Nets

The representation of a Petri Net is a graph which has two kinds of nodes: places and transitions. Places are usually related to conditions or states, whereas transitions are associated with events or actions, which cause the changes of state in a system. The arcs in the net represent the conditions that must be fulfilled for executing an action (firing a transition), and the new conditions or states obtained after firing that transition.

 ISBN 13: 978-84-613-5510-5

Definition 2. *An Ordinary Petri Net (OPN) is a triple $N = (P, T, F)$ consisting of two sets P and T, and a relation F defined over $P \cup T$, such that:*

1. $P \cap T = \emptyset$
2. $F \subseteq (P \times T) \cup (T \times P)$
3. $dom(F) \cup cod(F) = P \cup T$

P is said to be the *set of places*, T is called the *set of transitions* and F is named the *flow relation*. F relates places and transitions by arcs connecting them. In the classical representation of PNs, places are circles and transitions are rectangles.

Let X be the set $X = P \cup T$. Then, for all $x \in X$ two sets are defined:

- ${}^\bullet x = \{y \in X \,|\, (y, x) \in F\}$ (precondition set of x),
- $x^\bullet = \{y \in X \,|\, (x, y) \in F\}$ (postcondition set of x).

Example 1. *Let $N = (P, T, F)$ be an Ordinary Petri Net, where:*

$$P = \{p_1, p_2, p_3\}$$
$$T = \{t_1, t_2\}$$
$$F = \{(p_1, t_1), (p_2, t_1), (t_1, p_3), (p_3, t_2)\}$$

This Petri Net is graphically represented in Fig. 1.

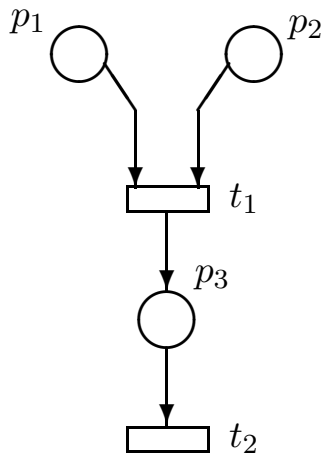


Figure 1: Example of Petri Net

The state of a system described by a PN is captured by means of the so called *Markings*. They are defined as follows.

 ISBN 13: 978-84-613-5510-5

Definition 3. *Let $N = (P, T, F)$ be an Ordinary Petri Net. A function $M : P \longrightarrow \mathbb{N}$ is a Marking of N. Thus, (P, T, F, M) is a Marked Ordinary Petri Net, MOPN.*

Markings of Petri Nets are graphically represented by including in the places as many points as tokens.

We are dealing with those Petri Nets which can have 1 token at most in any place, this fact imposes some requirements on the PN, but to begin with we think they are the appropriate set of PNs. These PNs are called *1-safe* or just *safe*

Given a MOPN (P, T, F, M) with $P = \{p_1, \ldots, p_n\}$, a Marking, $\mathcal{M}$ of it which has tokens (m) in places $p_{i_1}, \ldots, p_{i_m}$ with $m \leq n$, will be codified by a binary n-tuple containing $1's$ in $p_{i_1}, \ldots, p_{i_m}$ positions and the remainder $n - m$ positions contain $0's$.

The semantics of a MOPN is defined both by the following firing rule which establishes when a transition can be fired and by the marking obtained after firing.

Definition 4. *Let $N = (P, T, F, M)$ be a MOPN. A transition $t \in T$ is enabled at marking M, denoted by $M[t\rangle$, if for all place $p \in P$ such that $(p, t) \in F$, we have $M(p) > 0$ ($M(p) = 1$, in the particular case we are dealing with).*

An enabled transition M can be fired, thus producing a new marking, M':

$$M'(p) = M(p) - W_f(p, t) + W_f(t, p) \quad \forall p \in P$$

where

$$\begin{cases} W_f(x) = 1 & if\ x \in F \\ W_f(x) = 0 & if\ x \notin F, \end{cases}$$

for all $x \in (T \times P) \cup (P \times T)$. It is denoted by $M[t\rangle M'$.

We would like to note that since a place can belong to the precondition set of more than one different transitions, a token in it could potentially enable more than one transition and, after firing one of them (transitions), more than one different marking can be reached. This fact has lead us to consider as Phase Space not the set of binary n-tuples but the set of all its subsets, in order to properly capture these cases.

Example 2. *In the MOPN of figure 2 is presented an scenario with a place p_2 which belongs to the precondition set of both t_1 and t_2*

Both transitions t_1 and t_2 are enabled in the MOPN of fig.2 where the vector state would be $(1, 1, 1, 0, 0)$. Starting from this "state" and after firing transition t_1 the net will evolve to the one of figure 3, but this is not the only possible firing from such "marking", if t_2 would be fired the marking of the net would be that of figure 4.

 ISBN 13: 978-84-613-5510-5

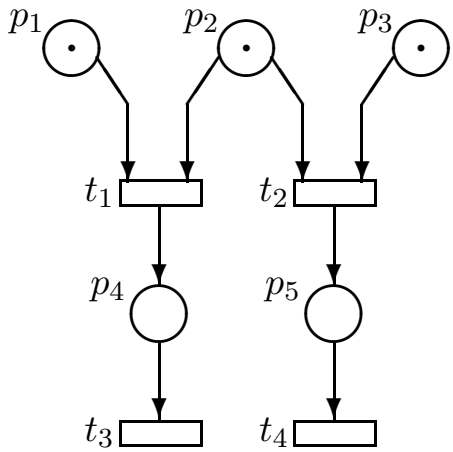


Figure 2: MOPN whose vector state is $(1, 1, 1, 0, 0)$

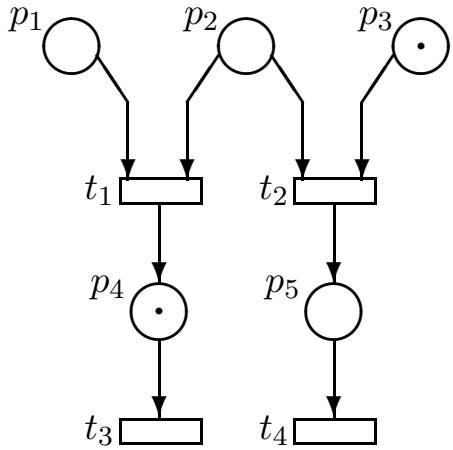


Figure 3: MOPN whose vector state is $(0, 0, 1, 1, 0)$

Therefore, in order to properly capture the whole dynamics of a PN, the DDS must be able to evolve from $(1, 1, 1, 0, 0)$ to both $(0, 0, 1, 1, 0)$ and $(1, 0, 0, 0, 1)$ and taking into account that the DDS is deterministic our new phase space is (for a PN like the one of figures 2 - 4) $\mathcal{P}(\{0, 1\}^5$ and from the "point" $\{(1, 1, 1, 0, 0)\}$ the DDS goes to the "point" $\{(0, 0, 1, 1, 0), (1, 0, 0, 0, 1)\}$. We would like to note that this phase space only has $2^{(2^5)}$ elements, i.e., around 4.3 billion states.

The formal definition of the DDS will be provided later on.

Coming back to the formal description of the evolution of a PN, we would like to note that the primary definition of firing a transition can be extended in order to consider the evolution by executing an arbitrary number of transitions simultaneously.

 ISBN 13: 978-84-613-5510-5

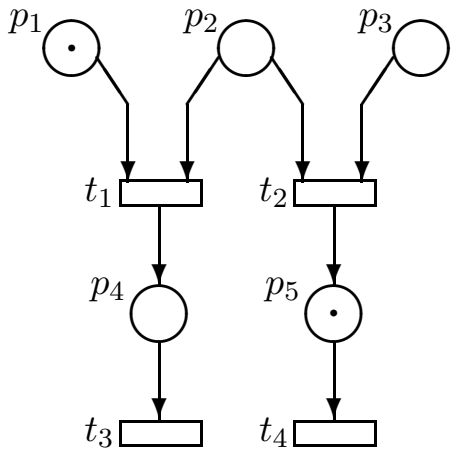


Figure 4: MOPN whose vector state is $(1, 0, 0, 0, 1)$

Definition 5. *Let $N = (P, T, F, M)$ be a MOPN. Let $R \subseteq T$ be a subset of transitions. It is said that all transitions in R are enabled at marking M, denoted by $M[R\rangle$, if and only if (iff)*

$$M(p) \geq \sum_{t \in R} W_f(p, t), \ \forall p \in P,$$

where $W_f(p, t)$ has been defined previously.

Moreover, we say that a multiset of transitions R is enabled at marking M iff

$$M(p) \geq \sum_{t \in T} W_f(p, t) \cdot R(t),$$

for all $p \in P$.

The firing of a multiset of transitions R at the marking M generates a new marking M', defined by:

$$M'(p) = M(p) - \sum_{t \in T} (W_f(p, t) - W_f(t, p)) \cdot R(t)$$

This evolution of the PN in a single step is denoted by $M[R\rangle M'$.

This is the way in which a PN evolves and it is assumed the best in terms of accuracy to the real behaviour of concurrent computing systems.

3 Petri Nets Modelled as Discrete Dynamical Systems

The discrete dynamical system which encodes the MOPN $N = (P, T, F, M)$ is the triple (X, τ, Φ), where:

- $X = \mathcal{P}(\{0, 1\}^n)$ is the set of all subsets of $\{0, 1\}^n$, being n the number of places of the MOPN. This is a finite set of $2^{(2^n)}$ elements.

 ISBN 13: 978-84-613-5510-5

- τ is the monoid $\mathbb{N} \cup \{0\}$
- $\Phi : \tau \times X \to X$ is the evolution operator Φ verifying:
  1. $\Phi(0, A) = A \quad \forall A \in X$, i.e., $\Phi_0 = id_X$
  2. $\Phi(1, A) = B \quad A, B \in X$ where:
     - $A = \{x_1, \ldots, x_k\}$ where $x_i \in \{0,1\}^n$ encodes Markings of the MOPN N
     - $B = \cup_{i=1}^{k} B_i$
     - $B_i = \cup_{j=1}^{t} \{y_i^j\}$, i.e. the union of all (t) possible reachable markings from x_i, defined by $x_i[R_i\rangle y_i^j$ being R_i the set of transitions of the net enabled at marking x_i
  3. $\Phi(t, \Phi(s, A)) = \Phi(t + s, A) \quad \forall t, s \in \tau, \forall A \in X$

As we commented above, for our pretensions to formalize Petri nets as discrete dynamical systems, we consider as phase space the set $\mathcal{P}(\{0,1\}^n)$. Now, we have to determine a topology on this set, such that the pair $(\mathcal{P}(\{0,1\}^n), \tau)$ be a complete or compact topological (state) space. In order to do that, following some of the ideas of reference [8], we begin defining on $\{0,1\}^n$ an original metric induced from the Bayre metric (see [2]) given by

$$d(x, y) = \frac{1}{2^{l(x \sqcap y)}} - \frac{1}{2^n}, \quad x, y \in \{0,1\}^n \tag{1}$$

where $l(x \sqcap y)$ is the length of the longest common initial part of the vectors x and y.

Theorem 1. *The function d defined in (1) is a metric.*

Proof. Effectively, from the definition it is obvious that d is symmetric and

$$d(x, y) = 0 \Leftrightarrow x = y$$

On the other hand, for all $x, y, z \in \{0,1\}^n$ it is true that

$$d(x, y) \leq d(x, z) + d(z, y)$$

To check this, it is sufficient to observe that for all $x, y \in \{0,1\}^n$ the function $d(x, y)$ "shows the coincidence grade of the initial part of x and y ". So, if by reduction to the absurd, we suppose that for some $x, y, z \in \{0,1\}^n$ is true that

$$d(x, y) > d(x, z) + d(z, y)$$

and we call k, l, m the length of the longest common initial part of the pairs of vectors (x, y), (x, z) and (z, y), then $k < l, m$. Note that if $k \geq l$ (or $k \geq m$) then

$$\frac{1}{2^k} - \frac{1}{2^n} \leq \frac{1}{2^l} - \frac{1}{2^n} \leq \frac{1}{2^l} - \frac{1}{2^n} + \frac{1}{2^m} - \frac{1}{2^n}$$

what is inconsistent with the supposition made before.

But, if $k < l, m$ the coincidence grade of the initial part of x and z and also z and y is greater than k. Thus, x, y, z have a initial coincident part whose length is the minimum of l and m, which is greater than k, what contradicts that k is the longest common initial part of the pair of vectors x and y. □

 ISBN 13: 978-84-613-5510-5

At this point, taking into account this metric d, we can define the distance between a vector $x \in \{0,1\}^n$ and a subset B of vectors of $\{0,1\}^n$, i.e., an element in $\mathcal{P}(\{0,1\}^n)$ in this manner

$$\mathtt{d}(x,B) = \min\{d(x,y) : y \in B\}$$

and, consequently, we can establish the distance between two elements A, B in $\mathcal{P}(\{0,1\}^n)$ as

$$D(A,B) = \max\{\mathtt{d}(x,B) : x \in A\} \tag{2}$$

Now, in view of the reasoning before, it is easy to check the following result

Corollary 1. *The application D defined in (2) determines a quasi-pseudometric on $\mathcal{P}(\{0,1\}^n)$, which is not a quasi-metric.*

Remark 1. *Observe that we do not have the property of quasi-metrics, neither symmetry. For instance, if we consider a Petri net with three nodes and consider the subsets $A, B \in \mathcal{P}(\{0,1\}^3)$, given by $A = \{(0,0,1)\}$, $B = \{(0,0,0),(0,0,1)\}$, then it is easy to check that $D(A,B) = 0$, while $D(B,A) = \frac{1}{2^2} - \frac{1}{2^3} \neq 0$*

In fact, as in [8], we have a quasi-pseudometric verifying $D(A,B) = 0$ if an only if $A \subseteq B$ and therefore we will denote it $d_{\subseteq}$.

Coming back to the PNs world, it is immediate that if $A \subseteq B$ then the reachable set of markings from every marking of A is included in the reachable set of markings from every marking of B.

Let us consider $A, B \in X$ with $A \subseteq B$ and $p \in \tau$ such that $\Phi(p,B) = A$, then we can assure that, as system evolves, the reachability of B (the union of the sets of reachability of each marking codified in B, is being reduced into the reachability of A and this fact introduces a notion quite similar to the periodicity of points of X by Φ, very interesting to research in.

Nevertheless, as $d_{\subseteq}(A,B)+d_{\subseteq}(B,A) > 0$ for every $A, B \in \mathcal{P}(\{0,1\}^n)$, $d_{\subseteq}$ separates points, therefore, the topology $\tau_{d_{\subseteq}}$ on X induced by $d_{\subseteq}$ is Hausdorff.

Theorem 2. *$(\mathcal{P}(\{0,1\}^n), D)$ is a compact (topological) space with the topology $\tau_{d_{\subseteq}}$ induced by the quasi-pseudometric $d_{\subseteq}$.*

Proof. Note that, if two subsets A, B are different, then the minimum distance between them could be

$$\frac{1}{2^{n-1}} - \frac{1}{2^n} = \frac{2-1}{2^n} = \frac{1}{2^n}$$

Thus every element $A \in \mathcal{P}(\{0,1\}^n)$ is an open set n the topology $\tau_{d_{\subseteq}}$ induced by the quasi-pseudometric $d_{\subseteq}$, because it coincides with the open ball $B(A, \frac{1}{2^{n+1}})$, with centre A and radius $\frac{1}{2^{n+1}}$. Therefore, using the topological notion of compact space introduced by Alessandrov and taking into account that $\mathcal{P}(\{0,1\}^n)$ is a finite set, for every cover of $\mathcal{P}(\{0,1\}^n)$ constituted by a family of open sets, we can found a finite number of open sets of this family which contain all the space $\mathcal{P}(\{0,1\}^n)$. Hence, $\mathcal{P}(\{0,1\}^n)$ is a compact (topological) space. □

 ISBN 13: 978-84-613-5510-5

4 Conclusions and further research directions

Following [11], the main goals in the study of a dynamical system are both giving a complete characterization of the geometry of its orbit structure and analyzing wether or not this structure remain when the system is perturbed slightly.

Remember that the ordered subset $Orb(x_0) = \{x \in X : x = \Phi(t, x_0), t \in \tau\}$ of the state space X is named the *orbit* of the present (or initial) state x_0. Note that orbits of continuous dynamical systems are curves in the state space, while orbits of discrete dynamical system are sequences of points in the state space.

Since in our particular case of discrete dynamical system, we have a finite state space, it is easy to know that every orbit is either periodic or eventually periodic. Therefore, every orbit is an invariant set of the system. However, it is not so easy to determine a priori the different coexistent periods of its orbits, neither the existence of fixed points and their basin of attraction (see [3], [9]).

Taking into account what we said above, we can conclude that the ω-limit set and the α-limit set of the system coincide with the set of periodic orbits.

On the other hand, observe that the quasi-pseudometric $d_{\subseteq}$ express the difference between each ordered pair of elements belonging to $\mathcal{P}(\{0,1\}^n)$. Besides, it is equals 0 when the first set is a subset of the second one. This infers a new possibility in the study of the dynamics, because when the evolution of the initial set arrives to a subset of it, we have "controlled" the future evolution. So, we could define a kind of quasi-pseudoperiodic orbits or quasi-pseudofixed points, which are in fact periodic or fixed orbits considering the quasi-pseudometric $d_{\subseteq}$.

On the other hand, not every result relative to orbit structure for the well known discrete dynamical systems given by continuous map of the interval, works here. For instance, the famous Sharkovskii theorem (see [10]) is not true for a system with a period three, because we can only have a finite number of different periodic orbits in the system (one for each initial state).

Also, those questions which can be studied by means of the differentiability of the evolution operator, as attraction of certain orbits, are now very difficult to state.

Obviously, one could count all the diverse orbits, but, for a state space big enough, it could be very hard.

Another open problem is to analyze the perturbations, or even what a perturbation could be here, of these kind of discrete dynamical systems. Often, this problem is formalized mathematically by adding a parameter in the expression of the evolution operator (see [5] or [6]). But in our case, the evolution operator is not given by a formula and even to formalize a perturbation of a system is a problem.

Acknowledgements

This work has been partially supported by projects TIN2009-14312, PAC08-0173-4838, PEII09-0184-7802, MTM2008-03679 & CGL07-66440-C04-03

 ISBN 13: 978-84-613-5510-5

References

[1] D. K. Arrowsmith and C. M. Place, *An Introduction to Dynamical Systems*, Cambridge University Press (1990).

[2] J.W. Baker and E.P. Vink, *A metric approach to control flow semantics*, Annals of the New York Academy of Sciences 806, 11–27 (1996).

[3] J.W. Baker and E.P. Vink, *Denotational models for programming languages: Applications of Banach's fixed point theorem*, Topology Appl. 85, 35–52, (1998).

[4] J.L. Guirao, F.L. Pelayo and J.C. Valverde, *Modeling dynamics of Concurrent Computing Systems*, Comput. Math. Appl. (to appear).

[5] J. Hale and H. Koak, *Dynamics and Bifurcations*, Springer, New York, Heidelberg, Berlin, (1991).

[6] P. Holmes and D. Whitey, *Bifurcations of one-and two-dimensional maps*, Phil. Trans. R. Soc. Lond. 311, 43-102 (1984).

[7] G. A. Petri, *Communications with Automata*, Technical Report RADC-TR-65-377, New York University (1966).

[8] J. Rodriguez-Lopez, S. Romaguera and O. Valero, *Denotational semantics for programming languages, balanced quasi-metrics and fixed points*, International Journal of Computer Mathematics, 85:3, 623 - 630 (2008).

[9] S. Romaguera and M. Sanchis, *Applications of utility functions defined on quasi-metric spaces*, J. Math. Anal. Appl., 283, 219–235 (2003).

[10] P. Stefan, *A theorem of Sarkovskii on the existence of periodic orbits of continuous endomorphisms of the real line*, Commun. Math. Phys. 54, 237-248 (1977).

[11] S. Wigginns, *Introduction to Applied Nonlinear Systems and Chaos*, Springer, New York (1990).

[12] W.A. Wilson *On quasi-metric spaces*, Amer. J. Math. 53, 675-684 (1931).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Non-bounded Petri Nets as Discrete Dynamical Systems

Juan L. G. Guirao[a], Fernando L. Pelayo[b] and Jose C. Valverde[c]

[a] *Department of Applied Mathematics, Polytechnic University of Cartagena*

[b] *Department of Computing Systems, University of Castilla - La Mancha*

[c] *Department of Applied Mathematics, University of Castilla - La Mancha*

emails: Juan.Garcia@upct.es, FernandoL.Pelayo@uclm.es*,
Jose.Valverde@uclm.es

Abstract

This paper provides a codification of Non-bounded Ordinary Petri Nets as Discrete Dynamical Systems. This work extends the considerations made by the authors in this line over safe Petri Nets. The *transition vector* plays a key role to provide an appropriate metric for the underlying phase space.

Key words: Formal Computing Science, Petri Net, Quasi-Pseudometric, Discrete Dynamical System.

1 Non-bounded Petri Nets

Given the MOPN $N = (P, T, F, M)$ where:

- P is the finite *set of places*
- T is the finite *set of transitions*
- $P \cap T = \emptyset$. In the classical representation of PNs, places are circles and transitions are rectangles
- F is the *flow relation* which relates places and transitions by arcs connecting them.
- $F \subseteq (P \times T) \cup (T \times P)$
- $dom(F) \cup cod(F) = P \cup T$
- $M : P \longrightarrow \mathbb{N}$ is a Marking of N. Markings of Petri Nets are graphically represented by including in the places as many points as tokens.

 ISBN 13: 978-84-613-5510-5

Example 1. *Figure 1 shows the MOPN which models the classical Producer/Consumer problem with a buffer of capacity 5, where transition* $t1$ *represents "producing" an item,* $t2$ *"putting" the item in the buffer,* $t3$ *"taking out" an item from the buffer and* $t4$ *"consuming" the item.*

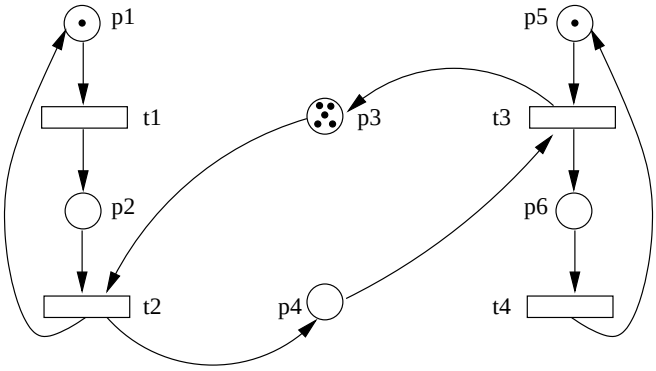


Figure 1: Initial Marking of the PN modelling Producer/Consumer with a 5-buffer

Given a MOPN (P, T, F, M) with $P = \{p_1, \ldots, p_n\}$, a Marking $\mathcal{M}$ of it which has tokens (m) in places $p_{i_1}, \ldots, p_{i_t}$ with $t \leq m$, will be codified by a n-tuple containing in every i_j-position $j \in \{1 \ldots t\}$ the number of tokens of place p_{i_j} and the remainder $n-t$ positions contain $0's$.

The codification of the MOPN modelling the initial state of the classical Producer/Consumer problem with a buffer of capacity 5, as shown in fig. 1, will be (1,0,5,0,1,0).

Given a MOPN $N = (P, T, F, M)$, a transition $t \in T$ is enabled at marking M, denoted by $M[t\rangle$, if for all place $p \in P$ such that $(p, t) \in F$, we have $M(p) > 0$.

An enabled transition M can be fired, thus producing a new marking, M':

$$M'(p) = M(p) - W_f(p, t) + W_f(t, p) \quad \forall p \in P$$

where

$$\begin{cases} W_f(x) = 1 & if\ x \in F \\ W_f(x) = 0 & if\ x \notin F, \end{cases}$$

for all $x \in (T \times P) \cup (P \times T)$. It is denoted by $M[t\rangle M'$

We would like to note that since a place can belong to the precondition set of more than one different transition, a token in it could potentially enable more than one transition and, after firing one of them (transitions), more than one different marking can be reached. This fact has lead us to consider as Phase Space not the set of n-tuples of natural numbers but the set of all its subsets, in order to properly capture these cases.

 ISBN 13: 978-84-613-5510-5

Let $N = (P, T, F, M)$ be a MOPN and $R \subseteq T$ be a subset of transitions. It is said that all transitions in R are enabled at marking M ($M[R\rangle$), if and only if (iff)

$$M(p) \geq \sum_{t \in R} W_f(p,t), \ \forall p \in P,$$

Moreover, we say that a multiset of transitions R is enabled at marking M iff

$$M(p) \geq \sum_{t \in T} W_f(p,t) \cdot R(t),$$

for all $p \in P$.

Transition Vector of a Marking: Given a MOPN $N = (P, T, F, M)$ the transition vector of the marking M is the binary $n-vector$, being n the number of transitions of N, containing 1 in position j iff transition t_j is enabled at marking M and 0 otherwise $\forall j \in \{1 \dots n\}$

The firing of a multiset of transitions R at the marking M generates a new marking M', defined by:

$$M'(p) = M(p) - \sum_{t \in T} (W_f(p,t) - W_f(t,p)) \cdot R(t)$$

This evolution of the PN in a single step is denoted by $M[R\rangle M'$.

2 Non-bounded PNs as Discrete Dynamical Systems

The discrete dynamical system which encodes the MOPN $N = (P, T, F, M)$ is the triple (X, τ, Φ), where:

- $X = \mathcal{P}(\mathbb{N}^n)$ is the set of all subsets of $\mathbb{N}^n$, being n the number of places of the MOPN.
- τ is the monoid $\mathbb{N} \cup \{0\}$
- $\Phi : \tau \times X \to X$ is the evolution operator Φ verifying:
  1. $\Phi(0, A) = A \quad \forall A \in X$, i.e., $\Phi_0 = id_X$
  2. $\Phi(1, A) = B \quad A, B \in X$ where:
     - $A = \{x_1, \dots, x_k\}$ where $x_i \in \mathbb{N}^n$ encodes Markings of the MOPN N
     - $B = \cup_{i=1}^{k} B_i$
     - $B_i = \cup_{j=1}^{t} \{y_i^j\}$, i.e. the union of all (t) possible reachable markings from x_i, defined by $x_i[R_i\rangle y_i^j$ being R_i the set of transitions of the net enabled at marking x_i
  3. $\Phi(t, \Phi(s, A)) = \Phi(t + s, A) \quad \forall t, s \in \tau, \forall A \in X$

 ISBN 13: 978-84-613-5510-5

As we commented above, for our pretensions to formalize Petri nets as discrete dynamical systems, we consider as phase space the set $\mathcal{P}(\mathbb{N}^n)$. Now, we have to determine a topology on this set, such that the pair $(\mathcal{P}(\mathbb{N}^n), \tau)$ be a complete or compact topological (state) space. In order to do that, following some of the ideas of reference [8], we have already defined on $\{0,1\}^n$, see [4], (for the case of 1-safe Petri Nets) a metric induced from the Bayre metric (see [2]) given by

$$d(x,y) = \frac{1}{2^{l(x \sqcap y)}} - \frac{1}{2^n}, \quad x, y \in \{0,1\}^n \tag{1}$$

where $l(x \sqcap y)$ is the length of the longest common initial part of the vectors x and y.

Now we have a quite different scenario since in the *binary* one a difference in a place means that every transition which has this place as precondition could potentially be enabled in one case and necessarily disabled in the other.

Nevertheless, in the case we are dealing with, figures 2, 3 and 4 show three markings of the same OPN whose codifications are (0,1,0,5,1,0), (0,1,1,4,1,0) and (0,1,2,3,1,0) respectively, if we compute the distance between the second (fig3) and the first (fig2) and the distance between the second (fig3) and the third (fig4) the same value is obtained, since the first difference in the codification of the markings appears in the third place in both cases, but meanwhile both the MOPN of fig. 3 and the the MOPN of fig. 4 are able to fire the transition $t2$, on the contrary, the MOPN of fig 2 is not able to fire this transition.

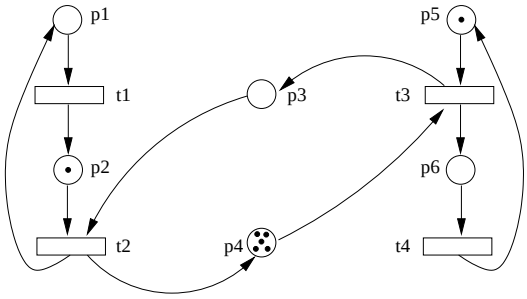


Figure 2: Marking (0,1,0,5,1,0) of the PN. Transition vector (0,0,1,0)

This means that the metric proposed for the binary case is blind to this huge differences on the behaviours of the MOPNs. Therefore, we have chosen as reference for the new metric the **transition vector** previously defined.

As shown in the feet of the figures, it is more accurate to the real behaviour of the PNs we are codifying.

At this point, we consider that the role of x and y (codification of the markings) in the definition of the metric can be played by their corresponding transition vectors of x and y, so we denote as $tv(x)$ the transition vector of the marking x.

Now we redefine the distance between markings:

$$d(x,y) = \frac{1}{2^{l(tv(x) \sqcap tv(y))}} - \frac{1}{2^m}, \quad x, y \in \mathbb{N}^n \tag{2}$$

 ISBN 13: 978-84-613-5510-5

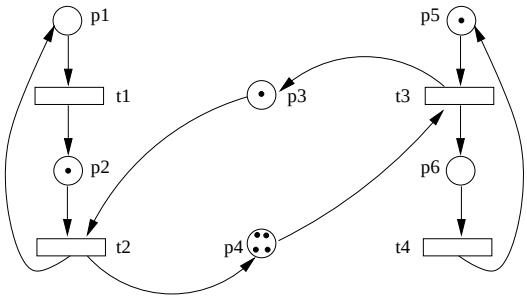


Figure 3: Marking (0,1,1,4,1,0) of the PN. Transition vector (0,1,1,0)

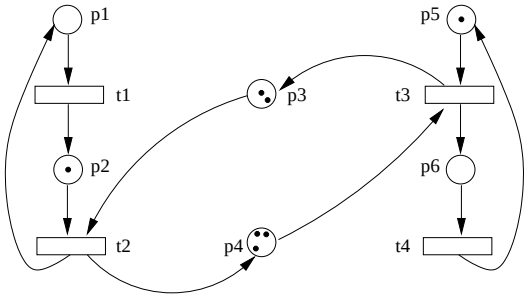


Figure 4: Marking (0,1,2,3,1,0) of the PN. Transition vector (0,1,1,0)

where $l(tv(x) \sqcap tv(y))$ is the length of the longest common initial part of the transition vectors of x and y and $m \in \mathbb{N}$ is the number of transitions of the PN. From this metric d, we can define the distance between a vector $x \in \mathbb{N}^n$ and a subset B of vectors of $\mathbb{N}^n$, i.e., an element in $\mathcal{P}(\mathbb{N}^n)$ in this manner

$$\mathtt{d}(x, B) = \min\{d(x, y) : y \in B\}$$

and, consequently, we can establish the distance between A, B in $\mathcal{P}(\mathbb{N}^n)$ as

$$D(A, B) = \max\{\mathtt{d}(x, B) : x \in A\} \tag{3}$$

The application D defined in (3) determines a quasi-pseudometric on $\mathcal{P}(\mathbb{N}^n)$.

3 Conclusions and future work

We have presented a metric for the set of markings of every Non-bounded Ordinary Petri Net. This metric is different to the one of the case of safe Petri Nets [4]. It is based on the (enabled) transition vector associated to each Marking, which gives very valuable information over the behaviour of the corresponding PN.

It has been defined the quasi-pseudometric D over the Phase Space $\mathcal{P}(\mathbb{N}^n)$. This quasi-pseudometric can give a partial order over the Phase Space. This induced partial order reflects behaviours more and less controlled on the original PNs we are codifying.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work has been partially supported by projects TIN2009-14312, PAC08-0173-4838, PEII09-0184-7802, MTM2008-03679 & CGL07-66440-C04-03

References

[1] D. K. Arrowsmith and C. M. Place, *An Introduction to Dynamical Systems*, Cambridge University Press (1990).

[2] J.W. Baker and E.P. Vink, *A metric approach to control flow semantics*, Annals of the New York Academy of Sciences 806, 11–27 (1996).

[3] J.W. Baker and E.P. Vink, *Denotational models for programming languages: Applications of Banach's fixed point theorem*, Topology Appl. 85, 35–52, (1998).

[4] J.L. Guirao, F.L. Pelayo and J.C. Valverde, *Modeling dynamics of Concurrent Computing Systems*, Comput. Math. Appl. (to appear).

[5] J. Hale and H. Koak, *Dynamics and Bifurcations*, Springer, New York, Heidelberg, Berlin, (1991).

[6] P. Holmes and D. Whitey, *Bifurcations of one-and two-dimensional maps*, Phil. Trans. R. Soc. Lond. 311, 43-102 (1984).

[7] G. A. Petri, *Communications with Automata*, Technical Report RADC-TR-65-377, New York University (1966).

[8] J. Rodriguez-Lopez, S. Romaguera and O. Valero, *Denotational semantics for programming languages, balanced quasi-metrics and fixed points*, International Journal of Computer Mathematics, 85:3, 623 - 630 (2008).

[9] S. Romaguera and M. Sanchis, *Applications of utility functions defined on quasi-metric spaces*, J. Math. Anal. Appl., 283, 219–235 (2003).

[10] P. Stefan, *A theorem of Sarkovskii on the existence of periodic orbits of continuous endomorphisms of the real line*, Commun. Math. Phys. 54, 237-248 (1977).

[11] S. Wigginns, *Introduction to Applied Nonlinear Systems and Chaos*, Springer, New York (1990).

[12] W.A. Wilson *On quasi-metric spaces*, Amer. J. Math. 53, 675-684 (1931).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A Piecewise-linearized Algorithm for solving stiff ODEs*

J. Javier Ibáñez[1], Vicente Hernández[1], Pedro A. Ruiz[1] and Enrique Arias[2]

[1] *Instituto de Aplicaciones de las Tecnologías de la Información y de las Comunicaciones Avanzadas, Technical University of Valencia, Camino de Vera s/n, 46022-Valencia (Spain)*

[2] *Albacete Research Institute of Informatics, University of Castilla-La Mancha, Avda. España s/n, 02071-Albacete (Spain)*

emails: jjibanez@dsic.upv.es, vhernand@dsic.upv.es, pruiz@dsic.upv.es, earias@dsi.uclm.es

Abstract

Numerical methods for solving Ordinary Differential Equations (ODEs) have received considerable attention in recent years. In this paper a piecewise-linearized algorithm based on Krylov subspaces for solving Initial Value Problems (IVPs) is proposed. MATLAB versions for autonomous and non-autonomous ODEs of this algorithm have been implemented. These implementations have been compared with other piecewise-linearized algorithms based on Padé approximants, recently developed by the authors of this paper, comparing both precision and computational costs in equality of conditions. Five case studies have been used in the tests that come from biology and chemical kinetics stiff problems. Experimental results show the advantages of the proposed algorithms, especially when the dimension is increased in stiff problems.

Key words: Initial Value Problem (IVP), Ordinary Differential Equation (ODE), Linear Differential Equation (LDE), Piecewise-linearized methods, Padé approximants, Krylov subspace

1 Introduction

Many scientific and engineering problems are described by ODEs where the analytic solution is unknown. In recent years many review articles and books have appeared

*This work has been supported by the Spanish CICYT project CGL2007-66440-C04-03.

 ISBN 13: 978-84-613-5510-5

on numerical methods for integrating ODEs, in stiff cases in particular [1]. Stiff problems appear in many fields of the applied sciences such as biology, chemical kinetics, electronic circuit theory, fluids, etc.

Numerical methods for solving ODEs can be classified into two groups: One step methods (Euler, Runge-Kutta, etc.) and multistep methods (Adams-Bashforth, Adams-Moulton, BDF, etc) [3]. A family of one step methods for solving ODEs are piecewise-linearized [4]. These methods solve an IVP by approximating the right hand side of the corresponding ODE by a Taylor polynomial of degree one. The resulting approximation can be integrated analytically to obtain the solution in each subinterval and yields the exact solution for linear problems. In [4, 5] an exhaustive study of this method is introduced. The proposed method requires a non-singular Jacobian matrix on each subinterval.

In [6] the authors presented a piecewise-linearized method for solving ODEs. This method uses a theorem proved in that article, which enables the approximate solution to be computed at each time step by a block-oriented approach based on diagonal Padé approximations. In this work another approach based on the piecewise-linearized method is introduced. In this case, the matrix-vector product $e^A v$, which appears in these methods, is computed by a Krylov subspace approach. Computational cost and precision of algorithms are compared in equal conditions.

The paper is structured as follows. In Section **??** a piecewise-linearized approach for solving IVPs for ODEs based on Padé approximants is introduced [6]. The new approach for solving ODEs based on the Krylov subspace approach is presented in Section 2. The experimental results are shown in Section 3. Finally, conclusions and future expectations are given in Section 4.

2 A piecewise-linearized algorithm for solving ODEs based on the Krylov subspace approach

In [6] the authors presented a piecewise-linearized method for solving ODEs, based on the following theorem which enables the approximate solution to be computed at each time step by a block-oriented approach based on diagonal Padé approximations.

Theorem 1 ([6]) *Let*

$$\dot{x}(t) = f(t, x(t)), t \in [t_0, t_f], \tag{1}$$

be an ODE with initial value

$$x(t_0) = x_0 \in \mathbb{R}^n,$$

so that the first order partial derivatives of $f(t, x)$ are continuous on $[t_0, t_f] \times \mathbb{R}^n$. Given a mesh $t_0 < t_1 < \cdots < t_{l-1} < t_l = t_f$, ODE (1) can be approximated by a set of LDEs obtained as a result of a linear approximation of $f(t, x(t))$ at each subinterval ([5, 7]),

$$\begin{gathered} \dot{y}(t) = f_i + J_i(y(t) - y_i) + g_i(t - t_i), t \in [t_i, t_{i+1}], \\ y(t_i) = y_i, \ i = 0, 1, \ldots, l-1, \end{gathered} \tag{2}$$

 ISBN 13: 978-84-613-5510-5

The solution of (2) *is*

$$y(t) = y_i + E_{12}^{(i)}(t-t_i)f_i + E_{13}^{(i)}(t-t_i)g_i, \tag{3}$$

where $E_{12}^{(i)}(t-t_i)$ *and* $E_{13}^{(i)}(t-t_i)$ *are blocks* $(1,2)$ *and* $(1,3)$ *of* $E = e^{C_i(t-t_i)}$, *where*

$$C_i = \begin{bmatrix} J_i & I_n & 0_n \\ 0_n & 0_n & I_n \\ 0_n & 0_n & 0_n \end{bmatrix}. \qquad \square$$

If t is replaced by t_{i+1} in (3), the approximate solution of ODE (1) at t_{i+1}, $i = 0, 1, \ldots, l-1$, is given by

$$y_{i+1} = y_i + \Delta t_i E_{12}^{(i)} f_i + \Delta t_i E_{13}^{(i)} g_i,\ \Delta t_i = t_{i+1} - t_i.$$

In this work, another approach based on the piecewise-linearized method is introduced as follows. The approximate solution y_{i+1} given in (3) can be expressed as follows

$$y_{i+1} = y_i + e^{C_i \Delta t_i} v_i, \tag{4}$$

where

$$C_i = \begin{bmatrix} J_i & I_n & 0_n \\ 0_n & 0_n & I_n \\ 0_n & 0_n & 0_n \end{bmatrix}, v_i = \begin{bmatrix} 0_{n\times 1} \\ f_i \\ g_i \end{bmatrix}.$$

The matrix-vector product $e^{C_i \Delta t_i} v_i$ can be obtained by a Krylov subspace method [8, 9]. Given $A \in \mathbb{R}^{n\times n}$ and $v \in \mathbb{R}^n$, it is possible to compute an approximation to vector $e^A v$ by using the Arnoldi method. This approximation is given by

$$e^A v \cong v_{opt} = \beta V_p e^{H_p} e_1, \tag{5}$$

where matrix $H_p = (h_{ij}) \in \mathbb{R}^{p\times p}$ is the Hessenberg matrix obtained from the Arnoldi method and $V_p = [v_1, v_2, \ldots, v_p] \in \mathbb{R}^{n\times p}$, with $\{v_i\}_{i=1,2,\ldots,p}$ an orthonormal basis of the Krylov subspace $K_p = \text{span}\{v, Av, \ldots, A^{p-1}v\}$.

In order to reduce computational and storage costs when we want to compute vector y_{i+1} from (4), it is necessary to modify the classical Arnoldi algorithm without explicitly forming matrix $C_i \Delta t_i$. Algorithm 1 solves IVPs for non-autonomous ODEs by the above piecewise-linearized method based on a Krylov subspace approach. This algorithm uses Algorithm 2 , which computes the approximate solution at t_{i+1} of IVP (1) for non-autonomous ODEs, obtained after the piecewise-linearized process, by a block-oriented implementation of the Krylov subspace approach. Its computational cost is $2n^2p + 6np(p+1) + 2(q + j_{H_p} + 1/3)p^3$ flops, where $j_{H_p} = \max(0, 1 + int(\log_2(||H_p||)))$. It is possible to reduce the computational and storage costs of Algorithm 1 when IVP (1) is autonomous.

 ISBN 13: 978-84-613-5510-5

Algorithm 1 solves IVP (1) by a piecewise-linearized method based on a Krylov subspace approach

Function $y = \texttt{inolkr}(t, \texttt{data}, x_0, p, tol, q)$
Inputs: Time vector $t \in \mathbb{R}^{l+1}$; function `data` computes $f(\tau, y) \in \mathbb{R}^n$, $J(\tau, y) \in \mathbb{R}^{n\times n}$ and $g(\tau, y) \in \mathbb{R}^n$ ($\tau \in \mathbb{R}$, $y \in \mathbb{R}^n$); vector $x_0 \in \mathbb{R}^n$; dimension $p \in \mathbb{N}$ of the Krylov subspace; tolerance $tol \in \mathbb{R}^+$; order $q \in \mathbb{N}$ of the diagonal Padé approximation of the exponential function
Output: Matrix $Y = [y_1, \ldots, y_l] \in \mathbb{R}^{n\times l}$, $y_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, l$
1: Compute the vectors c_1 and c_2 that contain the coefficients of terms of degree greater than 0 in the diagonal Padé approximation of the exponential function
2: $y_0 = x_0$
3: **for** $i = 0 : l - 1$ **do**
4: $\quad [J_i, f_i, g_i] = \texttt{data}(t_i, y_i)$
5: $\quad \Delta t_i = t_{i+1} - t_i$
6: $\quad y_{i+1} = \texttt{inlbkr}(J_i, f_i, g_i, y_i, \Delta t_i, p, tol, c_1, c_2)$ (Algorithm 2)
7: **end for**

3 Experimental results

The main objective of this section is to compare the MATLAB implementations of algorithm developed in Section 2 with the implementations developed by the authors of this paper in [6]. What follows is a short description of the implemented algorithms and its characteristic parameters:

- `iaolwp` and `inolwp` solve IVPs for ODEs by a piecewise-linearized approach and a block-oriented version without scaling-squaring implementation of the diagonal Padé approximation method:
  - order $q = 2$ of the diagonal Padé approximation of the exponential function.
- `iaolkr` and `inolkr` solve IVPs for ODEs by a piecewise-linearized method based on Krylov subspaces:
  - dimension $p = 4$ of the Krylov subspace.
  - tolerance $tol = 10^{-6} \in \mathbb{R}^+$.
  - order $q = 2$ of the diagonal Padé approximation of the exponential function.

As test battery four case studies of stiff ODEs were considered, which come from biology and chemical kinetics stiff problems. For each test, the following results are shown:

- Tables which contain the relative error

$$E_r = \frac{\|x - x^*\|_\infty}{\|x\|_\infty},$$

 ISBN 13: 978-84-613-5510-5

Algorithm 2 computes the approximate solution at t_{i+1} of IVP (1) for non-autonomous ODEs, obtained after the piecewise-linearized process, by a block-oriented implementation of the Krylov subspace approach

Function $y_{i+1} = \texttt{inlbkr}(J_i, f_i, g_i, y_i, \Delta t_i, p, tol, c_1, c_2)$
Inputs: Matrix $J_i \in \mathbb{R}^{n\times n}$; vector $f_i \in \mathbb{R}^n$; vector $g_i \in \mathbb{R}^n$; vector $y_i \in \mathbb{R}^n$; step size $\Delta t_i \in \mathbb{R}$; dimension $p \in \mathbb{N}$ of the Krylov subspace; tolerance $tol \in \mathbb{R}^+$; vectors $c_1, c_2 \in \mathbb{R}^q$ with the coefficients of terms of degree greater than 0 in the diagonal Padé approximation of the exponential function
Output: Vector $y_{i+1} \in \mathbb{R}^n$ given by expression (4)

$V(1:n,1) = 0_n$
$V(n+1:2n,1) = f_i$
$V(2n+1:3n,1) = g_i$
$\beta = ||V(n+1:3n,1)||_2$
if $\beta == 0$ **then**
    $y_{i+1} = y_i$
    Return
end if
$V(n+1:3n,1) = V(n+1:3n,1)/\beta$
for $j = 1:p$ **do**
    $w(1:n) = J_i V(1:n,j) + V(n+1:2n,j)$
    $w(n+1:2n) = V(2n+1:3n,j)$
    $w(1:2n) = \Delta t_i w(1:2n)$
    $w(2n+1:3n) = 0_n$
    for $i = 1:j$ **do**
        $H(i,j) = w^T V(1:3n,i)$
        $w = w - H(i,j)V(1:3n,i)$
    end for
    $s = \|w\|_2$
    if $s < tol$ **then**
        $p = j$
        Leave for loop
    end if
    $H(j+1,j) = s$
    $V(1:3n,j+1) = w/s$
end for
computes $E = e^{H_p}$
$y_{i+1} = y_i + \beta V(1:n,1:p)E$

 ISBN 13: 978-84-613-5510-5

E_r	Δt=0.1	Δt=0.05	Δt=0.01	Δt=0.005	Δt=0.001
`iaolwp`	2.809e-04	7.523e-05	2.390e-06	5.840e-07	2.366e-08
`iaolkr`	2.348e-04	6.928e-05	2.759e-06	6.423e-07	2.399e-08

Table 1: Relative error (E_r) with $t = 10$ and Δt variable (case study 2).

T_e	Δt=0.1	Δt=0.05	Δt=0.01	Δt=0.005	Δt=0.001
`iaolwp`	0.014	0.021	0.114	0.257	6.231
`iaolkr`	0.025	0.048	0.201	0.428	6.802

Table 2: Execution time (T_e) in seconds with $t = 10$ and Δt variable (case study 2).

where x^* is the computed solution and x is the analytic solution (case study 3) or the solution computed by the MATLAB function ode15s with a vector of relative error tolerances $rtol = 10^{-13}$ and a vector of absolute error tolerances $atol = 10^{-13}$ [12].

- Tables/Figures with the execution time.

The algorithms were implemented in Matlab 7.9 and tested on an Intel Core 2 Duo processor at 2.66 GHz with 2 GB main memory. Several tests have been developed in order to determine the accuracy and efficiency of the algorithms. The implemented algorithms are available online at http://www.grycap.upv.es/odelin.

3.1 Case of study 1 (Pollution problem [13])

This case study corresponds to a stiff IVP of dimension twenty. The problem describes a chemical process consisting of 25 reactions and 20 species. The following tests were done:

- First test (Tables 1 and 2): t=10 and Δt variable.
- Second test (Table 3 and Figure 1): Δt=0.01 and t variable.

3.2 Case of study 2 (Emep problem [13])

In this case study a stiff IVP for ODEs of dimension sixty-six is solved. The problem describes a problem which consists of 66 chemical species and about 140 reactions. The following tests were done:

E_r	t=20	t=30	t=40	t=50	t=60
`iaolwp`	2.015e-06	1.744e-06	1.537e-06	1.374e-06	1.240e-06
`iaolkr`	2.327e-06	2.013e-06	1.775e-06	1.585e-06	1.431e-06

Table 3: Relative error (E_r) $\Delta t = 0.01$ and t variable (case study 2).

 ISBN 13: 978-84-613-5510-5

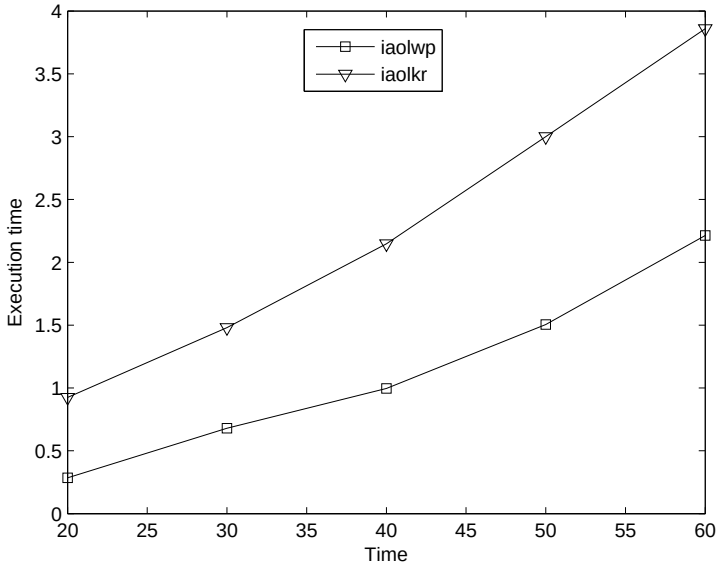


Figure 1: Execution time in seconds of the MATLAB implementations considering $\Delta t = 0.01$ and varying t (case study 2).

E_r	$t = 15400$	$t = 16400$	$t = 17400$	$t = 18400$	$t = 19400$
`inolwp`	4.410e-14	8.833e-14	1.431e-13	1.980e-13	2.528e-13
`inolkr`	4.410e-14	8.833e-14	1.431e-13	1.980e-13	2.528e-13

Table 4: Relative error (E_r) with $\Delta t = 0.1$ and t variable (case study 3).

- In the first test t=14450 was considered. With $\Delta t = 0.1$ the relative errors of the three implementation were equal to $2.219 \cdot 10^{-15}$, with executions times equal to 1.290 (`inolwp`) and 0.266 (`inolkr`) seconds.
- Second test (Tables 4 and 5, and Figure 2): Δt=0.1 and t variable.

3.3 Case study 3 (Medical Akzo Nobel problem [13])

This case study corresponds to a stiff non-autonomous ODE [13] of variable dimension $2N$. This problem studies the penetration of radio-labeled antibodies into a tissue infected by a tumor.

T_e	$t = 15400$	$t = 16400$	$t = 17400$	$t = 18400$	$t = 19400$
`inolwp`	52.830	157.465	316.257	529.469	790.217
`inolkr`	26.547	102.401	227.630	400.672	623.248

Table 5: Execution time (T_e) in seconds with $\Delta t = 0.1$ and t variable (case study 3).

 ISBN 13: 978-84-613-5510-5

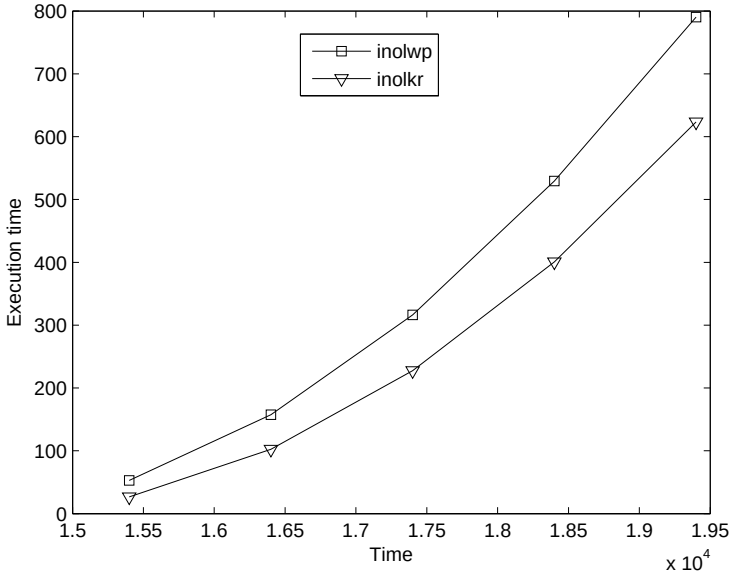


Figure 2: Execution time in seconds of the MATLAB implementations considering $\Delta t = 0.1$ and varying t between 15400 and 19400 (case study 3).

E_r	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00001
`inolwp`	1.572e-02	1.726e-03	1.741e-04	1.742e-05
`inolkr`	1.663e-02	1.728e-03	1.741e-04	1.742e-05

Table 6: Relative error (E_r) considering $n = 100$, $t = 1$ and Δt variable (case study 4).

The following tests were made:

- First test (Tables 6 and 7): $n = 100$ ($N = 50$), t=1 and Δt variable.
- Second test (Tables 8 and 9): Δt=0.001, $t = 1$ and varying n from 50 to 250 ($N = 25$ to 125).

3.4 Case of study 4 (Brusselator problem) [1, pp. 6]

This case study corresponds to a stiff non-autonomous ODE of variable dimension N. This problem comes from chemical kinetics where the model of Lefever and Nicolis [15]

T_e	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00001
`inolwp`	0.301	4.926	144.484	6362.490
`inolkr`	0.036	0.538	50.044	5263.304

Table 7: Execution time (T_e) in seconds considering $n = 100$, $t = 1$ and Δt variable (case study 4).

 ISBN 13: 978-84-613-5510-5

E_r	n=50	n=100	n=150	n=200	n=250
`inolwp`	1.636e-03	1.726e-03	1.746e-03	1.743e-03	1.736e-03
`inolkr`	1.637e-03	1.728e-03	1.752e-03	1.763e-03	1.781e-03

Table 8: Relative error (E_r) considering $\Delta t = 0.001$, $t = 1$ and n variable (case study 4).

T_e	n=50	n=100	n=150	n=200	n=250
`inolwp`	0.720	3.531	20.863	63.944	143.920
`inolkr`	0.288	0.482	0.740	1.159	1.367

Table 9: Execution time (T_e) in seconds considering $\Delta t = 0.001$, $t = 1$ and n variable (case study 4).

is used and the method of lines is applied on a grid of N points:

- First test (Tables 10 and 11): $n = 100$ ($N = 50$), t=1 and Δt variable.
- Second test (Tables 12 and 13): $t = 1$, Δt=0.001 and n variable.

4 Conclusions and future work

In this work a new piecewise-linearized approach for solving ODEs based on Krylov subspaces has been presented. Two algorithms based on this approach (`inolkr` and `iaolbk`) have been also proposed and compared to the piecewise-linearized algorithms `iaolwp` and `inolwp` based on Padé approximants developed by the authors of this paper in[6].

Numerous test have been made on four case studies that come from biology and chemical kinetics stiff problems. Experimental results show the advantages of the proposed algorithms, especially when they are integrating stiff problems. According to the experimental results, the new algorithms offer in general similar precision and smaller computational cost when the problem size is increased. For example, Algorithm 1 (`inolkr`) was up to 111 times faster than `inolwp` for $n = 250$ and $t = 1$ in case study 3. This is because in the new approach the vector $e^A v$, $A \in \mathbb{R}^{n\times n}, v \in \mathbb{R}^n$, is approximated by the expression $\beta V_p e^{H_p} e_1$, where $p << n$. Nevertheless, when the problems are of small dimension, computational costs of piecewise-linearized algorithms based on diagonal Padé approximants are smaller than the computational costs of piecewise-linearized algorithms based on Padé approximants. In general, all algorithms offer accuracy and good behaviour with stiff problems.

E_r	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00001
`inolwp`	2.162e-02	3.673e-04	3.715e-05	3.719e-06
`inolkr`	2.263e-02	3.672e-04	3.715e-05	3.719e-06

Table 10: Relative error (E_r) considering $n = 100$, $t = 1$ and Δt variable (case study 5).

 ISBN 13: 978-84-613-5510-5

T_e	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00001
`inolwp`	0.140	1.688	55.297	4106.738
`inolkr`	0.031	0.465	38.122	3710.951

Table 11: Execution time (T_e) in seconds considering $n = 100$, $t = 1$ and Δt variable (case study 5).

E_r	n=50	n=100	n=150	n=200	n=250
`inolwp`	5.033e-04	3.673e-04	3.308e-04	3.170e-04	3.108e-04
`inolkr`	5.033e-04	3.672e-04	3.307e-04	3.169e-04	3.107e-04

Table 12: Relative error (E_r) considering $\Delta t = 0.001$, $t = 1$ and n variable (case study 5).

T_e	n=50	n=100	n=150	n=200	n=250
`inolwp`	0.521	1.894	2.988	7.616	16.391
`inolkr`	0.279	0.506	0.701	0.963	1.256

Table 13: Execution time (T_e) in seconds considering $\Delta t = 0.001$, $t = 1$ and n variable (case study 5).

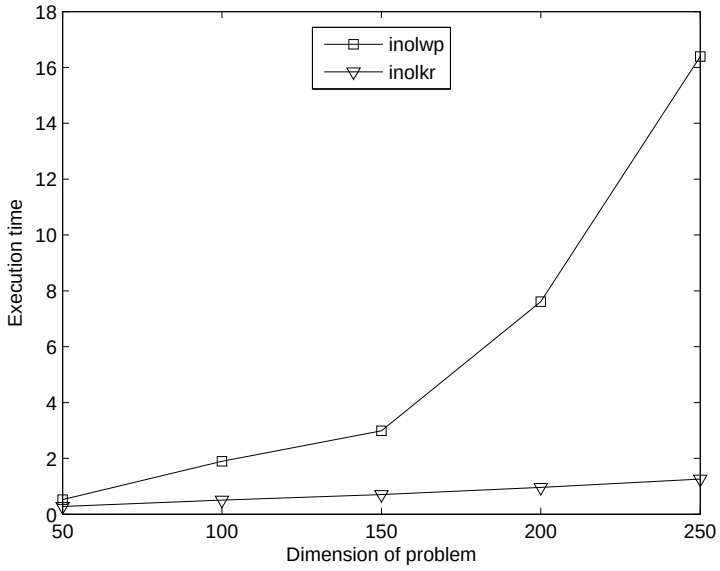


Figure 3: Execution time in seconds of the MATLAB implementations considering $\Delta t = 0.001$ and varying t between 50 and 250 (case study 5).

 ISBN 13: 978-84-613-5510-5

As future work new improvements will be developed such as:

1. To implement algorithms based on the piecewise-linearized approach with error control in order to vary step size dynamically. The tests reported here considered constant step size. It is possible to improve the developed algorithms, using a variable step size which can be used to estimate the error committed in each iteration [5].

2. To do parallel implementation of the algorithms presented in this work in a distributed memory platform, using the message passing paradigm MPI [16] and BLACS [17] for communications, and PBLAS [18] and ScaLAPACK [19] for computations.

References

[1] E. Hairer, G. Wanner, Solving ordinary differential equations II. Stiff and differential-algebraic problems., in: Springer Series in Computational Mathematics, Vol. 14, Springer-Verlag, 1996.

[2] J. D. Lambert, Numerical Methods for Ordinary Differential Systems: The Initial Value Problem, John Wiley & Sons, 1991.

[3] U. M. Ascher, L. R. Petzold, Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, SIAM, 1998.

[4] J. I. Ramos, C. M. García, Piecewise-linearized methods for initial-value problems, Applied Mathematics and Computation 82 (1997) 273–302.

[5] C. M. García, Métodos de linealización para la resolución numérica de ecuaciones diferenciales, Ph.D. thesis, Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga (1998).

[6] J. Ibáñez, V. Hernández, E. Arias, P. Ruiz, Solving initial value problems for ordinary differential equations by two approaches: BDF and piecewise-linearized methods, Computer Physics Communications 180 (5) (2009) 712–723.

[7] C. M. García, Piecewise-linearized and linearized θ-methods for ordinary and partial differential equation problems, Computer & Mathematics with Applications 45 (2003) 351–381.

[8] Y. Saad, Analysis of some Krylov subspace approximations to the matrix exponential operator, SIAM Journal on Numerical Analysis 29 (92) 209–228.

[9] R. B. Sidje, Expokit: A software package for computing matrix exponentials, ACM Trans. Math. Softw. 24 (1998) 130–156.

[10] C. B. Moler, C. V. Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Review 45 (2003) 3–49.

 ISBN 13: 978-84-613-5510-5

[11] N. J. Higham, The scaling and squaring method for the matrix exponential revisited, Tech. Rep. 452, Manchester Centre for Computational Mathematics (2004).

[12] L. S. Shampine, I. Gladwell, S. Thomson, Solving ODEs with MATLAB, Cambridge University Press, 2003.

[13] W. M. Lioen, J. J. B. de Swart, Test set for initial value problems solvers, release 2.0 (December 1998).

[14] F. Mazzia, C. Magherini, Test set for initial value problems solvers, Tech. Rep. 4/2008, Department of Mathematics, University of Bari (Italy), release 2.4 (1998).

[15] R. Lefever, G. Nicolis, Chemical instabilities and sustained oscillations, J. Theor. Biol 30 (1971) 267–284.

[16] W. Gropp, E. Lusk, A. Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, MIT Press, 1994.

[17] J. J. Dongarra, R. A. V. D. Geijn, Two dimensional basic linear algebra communications subprograms, Tech. rep., Department of Computer Science, University of Tennessee (1991).

[18] J. Choi, J. Dongarra, S. Ostrouchov, A. Petitet, D. Walker, A proposal for a set of parallel basic linear algebra subprogram, Tech. Rep. UT-CS-95-292, Department of Computer Science, University of Tennessee (1995).

[19] L. S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, ScaLAPACK Users' Guide, SIAM, 1997.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

On a Consistency Driven Pairwise Comparison Based Non-Numerical Ranking

Ryszard Janicki[1] and Yun Zhai[1,2]

[1] *Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada*

[2] *Artificial Intelligence in Medicine, Toronto, Ontario, Canada*

emails: janicki@mcmaster.ca, zhaiy@mcmaster.ca

Abstract

An abstract model of pairwise comparisons based non-numerical ranking is presented and discussed. An algorithm for enforcing consistency rules is given and analysed.

Key words: consistency driven, non-numerical ranking, pairwise comparisons

1 Introduction

A *ranking* or *preference* is usually defined as a weakly ordered relationship between a set of items such that, for any two items, the first is either "less preferred", "more preferred", or "indifferent" to the second one [8]. The ranking is numerical if numbers are used to measure importance and to create the ranking relation. Numerical rankings are usually totally ordered. Various kinds of *global indexes* are popular examples of numerical rankings.

The *Pairwise Comparisons* method is based on the observation that while ranking the importance of *several* objects is often problematic, it is much easier to do when restricted to *two* objects [3]. The problem is then reduced to constructing a global ranking from the set of partially ordered pairs. The method could be traced to the 1785 Marquis de Condorcet paper [15]), it was explicitly mentioned and analyzed by Fechner in 1860 [5], made popular by Thurstone in 1927 [20], and was transformed into a kind of semi-formal methodology by Saaty in 1977 (called *AHP*, Analytic Hierarchy Process, see [4, 8, 18]).

At present Pairwise Comparisons are practically identified with the controversial Saaty's AHP. On one hand AHP has respected practical applications, on the other hand it is still considered by many (see [4, 9, 14]) as a flawed procedure that produces

 ISBN 13: 978-84-613-5510-5

arbitrary rankings. For more details the reader is referred to [4, 9, 13, 14] (specially [4]).

Pairwise Comparisons based *non-numerical* solutions were proposed and discussed in [9, 11, 12, 13]. The model presented in this paper stems from [9], and is orthogonal to that of [11].

The model presented below uses no numbers and is entirely based on the concept of partial orders.

Non-numerical rankings should be *stratified* or *total* orders, but the initial empirical data may not even be partial orders, in general they are just arbitrary relations. This leads to the problem, what is the "best" partial order approximation of and arbitrary relation, and what is the "best" stratified order approximation of an arbitrary partial order? The latter problem is discussed in details in [6, 7], some solutions to the former one were proposed in [10, 11], where four different solutions were proposed and analyzed. In this paper we will use the approximation denoted as $(R^+)^\bullet$ (calculate the transition closure first and then remove all cycles), that was first proposed by Schröder in 1895 [19].

The paper is structured into four parts, from Sections 2 to 6. In Section 2 the basic notions of partially ordered relations are recalled. The formal definitions of the proposed model are given in Section 3. In Section 4 some algorithmic solutions are presented, while Section 5 is devoted to the problem of *testing* models like the one presented in this paper. Section 6 contains some final comments.

2 Relations and Partial Orders

In this section we recall some fairly known concepts and results that will be used in the following sections [6, 17].

Let X be a finite set, fixed for the rest of this paper. For every relation $R \subseteq X \times X$, let $R^+ = \bigcup_{i=1}^{\infty} R^i$, denote the *transitive closure* of R, $id = \{(x, x) \mid x \in X\}$ denote the identity relation, and let $R^\circ = R \cup id$ denote the *reflexive closure* of R (see [17] for details).

A relation $< \in X \times X$ is a (*sharp*) *partial order* if it is irreflexive and transitive, i.e. if $\neg(a < a)$ and $a < b < c \Longrightarrow a < c$, for all $a, b, c \in X$.

We write $a \sim_< b$ if $\neg(a < b) \wedge \neg(b < a)$, that is if a and b are either *distinctly incomparable* (w.r.t. $<$) or *identical* elements.

A partial order is

- *total* or *linear*, if $\sim_<$ is empty, i.e., for all $a, b \in X$. $a \neq b \Longrightarrow (a < b \vee b < a)$.
- *weak* or *stratified*, if $a \sim_< b \sim_< c \Longrightarrow a \sim_< c$, i.e. if $\sim_<$ is an equivalence relation (i.e. it is reflexive, symmetric and transitive).

If a partial order $<$ is weak than $a \equiv_< b \iff a \sim_< b$ (see [6]).

A relation R is *acyclic* if and only if $\neg x R^+ x$ for all $x \in X$.

 ISBN 13: 978-84-613-5510-5

For every relation R, define the relations R^{cyc}, R^{cyc}_{id} and $R^{\bullet}$ as

- $aR^{cyc}b \iff aR^{+}b \wedge bR^{+}a$,
- $aR^{\bullet}b \iff aRb \wedge \neg(aR^{cyc}b)$,

We will call $R^{\bullet}$ an *acyclic refinement of* R.

Corollary 1 *If R is a partial order then $R = R^{+} = R^{\bullet}$.* □

In this paper expressions like $(R^{+})^{\bullet}$ are interpreted as $(R^{+})^{\bullet} = Q^{\bullet}$ where $Q = R^{+}$.

The relation $(R^{+})^{\bullet}$ could be treated as a partial order approximation of R (see [10, 11] for detailed definitions and proofs).

Lemma 1 *1. $(R^{+})^{\bullet}$ is a partial order (Schröder 1895, [19]).*

2. $(R^{+})^{\bullet}$ is a partial order approximation of R (see [10, 11]). □

Approximations of partial orders by weak orders are just proper extensions. Various methods were proposed and discussed in [6] and specially in [7]. For our purposes, the best seems to be the method based on the concept of a *global score function* [6], which is defined as (for every finite set X, $\| X \|$ denotes its number of elements):

$$g_{<}(x) = \| \{z \mid z < x\} \| - \| \{z \mid x < z\} \| .$$

Given the global score function $g_{<}(x)$, we define the relation $<^{w} \subseteq X \times X$ as

$$a <^{w} b \iff g_{<}(a) < g_{<}(b).$$

We will use this technique is our model.

3 Consistency-Driven Non-Numerical Ranking : The Model

A *pairwise comparisons ranking data* [11] is a tuple $PCRD = (X, R'_0, R'_1, ..., R'_k)$, where X is the set of objects to be ranked, $k \geq 1$, and R'_i's are relations satisfying $R'_0 \cup R'_1 \cup ... \cup R'_k = X \times X$ and $R'_j \cap R'_j = \emptyset$ unless $i = j$. The relation R'_0, interpreted as *indifference*, is symmetric and reflexive, the relations $R'_1, ..., R'_k$, interpreted as *preferences*, are asymmetric and irreflexive.

The relations $(R'_0, R'_1, ..., R'_k)$ are based on empirical data or judgments, so no other specific properties are expected.

A tuple $PCCRS = (X, R_0, R_1, ..., R_k)$, is called a *pairwise comparisons consistent ranking system* when some additional *consistency* properties are satisfied, and is called *derived from* $PCRD$, if each R_i is some *approximation* of R_i.

 ISBN 13: 978-84-613-5510-5

We will devote the rest of this chapter to this problem. Quite often we will use the same symbol to denote both R_i and R'_i, our algorithm presented later will take R'_i's, and produced R_i's.

In [9], the case $PCCRS = (X, \approx, \sqsubset, \subset, <, \prec)$, with the following interpretation $a \approx b$: a and b are *indifferent*, $a \sqsubset b$: *slightly in favor* of b, $a \subset b$: *in favor* of b, $a < b$: b is *strongly better*, $a \prec b$: b is *extremely better*, was proposed and some (incomplete) axioms were proposed. For all practical applications, the list $\sqsubset, \subset, <, \prec$ may be shorter or longer, but not empty and not much longer (due to limitations of the human mind [2, 16]).

In this paper we will consider only the case $PCCRS = (X, \approx, \sqsubset, \subset, <, \prec)$, leaving the generalizations and special cases to the reader.

Definition 1 *Let X be a finite set of objects to be "ranked", and let $\approx$, $\sqsubset$, $\subset$, $<$ and $\prec$ be a family of disjoint relations on X such that $X = \approx \cup \sqsubset \cup \subset \cup < \cup \prec$.*

We define the relations $\widehat{\sqsubset}$, $\widehat{\subset}$, $\widehat{<}$, and $\widehat{\prec}$ as follows:

$$\begin{array}{lll} \widehat{\prec} = \prec & & \widehat{<} = \prec \cup < \\ \widehat{\subset} = \prec \cup < \cup \subset & & \widehat{\sqsubset} = \prec \cup < \cup \subset \cup \sqsubset \end{array}$$

The relations $\widehat{\sqsubset}$, $\widehat{\subset}$, $\widehat{<}$, and $\widehat{\prec}$ are interpreted as combined preferences*, i.e. $a \widehat{\sqsubset} b$:* at least slightly in favor *of b, $a \widehat{\subset} b$:* at least in favor *of b, $a \widehat{<} b$:* at least strongly in favor *of b, and $a \widehat{\prec} b$:* at least b is far superior than a*.*

The tuple $PCRS = (X, \approx, \sqsubset, \subset, <, \prec)$ is a Pairwise Comparison Ranking System *if the following two simple rules are satisfied:*

1. *$\widehat{\sqsubset}$, $\widehat{\subset}$, $\widehat{<}$, $\widehat{\prec}$ are partial orders*
2. *$\approx = \sim_{\widehat{\sqsubset}}$, i.e. $\approx \cup \widehat{\sqsubset} \cup \widehat{\sqsubset}^{-1} = X \times X$.*

The tuple $PCCRS = (X, \approx, \sqsubset, \subset, <, \prec)$ is a Pairwise Comparison **Consistent** Ranking System *if the additional* consistency rules *are satisfied:*

3. $(a \approx b \wedge b \approx c) \Rightarrow (a \approx c \vee a \sqsubset c \vee c \sqsubset a)$

4.1. $(a \approx b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \approx c) \Rightarrow (a \sqsubset c \vee a \subset c)$

4.2. $(a \approx b \wedge b \subset c) \vee (a \subset b \wedge b \approx c) \Rightarrow (a \subset c \vee a < c)$

4.3. $(a \approx b \wedge b < c) \vee (a < b \wedge b \approx c) \Rightarrow (a < c \vee a \prec c)$

5. $(a \approx b \wedge b \prec c) \vee (a \prec b \wedge b \approx c) \Rightarrow (a \prec c)$

6.1. $(a \sqsubset b \wedge b \sqsubset c) \Rightarrow (a \sqsubset c \vee a \subset c)$

6.2. $(a \subset b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \subset c) \Rightarrow (a \subset c \vee a < c)$

6.3. $(a < b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b < c) \Rightarrow (a < c \vee a \prec c)$

 ISBN 13: 978-84-613-5510-5

6.3. $(a < b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b < c) \Rightarrow (a < c \vee a \prec c)$

7.1. $(a \sqsubset b \wedge b \prec c) \vee (a \prec b \wedge b \sqsubset c) \Rightarrow (a \prec c)$

7.2. $(a \subset b \wedge b \prec c) \vee (a \prec b \wedge b \subset c) \Rightarrow (a \prec c)$

7.3. $(a < b \wedge b \prec c) \vee (a \prec b \wedge b < c) \Rightarrow (a \prec c)$

7.4. $(a \prec b \wedge b \prec c) \vee (a \prec b \wedge b \prec c) \Rightarrow (a \prec c)$

8.1. $(a \subset b \wedge b < c) \vee (a < b \wedge b \subset c) \Rightarrow (a \prec c)$

8.2. $(a < b \wedge b < c) \Rightarrow (a \prec c)$

9.1. $(a \subset b \wedge b \subset c) \Rightarrow (a < c) \vee (a \prec c)$

10.1. $(a \approx b \wedge b \sqsupset c) \vee (a \sqsupset b \wedge b \approx c) \Rightarrow (a \sqsupset c) \vee (a \supset c)$

10.2. $(a \approx b \wedge b \supset c) \vee (a \supset b \wedge b \approx c) \Rightarrow (a \supset c) \vee (a > c)$

10.3. $(a \approx b \wedge b > c) \vee (a > b \wedge b \approx c) \Rightarrow (a > c) \vee (a \succ c)$

10.4. $(a \approx b \wedge b \succ c) \vee (a \succ b \wedge b \approx c) \Rightarrow (a \succ c)$

11.1. $(a \sqsupset b \wedge b \sqsupset c) \Rightarrow (a \sqsupset c) \vee (a \supset c) \vee (a > c)$

11.2. $(a \sqsupset b \wedge b \supset c) \vee (a \supset b \wedge b \sqsupset c) \Rightarrow (a \supset c) \vee (a > c)$

11.3. $(a \sqsupset b \wedge b > c) \vee (a > b \wedge b \sqsupset c) \Rightarrow (a > c) \vee (a \succ c)$

11.4. $(a \sqsupset b \wedge b \succ c) \vee (a \succ b \wedge b \sqsupset c) \Rightarrow (a \succ c)$

12.1. $(a \supset b \wedge b \supset c) \Rightarrow (a > c) \vee (a \succ c)$

12.2. $(a \supset b \wedge b > c) \Rightarrow (a \succ c)$

12.3. $(a \supset b \wedge b \succ c) \Rightarrow (a \succ c)$

13.1. $(a > b \wedge b > c) \Rightarrow (a \succ c)$

13.2. $(a > b \wedge b \succ c) \vee (a \succ b \wedge b > c) \Rightarrow (a \succ c)$

14.1. $(a \succ b \wedge b \succ c) \Rightarrow (a \succ c)$

15.1. $(a \sqsupset b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \sqsupset c) \Rightarrow (a \approx c \vee a \sqsubset c \vee a \sqsupset c)$

15.2. $(a \sqsupset b \wedge b \subset c) \vee (a \subset b \wedge b \sqsupset c) \Rightarrow (a \sqsubset c \vee a \subset c)$

15.3. $(a \sqsupset b \wedge b < c) \vee (a < b \wedge b \sqsupset c) \Rightarrow (a \subset c \vee a < c)$

15.4. $(a \sqsupset b \wedge b \prec c) \vee (a \prec b \wedge b \sqsupset c) \Rightarrow (a < c \vee a \prec c)$

16.1. $(a \supset b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \supset c) \Rightarrow (a \sqsupset c \vee a \supset c)$

 ISBN 13: 978-84-613-5510-5

16.2. $(a \supset b \wedge b \subset c) \vee (a \subset b \wedge b \supset c) \Rightarrow (a \approx c \vee a \sqsubset c \vee a \sqsupset c)$

16.3. $(a \supset b \wedge b < c) \vee (a < b \wedge b \supset c) \Rightarrow (a \sqsubset c \vee a \subset c)$

16.4. $(a \supset b \wedge b \prec c) \vee (a \prec b \wedge b \supset c) \Rightarrow (a \subset c \vee a < c \vee a \prec c)$

17.1. $(a > b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b > c) \Rightarrow (a \supset c \vee a > c)$

17.2. $(a > b \wedge b \subset c) \vee (a \subset b \wedge b > c) \Rightarrow (a \sqsupset c \vee a \supset c)$

17.3. $(a > b \wedge b < c) \vee (a < b \wedge b > c) \Rightarrow (a \approx c \vee a \sqsubset c \vee a \sqsupset c \vee a \subset c \vee a \supset c)$

17.4. $(a > b \wedge b \prec c) \vee (a \prec b \wedge b > c) \Rightarrow (a \sqsubset c \vee a \subset c \vee a < c \vee a \prec c)$

18.1. $(a \succ b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \succ c) \Rightarrow (a > c \vee a \succ c)$

18.2. $(a \succ b \wedge b \subset c) \vee (a \subset b \wedge b \succ c) \Rightarrow (a \supset c \vee a > c \vee a \succ c)$

18.3. $(a \succ b \wedge b > c) \vee (a > b \wedge b \succ c) \Rightarrow (a \sqsupset c \vee a \supset c \vee a > c \vee a \succ c)$

18.4. $(a \succ b \wedge b \prec c) \vee (a \prec b \wedge b \succ c) \Rightarrow (a \approx c \vee a \sqsupset c \vee a \sqsubset c \vee a \supset c \vee a \subset c \vee a > c \vee a < c)$

The rules above implement the idea that composition may change precedences only by 'one step', or not at all. Consider for example the rule:

4.1. $(a \approx b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \approx c) \Rightarrow (a \sqsubset c \vee a \subset c)$.

In principle it says that a and b are indifferent and c is slightly better than b, then at most c is in favor of s, and similarly for a symmetric case. The rules (*1*) and (*2*) say that rankings are just partial orders.

Usually we need that the finest ranking, $\sqsubset$ is also a weak or total order (it is usually unreasonable to require the other orders to have any specific properties).

A ranking $PCCRS = (X, \approx, \sqsubset, \subset, <, \prec)$ is *weakly ordered* if the relation $\widehat{\sqsubset}$ is a weak order.

As we mentioned above, the starting data $\approx, \sqsubset, \subset, <, \prec$ may not even be partial orders. The following algorithm has been proposed in [10, 11] to obtain a Pairwise Comparisons Ranking System (not necessarily consistent) $PCRS = (X, \approx, \sqsubset, \subset, <, \prec)$ from a Pairwise Comparisons Ranking Data $PCRD = (X, R_0, R_1, R_2, R_3, R_4)$:

Algorithm 1

1. *Calculate* $\widehat{R_4} = R_4$, $\widehat{R_3} = R_3 \cup \widehat{R_4}$, $\widehat{R_2} = R_2 \cup \widehat{R_3}$, $\widehat{R_1} = R_1 \cup \widehat{R_3}$.
2. *Calculate* $\widehat{\prec} = ((\widehat{R_4})^+)^\bullet$.
3. *Calculate* $\widehat{<} = ((\widehat{R_3} \cup \widehat{\prec})^+)^\bullet$.
4. *Calculate* $\widehat{\subset} = ((\widehat{R_2} \cup \widehat{<})^+)^\bullet$.
5. *Calculate* $\widehat{\sqsubset} = ((\widehat{R_1} \cup \widehat{\subset})^+)^\bullet$.

 ISBN 13: 978-84-613-5510-5

6. *Calculate* $\approx = X \times X \setminus \widehat{\sqsubset}$.
7. *Calculate* $\prec = \widehat{\succ}$, $< = \widehat{\leq} \setminus \widehat{\succ}$, $\subset = \widehat{\subseteq} \setminus \widehat{\leq}$, $\sqsubset = \widehat{\sqsubseteq} \setminus \widehat{\subseteq}$. □

One may easily find that the complexity of the above algorithm is $O(n^3)$, where n is the size of X. It was also shown in [10, 11]that the technique used satisfies the requirements for computing 'partial order approximation'.

4 Enforcing Consistency

Algorithm 1 from the previous section gives a ranking system that may not be consistent. The following algorithm starts with a Pairwise Comparisons Ranking System and produces a Pairwise Comparisons **Consistent** Ranking System.

First we order consistency rules in a consecutive manner starting from the rules for $\approx$, than those for $\sqsubset$, etc., ending with the rules for $\prec$. The ordering given in Definition 1 is a possible choice.

Algorithm 2

1. *Find all pairs that violate consistency rules, if none, go* the end.
2. *Pick a pair which violates the rule with the highest number (arbitrary choice if more than one).*
3. *Revise the relation between the pairs violating a rule, by appropriate* lowering preferences*, for example from* $<$ *to* $\subset$*, etc.*
4. *go to 1.*
5. the end □

Proposition 1

1. *Algorithm 2 always converges.*
2. *The time complexity of Algorithm 2 is* $O(n^4)$.
3. *In the worst case the outcome is* $\approx = X \times X$.

Proof (sketch) (1) and (3) From step 3 of the algorithm. We always 'increase the disorder'. Ultimately, we may get that $\approx = X \times X$, but the procedure always stops.
(2) This is is principle analysis of triples (step 1 in the algorithm), so we need at least $O(n^3)$, but relatively easily it is sufficient to do it only n times. □

 ISBN 13: 978-84-613-5510-5

5 Testing

How can we test the results of the algorithms presented in this section? How do we know if they produce the results that make any sense? How can we compare them with the algorithms constructed using numerical ranking paradigms?

Testing means that there are some data and results that are known to be correct, and then the technique is applied to the same data. The differences between the correct results and those obtained by a given technique are used to judge the value of the technique. Hence testing models such as the one presented above is problematic since *it is not obvious what should be tested against.* What are the correct results for a given data? If the object has measurable attributes and there is a precise algorithm to calculate the value, the whole problem disappears. Nevertheless we think we have designed a proper test for these kinds of ranking techniques.

A blindfolded person compared the weights of stones. The person put one stone in his left hand and another in his right hand, and then decided which of the relations $\approx$, $\sqsubset$, $\subset$, $<$, or $\prec$ (interpreted as described in previous section) held. The experiment was repeated for the same set of stones by various people; and then again for different stones and different number of stones; and again for various subsets of $\{\approx, \sqsubset, \subset, <, \prec\}$. Those experiments have most likely been carried out by the prehistoric man. Our ancestors probably used this technique to decide which stone is better to kill an enemy or an animal.

In this experiment the stones can be weighted using precise scale, so we have the precise results to test against.

The complete analysis of those experiments has not been finished yet, especially the comparison with numerical ranking technique (the main goal of those experiments), but even though inconsistencies occur often in the ranking data, after correction we always obtain ranking that do not contradict the real weights of stones.

6 Final Comment

The concepts of consistent ranking and pairwise comparisons ranking data have been defined and analyzed in the setting of partial orders. Some algorithms have been presented. No numbers were used whatsoever, which we believe is more fair and objective approach. A method of testing has been proposed. The approach presented in this paper is an extension of models proposed in [9, 11].

Acknowledgment

Waldemar W. Koczkodaj is thanked for proposing experiments discussed in Section 5.

 ISBN 13: 978-84-613-5510-5

References

[1] Arrow K. J., *Social Choice and Individual Values*, J. Wiley, New York 1951.

[2] Cowan, N., The magical number 4 in short-term memory. A reconsideration of mental storage capacity, *Behavioural and Brain Sciences*, 24 (2001), 87-185.

[3] David H. A., *The Method of Paired Comparisons*, Oxford University Press, New York 1988.

[4] Dyer J. S., Remarks on the Analytic Hierarchy Process, *Management Sci.*, 36 (1990) 244-58.

[5] Fechner G. T., *Elemente der Psychophysik*, Breitkopf und Härtel, Leipzig 1860.

[6] Fishburn, P. C., *Interval Orders and Interval Graphs*, J. Wiley, New York 1985.

[7] Fishburn, P. C., Gehrlein W. V., A comparative analysis of methods for constructing weak orders from partial orders, *J. Math. Sociol.* 4 (1975), 93-102.

[8] French S., *Decision Theory*, Ellis Horwood, New York 1986.

[9] Janicki R., Pairwise Comparisons, Incomparability and Partial Orders, Proc. of *ICEIS'2007* (Int. Conference on Enterprise Information Systems), Vol. 2 (Artificial Intelligence and Decision Support System), Funchal, Portugal 2007, pp. 296-302.

[10] Janicki R., Ranking with Partial Orders and Pairwise Comparisons, *Lecture Notes in Artificial Intelligence* 5009, Springer 2008, pp. 442-451.

[11] Janicki R., Pairwise Comparisons Based Non-Numerical Ranking, *Fundamenta Informaticae* 94 (2009), 197-217.

[12] Janicki R., Koczkodaj W. W., Weak Order Approach to Group Ranking, *Computers Math. Applic.*, 32, 2 (1996) 51-59.

[13] Janicki R., Koczkodaj W. W., A Weak Order Solution to a Group Ranking and Consistency-Driven Pairwise Comparisons, *Applied Mathematics and Computation*, 94 (1998) 227-241.

[14] Koczkodaj W. W., A new definition of consistency of pairwise comparisons, *Mathematical and Computer Modelling*, 18 (1993) 79-84.

[15] Marquis de Condorcet, *Essai sur l'application de l'analyse a la probabilite des decisions rendue a la pluralite des voix*, Paris 1785 (see [1]).

[16] Miller G. A., The Magical Number Seven, Plus or Minus Two, *The Psychological Review*, 63, 2 (1956), 81-97.

[17] Rosen K. H., *Discrete Mathematics and Its Applications*, McGraw-Hill, New York 1999.

 ISBN 13: 978-84-613-5510-5

[18] Saaty T. L., A Scaling Methods for Priorities in Hierarchical Structure, *Journal of Mathematical Psychology*, 15 (1977) 234-281.

[19] Schröder E., *Algebra der Logik*, Teuber, Leipzig 1895.

[20] Thurstone L. L., A Law of Comparative Judgments, *Psychol. Reviews*, 34 (1927) 273-286.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Functions with Two Variables on Two Time Scales

Sinan Kapçak[1] and Ünal Ufuktepe[1]

[1] *Department of Mathematics, İzmir University of Economics, İzmir, TURKEY*

emails: sinankapcak@gmail.com, unal.ufuktepe@ieu.edu.tr

Abstract

The theory of time scales is the unification and generalization of various mathematical concepts from theories of discrete and continuous dynamical systems. In this paper, we extend our previous study [6]and develop some important concepts;tangent planes and partial derivatives for multivariable functions on time scales with mathematica.

Key words: partial delta derivative, time scale, mathematica, tangent plane
MSC 2000: AMS codes (optional)

1 Introduction

Let $\mathbb{T}$ be a time scale which is nonempty closed subset of real numbers. We define the *forward* and *backward jump operators* $\sigma, \rho : \mathbb{T} \to \mathbb{T}$ respectively as follows:

$$\begin{aligned} \sigma(t) &= \inf\{s \in \mathbb{T} : s > t\}, \text{ and} \\ \rho(t) &= \sup\{s \in \mathbb{T} : s < t\}. \end{aligned}$$

A point $t \in \mathbb{T}$ is called *right-scattered, right-dense, left-scattered, left-dense* if $\sigma(t) > t$, $\sigma(t) = t$, $\rho(t) < t$, $\rho(t) = t$ holds, respectively. The *graininess* $\mu : \mathbb{T} \to [0, \infty)$ is defined by

$$\mu(t) = \sigma(t) - t.$$

If $\mathbb{T}$ has left-scattered maximum then we define $\mathbb{T}^\kappa = \mathbb{T} - \{\max \mathbb{T}\}$, otherwise $\mathbb{T}^\kappa = \mathbb{T}$. If f is delta differentiable at $t \in \mathbb{T}^\kappa$ then *delta derivative of* f is

$$f^\Delta(t) = \begin{cases} \lim_{s \to t} \frac{f(t)-f(s)}{t-s} & , \quad \mu(t) = 0; \\ \frac{f(\sigma(t))-f(t)}{\sigma(t)-t} & , \quad \mu(t) > 0. \end{cases}$$

We refer to the original work by Hilger [5] and the recently appeared works can be found in [2], [3].

 ISBN 13: 978-84-613-5510-5

2 The Tangent Line with *Mathematica*

To describe a time scale in our *Mathematica* package, we use a collection of three lists: a list of right-dense left-scattered points, a list of left-dense right-scattered points and a list of isolated points. For example, we represent time scale $\mathbb{T}_1 = [-1,0] \cup [1,2] \cup \{\frac{1}{2}\}$ consisting of two closed intervals and one isolated point by

```
In[1]:= T1={{-1,1},{0,2},{1/2}};
```

We refer to [6], [7] for more details about the symbolic and numerical computations of jump operators, delta derivative and delta integral as well as their visual representations.

In the rest of this section, we give the definition of tangent line and visualize it by using our time scale package.

We consider the geometric sense of delta differentiability in simple variable functions on time scales. Consider $u = f(t)$ for $t \in \mathbb{T}$. Let Γ be the curve represented by the function f. Let t_0 be a fixed point in $\mathbb{T}^\kappa$. In this case $P_0 = (t_0, f(t_0))$ is a point on the curve.

Definition 1 *A line ℓ passing through the point P_0 is called the* delta tangent line *to the curve Γ at the point P_0 if*

a. *ℓ passes also through the point $P_0^\sigma = (\sigma(t_0), f(\sigma(t_0))$.*

b. *if P_0 is not an isolated point of the curve Γ, then*

$$\lim_{P \to P_0,\, P \neq P_0} \frac{d(P,\ell)}{d(P,P_0)} = 0,$$

where P is the moving point of the curve Γ, $d(P,\ell)$ is the distance from the point P to the line ℓ, and $d(P,P_0^\sigma)$ is the distance from the point P to the point P_0^σ.[3].

Now, we plot the tangent line to the function $f : \mathbb{T}_1 \to \mathbb{R}$, $f(x) = x^2$ at the point $x = 0$ where $\mathbb{T}_1$ is the time scale above:

```
In[2]:= TSTangentLine[T1,x^2,x,0]
```

```
Out[2]= (See Figure 1)
```

3 Functions with Two Variables on Two Time Scales

Let $\mathbb{T}_1$ and $\mathbb{T}_2$ be time scales. We define the product $\mathbb{T}_1 \times \mathbb{T}_2$ by

$$\mathbb{T}_1 \times \mathbb{T}_2 = \{(t,s) : t \in \mathbb{T}_1 \text{ and } s \in \mathbb{T}_2\}.$$

 ISBN 13: 978-84-613-5510-5

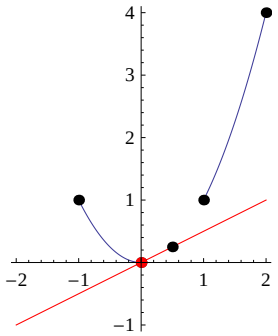


Figure 1: Tangent line to the curve $f(x) = x^2$ at the point $x = 0$

Let σ_1 and σ_2 be two forward jump operators for $\mathbb{T}_1$ and $\mathbb{T}_2$, respectively. Let ρ_1 and ρ_2 be two backward jump operators for $\mathbb{T}_1$ and $\mathbb{T}_2$, respectively. We consider the surface S which is defined by $u = f(t, s)$ continuous function on $\mathbb{T}_1 \times \mathbb{T}_2$: the set of points

$$\{(t, s, f(t, s)) : (t, s) \in \mathbb{T}_1 \times \mathbb{T}_2\}$$

in three space.

To get the visual representation of the product $\mathbb{T}_1 \times \mathbb{T}_2$ by Mathematica, we first input $\mathbb{T}_1 = [-1, 0] \cup [1, 2] \cup \{\frac{1}{2}\}$ and $\mathbb{T}_2 = [0, 1] \cup [2, 3] \cup [5, 6] \cup \{4, 8\}$ as follows:

```
In[3]:= T1={{-1,1},{0,2},{1/2}};

In[4]:= T2={{0,2,5},{1,3,6},{4,8}};
```

We plot the set of points $\mathbb{T}_1 \times \mathbb{T}_2$ by using the command

```
In[5]:= DrawTimeScale3D[T1,T2]

Out[5]=
```

Let $f : \mathbb{T}_1 \times \mathbb{T}_2 \to \mathbb{R}$ such that $f(t, s) = t^2\sqrt{s}$. We plot this surface on $\mathbb{T}_1 \times \mathbb{T}_2$ by using the command `TSPlot3D[T1,T2,f(t,s),t,s]` where `t` $\in$ `T1`, `s` $\in$ `T2`, as follows:

```
In[6]:= TSPlot3D[T1,T2,t^2 s^(1/2),t,s]

Out[6]=
```

 ISBN 13: 978-84-613-5510-5

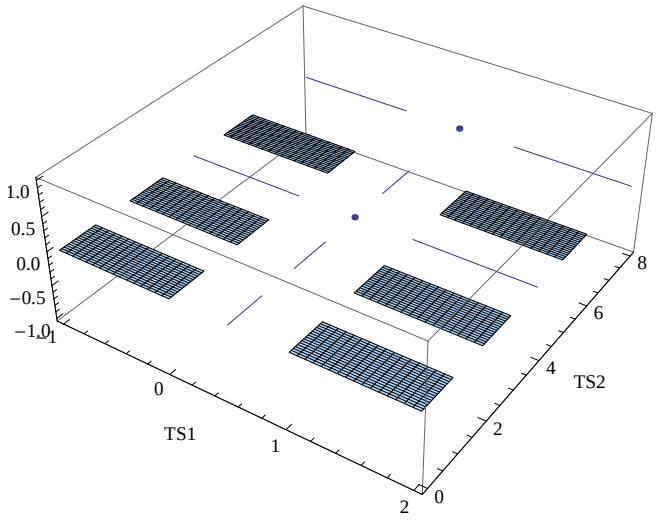


Figure 2: $\mathbb{T}_1 \times \mathbb{T}_2$

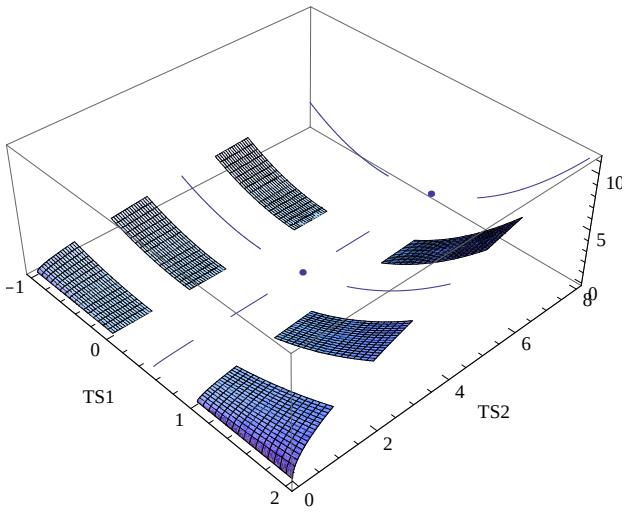


Figure 3: $\{t^2\sqrt{s} : t \in \mathbb{T}_1,\ s \in \mathbb{T}_2\}$

 ISBN 13: 978-84-613-5510-5

4 Partial Differentiation on Time Scales

Let $f : \mathbb{T}_1 \times \mathbb{T}_2 \to \mathbb{R}$ be a function. The first order delta derivatives of f at a point $(t_0, s_0) \in \mathbb{T}_1^\kappa \times \mathbb{T}_2^\kappa$ are defined by

$$\frac{\partial f(t_0,s_0)}{\Delta_1 t} = \lim_{t\to t_0,\, t\neq\sigma_1(t_0)} \frac{f(\sigma_1(t_0),s_0)-f(t,s_0)}{\sigma_1(t_0)-t}$$

and

$$\frac{\partial f(t_0,s_0)}{\Delta_2 s} = \lim_{s\to s_0,\, s\neq\sigma_2(s_0)} \frac{f(t_0,\sigma_2(s_0))-f(t_0,s)}{\sigma_2(s_0)-s}.$$

To compute the partial delta derivative of a given function at a given point, we use the command `PartialDeltaDerivative[{TS1,TS2},{t,s},f(t,s),var,p]`. Here, the partial delta derivative of `f`, a function of `t` $\in$ `TS1` and `s` $\in$ `TS2`, is taken with respect to `var` at the point `p`. Let us get some results for the time scales $\mathbb{T}_1$ and $\mathbb{T}_2$ we have given before, for a function $f(t,s) = t^2\sqrt{s}$:

```
In[7]:= PartialDeltaDerivative[{T1,T2},{t,s},t^2 s^(1/2),t,{1,2}]

Out[7]= 2√2

In[8]:= PartialDeltaDerivative[{T1,T2},{t,s},t^2 s^(1/2),t,{1/2,3}]

Out[8]= 3√3/2
```

Now, let us give the definition of the tangent plane to a surface defined on a product of two time scales.

Definition 2 *A plane Ω_0 passing through the point $P_0 = (t_0, s_0, f(t_0, s_0))$ is called the* delta tangent plane *to the surface $\mathcal{S}$ at the point P_0 if*

a. *Ω_0 passes also through the points $P_0^{\sigma_1} = (\sigma_1(t_0), s_0, f(\sigma_1(t_0), s_0))$ and $P_0^{\sigma_2} = (t_0, \sigma_2(s_0) f(t_0, \sigma_2(s_0)))$;*

b. *if P_0 is not an isolated point of the surface $\mathcal{S}$, then*

$$lim_{P\to P_0,\, P\neq P_0} \frac{d(P,\Omega_0)}{d(P,P_0)} = 0,$$

where P is the moving point of the surface $\mathcal{S}$, $d(P,\Omega_0)$ is the distance from the point P to the plane Ω_0, and $d(P,P_0)$ is the distance of the point P from the point P_0.

The command `DeltaTangentPlane[T1,T2,f(t,s),{t,s},p]` plots the function `f(t,s)` with `t` $\in$ `T1`, `s` $\in$ `T2` at the point `p`. For the previous function, let us plot the tangent plane at the point $\{\frac{1}{2}, 3\}$:

```
In[9]:= DeltaTangentPlane[T1,T2,t^2 s^(1/2),{t,s},{1/2,3}]

Out[9]=
```

 ISBN 13: 978-84-613-5510-5

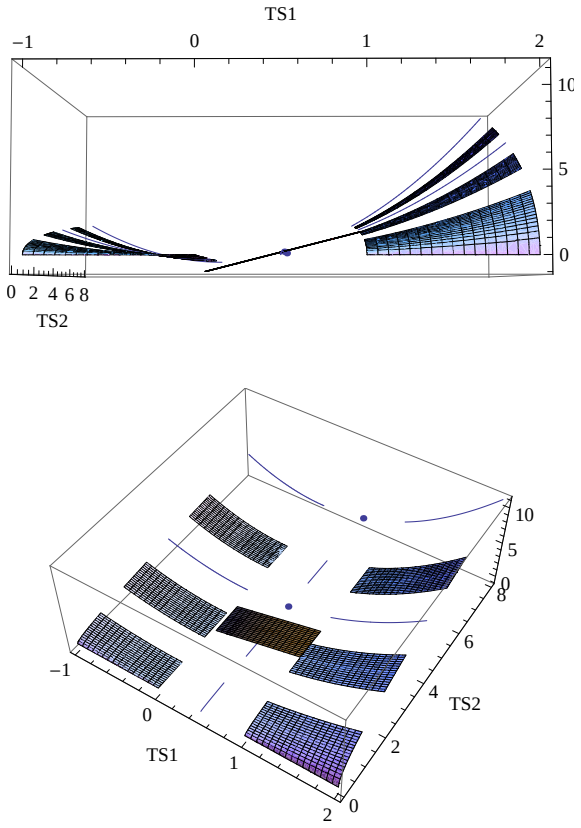


Figure 4: Delta tangent plane to the surface $f(t,s) = t^2\sqrt{s}$ at the point $(\frac{1}{2}, 3)$

 ISBN 13: 978-84-613-5510-5

References

[1] M. Bohner, A. Peterson, *Dynamic Equations on Time Scales: An Introduction with Applications*, Birkhäuser, Boston, 2001.

[2] M. Bohner and A. Peterson, *Advances in Dynamic Equations on Time Scales*, Birkhäuser, Boston, 2003.

[3] M. Bohner, G. Sh. Guseinov, *Patrial Differentiation on Time Scales*, Dynamic Systems and Applications **13** (2004) 351–379.

[4] S. Hilger, *Analysis on measure chains-a unified approach to continuous and discrete calculus*, Results Math. **35** (1990) 18–56.

[5] S. Hilger, *Differential and difference calculus — unified*, Nonlinear Analysis, Theory, Methods and Applications **30** (1997) 2683–2694.

[6] Ü. Ufuktepe, S. Kapçak, *Unification of Analysis with Mathematica*, Proceeding of the 2009 International Conference on Computational and Mathematical Methods in Science and Engineering, V 3 (2009) 1053–1063.

[7] A. Yantir, Ü. Ufuktepe, *Mathematica Applications on Time Scales for Calculus*, Lecture Notes in Computer Science (2005) 529–537, 3482.

[8] M. Bohner, G. Sh. Guseinov, *An Introduction to Complex Functions on Product of Two Time Scales*, J. Difference Equ. Appl. (2005).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

An embedding of ChuCors in L-ChuCors

Ondrej Krídlo[1], Stanislav Krajči[1] and Manuel Ojeda-Aciego[2]

[1] *Department of Computer Science, University of P. J. Šafárik in Košice, Slovakia*

[2] *Department of Applied Mathematics, University of Málaga, Spain*

emails: o.kridlo@gmail.com, stanislav.krajci@upjs.sk, aciego@uma.es

Abstract

An L-fuzzy generalization of the so-called Chu correspondences between formal contexts forms a category called L-ChuCors. In this work we show that this category naturally embeds ChuCors.

Key words: Formal Concept Analysis, Category theory, L-fuzzy logic

1 Preliminaries

Formal concept analysis (FCA) introduced by Ganter and Wille [6] has become an extremely useful theoretical and practical tool for formally describing structural and hierarchical properties of data with "object-attribute" character. Bělohlávek in [1, 2] provided an L-fuzzy extension of the main notions of FCA, such as context and concept, by extending its underlying interpretation on classical logic to the more general framework of L-fuzzy logic [7].

In this work, we aim at formally describing some structural properties of inter-contextual relationships [5, 11] of L-fuzzy formal contexts by using category theory [3], following the results in [12, 13]. The category L-ChuCors is formed by considering the class of L-fuzzy formal contexts as objects and the L-fuzzy Chu correspondences as arrows between objects.

The main result here is that L-ChuCors embeds the category ChuCors. This result is illustrated by showing different categories L-ChuCors built on different underlying truth-values sets L.

In order to make this contribution as self-contained as possible, we proceed now with the preliminary definitions of complete residuated lattice, L-fuzzy context, L-fuzzy concept and L-Chu correspondence.

Definition 1 *An algebra $\langle L, \wedge, \vee, \otimes, \to, 0, 1\rangle$ is said to be a* ***complete residuated lattice*** *if*

 ISBN 13: 978-84-613-5510-5

1. $\langle L, \wedge, \vee, 0, 1\rangle$ *is a complete bounded lattice with least element,* 0*, and greatest element,* 1*,*
2. $\langle L, \otimes, 1\rangle$ *is a commutative monoid,*
3. $\otimes$ *and* $\rightarrow$ *are adjoint, i.e.* $a \otimes b \leq c$ *if and only if* $a \leq b \rightarrow c$*, for all* $a, b, c \in L$*, where* $\leq$ *is the ordering in the lattice generated from* $\wedge$ *and* $\vee$*.*

Definition 2 *Let* L *be a complete residuated lattice, an* L***-fuzzy context*** *is a triple* $\langle B, A, r\rangle$ *consisting of a set of objects* B*, a set of attributes* A *and an* L*-fuzzy binary relation* r*, i.e. a mapping* $r\colon B \times A \rightarrow L$*, which can be alternatively understood as an* L*-fuzzy subset of* $B \times A$

We now introduce the L-fuzzy extension provided by Bělohlávek [1], where we will use the notation Y^X to refer to the set of mappings from X to Y.

Definition 3 *Consider an* L*-fuzzy context* $\langle B, A, r\rangle$*. A pair of mappings* $\uparrow\colon L^B \rightarrow L^A$ *and* $\downarrow\colon L^A \rightarrow L^B$ *can be defined for every* $f \in L^B$ *and* $g \in L^A$ *as follows:*

$$\uparrow f(a) = \bigwedge_{o \in B} (f(o) \rightarrow r(o, a)) \qquad \downarrow g(o) = \bigwedge_{a \in A} \big(g(a) \rightarrow r(o, a)\big) \tag{1}$$

Lemma 1 *Let* L *be a complete residuated lattice, let* $r \in L^{B \times A}$ *be an* L*-fuzzy relation between* B *and* A*. Then the pair of operators* $\uparrow$ *and* $\downarrow$ *form a Galois connection between* $\langle L^B; \subseteq\rangle$ *and* $\langle L^A; \subseteq\rangle$*, that is,* $\uparrow\colon L^B \rightarrow L^A$ *and* $\downarrow\colon L^A \rightarrow L^B$ *are anti tonic and, furthermore, for all* $f \in L^B$ *and* $g \in L^A$ *we have* $f \subseteq \downarrow\uparrow f$ *and* $g \subseteq \uparrow\downarrow g$*.*

Definition 4 *Consider an* L*-fuzzy context* $C = \langle B, A, r\rangle$*. An* L*-fuzzy set of objects* $f \in L^B$ *(resp. an* L*-fuzzy set of attributes* $g \in L^A$*) is said to be* ***closed in*** $\mathbf{C}$ *iff* $f = \downarrow\uparrow f$ *(resp.* $g = \uparrow\downarrow g$*).*

Lemma 2 *Under the conditions of Lemma 1, the following equalities hold for arbitrary* $f \in L^B$ *and* $g \in L^A$*,* $\uparrow f = \uparrow\downarrow\uparrow f$ *and* $\downarrow g = \downarrow\uparrow\downarrow g$*, that is, both* $\downarrow\uparrow f$ *and* $\uparrow\downarrow g$ *are closed in* C*.*

Definition 5 *An* L***-fuzzy concept*** *is a pair* $\langle f, g\rangle$ *such that* $\uparrow f = g, \downarrow g = f$*. The first component* f *is said to be the* ***extent*** *of the concept, whereas the second component* g *is the* ***intent*** *of the concept.*

The set of all L*-fuzzy concepts associated to a fuzzy context* (B, A, r) *will be denoted as* L*-*$FCL(B, A, r)$*.*

An ordering between L*-fuzzy concepts is defined as follows:* $\langle f_1, g_1\rangle \leq \langle f_2, g_2\rangle$ *if and only if* $f_1 \subseteq f_2$ *if and only if* $g_1 \supseteq g_2$*.*

Proposition 1 *The poset* $(L\text{-}FCL(B, A, r), \leq)$ *is a complete lattice where*

$$\bigwedge_{j \in J} \langle f_j, g_j\rangle = \Big\langle \bigwedge_{j \in J} f_j, \uparrow \big(\bigwedge_{j \in J} f_j\big)\Big\rangle$$

$$\bigvee_{j \in J} \langle f_j, g_j\rangle = \Big\langle \downarrow \big(\bigwedge_{j \in J} g_j\big), \bigwedge_{j \in J} g_j\Big\rangle$$

 ISBN 13: 978-84-613-5510-5

Finally, we proceed with the definition of L-Chu correspondences, for which we need the notion of L-multifunction.

Definition 6 *An L-**multifunction** from X to Y is a mapping $\varphi\colon X \to L^Y$. The set L-$Mfn(X,Y)$ of all the L-multifunctions from X to Y can be endowed with a poset structure by defining the ordering $\varphi_1 \leq \varphi_2$ as $\varphi_1(x)(y) \leq \varphi_2(x)(y)$ for all $x \in X$ and $y \in Y$.*

Definition 7 *Consider two L-fuzzy contexts $C_i = \langle B_i, A_i, r_i\rangle, (i = 1,2)$, then the pair $\varphi = (\varphi_l, \varphi_r)$ is called a **correspondence** from C_1 to C_2 if φ_l and φ_r are L-multifunctions, respectively, from B_1 to B_2 and from A_2 to A_1 (that is, $\varphi_l : B_1 \to L^{B_2}$ and $\varphi_r : A_2 \to L^{A_1}$).*

*The L-correspondence φ is said to be a **weak L-Chu correspondence** if the equality $\hat{r}_1(\chi_{o_1}, \varphi_r(a_2)) = \hat{r}_2(\varphi_l(o_1), \chi_{a_2})$ holds for all $o_1 \in B_1$ and $a_2 \in A_2$. By unfolding the definition of $\hat{r}_i$ this means that*

$$\bigwedge_{a_1 \in A_1} (\varphi_r(a_2)(a_1) \to r_1(o_1, a_1)) = \bigwedge_{o_2 \in B_2} (\varphi_l(o_1)(o_2) \to r_2(o_2, a_2)) \tag{2}$$

*A weak Chu correspondence φ is an **L-Chu correspondence** if $\varphi_l(o_1)$ is closed in C_2 and $\varphi_r(a_2)$ is closed in C_1 for all $o_1 \in B_1$ and $a_2 \in A_2$. We will denote the set of all Chu correspondences from C_1 to C_2 by L-$ChuCors(C_1, C_2)$.*

In the following definition and lemma, we introduce some connections between the right and the left sides of L-Chu correspondences.

Definition 8 *Given a mapping $\varpi : X \to L^Y$ we consider the following associated mappings $\varpi_*\colon L^X \to L^Y$ and $\varpi^* : L^Y \to L^X$, defined for all $f \in L^X$ and $g \in L^Y$ by*

1. $\varpi_*(f)(y) = \bigvee_{x \in X}(f(x) \otimes \varpi(x)(y))$
2. $\varpi^*(g)(x) = \bigwedge_{y \in Y} \varpi(x)(y) \to g(y)$

Lemma 3 *Let $C_i = \langle B_i, A_i, r_i\rangle$ for $i = 1,2$ be L-fuzzy contexts. Let $\varphi = (\varphi_l, \varphi_r) \in L$-ChuCors$(C_1, C_2)$. Then*

- *for all $f \in L^{B_1}$ and $g \in L^{A_2}$, the following equalities hold*

$$\uparrow_2 (\varphi_{l*}(f)) = \varphi_r^*(\uparrow_1 (f)) \quad \text{and} \quad \downarrow_1 (\varphi_{r*}(g)) = \varphi_l^*(\downarrow_2 (g))$$

- *for all $o_1 \in B_1$ and $a_2 \in A_2$, the following equalities hold*

$$\varphi_l(o_1) = \downarrow_2 (\varphi_r^*(\uparrow_1 (\chi_{o_1}))) \quad \text{and} \quad \varphi_r(a_2) = \uparrow_1 (\varphi_l^*(\downarrow_2 (\chi_{a_2})))$$

 ISBN 13: 978-84-613-5510-5

2 The category L-ChuCors

We introduce now the category of L-Chu correspondences between L-fuzzy formal contexts as follows:

- **objects** L-fuzzy formal contexts
- **arrows** L-Chu correspondences
- **composition** $\varphi_2 \circ \varphi_1 : C_1 \to C_3$ **of arrows** $\varphi_1 : C_1 \to C_2$, $\varphi_2 : C_2 \to C_3$ ($C_i = \langle B_i, A_i, r_i \rangle$, $i \in \{1, 2\}$)
  - $(\varphi_2 \circ \varphi_1)_l : B_1 \to L^{B_3}$ and $(\varphi_2 \circ \varphi_1)_r : A_3 \to L^{A_1}$
  - $(\varphi_2 \circ \varphi_1)_l(o_1) = {\downarrow_3}{\uparrow_3} (\varphi_{2l*}(\varphi_{1l}(o_1)))$, where
$$\varphi_{2l*}(\varphi_{1l}(o_1))(o_3) = \bigvee_{o_2 \in B_2} \varphi_{1l}(o_1)(o_2) \otimes \varphi_{2l}(o_2)(o_3)$$
  - $(\varphi_2 \circ \varphi_1)_r(a_3) = {\uparrow_1}{\downarrow_1} (\varphi_{1r*}(\varphi_{2r}(a_3)))$, where
$$\varphi_{1r*}(\varphi_{2r}(a_3))(a_1) = \bigvee_{a_2 \in A_2} \varphi_{2r}(a_3)(a_2) \otimes \varphi_{1r}(a_2)(a_1)$$

Theorem 1 *L-fuzzy Chu correspondences between L-fuzzy formal contexts form a category with the composition defined above.*

Proof: We just have to check the existence of identity arrows and the associativity of composition. The latter is just a matter of straightforward calculation, the identity arrows $\iota : C \to C$ are defined as follows for any given L-fuzzy context $C = \langle B, A, r \rangle$:

- $\iota_l(o) = {\downarrow}{\uparrow} (\chi_o)$, for all $o \in B$
- $\iota_r(a) = {\uparrow}{\downarrow} (\chi_a)$, for all $a \in A$. □

3 L-ChuCors **embeds** ChuCors

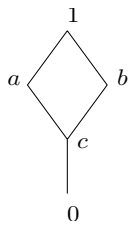
In the following paragraph, we sketchily argue that ChuCors can be embedded in any of the extensions L-ChuCors where L is a complete residuated lattice.

Assume that $\langle L_1, \wedge, \vee, \otimes, \to, 0, 1 \rangle$ and $\langle L_2, \wedge, \vee, \otimes, \to, 0, 1 \rangle$ are two complete residuated lattices, such that L_2 is a sublattice of L_1. Any L_2-fuzzy formal context $\langle B, A, r \rangle$ satisfies that $r \in L_2^{B \times A} \subseteq L_1^{B \times A}$. This inclusion implies that the class of all objects of L_2-ChuCors is a subclass of L_1-ChuCors. Moreover, every concept constructed in $\langle B, A, r \rangle$ by using the underlying logic provided by L_2 can be seen as well as a concept under the logic of L_1. As a result, the concept lattice L_2-$FCL(B, A, r)$ is a sublattice of the concept lattice L_1-$FCL(B, A, r)$.

The following example illustrates the previous results on the light of two particular cases for L_i.

Example 1 *Consider L_1 and L_2 the lattices shown to the left of the picture below, together with the two L_2-formal contexts shown in the right.*

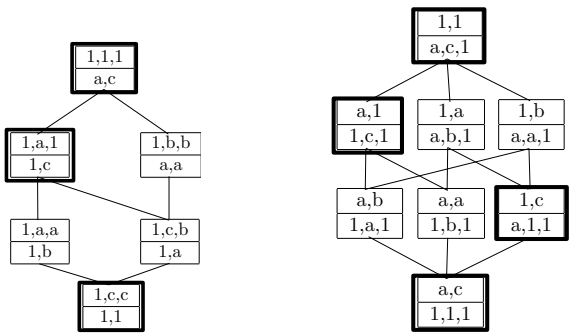
 ISBN 13: 978-84-613-5510-5



	a_{11}	a_{12}
o_{11}	*1*	*1*
o_{12}	*a*	*c*
o_{13}	*1*	*c*

	a_{21}	a_{22}	a_{23}
o_{21}	*a*	*1*	*1*
o_{22}	*1*	*c*	*1*

Consider two complete residuated lattice(s) to be consisting of the infimum on L_i, together with its residual implication defined as $k \to l = \bigvee\{m \in L \mid m \wedge k \leq l\}$, for all $k, l, m \in L_i$ where $i \in \{1, 2\}$. The concept lattices on the underlying logic of L_1 are shown in the pictures below, where the concepts in bold line are those in the frame associated to L_2.



The common L_2 and L_1-Chu correspondences are shown below:

φ_l	φ_r
1,c,c	*a,c,1*
1,c,c	*a,c,1*

,

φ_l	φ_r
1,c,c	*a,c,1*
1,a,1	*a,1,1*

The following result formally states the general relation between L_i-ChuCors.

Lemma 4 *Let C_1, C_2 be the L_2-contexts. L_2-ChuCors$(C_1, C_2) \subseteq L_1$-ChuCors(C_1, C_2).*

It is easy to see that the connection of two L_2-Chu correspondences make a new L_2-Chu correspondence. In addition, the set of L_2-Chu correspondences between two L_2-contexts is a subset of all L_1-Chu correspondences between the same contexts. L_2-Chu correspondences form a category, so the set of arrows is closed under the connections of arrows, as a result the set of L_1-Chu correspondences is closed under connections of L_2-Chu correspondences. Thus, we have just proved the following

Lemma 5 *Let C_i for $i \in \{1, 2, 3\}$ be two L_2-contexts. For every L_2-Chu correspondence $\varphi \in L_2$-ChuCors(C_1, C_2) and $\psi \in L_2$-ChuCors(C_2, C_3) holds $\psi \circ \varphi \in L_1$-ChuCors$(C_1, C_3)$.*

In consequence, we can state

 ISBN 13: 978-84-613-5510-5

Theorem 2 *Under the environment hypotheses of this section, the category* L_2-ChuCors *naturally embeds in* L_1-ChuCors.

As the category ChuCors of classical Chu correspondences are defined on classical, two-valued logic, which is a special case of any logic defined on complete residuated lattice, we obtain

Corollary 1 *The category* ChuCors *naturally embeds in* L-ChuCors

References

[1] R. Bělohlávek. Fuzzy concepts and conceptual structures: induced similarities. In *Joint Conference on Information Sciences*, pages 179–182, 1998.

[2] R. Bělohlávek. Lattices of fixed points of fuzzy Galois connections. *Mathematical Logic Quartely*, 47(1):111–116, 2001.

[3] M. Barr, Ch. Wells. *Category theory for computing science*. Prentice Hall, 1995.

[4] B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, second edition, 2002.

[5] B. Ganter. Relational Galois connections. *Lect Notes in Computer Science* 4390:1–17, 2007

[6] B. Ganter and R. Wille. *Formal concept analysis*. Springer-Verlag, 1999.

[7] P. Hájek. Metamathematics of fuzzy logic. Kluwer Academic Publisher, 2001.

[8] S. Krajči. Every concept lattice with hedges is isomorphic to some generalized concept lattice. In *Intl Workshop on Concept Lattices and their Applications*, pages 1–9, 2005.

[9] S. Krajči. A generalized concept lattice. *Logic Journal of IGPL*, 13(5):543–550, 2005.

[10] O. Kridlo, M. Ojeda-Aciego, On the L-fuzzy generalization of Chu correspondences, *International Journal of Computer Mathematics*, to appear.

[11] M. Krötzsch, P. Hitzler, G-Q. Zhang, Morphisms in Context. *Lecture Notes in Computer Science* 3596:223–237, 2005.

[12] H. Mori. Chu Correspondences. *Hokkaido Matematical Journal*, 37:147–214, 2008

[13] H. Mori. Functorial properties of Formal Concept Analysis. *Lecture Notes in Artificial Intelligence* 4604:505–508, 2007

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Sampling with the eigenfunctions of finite Hankel transform and the relevant calculations

Tatiana Levitina[1]

[1] *Institut Computational Mathematics, Technische Universität Braunschweig, Pockelsstraße 14, D-38106 Braunschweig*

emails: levitina@tu-bs.de

Abstract

As was recently shown by Walter and Shen, for band-limited functions mainly concentrated on some time interval, the truncation error of sampling expansions with Finite Fourier Transform Eigenfunctions may be essentially less than that of the conventional Shannon series.

Similar expansions can be written for Hankel–band–limited functions in terms of Finite Hankel Transform Eigenfunctions. Yet, practical implementation requires efficient and accurate numerical methods both for FHTE evaluation and for computation of sampling coefficients.

Key words: Kramer Sampling theorem, finite Hankel transform, Bessel functions

MSC 2000: 41A05, 65D05, 65D20

1 Introduction

The classical Whittaker-Kotelnikov-Shannon (WSK) sampling theorem claims that any Ω–band–limited function $f(x) \in L^2$, i.e. representable as

$$f(t) = \frac{1}{2\pi} \int_{-\Omega}^{\Omega} e^{i\,\omega\,t} F[f](\omega)\, d\omega$$

(here $F[f](\omega)$ stands for the Fourier transform of $f(t)$), can be reconstructed from its equally spaced samples

$$f(t) = \sum_{k=-\infty}^{\infty} f\left(\frac{\pi k}{\Omega}\right) \frac{\sin \Omega\,(x - \pi k/\Omega)}{\Omega\,(x - \pi k/\Omega)} \tag{1}$$

 ISBN 13: 978-84-613-5510-5

Another sampling formula for $f(t)$ invented and studied in [1, 2], is based on the eigenfunctions of the Finite Fourier Transform (FFT), first defined inside the interval $I_\Omega = [-\Omega, \Omega]$

$$\int_{-\Omega}^{\Omega} \exp(\mathrm{i}x\,y)\,\psi_l(\Omega, y)\,dy \;=\; \mu_l(\Omega)\,\psi_l(\Omega, x)\,, \tag{2}$$

and then continued analytically according to (2) to the whole real axis; the associated FFT eigenvalues $\mu_l = \mu_l(\Omega)$ are ordered by magnitude, $\sqrt{2\,\pi} > |\mu_0| > |\mu_1| > \ldots > 0$. In terms of these functions the reconstruction formula looks more complicated because of double summation:

$$f(t) = \frac{\pi}{\Omega} \sum_{n=0}^{\infty} \frac{|\mu_n|^2}{2\pi} \left\{ \sum_{k=-\infty}^{\infty} \psi_n\left(\frac{\pi k}{\Omega}\right) f\left(\frac{\pi k}{\Omega}\right) \right\} \psi_n(x)\,. \tag{3}$$

However in practice the latter series may converge much faster then the classical one, in case that the function $f(t)$ vanishes rapidly outside the interval I_Ω.

Thus in [1, 2] for Ω–band–limited functions $f(x) \in L^2$, which Fourier transform $F[f](\omega)$ is sufficiently smooth, truncation error estimates are given showing that double series (3) to a very good accuracy can be truncated at $k = \pm \left[\frac{\Omega}{\pi}\right]$, $n = 2\left[\frac{\Omega}{\pi}\right]$, provided f itself is mostly concentrated in the interval $[-\Omega, \Omega]$. Another example illustrating the advantages of formula (3) is given in recent papers [3, 4]. A convolution of two $\frac{\Omega}{\sqrt{2}}$–FFT eigenfunctions is an Ω–band–limited and may practically be reconstructed from its samples computed at $2\left[\frac{\Omega}{\pi}\right]$ points, although the Fourier transform of the convolution is even not continuous.

2 Kramer's sampling theorem and Finite Hankel Transform

Kramer's generalization of the WSK sampling theorem extends the variety of functions which can be reconstructed from their sampled values by mean of orthogonal sampling formulae and establishes general conditions that allow the reconstruction (see e.g. [5, 6, 7, 8]). Of special importance for image processing, in particular for computerized tomography, is the Hilbert space comprising Hankel–band–limited functions, each been expressed through a finite Fourier–Bessel integral:

$$f(r) = \int_0^1 \sqrt{r\,\rho}\; J_\nu(r\rho)\, H[f](\rho)\, d\rho,$$

above $H[f](\rho) = \int_0^\infty \sqrt{\rho\,\lambda}\; J_\nu(\rho\,\lambda)\, H[f](\lambda)\, d\lambda$, is the Hankel transform of $f(r)$, and $J_\nu(\cdot)$ is the νth–order Bessel function of the first kind, $\nu = 0, 1, 2, \ldots$.

 ISBN 13: 978-84-613-5510-5

Let us denote positive zeroes of $J_\nu(r)$ as $r_{\nu n},\ n = 0, 1, 2, \ldots$. Kramer's sampling formula for function $f(r)$ yields then [9, 10]

$$f(r) = \sum_{n=0}^{\infty} f(r_{\nu n}) \frac{2\sqrt{r\, r_{\nu n}}\ J_\nu(r)}{(r^2 - r_{\nu n}^2)\, J'_\nu(r_{\nu n})}. \tag{4}$$

If $r \to \infty$, Bessel function $J_\nu(r)$ vanishes as $\dfrac{1}{\sqrt{r}}$, which defines the rate of convergence of series (4). As in the case of the finite Fourier transform, sampling formula rewritten in terms of Finite Hankel Transform (FHT) eigenfunctions can converge much faster.

3 Sampling with the FHT eigenfunctions

Let $T_{\nu k}(r)$ be an eigenfunction of FHT associated with the eigenvalue $\mu_{\nu k}$, i.e.:

$$\int_0^1 \sqrt{r\,\rho}\ J_\nu(r\rho)\ T_{\nu\, k}(\rho)\, d\rho\ =\ \mu_{\nu k}\, T_{\nu\, k}(r), \tag{5}$$

the eigenvalues are ordered by magnitude, $|\mu_{\nu\, 0}| > |\mu_{\nu\, 1}| > \ldots > 0$.

In terms of $T_{\nu k}(r)$

$$f(r) = \sum_{n=0}^{\infty} f(r_{\nu\, n}) \sum_{k=0}^{\infty} 2\, \mu_{\nu\, k}\, \frac{T_{\nu\, k}(r)\ T_{\nu\, k}(r_{\nu\, n})}{(r^2 - r_{\nu\, n}^2)\, J'_\nu(r_{\nu\, n})}.$$

Special properties of functions $T_{\nu k}(r)$, known also as generalized prolate functions, were studied earlier in [11, 12]; one can also apply here the general results discussed in [13].

Thus, e.g. $T_{\nu k}(r)$ are simulteneously eigenfunctions of a singular self–adjoint Sturm–Liouville problem, which allows one to apply for thier evaluation a general approach developed in [14, 15, 16] for numerical solution of similar problems. The approach is efficient, robust and accurate and has previousely been applied e.g. for evaluation of prolate spheroidal and ellipsoidal wave functions.

4 Bessel functions and FHT eigenvalues

Numerical method for evaluation of FHT eigenfunctions $T_{\nu k}(r)$ was earlier described in [17]. Sampling coefficients of (4), however, require also eigenvalues $\mu_{\nu k}$, that may only be defined via integration of $T_{\nu k}(r)$ versus Bessel function J_ν.

Although Bessel functions are widely used, their evaluation remains problematic, especially if integrals involving Bessel functions are needed, rather than the functions themselves. In our case, in addition to the integral (5) determining the eigenvalue $\mu_{\nu k}$, we also need all zeros $r_{\nu n}$ of the Bessel function $J_\nu(r)$ inside the interval $(0, 1)$ (in general, inside $I = (0, a)$), as well as the values $J'_\nu(r_{\nu n})$ (see (4)).

Conventional Bessel function $J_n(r)$ is defined as the solution of the Bessel equation

$$\frac{1}{r}\left(r\, y'(r)\right)' + 1 - \frac{n^2}{r^2}\, y(r)\ =\ 0 \tag{6}$$

 ISBN 13: 978-84-613-5510-5

bounded at $r = 0$ and normalized in accordance with the assymptotical behaviour at infinity

$$J_n(r) = \sqrt{\frac{2}{\pi\, r}} \cos\left(r - \frac{\pi}{2}\, n - \frac{\pi}{4}\right) + O\left(\frac{1}{r^{3/2}}\right).$$

Yet, in (4) the latter normalization plays no role, the ratio $\dfrac{\mu_{\nu\, k}}{J'_\nu(r)}$ remains the same for all bounded solutions of (6), provided $\mu_{\nu\, k}$ is in a correspondence with this solution. This yields us a possibility to avoid explicit evaluation of conventional Bessel functions.

In the present talk we shall give a detailed description of all the relevant calculations. Prelimenary test calculations will illustrate the facilites of the numerical technique.

References

[1] G. G. Walter, X. Shen, *Sampling With Prolate Spheroidal Wave Functions* , Journal of Sampling Theory in Signal and Image Processing **2 (1)** (2003) 25–5.

[2] G. G. Walter, X. Shen, *Wavelets Based On Prolate Spheroidal Wave Functions*, Journal of Fourier Analysis and Applications **10 (1)** (2004) 1–26.

[3] T. V. Levitina, E. J. Brändas, *Sampling formula for convolution with a prolate*, International Journal of Computer Mathematics **85** (2008) 487–496

[4] T. V. Levitina, E. J. Brändas, *Filter diagonalization: Filtering and postprocessing with prolates*, Computer Physics Communications **180 (9)** (2009) 1448–1457

[5] H. P. Kramer, *A generalized sampling theorem*, Jour. Math. Phys **38** (1959)68–72

[6] A. J. Jerri, Applications for Kramer's Generalized Sampling Theorem, Journal of Engineering Mathematics **3(2)** (1969) 103–105

[7] A. J. Jerri, *The. Shannon sampling theorem-Its various extensions and applications: A tutorial review* Proc. IEEE **65(11)** (1977) 1565–1596

[8] A. I. Zayed, *On Kramer's sampling theorem associated with general Strum-Liouville problems and Lagrange interpolation*, SIAM Journal on Applied Mathematics **51 (2)** (1991) 575–604

[9] J.R.Higgins, *An interpolation series associated with the Bessel-Hankel transform*, Journal of the London Mathematical Society **5** (1972) 707–714

[10] J.R.Higgins, *Five short stories about the cardinal series*, Bull. Amer. Math. Soc. (N.S.) **12(1)** (1985) 45–89

[11] D. Slepian. , *Prolate spheroidal wave functions, Fourier analysis and uncertainty, IV: Extensions to many dimensions; generalized prolate spheroidal functions*, Bell System Technical Journal **43** (1964) 3009–3058

 ISBN 13: 978-84-613-5510-5

[12] I. V. Komarov, L. I. Ponomarev, S. Yu. Slavyanov, *GSpheroidal and Coulomb Spheroidal Functions* [in Russian], Nauka, Moscow, 1976.

[13] A. I. Zayed, *A generalization of the prolate spheroidal wave functions* , Proc. Amer. Math. Soc. **135** (2007) 2193–2203

[14] A. A. Abramov, A. L. Dyshko, N. B. Konyukhova, T. V. Pak, and B. S. Pariiskii, *Evaluation of prolate spheroidal function by solving the corresponding differential equations*, U.S.S.R. Comput. Math. and Math. Phys. **24(1)** (1984) 1–11.

[15] A. A. Abramov, A. L. Dyshko, N. B. Konyukhova, and T. V. Levitina, *Computation of radial wave functions for spheroids and triaxial ellipsoids by the modified phase function method*, [Comput. Math. and Math. Phys. **31(2)** (1991) 25–42.

[16] T. V. Levitina, E J. Brändas , *Computational techniques for Prolate Spheroidal Wave Functions in Signal Processing*, J.Comp.Meth.Sci. & Engrg. **1** (2001) 287–313

[17] B. Larsson, T. V. Levitina, E. J. Brandas, *Eigenfunctions of the 2D finite Fourier transform*, J. Comp.Meth.Sci. & Engrg. **4** (2004) 135–148

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A note on the construction of the semi-analytical integrators in celestial mechanics with aid of a Poisson series processors

J.A. López Ortí[1], V. Agost Gómez[1] and M. Barreda Rochera[1]

[1] *Department of Mathematics, University Jaume I of Castellón (Spain)*

emails: lopez@mat.uji.es, agostv@uji.es, barreda@mat.uji.es

Abstract

The main aim of this paper is to build up a set of semi-analytical integrators based in the use of several kind of anomalies as temporal variables. The integrators are based on the development of the selected anomaly according to the mean and anomaly.

To manipulate these developments a new Poisson procesor series will be used. This processor has been written as a C++ class and it contains a set of methods to manage the most common arithmetic and funtional operations with these objects.

Key words: Celestial Mechanics. Planetary Theories. Algorithms. Orbital Mechanics. Perturbation Theory. Computational Algebra.
MSC 2000: 70F05, 70F10, 70F15,70M20

1 Introduction

One of main problems in celestial mechanics is the study of the solutions of the perturbed motion of the celestial bodies in the solar system, that is, the so-called planetary theories. These theories can be classified into analytical theories, semi-analytical theories, and numerical theories

The motion of a body in the solar system is completly defined by the value of its elements [2],[7],[15]. These values are given by the planetary Lagrange equations [9]

$$\frac{d\overrightarrow{\sigma}_i}{dt} = \overrightarrow{f}_i(\overrightarrow{\sigma}_1, \ldots, \overrightarrow{\sigma}_n), \quad i = 1, \ldots, n \tag{1}$$

The analytical and semi-analytical theories involve the management of the Fourier and Poisson series, the appropriate techniques to develop the inverse of the distance between two planets according to the chosen anomaly [8],[1],[14],[10],[11].

 ISBN 13: 978-84-613-5510-5

A Poisson series of the type (s,l) is a mathematical object defined as

$$S=\sum_{i_1=0}^{\infty}\cdot\cdot\sum_{i_s=0}^{\infty}\cdot\cdot\sum_{j_1=-\infty}^{\infty}\cdot\cdot\sum_{j_l=-\infty}^{\infty}C_{i_1,..,i_s}^{j_1,..,j_l}x_1^{i_1}\cdot\cdot x_s^{i_s}\cos(j_1y_1+\cdot\cdot+j_ly_l+\Phi_{j_1,\cdots,j_l}) \quad (2)$$

where $C_{i_1,..,i_s}^{j_1,..,j_l}$ are real or complex numbers and $\Phi_{i_1,..,i_s}^{j_1,..,j_l}$ are real numbers. The variables $(x_1,\cdot\cdot,x_s)$ are called power variables, and the variables $(y_1,\cdot\cdot,y_l)$ are called angular variables.

The second member of the Lagrange planetary equations can be developed as

$$\frac{d\overrightarrow{\sigma}_i}{dt}=\sum_{k=0}^{\infty}\sum_{j_1=0}^{\infty}\cdot\cdot\sum_{j_l=-\infty}^{\infty}C_i^{j_1,..,j_l}t^k\cos(j_1\Psi_1+\cdot\cdot+j_l\Psi_l+\Phi_{j_1,\cdots,j_l}) \quad (3)$$

where t is the time and Ψ_i the anomaly of the body i.

2 semi-analitical integrators

To integrate the Lagrange planetary equations it is necessary the evaluation of the quantities

$$\int_{t_0}^{t}\cos(j_1\Psi_1+\cdot\cdot+j_l\Psi_l+\Phi_{j_1,\cdots,j_l})dt \quad (4)$$

This process is immediate if the mean anomalies $M_1,\ldots,M_k$ are used in the developmets. In this case $j_1M_1+\ldots+j_lM_l=(j_1n_1+\ldots+j_ln_l)\,t$ where $n_1,\ldots,n_l$ are the mean motions

$$\int_{t_0}^{t}\cos(j_1\Psi_1+\cdot\cdot+j_l\Psi_l+\Phi_{j_1,\cdots,j_l})dt=\left.\frac{\cos(j_1\Psi_1+\cdot\cdot+j_l\Psi_l+\Phi_{j_1,\cdots,j_l}+\frac{\pi}{2})}{j_1n_1+\ldots+j_ln_l}\right|_{t_0}^{t} \quad (5)$$

Let Ψ be an anomaly connected whit M through of the Kepler equation.

$$M=\Psi+\sum_{k=0}^{\infty}K_k(e)\cos k\Psi \quad (6)$$

and from this equation we have

$$n_idt=\left(1+\sum_{k=1}^{\infty}kK_k(e_i)\sin k\Psi_i\right)d\Psi_i \quad (7)$$

and by inversion

$$d\Psi_i=n_i\left(1+\sum_{k=0}^{\infty}T_k(e_i)\sin k\Psi_i\right)dt \quad (8)$$

Let us now $\xi=j_1\Psi_1+\ldots+j_l\Psi_l$, then

$$dt=\frac{1}{j_1n_1+\ldots+j_ln_l}d\xi+ \\ +\sum_{s=1}^{l}j_s\left[\sum_{k=1}^{\infty}kK_k(e_s)\sin k\Psi_s\right]\left[n_s(1+\sum_{k=0}^{\infty}T_k(e_s)\sin k\Psi_s)\right]dt \quad (9)$$

 ISBN 13: 978-84-613-5510-5

If Ψ is an anomaly connected with M through a Sundman transformation [13], [6], Kepler equation can be obtained through the eccentric anomaly

$$\Psi = E + \sum_{k=0}^{\infty} B_k(e) \cos kE \tag{10}$$

repalcing E and $\cos kE$ by their developments with respect to M [9] we obtain

$$\Psi = M + \sum_{k=0}^{\infty} C_k(e) \cos M \tag{11}$$

The Kepler equation can be obtained from this equation using the Deprit algorithm [4].

The operations described above involve a hard management of Poisson Series. For this purpose we use a new C++ class developed by the authors called poison.h [12].

The main public methods of poisson.h are the arithmetic operations $+$, $-$, $*$, *pow*; the extension of the most common functions sin cos, exp, ..., to be evaluated over Poisson series [3]; and functional operations as Taylor developments and series inversion procedures based on Lagrange and Deprit methods [4]. The operators and common functions have been overloaded in order to be more user friendly.

3 Concluding Remarks

The process described above is a suitable algorithm to construct a set of semi-analytical integrators using an extended class of anomalies as temporal variables. The use of the the C++ class poisson.h allows the managment of the equations.

4 Acknowledgments

This research has been partially supported by Grant GV/2009/027 from the Generalitat Valenciana and Grant P1-06I455.01/1 from Bancaja.

References

[1] P. Bretagnon and G. Francou , 'Variations Seculaires des Orbites Planetaires. Thèorie VSOP87', *Astron. Astrophys.*, **Vol. 114**, pp. 69-75. 1988

[2] D. Brower, G. M. Clemence, *Celestial Mechanics*, Ed Academic Press, New York, 1965.

[3] V.A.Brumberg,*Analytical Thechniques of Celestial Mechanics*, Ed Springer-Verlag, Berlin, 1995.

[4] Deprit, A.,"A Note on Lagrange's Inversion Formula",*Celestial Mechanics.* **20** (1979) 325–327

 ISBN 13: 978-84-613-5510-5

[5]

[6] Ferrándiz, J. M.; Ferrer, S; Sein-Echaluce, M.L."Generalized Elliptic Anomalies". *Celestial Mechanics.* **40** (1987) 315–328

[7] Y. Hagihara, *Celestial Mechanics*, Ed MIT Press, Cambridge MA, 1970.

[8] J. Kovalewsky, *Introduction to Celestial Mechanics*, Ed D. Reidel Publishing Company, DoDrecht-Holland, 1967.

[9] L.L Levallois, J. Kovalewsky, *Geodesie Generale Vol 4*, Ed Eyrolles, Paris, 1971.

[10] J.A. López, M. Barreda,"A Formulation to Obtain Semi-analytical Planetary Theories Using True Anomalies as Temporal Variables", *Journal of Computational and Applied Mathematics.* **6** In Press,(now available online).

[11] J.A. López, M. Barreda, J. Artes, "Integration Algorithms to Construct Semi-analytical Planetary Theories", *Wseas Transactions on Mathematics.* **6** (2006) 609-614.

[12] J.A.Lopez, V. Agost, M. Barreda"A new C++ Poisson series processor". *Proccedings of International conference on computational methods in science and engineering ICCMSE 2009 (In press).*

[13] P. Nacozy, "The intermediate anomaly." *Celestial Mechanics.* **16** (1977) 309–313.

[14] J.L. Simon,"Computation of the first and second derivatives of the Lagrange equations by harmonic analysis", *Astron&Astrophys.* **17** (1982) 661–692

[15] F. F. Tisserand, *Traité de Mecanique Celeste*, Ed Gauthier-Villars, Paris, 1896.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

On the existence of stable models in normal residuated logic programs

Nicolás Madrid[1] **and Manuel Ojeda-Aciego**[1]

[1] *Departamento de Matemática Aplicada, Universidad de Málaga, Spain*

emails: nmadrid@ctima.uma.es, aciego@ctima.uma.es

Abstract

We introduce a sufficient condition which guarantees the existence of stable models for a normal residuated logic program interpreted on the truth-space $[0,1]^n$. Specifically, the continuity of the connectives involved in the program ensures the existence of stable models.

1 Introduction

Similarly to classical logic programming, the existence of fuzzy stable models cannot be guaranteed for an arbitrary normal residuated logic program [11]. Necessary conditions to ensure the existence of stable models has been widely studied in classical logic programming. In fact, the syntactic characterization of normal programs with stable models can be found in [1].

However the characterization in the fuzzy framework is much more complicated since it involves two different dimensions: "the syntactic structure of the normal program" and "the choice of suitable connectives in the residuated lattice". For short, we will call them the *syntactic* and the *semantic dimension*, respectively.

In classical logic programming only syntactic conditions are available since the connectives are fixed. However, for normal residuated logic program the semantic dimension plays also a crucial role; for example the program with only one rule

$$\mathbb{P} = \{\langle p \leftarrow \neg p; 1\rangle\}$$

has a stable model if and only if the operator associated with $\neg$ has a fixpoint. As far as we know, establishing semantic conditions for guaranteeing the existence of stable models has not been directly attempted, although sufficient conditions underlie in some approaches; for example [12] proves that every normal logic program has stable models in the 3-valued Kleene logic and, more generally, [3, 8] show that every normal

 ISBN 13: 978-84-613-5510-5

residuated logic program has stable models if the underlying residuated lattice has an appropriate bilattice structure [5].

In this paper we provide another condition on the residuated lattice to enssure the existence of stable models, more specifically: if the lattice selected is an euclidean space and the connectives $*$ and $\neg$ in the residuated lattice are continuous, then the existence of at least a fuzzy stable model is guaranteed.

2 Preliminaries

Let us start this section by recalling the definition of residuated lattice, which fixes the set of truth values and the relationship between the conjunction and the implication (the adjoint condition) occurring in our logic programs.

Definition 1 *A* residuated lattice *is a tuple* $(L, \leq, *, \leftarrow)$ *such that:*

1. $(L, \leq)$ *is a complete bounded lattice, with top and bottom elements 1 and 0.*
2. $(L, *, 1)$ *is a commutative monoid with unit element 1.*
3. $(*, \leftarrow)$ *forms an adjoint pair, i.e.* $z \leq (x \leftarrow y)$ *iff* $y * z \leq x \quad \forall x, y, z \in L$.

In the rest of the paper we will consider a residuated lattice enriched with a negation operator, $(L, *, \leftarrow, \neg)$. The negation $\neg$ will model the notion of default negation often used in logic programming. As usual, a negation operator, over L, is any decreasing mapping $n\colon L \to L$ satisfying $n(0) = 1$ and $n(1) = 0$.

Definition 2 *Given a residuated lattice with negation* $(L, \leq, *, \leftarrow, \neg)$*, a* normal residuated logic program $\mathbb{P}$ *is a set of weighted rules of the form*

$$\langle p \leftarrow p_1 * \cdots * p_m * \neg p_{m+1} * \cdots * \neg p_n; \quad \vartheta \rangle$$

where ϑ *is an element of* L *and* $p, p_1, \ldots, p_n$ *are propositional symbols.*

It is usual to denote the rules as $\langle p \leftarrow \mathcal{B}; \vartheta \rangle$. The formula $\mathcal{B}$ is usually called the *body* of the rule, p is called its *head* and ϑ is called its weight.

A *fact* is a rule with empty body, i.e facts are rules with this form $\langle p \leftarrow \quad ; \vartheta \rangle$. The set of propositional symbols appearing in $\mathbb{P}$ is denoted by $\Pi_{\mathbb{P}}$.

Definition 3 *A fuzzy* L-interpretation *is a mapping* $I\colon \Pi_{\mathbb{P}} \to L$*; note that the domain of the interpretation can be lifted to any rule by homomorphic extension.*

We say that I satisfies *a rule* $\langle \ell \leftarrow \mathcal{B}; \quad \vartheta \rangle$ *if and only if* $I(\mathcal{B}) * \vartheta \leq I(\ell)$ *or, equivalently,* $\vartheta \leq I(\ell \leftarrow \mathcal{B})$*. Finally,* I *is a* model *of* $\mathbb{P}$ *if it satisfies all rules (and facts) in* $\mathbb{P}$*.*

Note that the ordering relation in the residuated lattice $(L, \leq)$ can be extended over the set of all L-interpretations as follows: *Let* I *and* J *be two* L*-interpretations, then* $I \leq J$ *if and only if* $I(p) \leq J(p)$ *for all propositional symbol* $p \in \Pi_{\mathbb{P}}$*.*

 ISBN 13: 978-84-613-5510-5

2.1 Stable Models

Our aim in this section is to recall the adaptation given in [10] of the original approach by Gelfond and Lifschitz [4] to the framework of normal residuated logic programs just defined in the section above.

Let us consider a normal residuated logic program $\mathbb{P}$ together with a fuzzy L-interpretation I. To begin with, we will construct a new normal program $\mathbb{P}_I$ by substituting each rule in $\mathbb{P}$ such as

$$\langle p \leftarrow p_1 * \cdots * p_m * \neg p_{m+1} * \cdots * \neg p_n; \quad \vartheta \rangle$$

by the rule[1]

$$\langle p \leftarrow p_1 * \cdots * p_m; \quad \neg I(p_{m+1}) * \cdots * \neg I(p_n) * \vartheta \rangle$$

Notice that the new program $\mathbb{P}_I$ is positive, that is, does not contain any negation; in fact, the construction closely resembles that of a reduct in the classical case, this is why we introduce the following:

Definition 4 *The program $\mathbb{P}_I$ is called the* reduct *of $\mathbb{P}$ wrt the interpretation I.*

As a result of the definition, note that given two fuzzy L-interpretations I and J, then the reducts $\mathbb{P}_I$ and $\mathbb{P}_J$ have the same rules, and might only differ in the values of the weights. By the monotonicity properties of $*$ and $\neg$, we have that if $I \leq J$ then the weight of a rule in $\mathbb{P}_I$ is greater or equal than its weight in $\mathbb{P}_J$.

It is not difficult to prove that every model M of the program $\mathbb{P}$ is a model of the reduct $\mathbb{P}_M$.

Recall that *a fuzzy interpreta*tion can be interpreted as a L-fuzzy subset. Now, as usual, the notion of reduct allows for defining a *stable set* for a program.

Definition 5 *Let $\mathbb{P}$ be a normal residuated logic program and let I be a fuzzy L-interpretation; I is said to be a* stable set *of $\mathbb{P}$ iff I is a minimal model of $\mathbb{P}_I$.*

Theorem 1 *Any stable set of $\mathbb{P}$ is a minimal model of $\mathbb{P}$.*

Thanks to Theorem 1 we know that every stable set is a model, therefore we will be able to use the term *stable model* to refer to a stable set. Obviously, this approach is a conservative extension of the classical approach.

In the following example we use a simple normal logic program with just one rule in order to clarify the definition of stable set (stable model).

Example 1 Consider the program $\langle p \leftarrow \neg q \quad ; \vartheta \rangle$. Given a fuzzy L-interpretation $I \colon \Pi \to L$, the reduct $\mathbb{P}_I$ is the rule (actually, the fact) $\langle p \quad ; \vartheta * \neg I(q) \rangle$ for which the least model is $M(p) = \vartheta * \neg I(q)$, and $M(q) = 0$. As a result, I is a stable model of $\mathbb{P}$ if and only if $I(p) = \vartheta * \neg I(0) = \vartheta * 1 = \vartheta$ and $I(q) = 0$. □

[1] Note the overloaded use of the negation symbol, as a syntactic function in the formulas and as the algebraic negation in the truth-values.

 ISBN 13: 978-84-613-5510-5

3 The Main Result

The existence of stable models can be guaranteed by simply imposing conditions on the underlying residuated lattice:

Theorem 2 *Let $\mathcal{L} \equiv ([0,1], \leq, *, \leftarrow, \neg)$ be a residuated lattice with negation. If $*$ and $\neg$ are continuous operators, then every finite normal program $\mathbb{P}$ defined over $\mathcal{L}$ has at least a stable model.*

Proof: The idea is to apply Brouwer's fix-point theorem. Specifically, we show that the operator assigning each interpretation I the interpretation $\mathcal{R}(I) = \text{lfp}(T_{\mathbb{P}_I})$ is continuous. Note that this operator can be seen as a composition of two operators $\mathcal{F}_1(I) = \mathbb{P}_I$ and $\mathcal{F}_2(\mathbb{P}) = \text{lfp}(T_{\mathbb{P}})$. Actually, we will show that $\mathcal{F}_1$ and $\mathcal{F}_2$ are continuous.

To begin with, note that $\mathcal{F}_1$ can be seen as an operator from the set of $[0,1]$-interpretations to the Euclidean space $[0,1]^k$ where k is the number of rules in $\mathbb{P}$. This is due to the fact that $\mathcal{F}_1$ just changes the weights of $\mathbb{P}$, and nothing else. Now, the continuity of $\mathcal{F}_1$ is trivial since the weight of each rule in $\mathbb{P}$ is changed only by using the continuous operator $\neg$.

Concerning $\mathcal{F}_2$, the syntactic part of $\mathbb{P}$ can be considered fixed and positive. This is due to the fact that its only inputs are of the form $\mathbb{P}_I$, therefore, the number of rules is fixed, negation does not occur in $\mathbb{P}$, and the only elements which can change are the weights. As a result, $\mathcal{F}_2$ can be seen as a function from $[0,1]^k$ to the set of interpretations. Note that this restriction over $\mathcal{F}_2$ does not disallow the composition between $\mathcal{F}_1$ and $\mathcal{F}_2$. To prove that $\mathcal{F}_2$ is continuous note, firstly, that the immediate consequence operator is continuous with respect to the weights in $\mathbb{P}$, since every operator in the definition of $T_{\mathbb{P}}$ (namely sup and $*$) is continuous. Secondly, a direct consequence of the termination result introduced in [2, see Cor. 29] ensures that if $\mathbb{P}$ is a finite positive program, then $\text{lfp}(T_{\mathbb{P}})$ can be obtained by iterating finitely many times the immediate consequence operator; in other words, $\text{lfp}(T_{\mathbb{P}}) = T_{\mathbb{P}}^k(I_\perp)$ where k is the number of rules in $\mathbb{P}$. Therefore, as the operator $\mathcal{F}_2$ is a finite composition of continuous operators, $\mathcal{F}_2$ is also continuous.

Finally, as $\mathcal{R}(I) = \text{lfp}(T_{\mathbb{P}_I})$ is a composition of two continuous operators, $\mathcal{R}(I)$ is continuous as well. Hence we can apply Brouwer's fix-point theorem to $\mathcal{R}(I)$ and ensure that it has at least a fix-point. To conclude, we only have to note that every fix-point of $\mathcal{R}(I)$ is actually a stable model of $\mathbb{P}$. □

Example 2 The existence of stable model for the normal residuated logic program below

$$\langle p \leftarrow \neg q \ ; 0.8 \rangle$$
$$\langle q \leftarrow \neg r \ ; 0.7 \rangle$$
$$\langle r \leftarrow \neg p \ ; 0.9 \rangle$$

 ISBN 13: 978-84-613-5510-5

is not always guaranteed. For example, if we consider the residuated lattice $L = ([0,1], *, \leftarrow, \neg)$ determined by $x * y = x \cdot y$ and

$$\neg(x) = \begin{cases} 0 & \text{if } x > 0.5 \\ 1 & \text{if } x \leq 0.5 \end{cases}$$

then the program has not stable models. However, if we consider the residuated lattice $L = ([0,1], *, \leftarrow, \neg)$ determined by $x * y = x \cdot y$ and $\neg(x) = 1 - x$ the normal residuated logic program has the following stable model

$$M = \{(p, 0.4946808); (q, 0.3816489); (r, 0.4547872)\}$$

Obviously, the sufficient condition provided in Theorem 2 is not a necessary condition. Considering the residuated lattice $L = ([0,1], *, \leftarrow, \neg)$ determined by

$$x * y = \begin{cases} x & \text{if } y = 1 \\ y & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases} \qquad \neg(x) = \begin{cases} 0 & \text{if } x > 0.5 \\ 1 & \text{if } x \leq 0.5 \end{cases}$$

the program above has one stable model, $M = \{(p, 0); (q, 0); (r, 0)\}$; although the connectives $*$ and $\neg$ are not continuous.

Remark 1 It is important to recall that most connectives in fuzzy logic are defined on the unit interval $[0, 1]$. Thus the condition about continuity on a Euclidean space as sets of truth-values is not much restrictive. Moreover, most t-norms used currently in fuzzy logic are continuous (Gödel, Łukasiewicz, product, ...), therefore the theorem establishes that in the most used fuzzy frameworks, the existence of fuzzy stable models is always guaranteed.

4 Related Work

As stated in the introduction, one can find several conditions in the literature which guarantee the existence of stable models. Whereas in logic programming the syntactic characterization of consistent normal program was done in [1], the existence of stable models in fuzzy logic programming is an open problem; and apparently more complicated.

To the best of our knowledge, there are only other two sufficient conditions in fuzzy logic programming to guarantee the existence of stable model. The first one is given in the fuzzy description logic paradigm, and can be found in [9]. It is done at the syntactic dimension and extends a result already known in logic programming [6].

Definition 6 *A normal residuated logic program $\mathbb{P}$ is called* locally stratified *if there is a level function $||\cdot||$ such that for every rule $\langle p \leftarrow p_1 * \cdots * p_k * \neg p_{k+1} * \cdots * \neg p_n; \vartheta \rangle$ of $\mathbb{P}$:*

- $||p|| \geq ||p_i||$ *for all* $i \in \{1, \ldots, k\}$

 ISBN 13: 978-84-613-5510-5

- $||p|| > ||p_i||$ *for all* $i \in \{k+1, \ldots, n\}$

Proposition 1 *A stratified normal residuated logic program has one, and only one, stable model.*

The other result appears in [8], and is due to the use of *bilattices* as the set of truth-values. Briefly, a bilattice is a tuple $(L, \leq_t, \leq_k)$ where $(L, \leq_t)$ and $(L, \leq_k)$ form two complete lattices. Such a structure is used in [7] in order to define the well-founded semantics in fuzzy logic programming through the least stable model under the ordering $\leq_k$; i.e by generalizing a result provided in [3] for the classical case which relates the well-founded semantics and stable model semantics.

Proposition 2 *Let* $\mathbb{P}$ *be a normal logic program defined over the residuated lattice* $(L, \leq, *, \leftarrow, \neg)$. *If there is an ordering* $\leq_k$ *such that:*

- $(L, \leq, \leq_k)$ *is a bilattice*
- $*$ *and* $\neg$ *are monotonic w.r.t.* $\leq_k$

then, there exists at least one stable model of $\mathbb{P}$.

The key-point of proposition 2 is to find an ordering over L such that $*$ and $\neg$ are monotonic with respect it, thus we can to asure the existence of stable model.

5 Future Work

The result of Theorem 2 has interesting potential applications. To begin with, we can avoid the inconsistency in fuzzy logic programs by using continuous connectives. Moreover, the result is useful to resolve inconsistencies of normal programs defined on a linear lattice by extending them over $[0, 1]$ with continuous connectives. For example, consider the following normal program in classical logic programming:

$$r_1 : p \leftarrow \neg q, s \qquad r_2 : r \leftarrow \neg t, \neg p$$
$$r_3 : q \leftarrow \neg r \qquad r_5 : s \leftarrow$$
$$r_5 : u \leftarrow \neg t, s \qquad r_6 : v \leftarrow \neg v, \neg r$$

where "," denotes the classical conjunction. Clearly the program is inconsistent but we can assign a fuzzy stable model semantics by embedding the program into a residuated lattice $([0, 1], *, \leftarrow, \neg)$. For example, consider the connectives $x * y = x \cdot y$ and $\neg(x) = 1 - x$, then the program above (substituting "," by "$*$" and including the weight 1 in each rule) has the following stable model:

$$M = \{(p, 0.5); (q, 0.5); (r, 0.5); (s, 1); (t, 0); (u, 1); (v, 1/3)\}$$

Notice that if we collapse each $x \in (0, 1)$ to one undefined value (that is, M assigns to p, q, r and v the same truth-value "undefined") the semantics is equivalent to the well-founded semantics. Notice, however, that the residuated semantics is slightly more expressive, due to its ability to assign any value in the unit interval.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work has been partially supported by Junta de Andalucía grant P09-FQM-5233, and by the EU (FEDER), and the Spanish Science Ministry under grant TIN2009-14562-C05-01.

References

[1] S. Costantini. On the existence of stable models of non-stratified logic programs. *Journal of Theory and Practice of Logic Programming*, 6(1-2):169–212, 2006.

[2] C. Damásio, J. Medina, and M. Ojeda-Aciego. Termination of logic programs with imperfect information: applications and query procedure. *Journal of Applied Logic*, 5(3):435–458, 2007.

[3] M. Fitting. The family of stable models. *The Journal of Logic Programming*, 17(2-4):197 – 225, 1993.

[4] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proc. of ICLP-88*, pages 1070–1080, 1988.

[5] M. L. Ginsberg. Multivalued logics: a uniform approach to reasoning in artificial intelligence. *Computational Intelligence*, 4:265–316, 1988.

[6] J. Lloyd. *Foundations of Logic Programming*. Springer Verlag, 1987.

[7] Y. Loyer and U. Straccia. The well-founded semantics in normal logic programs with uncertainty. *Lect. Notes in Computer Science*, 2441:152–166, 2002.

[8] Y. Loyer and U. Straccia. Epistemic foundation of stable model semantics. *Journal of Theory and Practice of Logic Programming*, 6:355–393, 2006.

[9] T. Lukasiewicz. Fuzzy description logic programs under the answer set semantics for the semantic web. *Fundamenta Informaticae*, 82(3):289–310, 2008.

[10] N. Madrid and M. Ojeda-Aciego. Towards a fuzzy answer set semantics for residuated logic programs. In *Proc of WI-IAT'08. Workshop on Fuzzy Logic in the Web*, pages 260–264, 2008.

[11] N. Madrid and M. Ojeda-Aciego. On coherence and consistence in fuzzy answer set semantics for residuated logic programs. *Lect. Notes in Computer Science*, 5571:60–67, 2009.

[12] T. Przymusinski. Well-founded semantics coincides with three-valued stable semantics. *Fundamenta Informaticae*, 13:445–463, 1990.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

FEM approximation of the stochastic Stokes problem involving multiplicative white noise

Hassan Manouzi[1]

[1] *Department of mathematics and statistic, Laval University, Quebec, Canada*

emails: hm@mat.ulaval.ca

Abstract

We propose a finite element method for the numerical solution of the stochastic Stokes equations driven by a multiplicative white noise. We give existence and uniqueness results for the continuous problem and its approximation. Optimal error estimates are derived and algorithmic aspects of the method are discussed. Our method will reduce the problem of solving stochastic Stokes equations to solving a set of deterministic ones. Moreover, one can reconstruct particular realizations of the solution directly from Wiener chaos expansions once the coefficients are available.

1 Introduction

In the recent years, there has been an increased interest in applications of stochastic partial differential equations. Stochastic partial differential equation (SPDEs) can essentially be viewed as PDEs perturbed by some stochastic noise. For a given physical system, many different stochastic perturbations may be considered, which may include inexact knowledge of systems forcing, initial and boundary conditions, parametric uncertainties in the physical model and in physical properties of the medium. Moreover, internal randomness often reflects itself in additive noise terms, while external fluctuations give rise to multiplicative noise terms.

The mathematical treatment of SPDEs is more involved than deterministic PDEs. An early systematic introduction to SPDEs was given by Walsh. He considered a linear SPDE with additive white noise. He showed that for spatial-dimension greater than one, it is in general not possible to represent the solution as an ordinary stochastic field, but only as a distribution. Similar behavior can be observed in many other examples. This fact has motivated the introduction of weaker solution concepts. In the literature, various stochastic Galerkin methods have been applied to various linear and nonlinear problems.

 ISBN 13: 978-84-613-5510-5

Of particular interest to us is the approach of Walsh which treats the solution $u(x,\omega)$ of an SPDE as a distribution, i.e., the map $x \longrightarrow u(x,\omega)$ is a distribution for a.a. ω. Walsh considered the solution as a stochastic variable with values in a Sobolev-type distribution space. This approach is well suited for problems where the noise is additive; it has been explored extensively in the literature.

Walsh's approach does not handle problems where the noise appears multiplicative. Although the construction of Walsh supplies a useful tool for the study of linear SPDEs, its applicability to nonlinear equations is limited. This is due to the difficulty of defining nonlinear operations on distributions. To analyze SPDEs with a multiplicative noise or nonlinear SPDEs we consider SPDEs involving Wick products. This approach comes from white noise analysis where generalized solutions say $u(x,\omega)$ are treated in the sense that $\omega \longrightarrow u(x,\omega)$ is a stochastic distribution for a.a. x. An advantage of this approach is that one can establish a theory of nonlinear operations on distributions in order to handle a wide class of nonlinear SPDEs by using the Wick product $\diamond$. This product can be viewed as a regularization of the ordinary point-wise product, and it furnishes an interpretation of nonlinearities and multiplicative noises. In this framework, and by interpreting all products as Wick products, we obtain a well-defined solution concept for a range of different problems, both linear and nonlinear. Furthermore, it provides a nice structure making the equations tractable by Wick-calculus techniques.

In this paper we consider the case of the Stochastic Stokes equations driven by a multiplicative white noise. In particular we shall study the boundary value problem

$$\begin{aligned} -\nabla\cdot(\kappa(x,\omega)\diamond\nabla u)+\nabla p+\lambda u\diamond W(x,\omega) &= f(x,\omega), \qquad \text{in } \mathcal{D}\times\Omega, &(1)\\ \nabla.u(x,\omega) &= 0, \qquad \text{in } \mathcal{D}\times\Omega, &(2)\\ u &= 0, \qquad \text{on } \mathcal{D}\times\Omega, &(3) \end{aligned}$$

where $\mathcal{D}$ is a bounded open subset of $\mathbb{R}^d$ with Lipschitz continuous boundary $\partial\mathcal{D}$ and $\Omega = \mathcal{S}'(\mathbb{R})$ is the white noise probability space. The variables u, p, f denote respectively the velocity, the pressure and the external forces, $W(x,\omega)$ is a space white noise , where κ is the stochastic viscosity.

For technical reasons, since the noise appears multiplicatively, it is necessary to regularize the product between the viscosity κ and the gradient deformation ∇u. This regularization is achieved by replacing the ordinary product with the Wick product, and leads to (1).

We shall reformulate this stochastic variational problem as an infinite set of deterministic variational problems, using the properties of the Wick product. Each of these variational problems will give one of the coefficients in the Wiener-Itô chaos expansion of the solution of (1). The method we shall use is based on the ideas of Fourier analysis on Wiener space. In fact, Wiener Chaos expansion represents a stochastic function $u(x,\omega)$ as a Fourier series with respect to an orthonormal basis $\mathcal{H}_\alpha$, i.e.,

$$u(x,\omega)=\sum_{\alpha\in\mathcal{I}} u_\alpha(x)\mathcal{H}_\alpha(\omega)$$

 ISBN 13: 978-84-613-5510-5

where $\mathcal{I}$ denotes the set of multi-indices $\alpha = (\alpha_1, \alpha_2, \cdots)$ where all $\alpha_i \in \mathbb{N}$ and only finitely many $\alpha_i \neq 0$, u_α's are deterministic coefficients and $\mathcal{H}_\alpha$'s are the stochastic variables

$$\mathcal{H}_\alpha(\omega) = \Pi_{j=1}^{\infty} h_{\alpha_j}(< \omega, \eta_j >) \ , \ \omega \in \mathcal{S}'(\mathbb{R})$$

where h_n denotes the Hermite polynomial and the family $\{\eta\}_{j=1}^{\infty}$ forms an orthonormal basis for $L^2(\mathbb{R}^d)$. This decomposition separates the deterministic effects (described by the coefficients u_α) from the randomness (that is covered by the base $\mathcal{H}_\alpha$). The orthogonality of $\mathcal{H}_\alpha$ and the properties of the Wick product enable us to reduce SPDEs like (1) to a system of coupled deterministic equations for the chaos coefficients $u_\alpha(x)$. The propagator is a deterministic mechanism responsible for the evolution of randomness inherent to the original SPDE. Once the propagator is obtained, standard deterministic numerical methods can be applied to solve it sufficiently. The main statistics, such as mean, covariance and higher order statistical moments can be calculated by simple formulas involving only these deterministic coefficients. Moreover, in the procedure described above, there is no randomness directly involved in the simulations. One does not have to deal with the selection of random number generators, and there is no need to solve the SPDE equations realization by realization. Instead, coupled coefficient equations are solved once and for all. Moreover, one can reconstruct particular realizations of the solution directly from Wiener chaos expansions once the coefficients are available.

The idea to transform a SPDE into a hierarchy of deterministic PDEs for the Wiener-Itô chaos decomposition seems promising. It is thus important to devote some effort to propose concrete schemes based on that idea and to analyze them.

An outline of the paper will be as follows. In Section 1 we review notation and introduce some functional spaces. In Section 2 we first formulate the problem in a weak sense over stochastic Hilbert spaces, and prove existence of a unique solution to this formulation under suitable assumptions on the data. In Section 3, using a mixed finite element approach, we construct approximations of the solution and discuss convergence properties for these approximations. We establish estimates for the rate of convergence in both the spatial and stochastic dimension. Finally in Section 4 we discuss algorithmic aspects of this numerical method. In particular, we show how approximations of the solution can be constructed through the solution of a sequence of deterministic mixed variational problems, each giving an approximation to a chaos coefficient of the solution.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Improving the Scheduling of Parallel Applications using Accurate AIC-based Performance Models

D. R. Martínez[1], J. L. Albín[1], V. Blanco[2], T. F. Pena[1], J. C. Cabaleiro[1] and F. F. Rivera[1]

[1] *Dept. Electronics and Computing, University of Santiago de Compostela*

[2] *Dept. Statistics and Computer Science, La Laguna University*

emails: diego.rodriguez@usc.es, xulio.lopez@usc.es, Vicente.Blanco@ull.es, tf.pena@usc.es, jc.cabaleiro@usc.es, ff.rivera@usc.es

Abstract

Predictions based on analytical performance models can be used on efficient scheduling policies in order to select the adequate resources for an optimal execution in terms of throughput and response time. However, it is a hard issue developing accurate analytical models of parallel applications. In this paper, an accurate performance model of the HPL benchmark is obtained in a easy way by means of AIC-based model selection methods provided by the TIA framework. The performance of backfilling policy algorithms on schedulers using this AIC-based model is analyzed in the GridSim simulator and compared with the results obtained using the theoretical analytical model provided by the authors of the benchmark.

Key words: Performance analytical models, model selection, scheduling

1 Introduction

Despite hardware developers efforts, the efficient exploitation of parallel systems implies the understanding of the behavior of parallel applications. Therefore, performance modeling of parallel applications becomes a crucial issue in High Performance Computing, and different modeling approaches have been proposed in the literature. Although they are less accurate than other modeling methods, analytical models have the advantage of being able to evaluate the model in less time. This feature is essential in certain time-limited problems such as scheduling or dynamic load balancing. In most cases, the development of analytical models requires a hard effort as well as a deep knowledge of the parallel algorithm. Anyway, an exact analytical model is so complex that affordable analytical models has to trade-off accuracy for simplicity.

The TIA modeling framework [4] provides an environment to obtain an accurate analytical model of parallel applications in an easy way by means of model selection

 ISBN 13: 978-84-613-5510-5

methods. This environment is particularly useful when a deep knowledge about the parallel application is not available. In addition, it is also useful for expert analysts because the model produced by the framework actually reflects the experimental behavior of the application and, therefore, it allows a deep analysis of the theoretical analytical models. This paper shows how these accurate analytical models can improve the behavior of scheduling strategies using accurate runtime predictions.

2 The Modeling Framework

The TIA (Tools for Instrumentation and Analysis) modeling framework [4] provides the user with a simple but powerful environment to analyze the performance of parallel applications. It consists of two main connected stages. The first stage (*instrumentation*) implements the user-driven instrumentation of the source code, being the information about the performance in each execution of the application stored in XML files. In the second stage (*analysis*), an analytical model is calculated by analyzing the performance data obtained from multiple executions of the instrumented code. This stage is based on R, a language and environment for statistical analysis that provides specific functions to deal with model selection and AIC.

Model selection seeks for models that are good approximations to the truth and from which valid inferences about the system or process under study can be made. This search is based on analyzing data to aid in the selection of a parsimonious model. Parsimony is usually visualized as a suitable trade-off between squared bias and variance of parameter estimators. In [2], Burnham and Anderson propose a general strategy for modeling and data analysis using information theory and, in particular, the Akaike's information criterion (*An Information Criterion*, AIC). AIC provides a simple, effective, and objective means for the selection of an estimated *best approximating model* for data analysis and inference. A model selection method based on the AIC has been implemented in the analysis stage of the TIA framework [4]. This method performs an AIC selection using a finite set of candidate models which are generated from information provided by the user. The procedure automatically proposes the model with the lowest AIC score, but also provides some statistical information to help the user to decide the suitability of the model.

3 Scheduling of Parallel Applications

Production clusters usually use batch queuing systems to maximize their resources, so having accurate information about the runtime of the applications can improve the performance of scheduling strategies. In particular, backfilling algorithms are common on sites with parallel jobs because those algorithms improve starvation cases without significant changes in the priorities of the jobs [5]. In the backfilling policy, the arrival order is used to schedule the jobs. Nevertheless, if some local nodes are empty because the next work in the queue is not suitable, the queue is examined to find another job to be executed. The selection of this job should not delay the execution of the rest of

 ISBN 13: 978-84-613-5510-5

the queued jobs. EASYBackfilling is an aggressive backfilling in which only the first element of the queue is considered. This means that a job is promoted if it does not delay the execution start of the next one in the queue without taking into account the rest of entries in the queue. Anyway, precise runtime estimations must be provided to these policies for an efficient exploitation of the cluster resources.

In this work the behavior of backfilling schedulers were simulated using GridSim [1], a Java-based discrete event Grid simulation toolkit. This toolkit supports modeling and simulation of heterogeneous Grid resources (both time- and space-shared), users and application models. It provides primitives to create of application tasks, mapping of tasks to resources, and their management.

4 Case of Study

High Performance Linpack (HPL) [3] uses a LU factorization with row partial pivoting to solve a dense linear system while mapping a two-dimensional block-cyclic data distribution for load balance and scalability. Using the TIA framework, an analytical model of the performance of HPL is calculated from multiple executions of HPL in an cluster of seven Intel Xeon Quadcore biprocessors. In particular, this model characterizes the behavior of HPL, with the modified increasing two-ring broadcast option, for different matrix sizes, block sizes and process grid configurations. The fit of this model to the experimental data have been compared with the fit of the theoretical model provided by the developers of the HPL benchmark [3], and we have found that the standard deviation of the AIC model fit is roughly half of the standard deviation of the theoretical model fit.

The simulation results show that the models automatically obtained by our approach provide an accurate input for the schedulers. In fact, these models improve the behavior of the schedulers when they are compared with the theoretical models. Figure 1 shows an example of these results, using a queue of 1000 HPL executions with different parameter configurations, and for predictions based on the estimated time of both the AIC model (T_{AIC}) and the theoretical model (T_{Theo}). These predictions are overestimated with a multiple of the standard deviation (σ) of each model. For each case, the simulated CPU time needed to finish all the jobs ($T_{finished}$), the CPU time consumed by jobs that were canceled by the scheduler ($T_{canceled}$), and the idle time throughout the simulation (T_{idle}) are shown. The AIC-based predictions are more efficient than the theoretical-based ones because of their more precise estimations that reduce the number of canceled jobs. Note that, even without detailed information about the codes, our automatic models outperform the models provided by the developers of the benchmark, and, as a consecuence, they can be efficiently used by the schedulers.

Acknowledgements

This work was supported by the Spanish Ministry of Education and Science through TIN2007-67537-C03-01 and TIN2008-06570-C04-03 projects and through the FPI pro-

 ISBN 13: 978-84-613-5510-5

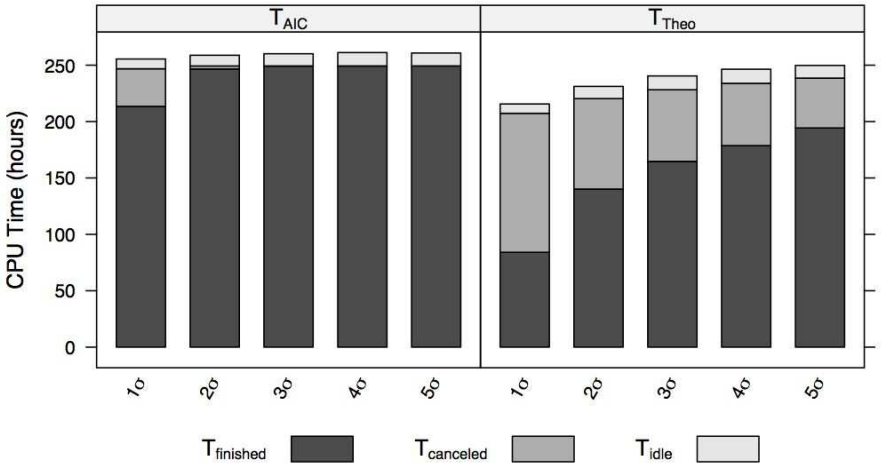


Figure 1: Simulation results of 1000 HPL executions

gram. It has been developed in the framework of the European network HiPEAC-2 and the Spanish network CAPAP-H.

References

[1] J. L. Albín, J. A. Lorenzo, J. C. Cabaleiro, T. F. Pena, and F. F. Rivera, *Simulation of parallel applications in GridSim*, In 1st Iberian Grid Infrastructure Conference Proceedings, Santiago de Compostela (Spain), 2007.

[2] Kenneth P. Burnham and David R. Anderson, *Model Selection and Multimodel Inference. A Practical Information-Theoretic Approach*, Spring Science + Bussiness Media, LLC, New York, 2002.

[3] Jack J. Dongarra, Piotr Luszczek, and Antoine Petitet, *The LINPACK Benchmark: past, present and future*, Concurrency and Computation: Practice and Experience, **15** (2003) 803–820.

[4] Diego R. Martínez, Tomás F. Pena, José C. Cabaleiro, Francisco F. Rivera, and V. Blanco, *Performance modeling of MPI applications using model selection*, In 18th Euromicro Conference on Parallel, Distributed and Network-based Processing, 2010.

[5] Dan Tsafrir, Yoav Etsion, and Dror G. Feitelson, *Backfilling using system-generated predictions rather than user runtime estimates*, IEEE Transactions on Parallel and Distributed Systems, **18** (2007) 789–803.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A parallel algorithm for LDPC decoding on GPUs

F.J. Martínez-Zaldívar[1], A.M. Vidal-Maciá[2], A. González-Salvador[1] and V. Almenar-Terre[1]

[1] *Departamento de Comunicaciones, Universidad Politécnica de Valencia (Spain)*

[2] *Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia (Spain)*

emails: fjmartin@dcom.upv.es, avidal@dsic.upv.es, agonzal@dcom.upv.es, valmenar@dcom.upv.es

Abstract

In this paper we describe a parallel algorithm for LDPC (*Low Density Parity Check* codes) decoding on a GPU (*Graphics Processing Unit*) using CUDA (*Compute Unified Device Architecture*). The strategy of the kernel grid and block design is shown and the multiword decoding solution is described using tridimiensional blocks.

Key words: LDPC, GPU, CUDA, Sum-Product Algorithm, Parallel Algorithm

1 Introduction

Low-Density Parity-Check codes (LDPC codes) are linear block channel codes for error control coding with a sparse parity-check matrix (a matrix that contains few 1's in comparison to the amount of 0's). They have recently been adopted by several data communication standards such as DVB-S2 [1], 10GBase-T [2], WiMax (or IEEE802.16e [3]), etc.

The concept of LDPC coding was firsrt developed by Robert G. Gallager in his doctoral dissertation at MIT in the begining of sixties [4] but forgotten due to its impractical implementation at that moment and the introduction of the Reed-Solomon codes. They were rediscovered by MacKay and Neal in 1996 [5]. These codes provide a performance very close to the Shannon capacity limit of the channel [11], low error floor, and linear time complexity for decoding (lower than turbocodes [12]). LDPC codes are inherently suited for parallel hardware and software implementations, but with some implementation difficulties when an optimum algorithmic behavior is desired on a GPU, as we can read in [6, 7, 8, 9] using CUDA and other GPU programming tools.

 ISBN 13: 978-84-613-5510-5

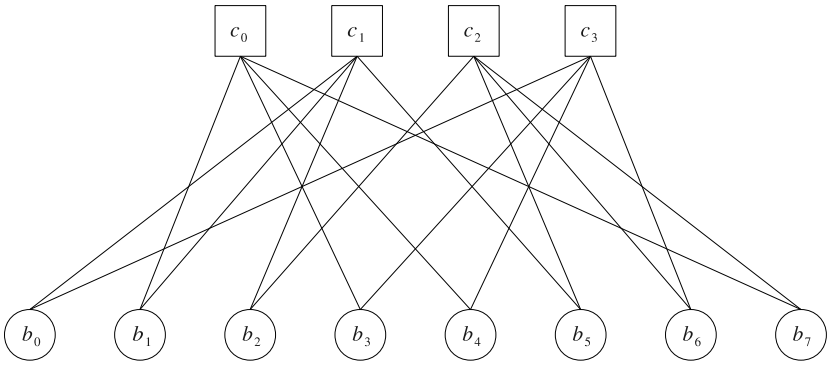


Figure 1: Tanner graph of the linear block code parity-check matrix in (1).

2 A LDPC decoding algorithm

LDPC codes can be represented graphically by a Tanner graph [10] (a bipartite graph with variable or bit nodes, b_i, and check nodes, c_j) . An example is shown in Figure 1, that corresponds with the next parity-check matrix $\mathbf{H} \in \mathbb{B}^{m \times n}$, where $\mathbb{B}$ denotes the field of the binary digits:

$$\mathbf{H} = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \tag{1}$$

The length of a codeword and the number of the parity bits match the number of columns and rows of the parity-check matrix respectively.

Let Φ_j denote the set of indices of the bit nodes connected to the check node c_j, and $\Phi_j^{\sim i}$, the same set of indices excluding i. Let Ω_i denote the set of indices of the check nodes connected to the bit node b_i, and $\Omega_i^{\sim j}$, the same set, excluding j.

LDPC decoders are based on variations of belief propagation, sum-product or message passing algorithms. In any of these algorithmic denominations, information flows to/from bit nodes and from/to check nodes until the algorithm converges to a stable state, finding the most likelihood transmitted codeword. There are many variations but we will focus our attention in a simple sum-product algorithmic version:

- Let $\mathbf{x} \in \mathbb{B}^n$ denote the transmitted n-dimensional codeword and $\mathbf{y} \in \mathbb{B}^n$ the received n-dimensional word.
- Let us suppose that the channel can be modeled as a BSC (Binary Symmetric Channel) with a bit error probability p and that the input bits are equiprobable (for other channels the formulation is analogous). The likelihoods are:

$$\mathrm{P}(y_i = y | x_i = x) = \begin{cases} 1-p & x, y \in \mathbb{B}, x = y \\ p & x, y \in \mathbb{B}, x \neq y \end{cases}, \forall i = 0, \ldots, n-1$$

 ISBN 13: 978-84-613-5510-5

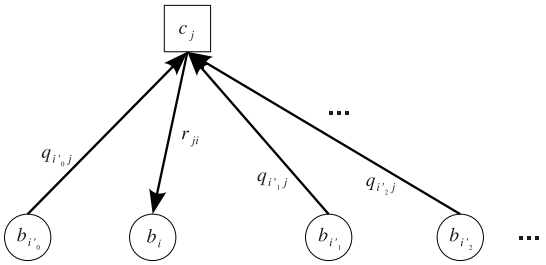


Figure 2: Computation in the check nodes

and

$$\mathrm{P}(x_i = 0) = \mathrm{P}(x_i = 1) = 1/2,\ \forall i = 0, \ldots, n-1$$

- Let P_i be the *likelihood* that the i^{th} bit of the transmitted codeword is 1:

$$P_i = \mathrm{P}(y_i = y | x_i = 1),\ \forall i = 0, \ldots, n-1.$$

  Hence, if the received bit is 1 ($y_i = 1$) then $P_i = 1 - p$; otherwise $P_i = p$.

- Let $q_{ij} \equiv (q_{ij}(0), q_{ij}(1))$ be the message sent by the bit node b_i to the check node c_j, denoting the belief that the bit node has about the transmitted bit (the belief of being 0 and 1, respectively).

- Let $r_{ji} \equiv (r_{ji}(0), r_{ji}(1))$ be the message sent by the check node c_j to the bit node b_i, denoting the belief that the check node has about the transmitted bit.

The steps of the algorithm are:

1. Each check node $c_j, \forall j = 0, \ldots m-1$, computes $r_{ji}, \forall i \in \Phi_j$ as:

$$r_{ji}(0) = \frac{1}{2} + \frac{1}{2} \prod_{i' \in \Phi_j^{\sim i}} \left(1 - 2q_{i'j}(1)\right) \tag{2}$$

$$r_{ji}(1) = 1 - r_{ji}(0) \tag{3}$$

   and sends it to each bit node b_i connected to the check node c_j (see Figure 2). For the first iteration, $q_{ij}(0) = 1 - P_i$, and $q_{ij}(1) = P_i$.

2. Each bit node $b_i, \forall i = 0, \ldots n-1$, computes $q_{ij}, \forall j \in \Omega_i$ as:

$$q_{ij}(0) = K_{ij}(1 - P_i) \prod_{j' \in \Omega_i^{\sim j}} r_{j'i}(0) \tag{4}$$

$$q_{ij}(1) = K_{ij} P_i \prod_{j' \in \Omega_i^{\sim j}} r_{j'i}(1) \tag{5}$$

 ISBN 13: 978-84-613-5510-5

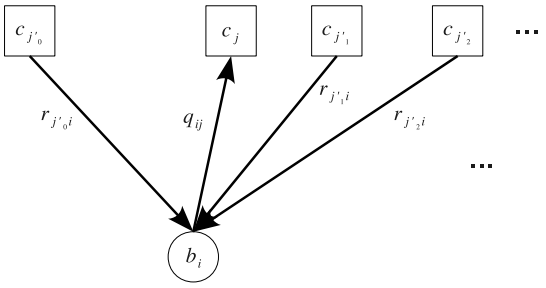


Figure 3: Computation in the bit nodes

and sends it to each check node c_j connected to the bit node b_i (see Figure 3). K_{ij} is chosen to ensure that $q_{ij}(0) + q_{ij}(1) = 1$. Now, bit nodes update their estimation of the transmitted codeword bits $\hat{x}_i$ computing:

$$Q_i(0) = K_i(1 - P_i) \prod_{j \in \Omega_i} r_{ji}(0) \tag{6}$$

$$Q_i(1) = K_i P_i \prod_{j \in \Omega_i} r_{ji}(1) \tag{7}$$

similarly K_i is chosen to ensure that $Q_i(0) + Q_i(1) = 1$. If $Q_i(0) > Q_i(1)$ the estimated transmitted bit $\hat{x}_i$ is 0, otherwise it is 1.

Clearly, (6) and (7) can be computed easily from (4) and (5), and it is not necessary to store (4) and (3) for the next iteration.

If the current estimated transmitted word is a codeword (this can be verified checking that $\mathbf{H}\hat{\mathbf{x}} = \mathbf{0}$) or the maximum number of iterations is reached, the algorithm ends, otherwise another iteration starts going to step 1.

It can be observed that the computations within the check nodes and the bit nodes are alternated and interdependent in time, so they must be executed one after another because of their inherent dependence. The computations in each check node are mutually independent, so they are perfectly parallelizable; the same happens with the variable nodes computations. Within a check node, a different result must be computed and sent to each bit node that is connected to it. Something similar is observed in the bit node computations. Figure 4 shows the dependency graph of the parallel algorithm.

3 CUDA algorithm

The parity check matrix is sparse in LDPC codes. The LDPC code is regular if the number of ones in every column w_c and the number of ones in every row w_r are constant and $w_c/m = w_r/n$; otherwise the code is irregular. Anyway, w_c and w_r are usually small integers with $w_c \ll m$ and $w_r \ll n$. In general, irregular LDPC codes have better

 ISBN 13: 978-84-613-5510-5

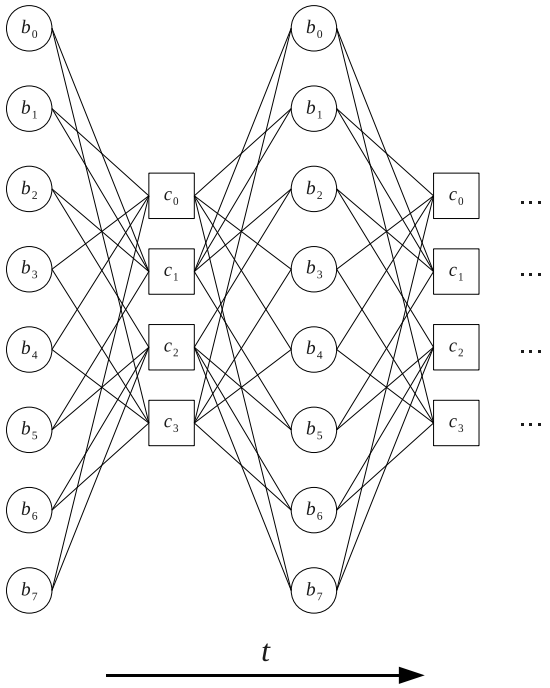


Figure 4: Computation and message passing in the parallel algorithm.

performance than regular ones. With the proposed data structure, the code can be regular or irregular, but if it is irregular the storage performance is worse.

The matrix $\mathbf{H}$ will be implicitly represented in the GPU memory with the matrices $\mathbf{H}_{\text{ind}_{r_0}} \in \mathbb{Z}^{m \times w_r}$ and $\mathbf{H}_{\text{ind}_{q_1}} \in \mathbb{Z}^{w_c \times n}$ that store the indices where we must store the results of the computations. Additionally, matrices $\mathbf{H}_{q_1} \in \mathbb{R}^{m \times w_r}$ and $\mathbf{H}_{r_0} \in \mathbb{R}^{w_c \times n}$ store the q_1 and r_0 computed values respectively. In the case of irregular codes, w_r and w_c denote the maximum number of ones in any row and any column respectively of $\mathbf{H}$.

3.1 CUDA kernels, grids, blocks and threads

Due to the different computing pattern of the steps 1 and 2 of the LDPC decoding algorithm, it is convenient to separate the iteration into two different kernels, A and B, with different thread distributions. Every $r_{ji}(0)$ or $q_{ij}(1)$ value will be computed by one thread.

In the A kernel, the grid is an unidimensional grid with only one column of blocks. The dimension (rows $\times$ columns) of every block is $(\texttt{MAX\_THREADS}/w_r \times w_r)$, and there will be $m/(\texttt{MAX\_THREADS}/w_r)$ blocks. The gray subarea of the grid denotes the active threads in the block and matches the nonzero element pattern of $\mathbf{H}_{q_1}$. If the code would be regular the whole grid would be gray. Here, `MAX_THREADS` is the maximum number of threads that can be executed per block with enough resources. Figure 5 shows these blocks.

In the B kernel, the grid has only one row of blocks. The dimension of every

 ISBN 13: 978-84-613-5510-5

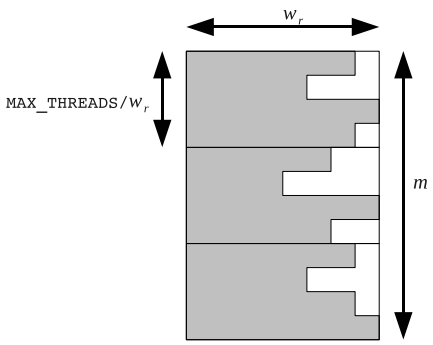


Figure 5: Blocks in the grid of kernel A.

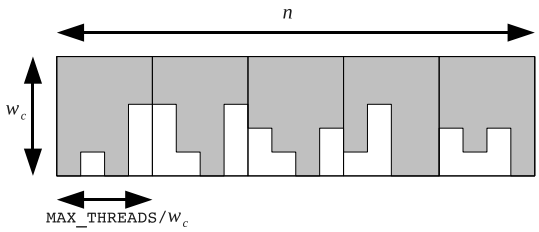


Figure 6: Blocks in the grid of kernel B.

block is ($w_c \times$MAX_THREADS$/w_c$). Figure 6 shows this grid. The gray subarea of the grid denotes the active threads in the block and matches the nonzero element pattern of $\mathbf{H}_{r_0}$. If the code would be regular the whole grid would be gray.

Any block of the A kernel needs to read certain submatrix of $\mathbf{H}_{q_1}$ with the same dimension as the block. Every row of this submatrix is read w_r times by the algorithm, so it is convenient to read only once and store it in the shared memory in order to minimize the memory latency time [13]. The required size to store this submatrix is MAX_THREADS elements (simple precision floating point numbers in our case). The access to $\mathbf{H}_{q_1}$ in global memory can be coalesced [13]. The pattern of the resulting $r_{ji}(0)$ values to write in global memory (in $\mathbf{H}_{r_0}$) is not regular so it cannot be coalesced.

Similarly, any block of the B kernel need to read certain submatrix of $\mathbf{H}_{r_0}$ with the same dimension as the block. Every column of this submatrix is read w_c times, so again it is convenient to read once and store it in the shared memory (MAX_THREADS elements). Again, the access to $\mathbf{H}_{r_0}$ can be coalesced. The pattern of the resulting $q_{ij}(1)$ values to write in $\mathbf{H}_{q_1}$ is not regular so it cannot be coalesced, as in kernel A.

With this framework it is relatively easy to solve a multiword decoding scheme, i.e., to decode several received words simultaneously. We need just to make up tridimensional blocks instead of bidimiensional blocks. The third dimension is the number of received words, n_w that are being decoded. Now, the dimension of the blocks for the A kernel would be (MAX_THREADS$/w_r/n_w \times w_r \times n_w$) and for the B kernel, ($w_c \times$MAX_THREADS$/w_c/n_w \times n_w$). This can be observed in Figures 7 and 8.

 ISBN 13: 978-84-613-5510-5

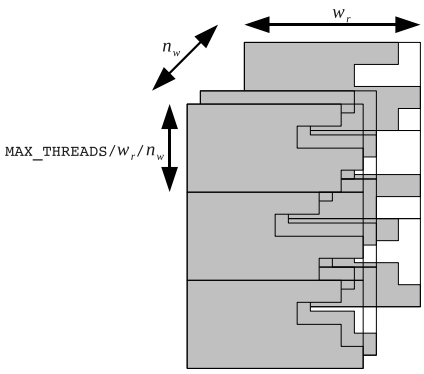


Figure 7: Tridimensional blocks in the grid of kernel A with multiword decoding.

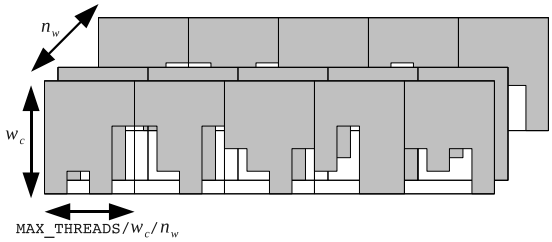


Figure 8: Tridimensional blocks in the grid of kernel B with multiword decoding.

4 Experimental results and conclusions

We have tested the proposed parallel algorithm in the next computing platform:

- CPU: two quad-core Intel Xeon E5530 @ 2.4 GHz and 48 GB of RAM
- GPU: NVidia Tesla C1060 with 4 GB of DDR3 RAM and 240 streaming processor cores @ 1.3 GHz with PCIe 2.0 x16 interface.

The CPU/GPU software was compiled with CUDA 3.0 `nvcc` with `-O3` compiler switch.

Table 1 summarizes the speedup of the parallel algorithm for several LDPC code dimensions m and n, using a column weight of $w_c = 3$ and decoding $n_w = 1$ received words simultaneously.

We can observe how the speedup increases with the dimensions of the LDPC code because the workload becomes more and more intensive. This is advantageous for both systems but the performance is higher for the GPU. It is expected that the effciency increases with an increase of the column weight and the number of simultaneously decoded words upto a limit related to the maximum amount of resources that can be available for a block.

 ISBN 13: 978-84-613-5510-5

m	500	1000	2000	4000
n	1000	2000	4000	8000
w_c	3	3	3	3
n_w	1	1	1	1
Speedup	0.56	1.1	1.96	2.2

Table 1: Speedup of the GPU execution respect to the CPU execution

Acknowledgements

This work was financially supported by the Spanish Ministerio de Ciencia e Innovación (Projects TIN2008-06570-C04-02 and TEC2009-13741), Universidad Politécnica de Valencia through "Programa de Apoyo a la Investigación y Desarrollo (PAID-05-09)" and Generalitat Valenciana through project PROMETEO/2009/013.

References

[1] ETSI, *Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications.*. Available at http://www.dvb.org

[2] IEEE, *P802.3an (10GBASE-T) Task Force*, Available at http://www.ieee802.org/3

[3] IEEE, *P802.16e. Air interface for fixed and mobile broadband wireless access systems*, Available at http://www.ieee802.org/16

[4] R.G. Gallager, *Low density parity check codes*, Ph.D. diss., Massachusetts Institute of Technology, 1963. Available at http://www.ldpc-codes.com/papers/Robert_Gallager_LDPC_1963.pdf

[5] D.J.C. Mackay and R.M. Neal, *Near Shannon limit performance of low density parity check codes*, Electron. Lett., vol. 32, no. 18, pp. 1645–1646, 1996.

[6] G. Falcão, L. Sousa, and V. Silva *Massive parallel LDPC decoding on GPU*, Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, Salt Lake City, Ut (USA), February 20 - 23, 2008.

[7] G. Falcão, V. Silva, and L. Sousa, *How GPUs can outperform ASICs for fast LDPC decoding*, Proceedings of the 23rd International Conference on Supercomputing, Yorktown Heights, NY (USA), 2009.

[8] G. Falcão, S. Yamagiwa, V. Silva and L. Sousa, *Parallel LDPC decoding on GPUs using a stream-based computing approach*, Journal of Computer Science and Technology, Vol. 24, Issue 5 (September 2009), pag.: 913–924, 2009.

 ISBN 13: 978-84-613-5510-5

[9] S. Wang, S. Cheng and Q. Wu, *A parallel decoding algorithm of LDPC codes using CUDA*, in Proc. Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, October 2008.

[10] R. Tanner, *A recursive approach to low complexity codes*, IEEE Transactions on Information Theory, vol. 27, no. 5, pp: 533-547, 1981.

[11] C. Shannon, *A mathematical theory of communication*, Bell System Technical Journal, vol. 27, pp. 379–423 and 623–656, July and October 1948.

[12] C. Berrou, A. Glavieux and P. Thitimajshima, *Near Shannon limit error-correcting coding and decoding: Turbo-codes*, in International Conference on Communications, Geneva, 1993

[13] D.B. Kirk and W.W. Hwu, *Programming Massively Parallel Processors. A hands on approach*, NVidia, Morgan Kaufmann, 2010.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Attribute-based group key establishment: a non-technical introduction

Consuelo Martínez López[1], Rainer Steinwandt[2] and Adriana Suárez Corona[1]

[1] *Departamento de Matemáticas, Universidad de Oviedo*

[2] *Department of Mathematical Sciences, Florida Atlantic University*

emails: chelo@orion.ciencias.uniovi.com, rsteinwa@fau.edu, adriana@orion.ciencias.uniovi.com

Abstract

Attribute-based cryptography adresses scenarios where the intended partner(s) of a communication are identified through the possession of certain attributes. We discuss two recent constructions for realizing group key establishment in such a setting and describe the underlying cryptographic primitives at a non-technical level. So far only a small number of constructions for attribute-based group key establishment have been identified, and we hope that this short introduction helps to stimulate further work in this young area of research in key establishment.

Key words: attribute-based cryptography, group key establishment
MSC 2000: 94A60

1 Introduction

In the standard public key scenario, public keys are used to verify signatures from or to encrypt messages for a particular user. Avoiding the need of a public key infrastructure with a certification authority is one motivation for *identity-based cryptography*. In the latter setting, certified public keys can be replaced with unique (public) identities, e. g., an e-mail address or a screen name: an arbitrary bitstring can be used as public identifier for a user. To send a message to a particular user or verify his signature, only his identity and the public system parameters are needed, no directory with certified public keys is necessary.

The idea of identity-based cryptogaphy has been introduced by Shamir more than 25 years ago [10] already. The first identity-based encryption scheme considered as efficient was not proposed until much later, however: in 2001 Boneh and Franklin [3, 4] proposed a pairing-based construction in the random oracle model whose security

 ISBN 13: 978-84-613-5510-5

is relying on a *Bilinear Diffie-Hellman* assumption. Since then various constructions have been identified and identity-based cryptography seems a rather convenient tool for communicating with a specific user or a set of users with known identities. For some situations, this way of specifying communication partners does not seem ideal though. Specifically, if the intended communication partner is not given as a particular identity, but rather by the requirement that the recipient is equipped with certain attributes—not being important who he is in particular. This is the case, for example if an enterprise wants to send a message to its employees or to the members of a particular department.

To deal with this kind of situations, Sahai and Waters [8] introduced in 2005 *fuzzy-identity based encryption*, a new type of identity-based encryption, where sets of descriptive attributes are used to specify a, not necessarily unique, communication partner. For example, sticking to the examples mentioned, the attributes of a user could consist of a description of his enterprise, department, position, etc. A message encrypted with a set of attributes W could only be decrypted by users possessing the private key for an attribute set W', such that W and W' have at least d elements in common, i. e., such that the *threshold condition* $|W \cap W'| \geq d$ is fulfilled. This allows some error tolerance, rendering the construction an interesting tool for applications involving biometric data as well. Since 2005, the idea of attribute-based cryptography has been extended in several directions, including efficiency improvements of the scheme by Sahai and Waters, the use of access structures that are more flexible than threshold based ones, or the use of attributes in schemes other than encryption.

In this note we will focus on *attribute-based group key establishment*, where some of these generalizations have been brought to use. Specifically, we take a look at the use of a *policy attribute-based key encapsulation mechanism* in a proposal of Gorantla et al. [7] and at the use of *attribute-based signcryption* in a proposal from [11]. One possible example where attribute-based group key establishment seems a natural tool is an Internet discussion forum, where users could only write or read messages if they possess the credentials necessary to derive the common symmetric key (for authentication and/or encryption). Going beyond attribute-based scenarios, one can consider *predicate-based* cryptographic primitives—[2] discusses such a setting for the two-party case, using a security model similar to what we discuss below. As an example of work in this context that uses an alternate approach to model security, Camenisch et al.'s work on *credential-authenticated key exchange* [5] can be mentioned; here Canetti's Universal Composability (UC) framework [6] is used. For the purpose of this introductory exposition, we restrict to attribute-based group key establishment, using an "oracle-based security model".

2 Cryptographic primitives used in attribute-based group key establishment

Similarly as in ordinary public key cryptography or in an identity-based setting, one can define attribute-based variants of asymmetric encryption, digital signatures and signcryption. These form the basis for the approach to attribute-based group key

 ISBN 13: 978-84-613-5510-5

establishment taken in [11]:

Encryption. A basic security requirement for an attribute-based encryption scheme is *one-wayness under chosen plaintext attacks* (OW-CPA) in the *selective access structure model*. Here a (probabilistic polynomial time) adversary $\mathcal{A}$ commits to a universe of attributes she wants to work with and selects an access structure $\mathbb{A}$ she wants to be challenged on. The adversary $\mathcal{A}$ is allowed to obtain secret decryption keys for arbitrary attribute sets that are not part of $\mathbb{A}$. The goal of $\mathcal{A}$ is to find the plaintext underlying an encryption of a random value under $\mathbb{A}$ (see [11] for more details) with non-negligible probability. If no successful adversary $\mathcal{A}$ exists, the corresponding encryption scheme is referred to as OW-CPA *secure*. For a threshold setting, a specific example of an encryption scheme meeting this security requirement is a pairing-based construction of Sahai and Waters in the above-mentioned paper from EUROCRYPT 2005 [8].

Digital signatures. Existential unforgeability for an attribute-based signature scheme can be defined analogously as for ordinary signature schemes. More specifically, to define *existential unforgeability under chosen message and attribute attacks* (UF-CMAA), a (probabilistic polynomial time) adversary $\mathcal{A}$ can extract private signing keys for arbitrary attribute sets and also obtain signatures for messages/attribute set pairs of her choice. The goal of $\mathcal{A}$ is to output a message μ, an access structure $\mathbb{A}$ and a signature σ, such that σ is a valid signature for μ under $\mathbb{A}$. Moreover, the signature must not have been obtained "trivially", i. e., after a signature query for μ under an attribute set in $\mathbb{A}$ or after a query for a private signing key for an attribute set in $\mathbb{A}$. To prove an attribute-signature scheme UF-CMAA *secure*, it has to be shown that the success probability for any adversary $\mathcal{A}$ as just discussed is negligible. For a threshold setting, a construction by Shahandashti and Safavi-Naini from AFRICACRYPT 2009 offers a specific example of a scheme that is secure in the aforementioned sense [9].

Signcryption. As detailed in [11], a OW-CPA secure attribute-based encryption scheme $\mathcal{E}$ and a UF-CMAA secure attribute-based signature scheme $\mathcal{S}$ can be combined, following the encrypt-then-sign paradigm, yielding an attribute-based analog $\mathcal{E}t\mathcal{S}$ of an ordinary signcryption scheme. The unforgeability requirement UFS-CMAA for an *attribute-based signcryption* scheme formalizes the idea that any probabilistic polynomial time adversary has only a negligible success probability in creating a valid signcryption, unless a query to a signcryption oracle or for a private key has been issued that makes the forgery trivial. Similarly, OWS-CPA *security in the selective access structure model* ensures that unsigncrypting the signcryption of a random plaintext succeeds with negligible probability only, unless a secret key query has been issued that makes the challenge trivial. As in the definition of OW-CPA discussed above, in the selective access structure model, the adversary has to commit in advance to the access structure on which she will be challenged.

 ISBN 13: 978-84-613-5510-5

Key encapsulation. In [7] the notion of an *encapsulation policy attribute-based key encapsulation mechanism* is introduced. In analogy to an ordinary key encapsulation mechanism, it allows the creation of an encapsulation of a symmetric key such that only users who are part of a specified access structure can recover the encapsulated key. Gorantla et al. consider monotone access structures and include a possibility for users to delegate the decapsulation capability. This idea is captured by an algorithm that on input the public parameters and a secret user key allows the derivation of a secret key for a subset of the credentials a user possesses. The definition of security in the sense of *indistinguishability under chosen ciphertext attacks (*IND-CCA*)* builds on earlier work by Bethencourt et al. [1]. The idea is to give a (probabilistic polynomial time) adversary $\mathcal{A}$ access to a secret key extraction oracle and a decapsulation oracle. With access to these oracles $\mathcal{A}$ selects an access structure and then has (with suitable restrictions on her oracle access) to decide if a challenge pair (K, C) consists of a symmetric key K along with an encapsulation C of K, or rather of a random key K and an encapsulation C of a key that has been computed independently of K.

In the next section we take a look at how the above primitives have been brought to use to construct attribute-based group key establishment protocols.

3 Attribute-based group key establishment

To formalize the security of a key establishment protocol, a number of approaches have been considered. Both Gorantla et al. in [7] and the authors of [11] decided for a model where adversarial capabilities are captured through certain oracles. This is different from the UC framework which has been used by Camenisch et al. [5], for instance.

3.1 Modelling security

Essentially, an adversary $\mathcal{A}$ (which is modeled as a probabilistic polynomial time algorithm) has access to `Send`, `RevealKey` and `Corrupt` oracles:

- `Send` materializes $\mathcal{A}$'s capability to initiate, insert, delete or modify protocol messages; the intuition is that the communication network is controlled by the adversary, and hence messages may be deleted, altered, or made up by $\mathcal{A}$.
- `Corrupt` captures $\mathcal{A}$'s capacity to control some users and the compromise of long-term secret keys of users.
- `RevealKey` (or `Reveal`) models a situation in which an (old) session key has been compromised. The adversary can query this oracle to learn successfully established session keys without corrupting a user.

One could think of adding a separate `Execute` oracle to model a passive eavesdropping, but $\mathcal{A}$ can simulate the latter by simply forwarding messages faithfully, using her `Send` oracle: neither [7] nor [11] use low-entropy secrets for authentication, and there seems no need for distinguishing "offline" and "online" attacks of $\mathcal{A}$. An option that could be

 ISBN 13: 978-84-613-5510-5

considered is to use a stronger form of `Corrupt`, revealing not only long-term secrets, but also state information of a user.

To express the basic security goal of a group key establishment—secrecy of the session key—another oracle is introduced, which does not really reflect an adversarial capability but is more a technical tool:

- `Test`: When being queried with a protocol instance that has accepted a session key, `Test` chooses uniformly at random a value $b \in \{0, 1\}$. If $b = 0$ the session key accepted by the respective instance is returned. If $b = 1$, a uniformly at random chosen element from the space of possible session keys is returned.

By means of `Test`, security of a group key establishment can be characterized as follows: the protocol is secure (in the sense of key secrecy) if no probabilistic polynomial time adversary with access to the above tools can distinguish if the output of the `Test` oracle is the established key or a random one with more than negligible probability. Obviously, some restrictions have to be imposed to exclude trivial attacks like revealing a session key from a protocol instance and then querying `Test` with the very same instance. Technically this is captured by introducing a notion of *freshness* for protocol instances and only *fresh instances* may be used in a `Test` query. While being technical, the definition of freshness is crucial, as it influences which attacks are considered successful. For instance, the attribute-based key establishment protocols put forward in [7] and [11] handle Corrupt queries differently, and this has consequences for another desirable security goal: *forward security*

Forward security of a key establishment protocol ensures that (old) session keys remain secure, even if later long-term secrets of users are compromised. In [11] a freshness definition is used which takes forward security into account—an adversary is allowed to corrupt users after querying `Test`, as long as her use of `Send` satisfies appropriate restrictions. Consequently, the security proof for the proposed two-round solution implies forward security. The authors of [7] treat forward security as a separate issue, starting out with a one-round solution with a freshness definition not implying forward security. In a second step, they explore possibilities to augment the one-round solution in such a way that forward security is achieved—possibly at the cost of increasing the round complexity.

In addition to key secrecy and forward security, further protocol properties can be explored, which in some applications might be of interest. For instance, [11] addresses topics like *attribute privacy* and *deniability*.

3.2 Two proposed protocols for attribute-based group key establishment

The protocol proposed in [7] is based on an *attribute-based key encapsulation mechanism* as outlined in Section 2. Before starting the actual protocol, each user is issued a private key for his attributes and they are given (or agree on) an access structure which their attributes have to be part of. The protocol has only one round, where each party U_i broadcasts an encapsulation C_i on input a session key contribution k_i

 ISBN 13: 978-84-613-5510-5

and the access structure agreed upon. Once a user U_i has received all encapsulations, U_i can decapsulate them by means of his private key (if the attributes are part of the respective access structure). Therewith U can derive the common session key

$$\mathtt{sk} = f_{k_1}(\mathtt{sid}) \oplus f_{k_2}(\mathtt{sid}) \oplus \cdots \oplus f_{k_n}(\mathtt{sid}),$$

where f is a pseudorandom function and the *session identifier* $\mathtt{sid} = (C_1 \| \cdots \| C_n)$ is the concatenation of all encapsulations. Gorantla et al. show that if the underlying attribute-based key encapsulation mechanism is secure in the sense of indistinguishability under chosen ciphertext attacks (IND-CCA), the resulting attribute-based group key establishment offers key secrecy. To achieve forward secrecy, [7] discusses different options: when restricting to groups with $n \leq 3$ users, dedicated constructions are suggested (e. g., using pairings) which avoid the addition of an additional round, whereas for more general values of n no one-round construction with forward security is offered.

The approach to attribute-based group key establishment taken in [11] is different from the one just described, resulting in a two-round solution with forward security, independent of the number of participants. The main technical tool is *attribute-based signcryption* as outlined in Section 2. Before starting the protocol, each user obtains the private key corresponding to his attributes (users with the same attributes are considered the same user). Every user chooses a random string k_U and an element $x_U \in \{1, \ldots, \mathrm{ord}(g)\}$, and computes $y_U = g^{x_U}$, where g is the generator of a suitable cyclic group. There is a special user U_{init}, called *initiator*, who in addition selects a random value r and signcrypts his $k_{U_{\mathrm{init}}}$ with the private key of his attributes and with access structure his $\mathtt{pid}_{U_{init}}$ (which describes the set of acceptable communication partners). In the first protocol round, all users broadcast their k_U- and y_U-values, except the initiator, who broadcasts a signcryption c of $k_{U_{\mathrm{init}}}$, the $y_{U_{\mathrm{init}}}$, $H(r)$ and his $\mathtt{pid}_{U_{\mathrm{init}}}$. After having received these messages, every user U—except the initiator—unsigncrypts c with the private key for his attributes and his $\mathtt{pid}_U$ as the verification access structure, therewith recovering the initiator's key contribution $k_{U_{\mathrm{init}}}$. Ordering the users U lexicographically according to their k_U-value, user no. i computes the values $t_i^{\mathrm{L}} := H(y_{i-1}^{x_i} \| k_0)$, $t_i^{\mathrm{R}} := H(y_{i+1}^{x_i} \| k_0)$ and $X_i := t_i^{\mathrm{L}} \oplus t_i^{\mathrm{R}}$ (here U_0 is assumed to be the initiator). The initiator U_0 computes additionally $e := k_0 \oplus r \oplus t_0^{\mathrm{R}}$.

In Round 2 the values (X_i, i) are broadcast, and the initiator also broadcasts e. Once a user has received these messages, he checks if the sum of the X_i values is 0 and computes r by first computing $t_0^{\mathrm{R}} = t_i^{\mathrm{L}} \oplus X_0 \oplus \bigoplus_{j=i}^{n-1} X_j$. Then U checks if the commitment $H(r)$ is correct. If any check fails, U aborts the protocol. Otherwise the session key is

$$\mathtt{sk} = H(r \| k_0 \| k_1 \| \cdots \| k_{n-1} \| \mathtt{pid}_{U_0} \| 0)$$

and the session identifier is $\mathtt{sid} = H(r \| k_0 \| k_1 \| \cdots \| k_{n-1} \| \mathtt{pid}_{U_0} \| 1)$.

If the group generated by g chosen is such that the computational Diffie- Hellman assumption holds, $H(\cdot)$ is a random oracle and the attribute-based signcryption scheme used is secure in the sense of OWS-CPA and UFS-CMAA as outlined in Section 2, then this protocol can be shown to provide key secrecy (see [11] for details).

 ISBN 13: 978-84-613-5510-5

4 Conclusion

We hope that the above discussion gives an idea of what kind of techniques are currently used in attribute-based group key establishment. Only few proposals are available so far, and we hope that this informal introduction helps to stimulate further work in this line of research. The work presented in [7, 11] suggests several natural directions for follow-up work: "direct" constructions of attribute-based group key establishments achieving forward security and one-round constructions which, through appropriate protocol compilers, are lifted to forward secure schemes. Finally, going beyond purely attribute-based settings to predicate-based versions seems an area in group key establishment which is not very well-explored yet and has potential for interesting follow-up work. Moreover, for "non-standard" guarantees like deniability, already the question of finding adequate formalizations seems to be an interesting one.

Acknowledgments

This work has been partially supported by grants MTM 2007-67884-C04-01, IB-08-147 and FPU grant AP2007-03141, cofinanced by the European Social Fund.

References

[1] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-Policy Attribute-Based Encryption. In *IEEE Symposium on Security and Privacy*, pages 321–334, 2007.

[2] James Birkett and Douglas Stebila. Predicate-Based Key Exchange. Cryptology ePrint Archive, Report 2010/082, February 2010. Available at `http://eprint.iacr.org/2010/082/`.

[3] Dan Boneh and Matthew Franklin. Identity-Based Encryption from the Weil Pairing. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag, 2001.

[4] Dan Boneh and Matthew Franklin. Identity-Based Encryption from the Weil Pairing. *SIAM J. of Computing*, 32(3):586–615, 2003.

[5] Jan Camenisch, Nathalie Casati, Thomas Gross, and Victor Shoup. Credential Authenticated Identification and Key Exchange. Cryptology ePrint Archive: Report 2010/055, February 2010. Available at `http://eprint.iacr.org/2010/055/`.

[6] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067, December 2005. Available at `http://eprint.iacr.org/2000/067/`.

[7] Malakondayya Choudary Gorantla, Colin Boyd, and Juan Manuel González Nieto. Attribute-based Authenticated Key Exchange. Cryptology ePrint Archive: Report 2010/084, February 2010. To appear at ACISP 2010.

 ISBN 13: 978-84-613-5510-5

[8] Amit Sahai and Brent Waters. Fuzzy Identity-Based Encryption. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2005.

[9] Siamak F. Shahandashti and Reihaneh Safavi-Naini. Threshold Attribute-Based Signatures and Their Application to Anonymous Credential Systems. In Bart Preneel, editor, *Progress in Cryptology – AFRICACRYPT 2009*, volume 5580 of *Lecture Notes in Computer Science*, pages 198–216. Springer, 2009. Full version available as Cryptology ePrint Archive Report 2009/126, `http://eprint.iacr.org/2009/126/`.

[10] Adi Shamir. Identity-Based Cryptosystems and Signature Schemes. In G.R. Blakley and D. Chaum, editors, *Advances in Cryptology – CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1985.

[11] Rainer Steinwandt and Adriana Suárez Corona. Attribute-based group key establishment. *Advances in Mathematics of Communications*, to appear.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Block Matrices and Stability Theory II

Fernando Martins[*1], **Edgar Pereira**[*2] **and José Vitória**[3]

[1] *Coimbra College of Education, Polytechnic Institute of Coimbra*

[2] *Department of Informatics, University of Beira Interior*

[3] *Department of Mathematics, University of Coimbra.*
[*] *Member of Instituto de Telecomunicações, Pólo de Coimbra, Delegação da Covilhã.*

emails: fmlmartins@esec.pt, edgar@di.ubi.pt, jvitoria@mat.uc.pt

MSC2010: 15A57; 34D99, 93D05

Key words: Stability theory; matrix differential equation; block Hermite matrix; block Hurwitz matrix; block Routh matrix; block Schwarz matrix; matrix polynomials; block minor;

1 Introduction

In a previous work [4] we studied the relations between block matrices and differential equations in the stability theory. We gave sufficient conditions in terms of blocks for the Lyapunov matrix equation in order to obtain stability criteria. The classical Hermite, Routh, Hurwitz and Schwarz matrices were generalized to block contexts and used to verify under certain conditions the stability of matrix differential equations [3, 5, 6]. The two main limitations of that work are that the differential matrix equation has to be in a companion form and that the criteria based in the minors, of the classical matrices mentioned before, can not be generalized to block contexts considering that we do not have block minor for a general block matrix.

Here, we will restrict our matrices to commuting blocks so we can work with block determinants and obtain a generalized characteristic polynomial, thus permitting, firstly, the work of differential equations in the form:

$$y'(t) = A_b y(t) \tag{1}$$

where A_b is a general block matrix and, secondly, to develop the stability theory in terms of the block minors.

 ISBN 13: 978-84-613-5510-5

2 Results

Let A_b be a real matrix partitioned into $m \times m$ commuting blocks of order n, which we represent by $A_b \in M_m(B_n(\mathbb{R}))$. The block determinant of A_b is a block, $det_b(A_b)$ computed the usual determinant as the block were scalars [7]. The characteristic matrix polynomial of A_b is $P(X) = det_b(I_m \otimes X - A_b)$ [8].

In a similar way we say $det_b(A_b)_{(j)}$ is the (j) block minor of A_b, that is: the block determinant computed for the principal block sub-matrix of $j \times j$ blocks.

A matrix $E \in \mathbb{R}^{n\times n}$ is said to be symmetrizable if there exists a matrix $F = F^T \in \mathbb{R}^{n\times n}$ positive definite such that $E^T F = FE$ [2, p. 67]. Furthermore, two matrices $E_1, E_2 \in \mathbb{R}^{n\times n}$ are said to be simultaneously symmetrizable if there exists a matrix $F = F^T \in \mathbb{R}^{n\times n}$ positive definite such that $E_1^T F = FE_1 \quad and \quad E_2^T F = FE_2$ [1]. It follows directly from this definition that $E_1 + E_2$ and $E_1 - E_2$ are simultaneously symmetrizable.

Now, given a matrix polynomial

$$P(X) = X^m + A_1 X^{m-1} + \cdots + A_{m-1}X + A_m, \tag{2}$$

where $A_1, \ldots, A_m \in B_n(\mathbb{R})$ and X is also a matrix of order n. Then, using the block versions of Hermite, Routh, Hurwitz and Schwarz associated with $P(X)$ ([4] and references there in) we present:

Proposition 2.1 *Let $H_b^e \in M_m(B_n(\mathbb{R}))$ be the block Hermite matrix associated with $P(X)$. Then the block minors $H_{b(j)}^e$ of H_b^e are:*
for $j = 1$,

$$H_{b(1)}^e = H_{b(11)}^e; \tag{3}$$

for even $j \geq 2$,

$$H_{b(j)}^e = det_b\left[N_{uv}^{(1)}\right] det_b\left[N_{uv}^{(2)}\right], \tag{4}$$

which

$$\begin{aligned} &N_{uv}^{(1)} = H_{b(2u2v)}^e, \quad u, v = 1, \ldots, \frac{j}{2} \\ &and \\ &N_{uv}^{(2)} = H_{b((2u-1)(2v-1))}^e, \quad u, v = 1, \ldots, \frac{j}{2}; \end{aligned} \tag{5}$$

for odd $j \geq 3$,

$$H_{b(j)}^e = det_b\left[N_{uv}^{(1)}\right] det_b\left[N_{uv}^{(2)}\right], \tag{6}$$

which

$$\begin{aligned} &N_{uv}^{(1)} = H_{b(2u2v)}^e, \quad u, v = 1, \ldots, \frac{j-1}{2} \quad and \\ &N_{uv}^{(2)} = H_{b((2u-1)(2v-1))}^e, \quad u, v = 1, \ldots, \frac{j+1}{2}. \end{aligned} \tag{7}$$

Proposition 2.2 *Let $H_b \in M_m(B_n(\mathbb{R}))$ be the block Hurwitz matrix associated with $P(X)$. Then the block minors $H_{b(j)}$ of H_b are:*
for odd $j \geq 1$,

$$H_{b(j)} = A_0^{-1} det_b\left[N_{uv}^{(2)}\right], \tag{8}$$

 ISBN 13: 978-84-613-5510-5

which

$$N_{uv}^{(2)} = H^e_{b((2u-1)(2v-1))} \quad u,v = 1,\ldots,\frac{j+1}{2} \quad \textit{and} \quad j = 2k-1, \quad k = 1,\ldots,\left[\frac{m+1}{2}\right];$$

for even $j \geq 2$,

$$H_{b[j]} = det_b\left[N_{uv}^{(1)}\right], \tag{9}$$

which

$$N_{uv}^{(1)} = H^e_{b(2u2v)}, \quad u,v = 1,\ldots,\frac{j}{2} \quad \textit{and} \quad j = 2k, \quad k = 1,\ldots,\left[\frac{m}{2}\right]. \tag{10}$$

Corollary 2.1 *If* H_b *and* H_b^e *are associated with* $P(X)$. *Then, the respective block minors satisfy:*

$$H^e_{b(j)} = A_0 H_{b(j-1)} H_{b(j)}, \quad (j = 1,\ldots,m; \quad H_0 = I_n). \tag{11}$$

Proposition 2.3 *Let* $S_b \in M_m(B_n(\mathbb{R}))$ *be the block Schwarz matrix associated with* $P(X)$. *Then, its characteristic matrix polynomial is*

$$P_m(X) = P_{m-1}(X)\Lambda + P_{m-2}(X)S_m, \tag{12}$$

with $P_{-1}(X) = P_0(X) = I_n$ *and* $m \geq 1$, *where* $P_{m-1}(\Lambda)$ *and* $P_{m-2}(\Lambda)$ *are the characteristic matrix polynomials of* $S_{b(m-1)}$ *and* $S_{b(m-2)}$, *respectively.*

Proposition 2.4 *Let* $S_b \in M_m(B_n(\mathbb{R}))$ *and* $H_b \in M_m(B_n(\mathbb{R}))$ *be the block Schwarz and the block Hurwitz matrices associated with* $P(X)$. *If the blocks* S_j $(j = 1,\ldots,m)$ *of* S_b *are constructed by:*

$$S_1 = H_{b(1)}, S_2 = H^{-1}_{b(1)}H_{b[2]},\ S_3 = H^{-1}_{b(1)}H^{-1}_{b(2)}H_{b(3)},\ S_r = H^{-1}_{b(r-1)}H^{-1}_{b(r-2)}H_{b(r-3)}H_{b(r)},$$

with $r = 4,\ldots,m$, *in which the block minors* $H_{b(j)}$, *are nonsingular. Then, the characteristic matrix polynomial of* S_b *is* $P(X)$.

Proposition 2.5 *Let* $S_b \in M_m(B_n(\mathbb{R}))$ *and* $H_b^e \in M_m(B_n(\mathbb{R}))$ *be the block Schwarz and the block Hermite matrices associated with* $P(X)$. *If the blocks* S_j $(j = 1,\ldots,m)$ *of* S_b *are constructed by:*

$$S_1 = H^e_{b(1)}, \quad S_2 = \left(H^e_{b(1)}\right)^{-2} H^e_{b(2)}, \quad S_r = \left(H^e_{b(r-1)}\right)^{-2} H^e_{b(r-2)}H^e_{b(r)},$$

with $r = 3,\ldots,m$, *in which the block minors* $H^e_{b(j)}$, *are nonsingular. Then, the characteristic matrix polynomial of* S_b *is* $P(X)$.

Proposition 2.6 *Let* $y'(t) = A_b y(t)$ *be a matrix differential equation, let* $P(X)$ *be the characteristic matrix polynomial of* A_b *and let* $H_b^e \in M_m(B_n(\mathbb{R}))$, $H_b \in M_m(B_n(\mathbb{R}))$ *and* $R_b \in M_m(B_n(\mathbb{R}))$ *be the block Hermite, the block Hurwitz and the block Routh matrices, respectively, associated with* $P(X)$. *If one of the following conditions holds:*

 ISBN 13: 978-84-613-5510-5

(i)

$$H^e_{b(1)}, \quad \left(H^e_{b(1)}\right)^{-1} H^e_{b(2)}, \quad \ldots, \quad \left(H^e_{b(m-1)}\right)^{-1} H^e_{b(m)},$$

are simultaneously symmetrizable and positive definite, where $H^e_{b(j)}$ *are the block minors of* H^e_b*;*

(ii)

$$H_{b(1)}, \quad H_{b(2)}, \quad H^{-1}_{b(1)} H_{b(3)}, \quad \ldots, \quad H^{-1}_{b(m-2)} H_{b(m)},$$

are simultaneously symmetrizable and positive definite, where $H_{b(j)}$ *are the block minors of* H_b*;*

(iii)

$$C_{21}, \quad C_{21}C_{31}, \quad C_{31}C_{41}, \quad \ldots, \quad C_{m1}C_{(m+1)1},$$

are simultaneously symmetrizable and positive definite, where C_{ij} *are block elements of* R_b*;*

Then, the equilibrium of the matrix differential equation is asymptotically stable.

References

[1] S. Adhikari, *On Symmetrizable Systems of Second Kind*, Journal of Applied Mechanics **67** (2000) 797–802.

[2] R. Bellman, *Introduction to Matrix Analysis*, McGraW-Hill Book Company, New York, 1960.

[3] S. H. Lehnigk, *Stability Theorems for Linear Motions with an Introduction to Lyapunov's Direct Method* , Prentice-Hall, Englewood Cliffs, N. J., London, 1966.

[4] F. Martins and E. Pereira, *Block Matrices and Stability Theory*, Tatra Mountains Mathematical Publications **38** (2007) 147–162.

[5] E. J. Routh, *A Treatise on the Stability of a Given State of Motion*, Adams Prize Essay, Univ. Cambridge, England, 1877.

[6] H. R. Schwarz, *Ein Verfahren zur Stabilitatsfrage bei Matrizen-Eigenwort-Problemen*, ZAMP, **7** (1956) 473–500 .

[7] J. Vitória, *Block Eigenvalues of Block Compound Matrices*, Linear Algebra and its Applications **47** (1982) 23–34.

[8] J. Vitória, *A Block Cayley Hamilton Theorem*, Bulletin Mathemátique (Roumanie) **26** (1982) 93–97 .

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

On multi-adjoint concept lattices based on heterogeneous conjunctors

J. Medina[1] and M. Ojeda-Aciego[2]

[1] *Dept. Matemáticas, Universidad de Cádiz. Spain*

[2] *Dept. Matemática Aplicada, Universidad de Málaga. Spain*

emails: jesus.medina@uca.es, aciego@uma.es

Abstract

In formal concept analysis, the sets of attributes and objects are usually different, with different meaning and, hence, it might not make sense to evaluate them on the same carrier. In this context, the operators used to obtain the concept lattice could be defined by considering different lattices associated to attributes and objects. Anyway there exist several reasons for which we need to evaluate the set of attributes and objects in the same carrier. In this direction, we present in this paper a new concept lattice, where the objects and attributes are evaluated on the same lattice L, although operators which evaluate objects and attributes in different carriers are used. Moreover, we have studied the relationship between the new concept lattice and the other one obtained directly considered different carriers to both set of attributes and objects.

Key words: Concept lattices, multi-adjoint lattices, Galois connection, implication triples

1 Introduction

Formal concept analysis was introduced by Wille in the eighties and it has become an important and appealing research topic both from the theoretical perspective [13,23,26] and from the applicative one [7, 9, 10, 12, 22].

Soon after the introduction of "classical" formal concept analysis, a number of different approaches for its generalization were introduced and, nowadays, there are works which extend the theory with ideas from fuzzy set theory [3,17,18] or fuzzy logic reasoning [2, 4, 8] or from rough set theory [16, 24, 27] or some integrated approaches such as fuzzy and rough [25], or rough and domain theory [14].

Recently, a new fuzzy framework has been introduced which is more general and flexible than other fuzzy extensions, see [20]. In this framework, we can evaluate the

 ISBN 13: 978-84-613-5510-5

set of objects and attributes on different lattices L_1, L_2, because it might not make sense to evaluate objects and attributes on the same carrier.

It is convenient to recall that, sometimes, it could be interesting to weaken this framework. For instance, given a group of experts that need to evaluate a knowledge system, they could believe that the carriers associated to the set of objects and attributes should not be different, or some of them believe that the attributes should be evaluated on L_1 and some others believe that they should be evaluated on L_2 and, once the evaluation is finished, the results should be homogenized. An interesting possibility is to embed both L_1 and L_2 into a set L, and to obtain a new concept lattice $\mathcal{M}_L$, in which the set of objects and attributes are evaluated in the *same* lattice, albeit using the operators which evaluate objects and attributes in different carriers.

Firstly, we will introduce the notion of P-connected poset, which will be used to define the concept lattice $\mathcal{M}_L$, when the set of attributes and objects are evaluated in L_1 and L_2, respectively, and L_1, L_2 are L-connected. Later, the new concept lattice is related with the concept lattice introduced in [20]. Finally, some conclusions and future work are presented.

2 P-connected posets

The main notion in this contribution refers to the notion of P-connection between two complete lattices. As we will see later, this condition will allow to somehow conciliate the different values generated by the consideration of a non-commutative conjunctor in the construction of a concept lattice.

Definition 1 *Given the posets $(P_1, \leq_1)$, $(P_2, \leq_2)$ and $(P, \leq)$, we say that P_1 and P_2 are* P-connected *if there exist increasing mappings $i_1 \colon P_1 \to P$, $\phi_1 \colon P \to P_1$, $i_2 \colon P_2 \to P$ and $\phi_2 \colon P \to P_2$ verifying that $\phi_1(i_1(x)) = x$, and $\phi_2(i_2(y)) = y$, for all $x \in P_1$, $y \in P_2$.*

Example 1 *Any pair of posets $(P_1, \leq_1)$, $(P_2, \leq_2)$ with top elements $\top_1$ and $\top_2$, respectively, are $P_1 \times P_2$-connected, with the pairwise ordering, where $P_1 \times P_2$ is the Cartesian product, and by considering the mappings ϕ_i as the projections π_i, and i_1, i_2 as the inclusions defined as $i_1(x) = (x, \top_2)$, $i_2(y) = (\top_1, y)$, for all $x \in P_1$, $y \in P_2$.*

A more complex example is presented below:

Example 2 *Assume that, in order to perform an evaluation of a product, for which we have to assign one value out of four possible ones. We ask two experts to collaborate in this task and, only when collecting the feedback from each expert, we notice that one expert has considered the ordering of values as in Fig. 1, whereas the other has considered that in Fig. 2. In both cases, the expert has used a suitable poset in order to obtain the final result of the evaluation.*

In order to unify both evaluations, we want to embed the posets in Figs. 1 and 2 into another one, for example, we might consider that given in Fig. 3.

 ISBN 13: 978-84-613-5510-5

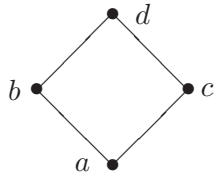


Figure 1: Poset $(P_1, \leq_1)$

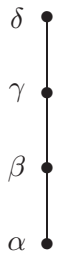


Figure 2: Poset $(P_2, \leq_2)$

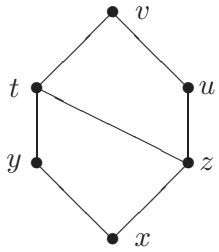


Figure 3: Poset $(P, \leq)$

	a	b	c	d
i_1	x	y	u	v

	α	β	γ	δ
i_2	x	y	t	v

Figure 4: Definition of i_1 and i_2

We can define two mappings $i_1 \colon P_1 \to P$, $i_2 \colon P_2 \to P$ as in Fig. 4; moreover, there exist several possibilities for the mappings $\phi_1 \colon P \to P_1$, $\phi_2 \colon P \to P_2$ in order to satisfy the properties in Definition 1, one of them is shown below:

	x	y	z	t	u	v
ϕ_1	a	b	c	d	c	d

	x	y	z	t	u	v
ϕ_2	α	β	γ	γ	δ	δ

As a result, P_1 and P_2 are P-connected.

Example 3 *A different example arises when we consider the posets $([0,1]_2, \leq)$ and $([0,1]_4, \leq)$, where $[0,1]_n$ is a regular partition of $[0,1]$ into n pieces, for instance $[0,1]_2 = \{0, 0.5, 1\}$, $[0,1]_4 = \{0, 0.25, 0.5, 0.75, 1\}$.*

We have that $[0,1]_2$, $[0,1]_4$ are $[0,1]$-connected, under the usual ordering, considering the mappings i_1, i_2 as the inclusions $i_1(x) = x$, $i_2(y) = y$, for all $x \in L_1$, $y \in L_2$; and ϕ_1, ϕ_2 defined as $\phi_1(t) = \lceil 2 \cdot t \rceil / 2$, $\phi_2(t) = \lceil 4 \cdot t \rceil / 4$, where $\lceil \_ \rceil$ is the ceiling function. For example, if $t = 0.55$, $\phi_1(0.55) = 1$, $\phi_2(0.55) = 0.75$,

3 Concept lattices on L-connected lattices

Firstly, we will recall the definition of adjoint triple, multi-adjoint frame and context, in order to define a new concept lattice where the objects and attributes are evaluated on the same lattice L. This new standpoint has several applications: for instance, although

 ISBN 13: 978-84-613-5510-5

operators which evaluate objects and attributes in different carriers are used, we will show later that it is possible to evaluate both objects and attributes in a common lattice obtained from the original ones and, using the methods introduced in [19], to obtain a certain class of t-concepts. Originally, the condition $L_1 = L_2$ was assumed; but the construction can be extended to the cases in which $L_1 \neq L_2$, as the only requirement is that both lattices should be L-connected.

Assuming a conjunctor defined on, say $P_1 \times P_2$, directly provides two different ways of generalising the well-known adjoint property between a t-norm and its residuated implication [1, 21], depending on which argument is fixed.

Definition 2 *Let $(P_1, \leq_1)$, $(P_2, \leq_2)$, $(P_3, \leq_3)$ be posets and $\&\colon P_1 \times P_2 \to P_3$, $\swarrow\colon P_3 \times P_2 \to P_1$, $\nwarrow\colon P_3 \times P_1 \to P_2$ be mappings, then $(\&, \swarrow, \nwarrow)$ is an* adjoint triple *with respect to P_1, P_2, P_3 if:*

1. *$\&$ is order-preserving in both arguments.*
2. *$\swarrow$ and $\nwarrow$ are order-preserving in the consequent and order-reversing in the antecedent.*
3. *$x \leq_1 z \swarrow y$ iff $x \,\&\, y \leq_3 z$ iff $y \leq_2 z \nwarrow x$, where $x \in P_1$, $y \in P_2$ and $z \in P_3$.*

The general theory formal concept analysis needs that the underlying posets have the structure of a lattice. Therefore, we will assume hereafter that we are working on lattices instead of on posets.

The multi-adjoint framework allows the existence of several adjoint triples for a given triplet of lattices.

Definition 3 *A* multi-adjoint frame $\mathcal{L}$ *is a tuple*

$$(L_1, L_2, P, \preceq_1, \preceq_2, \leq, \&_1, \swarrow^1, \nwarrow_1, \ldots, \&_n, \swarrow^n, \nwarrow_n)$$

where $(L_1, \preceq_1)$ and $(L_2, \preceq_2)$ are complete lattices, $(P, \leq)$ is a poset and, for all $i = 1, \ldots, n$, $(\&_i, \swarrow^i, \nwarrow_i)$ is an adjoint triple with respect to L_1, L_2, P.
Multi-adjoint frames are denoted $(L_1, L_2, L, \&_1, \ldots, \&_n)$.

Given a frame, a *multi-adjoint context* is a tuple consisting of sets of objects and attributes and a fuzzy relation among them; in addition, the multi-adjoint approach also includes a function which assigns an adjoint triple to each object (or attribute).

Definition 4 *Let $(L_1, L_2, P, \&_1, \ldots, \&_n)$ be a multi-adjoint frame, a* multi-adjoint context *is a tuple (A, B, R, σ) such that A and B are non-empty sets (usually interpreted as attributes and objects, respectively), R is a P-fuzzy relation $R\colon A \times B \to P$ and $\sigma\colon B \to \{1, \ldots, n\}$ is a mapping which associates any element in B with some particular adjoint triple in the frame.*[1]

[1] A similar theory could be developed by considering a mapping $\tau\colon A \to \{1, \ldots, n\}$ which associates any element in A with some particular adjoint triple in the frame.

 ISBN 13: 978-84-613-5510-5

In order to make this contribution self-contained and since we will provide a specific construction of a Galois connection, we recall its formal definition below:

Definition 5 *Let $(P_1, \leq_1)$ and $(P_2, \leq_2)$ be posets, and $^{\downarrow}\colon P_1 \to P_2$, $^{\uparrow}\colon P_2 \to P_1$ mappings, the pair $(^{\uparrow}, ^{\downarrow})$ forms a* Galois connection *between P_1 and P_2 if and only if:*

1. *$^{\uparrow}$ and $^{\downarrow}$ are order-reversing.*
2. *$x \leq_1 x^{\downarrow\uparrow}$ for all $x \in P_1$.*
3. *$y \leq_2 y^{\uparrow\downarrow}$ for all $y \in P_2$.*

In the following paragraphs, we define a suitable Galois connection on which the new concept lattice structure will be built.

Given a complete lattice $(L, \preceq)$ such that L_1 and L_2 are L-connected, a multi-adjoint frame $(L_1, L_2, P, \&_1, \ldots, \&_n)$, and a context (A, B, R, σ), we can define the mappings[2] $^{\uparrow_{c\sigma}}\colon L^B \to L^A$ and $^{\downarrow^{c\sigma}}\colon L^A \to L^B$ defined for all $g \in L^B$ and $f \in L^A$ as follows:

$$g^{\uparrow_{c\sigma}}(a) = i_1(\inf\{R(a,b) \swarrow^{\sigma(b)} \phi_2(g(b)) \mid b \in B\}) \tag{1}$$
$$f^{\downarrow^{c\sigma}}(b) = i_2(\inf\{R(a,b) \nwarrow_{\sigma(b)} \phi_1(f(a)) \mid a \in A\}) \tag{2}$$

Note that these definitions can be related to those given in [20] in that, for each adjoint triple $(\&, \swarrow, \nwarrow)$ of the multi-adjoint frame, we can define the mappings $\&^*\colon L \times L \to P$, $\swarrow^*\colon P \times L \to L$ and $\nwarrow_*\colon P \times L \to L$ for all $x, y \in L$ and $z \in P$ as follows:

$$x \,\&^*\, y \;=\; \phi_1(x) \,\&\, \phi_2(y) \qquad\qquad \begin{array}{rcl} z \swarrow^* y & = & i_1(z \swarrow \phi_2(y)) \\ z \nwarrow_* x & = & i_2(z \nwarrow \phi_1(x)) \end{array}$$

which, under the requirements $t \leq i_1(\phi_1(t))$ and $t \leq i_2(\phi_2(t))$, for all $t \in L$, forms another adjoint triple $(\&^*, \swarrow^*, \nwarrow^*)$. Under the additional assumption that the mappings i_j are inf-preserving, the mappings $^{\uparrow_{c\sigma}}\colon L^B \to L^A$ and $^{\downarrow^{c\sigma}}\colon L^A \to L^B$ can be written as

$$g^{\uparrow_{c\sigma}}(a) \;=\; \inf\{R(a,b) \swarrow^{*b} g(b) \mid b \in B\} \tag{3}$$
$$f^{\downarrow^{c\sigma}}(b) \;=\; \inf\{R(a,b) \nwarrow_{*b} f(a) \mid a \in A\} \tag{4}$$

and coincide with the Galois connection associated to the frame $(L_1, L_2, P, \&_1^*, \ldots, \&_n^*)$ introduced in [20]. As our construction of the new concept lattice will not need either of the requirements above, the proposed framework is strictly more general than the previous one.

Expressions (1), (2) do not coincide with those given in [20], because they are not defined directly from a residuated implication, although the mappings i_1, i_2, ϕ_1 and ϕ_2 are involved as well. Hence, we need to prove that these mappings form a Galois connection.

[2]The subscript c refers to the L-connection, since we are using the mappings ϕ_j and i_j; on the other hand, σ is needed to refer the particular choice of adjoint triple for a given b.

 ISBN 13: 978-84-613-5510-5

Proposition 1 *Let $(L_1, L_2, P, \&_1, \ldots, \&_n)$ be a multi-adjoint frame, where L_1 and L_2 are L-connected, and a context (A, B, R, σ), the pair $(^{\uparrow_{c\sigma}}, ^{\downarrow^{c\sigma}})$ is a Galois connection between L^A and L^B.*

The Galois connection just obtained is defined on a frame where $(L_1, \preceq_1)$ and $(L_2, \preceq_2)$ are L-connected. This Galois connection allows for defining a new concept lattice following the usual construction: a *concept* is a pair $\langle g^*, f^* \rangle$ satisfying $g^* \in L^B$, $f^* \in L^A$ and that $(g^*)^{\uparrow_c} = f^*$ and $(f^*)^{\downarrow^c} = g^*$; with $(^{\uparrow_c}, ^{\downarrow^c})$ being the Galois connection defined above.[3]

Definition 6 *The* multi-adjoint abelianized concept lattice *associated to a multi-adjoint frame $(L_1, L_2, P, \&_1, \ldots, \&_n)$ and context (A, B, R, σ), where L_1 and L_2 are L-connected, is the set*

$$\mathcal{M}_L = \{\langle g^*, f^* \rangle \mid \langle g^*, f^* \rangle \text{ is a concept}\}$$

in which the ordering is defined by $\langle g_1^*, f_1^* \rangle \preceq \langle g_2^*, f_2^* \rangle$ if and only if $g_1^* \preceq g_2^*$ (equivalently $f_2^* \preceq f_1^*$).

Note that as $(^{\uparrow_c}, ^{\downarrow^c})$ is a Galois connection, the pair $(\mathcal{M}_L, \preceq)$ is, indeed, a complete lattice [6].

In the rest of the section, we establish a comparison between the concept lattices $\mathcal{M}_L$ (defined above) and $\mathcal{M}$ (defined in [20]). Hence, we will fix a context (A, B, R, σ), a frame $(L_1, L_2, P, \&_1, \ldots, \&_n)$, where L_1 and L_2 are L-connected, and the corresponding multi-adjoint concept lattices $\mathcal{M}$ and $\mathcal{M}_L$.

Firstly we will prove, in the following result, that each concept $\langle g, f \rangle$ in $\mathcal{M}$ determines a concept in $\mathcal{M}_L$.

Proposition 2 *If $\langle g, f \rangle \in \mathcal{M}$, then the mappings $g^*\colon B \to L$, $f^*\colon A \to L$, defined as $g^* = i_2 \circ g$, $f^* = i_1 \circ f$, form a concept of the multi-adjoint concept lattice $\mathcal{M}_L$.*

Now, given a mapping $g\colon B \to L_2$, we have to possible ways to construct the smallest concept in $\mathcal{M}_L$ containing g:

- Considering the mapping $i_2 \circ g \in L^B$ and obtaining the corresponding concept in $\mathcal{M}_L$, that is, $\langle (i_2 \circ g)^{\uparrow_c\downarrow^c}, (i_2 \circ g)^{\uparrow_c} \rangle$.
- Obtaining the corresponding concept in $\mathcal{M}$ and, by Proposition 2, considering the concept $\langle i_2 \circ (g)^{\uparrow\downarrow}, i_2 \circ (g)^{\uparrow} \rangle$ in $\mathcal{M}_L$.

The following proposition states that the two constructions given above coincide.

Proposition 3 *Given a mapping $g\colon B \to L_2$, the concepts $\langle (i_2 \circ g)^{\uparrow_c\downarrow^c}, (i_2 \circ g)^{\uparrow_c} \rangle$ and $\langle i_2 \circ (g)^{\uparrow\downarrow}, i_2 \circ (g)^{\uparrow} \rangle$ coincide.*

[3] We include $*$ as a superscript in this new construction so that we can distinguish this new approach from that in [20]. Note that, in order to simplify the notation, references to σ have been omitted.

 ISBN 13: 978-84-613-5510-5

Similarly, we obtain a concept of $\mathcal{M}$ from each concept of $\mathcal{M}_L$, and the two possible construction of the smallest concept containing $g^*\colon B \to L$ coincide.

Proposition 4 *If $\langle g^*, f^*\rangle \in \mathcal{M}_L$, then the mappings $g\colon B \to L_2$, $f\colon A \to L_1$, defined as: $g = \phi_2 \circ g^*$, $f = \phi_1 \circ f^*$, form a concept of the multi-adjoint concept lattice $\mathcal{M}$. Moreover, given a mapping $g^*\colon B \to L$, the concepts $\langle (\phi_2 \circ g^*)^{\uparrow\downarrow}, (\phi_2 \circ g^*)^{\uparrow}\rangle$ and $\langle \phi_2 \circ (g^*)^{\uparrow_c\downarrow^c}, \phi_2 \circ (g^*)^{\uparrow_c}\rangle$ coincide.*

It is worth to take into account that the result above can be given analogously for any $f\colon A \to L_1$ as well.

Finally, as a consequence of the definition of L-connection and the above result, we have that the following theorem.

Theorem 1 *The mappings $\Phi\colon \mathcal{M}_L \to \mathcal{M}$ and $\mathcal{I}\colon \mathcal{M} \to \mathcal{M}_L$ defined, for each $\langle g, f\rangle \in \mathcal{M}$ and $\langle g^*, f^*\rangle \in \mathcal{M}_L$, as follows*

$$\begin{aligned} \Phi(\langle g^*, f^*\rangle) &= \langle \phi_2 \circ g^*, \phi_1 \circ f^*\rangle \\ \mathcal{I}(\langle g, f\rangle) &= \langle i_2 \circ g, i_1 \circ f\rangle \end{aligned}$$

are well-defined and $\Phi \circ \mathcal{I}\colon \mathcal{M} \to \mathcal{M}$ is the identity mapping. However, in general, $\mathcal{I} \circ \Phi\colon \mathcal{M}_L \to \mathcal{M}_L$ is not the identity mapping, but a closure operator.

Let $Fp(\mathcal{M}_L)$ be the subset of $\mathcal{M}_L$ consisting of all the fix-points of the $\mathcal{I} \circ \Phi$, i.e.

$$Fp(\mathcal{M}_L) = \{\langle g^*, f^*\rangle \in \mathcal{M}_L \mid \mathcal{I} \circ \Phi(\langle g^*, f^*\rangle) = \langle g^*, f^*\rangle\}$$

With this notation, the theorem above guarantees the following result:

Corollary 1 *The concept lattices $\mathcal{M}$ and $Fp(\mathcal{M}_L)$ are isomorphic.*

As a consequence of the previous isomorphism, several existing algorithms developed to obtain concept lattices where the conjunctors have the same carrier for both arguments can be applied; for instance, Lindig's algorithm [15], or its extension for graded attributes [5]. In order to obtain the concept lattice $\mathcal{M}$, we firstly use a fast algorithm to build the concept lattice $\mathcal{M}_L$ and then, compute the set $Fp(\mathcal{M}_L)$ of all fix-points of $\mathcal{I} \circ \Phi$, perhaps applying the algorithm once more. Finally, we apply Φ to obtain $\mathcal{M}$.

As the complexity of the algorithm used depends on the size of L, we should find, whenever possible, the least lattice L such that L_1 and L_2 are L-connected.

4 Conclusion

Usually, in formal concept analysis, the sets of attributes and objects are different, with different meaning and, hence, it might not make sense to evaluate them on the same carrier. In this context, the operators used to obtain the concept lattice could be defined considering different lattices associated to attributes and objects, see [20].

 ISBN 13: 978-84-613-5510-5

Anyway there exist several reasons for which we need to evaluate the set of attributes and objects in the same carrier. In this direction, a new concept lattice, where the objects and attributes are evaluated on the same lattice L, has been introduced, although operators which evaluate objects and attributes in different carriers are used.

Moreover, we have studied the relationship between the new concept lattice and the other one obtained directly considered different carriers to both set of attributes and objects, introduced in [20].

As future work, we want to study how the theory presented here can be applied to obtain t-concepts [11,19] when, originally, the set of attributes and objects are evaluated in different lattices. Another aim is to obtain mechanisms to find the least lattice L such that L_1 and L_2 are L-connected.

Acknowledgements

This work has been partially supported by Junta de Andalucía grant P09-FQM-5233, and by the EU (FEDER), and the Spanish Science and Education Ministry (MEC) under grant TIN2009-14562-C05-01 and TIN2009-14562-C05-03.

References

[1] A. Abdel-Hamid and N. Morsi. Associatively tied implicacions. *Fuzzy Sets and Systems*, 136(3):291–311, 2003.

[2] C. Alcalde, A. Burusco, R. Fuentes-González, and I. Zubia. Treatment of L-fuzzy contexts with absent values. *Information Sciences*, 179:1–15, 2009.

[3] R. Bělohlávek. Fuzzy concepts and conceptual structures: induced similarities. In *Joint Conference on Information Sciences*, pages 179–182, 1998.

[4] R. Bělohlávek. Concept lattices and order in fuzzy logic. *Annals of Pure and Applied Logic*, 128:277–298, 2004.

[5] R. Bělohlávek, B. D. Baets, J. Outrata, and V. Vychodil. Lindig?s algorithm for concept lattices over graded attributes. *Lecture Notes in Computer Science*, 4617:156–167, 2007.

[6] B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, second edition, 2002.

[7] P. du Boucher-Ryana and D. Bridge. Collaborative recommending using formal concept analysis. *Knowledge-Based Systems*, 19(5):309–315, 2006.

[8] S.-Q. Fan, W.-X. Zhang, and W. Xu. Fuzzy inference based on fuzzy concept lattice. *Fuzzy Sets and Systems*, 157(24):3177–3187, 2006.

[9] A. Formica. Ontology-based concept similarity in formal concept analysis. *Information Sciences*, 176(18):2624–2641, 2006.

[10] A. Formica. Concept similarity in formal concept analysis: An information content approach. *Knowledge-Based Systems*, 21(1):80–87, 2008.

[11] G. Georgescu and A. Popescu. Concept lattices and similarity in non-commutative fuzzy logic. *Fundamenta Informaticae*, 53(1):23–54, 2002.

 ISBN 13: 978-84-613-5510-5

[12] G. Jiang, K. Ogasawara, A. Endoh, and T. Sakurai. Context-based ontology building support in clinical domains using formal concept analysis. *International Journal of Medical Informatics*, 71(1):71–81, 2003.

[13] S. O. Kuznetsov. Complexity of learning in concept lattices from positive and negative examples. *Discrete Applied Mathematics*, 142:111–125, 2004.

[14] Y. Lei and M. Luo. Rough concept lattices and domains. *Annals of Pure and Applied Logic*, 2009. Article in press (http://dx.doi.org/10.1016/j.apal.2008.09.028).

[15] C. Lindig. Fast concept analysis. In S. G., editor, *Working with Conceptual Structures-Contributions to ICCS 2000*, pages 152–161, 2000.

[16] M. Liu, M. Shao, W. Zhang, and C. Wu. Reduction method for concept lattices based on rough set theory and its application. *Computers & Mathematics with Applications*, 53(9):1390–1410, 2007.

[17] X. Liu, W. Wang, T. Chai, and W. Liu. Approaches to the representations and logic operations of fuzzy concepts in the framework of axiomatic fuzzy set theory I. *Information Sciences*, 177(4):1007–1026, 2007.

[18] X. Liu, W. Wang, T. Chai, and W. Liu. Approaches to the representations and logic operations of fuzzy concepts in the framework of axiomatic fuzzy set theory II. *Information Sciences*, 177(4):1027–1045, 2007.

[19] J. Medina and M. Ojeda-Aciego. Multi-adjoint t-concept lattices. *Information Sciences*, 5(180):712–725, 2010.

[20] J. Medina, M. Ojeda-Aciego, and J. Ruiz-Calviño. Formal concept analysis via multi-adjoint concept lattices. *Fuzzy Sets and Systems*, 160(2):130–144, 2009.

[21] J. Medina, M. Ojeda-Aciego, A. Valverde, and P. Vojtáš. Towards biresiduated multi-adjoint logic programming. *Lect. Notes in Artificial Intelligence*, 3040:608–617, 2004.

[22] V. Phan-Luong. A framework for integrating information sources under lattice structure. *Information Fusion*, 9:278–292, 2008.

[23] K.-S. Qu and Y.-H. Zhai. Generating complete set of implications for formal contexts. *Knowledge-Based Systems*, 21:429–433, 2008.

[24] M.-W. Shao, M. Liu, and W.-X. Zhang. Set approximations in fuzzy formal concept analysis. *Fuzzy Sets and Systems*, 158(23):2627–2640, 2007.

[25] L. Wang and X. Liu. Concept analysis via rough set and afs algebra. *Information Sciences*, 178(21):4125–4137, 2008.

[26] X. Wang and W. Zhang. Relations of attribute reduction between object and property oriented concept lattices. *Knowledge-Based Systems*, 21(5):398–403, 2008.

[27] Q. Wu and Z. Liu. Real formal concept analysis based on grey-rough set theory. *Knowledge-Based Systems*, 22(1):38–45, 2009.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Computing the near interactions of the FMM for acoustic scattering using GPUs

J. Menéndez Canal[1], M. López Portugués[2], J.A. López Fernández[2], A. Rodríguez-Campa[1] and José Ranilla[1]

[1] *Departamento de Informática, Universidad de Oviedo, Spain*

[2] *Departamento de Ingeniería Eléctrica, Electrónica, de Computadores y de Sistemas, Universidad de Oviedo, Spain*

emails: uo189380@uniovi.es, mlopez@tsc.uniovi.es, lopezjesus@uniovi.es, rodriguezcalberto@uniovi.es, ranilla@uniovi.es

Abstract

In this work, we suggest some techniques to compute the near interactions of the Fast Multipole Method (FMM) applied to acoustic scattering problems using Graphical Processing Units (GPUs). In our implementation of the FMM, the calculation of the near interactions is the most computationally demanding process. In addition, our design of the near interactions step matches properly with the Single Instruction Multiple Threads paradigm. As a consequence, the mentioned process seems to be prone to run in GPUs.

Key words: FMM, GPU, Many-core

1 Introduction

There are a number of applications in which is fundamental to control the noise scattering. In fact, increasingly stringent requirements for environmental noise is a major design driver for new aircraft configurations [2]. In such cases, it is mandatory to find a proper model of the acoustical behaviour of the system under analysis.

Low-frequency methods, such as the *Boundary Elements Method* (BEM) [10], offer the potential of precise numerical formulations of acoustic scattering problems. However, these methods yield a linear system with N equations and a dense coupling matrix, whose direct solution has a time cost $\mathcal{O}\left(N^3\right)$ and a memory cost $\mathcal{O}\left(N^2\right)$. By means of iterative solvers, the time cost is reduced to $\mathcal{O}\left(N^2\right)$ per iteration.

The complexity, $\mathcal{O}\left(N^2\right)$ per iteration, of iterative solvers is due to the computation of a *Matrix-Vector Product* (MVP). The *Fast Multipole Method* (FMM) [7] and its multilevel version, known as *Multilevel Fast Multipole Algorithm* (MLFMA) [9], avoid

 ISBN 13: 978-84-613-5510-5

matrix explicit calculation yielding a dramatic reduction in the MVP time without significantly affecting to BEM's iterative solution accuracy. The FMM and the MLFMA reduce the cost per iteration to $\mathcal{O}\left(N^{1.5}\right)$ and to $\mathcal{O}\left(N\log\left(N\right)\right)$, respectively.

For the last years, the interest on many-core architectures, such as *Graphical Processing Units* (GPUs) [6], has been growing. These systems are specially suited in applications where there are at least as many running processes or independent threads as cores. Hence, Shared Memory programming paradigm can be considered the natural way of programming these systems. The GPUs offer a huge raw processing power while keeping a reduced energy usage yielding a higher performance per Watt than that of CPUs.

In this work, a heterogeneous parallel scattering solver is suggested. The GPU deals with the step of the FMM algorithm (near interactions) that best matches the *Single Instruction Multiple Threads* (SIMT) paradigm. We use the *Generalized Minimum Residual* (GMRES) method [8] due to its robustness for iteratively solving acoustic scattering problems [5].

2 FMM applied to acoustic scattering

The physical problem studied here consists in predicting the acoustic pressure on the space that surrounds a 3-D obstacle on which an incident acoustic wave is impinging. This problem may be posed in terms of the integral form of the Helmholtz equation [10] which is also known as *Conventional Boundary Integral Equation* (CBIE). In order to overcome the *non-uniqueness difficulty* [3] that appears in the CBIE at resonant frequencies, the CBIE is linearly combined with its normal derivative yielding the Burton and Miller equation [3]. By means of the BEM, the Burton and Miller equation may be discretised over S and formulated in terms of a linear system of equations.

In the sequel, the surface of the obstacle is considered to be discretised into N elements which means that the above-mentioned system of equations has N unknowns. In order to produce accurate results, S must be discretised into 6 to 10 elements per linear wavelength (λ). Since the wavelength and the frequency f are in inverse proportion, N increases, for a given obstacle, with the frequency squared $\left(f^2\right)$.

The FMM computes a *Matrix Vector Product* (MVP) without explicit calculation of the system matrix. The algorithm requires grouping the N elements of the problem into N_{g} groups. Then, it efficiently calculates *far interactions*, between non-neighboring groups, by means of the *Addition Theorems* and *plane wave decomposition* [1]. *Near interactions*, between elements that pertain to the same group or to neighboring groups, do not satisfy the conditions associated with the transformations described by the Addition Theorems [7]. As a consequence, near (or local) interactions must be computed directly evaluating its corresponding part of the system matrix.

 ISBN 13: 978-84-613-5510-5

3 Computing Near Interactions using GPUs

As it is shown in the previous section, groups size is critical to minimize the time to solution. On the one hand, if the group size is small then many groups are produced. This yields many far interactions and a high time cost. On the other hand, if the group size is big then there are many of groups should increase proportionally to $\sqrt{N}$ to achieve a $\mathcal{O}\left(N^{1.5}\right)$ time cost.

In order to achieve this aim, an accurate model is used to calculate the time cost of the iterations, taking into account both theoretical and empirical data. We use the following equation to compute the time cost for the near interactions:

$$T_{near}(N) \;\approx\; K_{near} \sum_{i=1}^{N_g} n_i \cdot n_i^{near}, \tag{1}$$

where N is the size of the problem (number of unknowns), N_g is the number of groups, n_i is the number i^{th} group, n_i^{near} is the number of elements in the neighbourhood of the i^{th} group, and K_{near} is a hardware dependent constant factor which can be estimated using empirical data. It is worth mentioning that n_i and n_i^{near} are efficiently obtained at runtime by means of oct-tree theory [4].

In addition, at runtime the algorithm can exhaustively test different group sizes and chooses the optimum group size -in the sense of minimizing the time cost- for an arbitrary problem solved in any CPU-GPU combination.

Since the number of groups and the number of elements per group are close to $\sqrt{N}$, near interactions require to calculate $\sqrt{N}$ matrices of size $\sqrt{N} \times \sqrt{N}$. That is, the calculation of the near interactions is the most demanding step, in terms of computational time, of the FMM. Hence, this step is prone to run in GPUs.

The computation of the near interactions involve complex computing with a low memory footprint. As a consequence, the size of the shared memory is not a limiting factor. In our implementation, all data needed to compute each matrix element are previously packed together. Therefore, threads have a regular access pattern that avoids possible conflicts between threads accessing to the banks of the shared memory. Since the calculation of each matrix element ($\sqrt{N}$ matrices of size $\sqrt{N} \times \sqrt{N}$) can be accomplished independently, threads may run asynchronously.

In this work *Compute Unified Device Architecture* (CUDA) is used to take advantage of NVIDIA[1] GPUs. It should be noted that FMM needs to use double precision to avoid the numerical instability of the method, thus, the new Fermi architecture is specially suitable. Our preliminary results, using two GeForce GTX 480 in SLI, are promising and now we are tuning some aspects related to the configuration of the new GTX 480.

[1] www.nvidia.com

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work has been partially supported by the "Ministerio de Ciencia e Innovación" from Spain and by FEDER, under the research projects TEC2008-01638/TEC (INVEMTA) and TIN2007-61273, and by infraestructure project EQP06-015 co-financed by the European Union. The Airbus A300 series geometry has been provided by the research project GRD1-2001-40147 financed by the European Union.

References

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, Dover Publications, 1972.

[2] Advisory Council for Aeronautics Research in Europe, *2008 Addendum to the Strategic Research Agenda*, available online at: www.acare4europe.org, 2008.

[3] A. J. Burton and G. F. Miller, *The Application of Integral Equation Methods to the Numerical Solution of Some Exterior Boundary-Value Problems*, Proc. of the Royal Society of London, **323(1553)**, (1971), 201–210.

[4] Nail A. Gumerov, Ramani Duraiswami and Eugene A.Borovikov, *Data Structures, Optimal Choice of Parameters, and Complexity Results for Generalized Multilevel Fast Multipole Methods in d Dimensions*, Institute for Advanced Computer Studies, 2003.

[5] S. Marburg and S. Schneider, *Performance of iterative solvers for acoustic problems. Part I. Solvers and effect of diagonal preconditioning*, Engineering Analysis with Boundary Elements, **27(7)**, (2003), 727–750.

[6] J. D. Owens, M. Houston, D. Luebke, S. Green, J. E. Stone and J. C. Phillips, *GPU Computing*, Proceedings of the IEEE, **96(5)**, (2008), 879–899.

[7] V. Rokhlin, *Diagonal Forms of Translation Operators for the Helmholtz Equation in Three Dimensions*, Applied and Computational Harmonic Analysis, **1(1)** (1993) 82–93.

[8] Y. Saad and M. H. Schultz, *GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems*, SIAM J. of Sci. and Statist. Comput., **7**, (1986) 856–869.

[9] J. Song and W. Chew, *Multilevel Fast-Multipole Algorithm for Solving Combined Field Integral Equations of Electromagnetic Scattering*, Microwave and Optical Technology Letters, **10(1)**, (1995), 14-19.

[10] T. W. Wu, *Boundary Element Acoustics*, Advances in Boundary Elements, WIT Press, 2000.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

PyPANCG: A Parallel Python Interface-Library for solving Mildly Nonlinear Systems

Héctor Migallón[1], Violeta Migallón[2] and José Penadés[2]

[1] *Departamento de Física y Arquitectura de Computadores, Universidad Miguel Hernández, 03202 Elche, Alicante, Spain*

[2] *Departamento de Ciencia de la Computación e Inteligencia Artificial, Universidad de Alicante, 03071 Alicante, Spain*

emails: hmigallon@umh.es, violeta@dccia.ua.es, jpenades@dccia.ua.es

Abstract

In this paper we present a parallel library, PyPANCG, treated as a high-level interface for solving nonlinear systems. This library consists of two modules, PySParNLCG and PySParNLPCG. The PySparNLCG module parallelizes the conjugate gradient method for solving mildly nonlinear system, and the PySParNLPCG module implements the preconditioning technique based on block two-stage methods. In order to create the high-level interfaces, we have chosen the Python language. On the other hand, the developed Fortran routines offer all the performance of the low-level language. Experimental results report the numerical accuracy and the parallel performance of our approach on different parallel computers.

Key words: parallel libraries, nonlinear algorithms, Python high-level interfaces

1 Introduction

The goal of this paper is to present PyPANCG (http://atc.umh.es/PyPANCG), a Python based high-level parallel interface-library for solving mildly nonlinear systems of the form

$$Ax = \Phi(x), \tag{1}$$

where $A \in \Re^{n\times n}$ and $\Phi : \Re^n \to \Re^n$ is a nonlinear diagonal mapping, i.e., the ith component ϕ_i of ϕ is a function only of the ith component x_i of x.

This library, distributed as a standard Python package, provides parallel implementations of both the nonlinear conjugate gradient method (NLCG) and the nonlinear preconditioned conjugate gradient method (NLPCG). PyPANCG can work with different tools to manage the parallel environment through *MPI* (www-unix.mcs.anl.gov/mpi), by using *PyMPI* or *mpipython* included in *Scientific Python* [4].

 ISBN 13: 978-84-613-5510-5

This paper is structured as follows. Section 2 introduces the nonlinear conjugate gradient method (NLCG) and the parallelization we have performed in the PySParNLCG module of PyPANCG. The nonlinear preconditioned conjugate gradient method and the parallelization performed in the PySParNLPCG module of PyPANCG are introduced in Section 3. In Sections 4, 5 and 6 we explain the main tools used to build PyPANCG, the involved parameters and different ways to implement the nonlinearity, respectively. In Section 7 some examples of using PyPANCG are reported while in Section 8 the behavior of this library is illustrated by means of numerical experiments. Finally, concluding remarks are presented in Section 9.

2 Nonlinear Conjugate Gradient Method

Consider the problem of solving the nonlinear system (1), where $A \in \Re^{n\times n}$ is a symmetric positive definite matrix. An effective approach to solve this nonlinear system is the Fletcher-Reeves version [3] of the nonlinear conjugate gradient method (NLCG). In order to describe the parallelization performed of this method, we consider that A is partitioned into $p \times p$ blocks, with square diagonal blocks of order n_j, $\sum_{j=1}^{p} n_j = n$, such that system (1) can be written as

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1p} \\ A_{21} & A_{22} & \cdots & A_{2p} \\ \vdots & \vdots & & \vdots \\ A_{p1} & A_{p2} & \cdots & A_{pp} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} = \begin{bmatrix} \Phi_1(x) \\ \Phi_2(x) \\ \vdots \\ \Phi_p(x) \end{bmatrix}, \tag{2}$$

where x and $\Phi(x)$ are partitioned according to the size of the blocks of A. Analogously, we consider $x^{(i)}$, $r^{(i)}$, $p^{(i)}$ and $\Phi(x^{(i)})$ partitioned according to the block structure of A in (2). With this notation we construct the following parallel algorithm.

Algorithm 1 *(Parallel Nonlinear Conjugate Gradient)*
Given an initial vector $x^{(0)}$
  In processor j, $j = 1, 2, \ldots, p$
    $r_j^{(0)} = \Phi_j(x^{(0)}) - [A_{j1}\ A_{j2} \cdots A_{jp}]x^{(0)}$
    $p_j^{(0)} = r_j^{(0)}$
  For $i = 0, 1, \ldots,$ *until convergence*
    In processor j, $j = 1, 2, \ldots, p$
      $\alpha_i =\rightarrow$ *see Algorithm 2*
      $x_j^{(i+1)} = x_j^{(i)} + \alpha_i p_j^{(i)}$
      $r_j^{(i+1)} = r_j^{(i)} - \Phi_j(x^{(i)}) + \Phi_j(x^{(i+1)}) - \alpha_i[A_{j1}\ A_{j2} \cdots A_{jp}]p^{(i)}$
    Convergence test
    In processor j, $j = 1, 2, \ldots, p$
      $\vartheta_j = \langle r_j^{(i+1)}, r_j^{(i+1)} \rangle$
      $\sigma_j = \langle r_j^{(i)}, r_j^{(i)} \rangle$
    Processor 1 computes and broadcasts $\beta_{i+1} = -\sum_{j=1}^{p} \vartheta_j / \sum_{j=1}^{p} \sigma_j$

 ISBN 13: 978-84-613-5510-5

In processor j, $j = 1, 2, \ldots, p$
Compute and perform an allgather $p_j^{(i+1)} = r_j^{(i+1)} - \beta_{i+1} p_j^{(i)}$

Note that, in Algorithm 1, α_i is obtained as follows:

Algorithm 2 *(Computing* α*)*
$\alpha_i^{(0)} = 0$
For $k = 0, 1, 2, \ldots,$ *until convergence*
$\delta^{(k)} = \frac{\alpha_i^{(k)} \left\langle Ap^{(i)}, p^{(i)} \right\rangle - \left\langle r^{(i)}, p^{(i)} \right\rangle + \left\langle \Phi(x^{(i)}) - \Phi(x^{(i)} + \alpha_i^{(k)} p^{(i)}), p^{(i)} \right\rangle}{\left\langle Ap^{(i)}, p^{(i)} \right\rangle - \left\langle \Phi'(x^{(i)} + \alpha_i^{(k)} p^{(i)}) p^{(i)}, p^{(i)} \right\rangle}$
$\alpha_i^{(k+1)} = \alpha_i^{(k)} - \delta^{(k)}$
Stopping criterion $\left(\left|\delta^{(k)}\right| < \zeta\right)$

3 Nonlinear Preconditioned Conjugate Gradient Method

Preconditioning is a technique for improving the condition number (cond) of a matrix. Suppose that M is a symmetric positive definite matrix that approximates A, but is easier to invert. We can solve $Ax = \Phi(x)$ indirectly by solving $M^{-1}Ax = M^{-1}\Phi(x)$. If $\text{cond}(M^{-1}A) << \text{cond}(A)$ we can iteratively solve $M^{-1}Ax = M^{-1}\Phi(x)$ more quickly than the original problem. In this case we obtain the following nonlinear preconditioned conjugate gradient algorithm.

Algorithm 3 *(Nonlinear Preconditioned Conjugate Gradient)*
Given an initial vector $x^{(0)}$
$r^{(0)} = \Phi(x^{(0)}) - Ax^{(0)}$
Solve $Ms^{(0)} = r^{(0)}$
$p^{(0)} = s^{(0)}$
For $i = 0, 1, \ldots,$ *until convergence*
$\alpha_i =\rightarrow$ *see Algorithm 2*
$x^{(i+1)} = x^{(i)} + \alpha_i p^{(i)}$
$r^{(i+1)} = r^{(i)} - \Phi(x^{(i)}) + \Phi(x^{(i+1)}) - \alpha_i Ap^{(i)}$
Solve $Ms^{(i+1)} = r^{(i+1)}$
Convergence test
$\beta_{i+1} = -\frac{\left\langle s^{(i+1)}, r^{(i+1)} \right\rangle}{\left\langle s^{(i)}, r^{(i)} \right\rangle}$
$p^{(i+1)} = r^{(i+1)} - \beta_{i+1} p^{(i)}$

Since the auxiliary system $Ms = r$ must be solved at each conjugate gradient iteration, this system needs to be easily solved. Moreover, in order to obtain an effective preconditioner, a good approximation M to the matrix A is needed. One of the general preconditioning techniques for solving linear systems [1] consists of considering a splitting of the matrix A as

$$A = P - Q \tag{3}$$

and performing m steps of the iterative procedure defined by this splitting toward the solution of $As = r$, choosing $s^{(0)} = 0$. In order to obtain the preconditioners suppose

 ISBN 13: 978-84-613-5510-5

that system (1) is partitioned as in (2). Let us consider the splitting (3) consists of the diagonal blocks of A in (2), that is $P = \mathrm{diag}(A_{11}, \ldots, A_{pp})$. Note that in this case, performing m steps of the iterative procedure defined by the splitting (3) to approximate the solution of $As = r$, corresponds to perform m steps of the Block Jacobi method. Thus, at each step l, $l = 1, 2, \ldots$, of a Block Jacobi method, p independent linear systems of the form

$$A_{jj} s_j^{(l)} = (Q s^{(l-1)} + r)_j, \quad 1 \leq j \leq p, \tag{4}$$

need to be solved; therefore each linear system (4) can be solved by a different processor. However, when the order of the diagonal blocks A_{jj}, $1 \leq j \leq p$, is large, it is natural to approximate their solutions by using an iterative method, and thus we are in the presence of a two-stage iterative method; see e.g., [6]. In a formal way, let us consider the splittings

$$A_{jj} = B_j - C_j, \quad 1 \leq j \leq p, \tag{5}$$

and at each lth step perform, for each j, $1 \leq j \leq p$, $q(j)$ iterations of the iterative procedure defined by the splittings (5) to approximate the solution of (4). That is, to solve the auxiliary system $Ms = r$ of Algorithm 3, we use m steps of the following algorithm, choosing $s^{(0)} = 0$.

Algorithm 4 *(Parallel Block Two-Stage)*
Given an initial vector $s^{(0)} = \left((s_1^{(0)})^T, (s_2^{(0)})^T, \ldots, (s_p^{(0)})^T\right)^T$, *and a sequence of numbers of inner iterations* $q(j)$, $1 \leq j \leq p$

For $l = 1, 2, \ldots,$ *until convergence*
  In processor j, $j = 1, 2, \ldots, p$
    $y_j^{(0)} = s_j^{(l)}$
    For $k = 1$ *to* $q(j)$
      $B_j y_j^{(k)} = C_j y_j^{(k-1)} + (Q s^{(l-1)} + r)_j$
    $s^{(l)} = \left((y_1^{(q(1))})^T, (y_2^{(q(2))})^T, \ldots, (y_p^{(q(p))})^T\right)^T$

Therefore, using similar notation as in Section 2, we construct the following parallel nonlinear algorithm.

Algorithm 5 *(Parallel Nonlinear Preconditioned Conjugate Gradient)*
Given an initial vector $x^{(0)}$

In processor j, $j = 1, 2, \ldots, p$
  $r_j^{(0)} = \Phi_j(x^{(0)}) - [A_{j1}\ A_{j2} \cdots A_{jp}] x^{(0)}$
Use m *steps of Alg. 4 to approximate* $As^{(0)} = r^{(0)}$
$p^{(0)} = s^{(0)}$
For $i = 0, 1, \ldots,$ *until convergence*
  In processor j, $j = 1, 2, \ldots, p$
    $\alpha_i =\rightarrow$ *see Algorithm 2*
    $x_j^{(i+1)} = x_j^{(i)} + \alpha_i p_j^{(i)}$
    $r_j^{(i+1)} = r_j^{(i)} - \Phi_j(x^{(i)}) + \Phi_j(x^{(i+1)}) - \alpha_i [A_{j1}\ A_{j2} \cdots A_{jp}] p^{(i)}$

 ISBN 13: 978-84-613-5510-5

Use m steps of Alg. 4 to approximate $As^{(i+1)} = r^{(i+1)}$
Convergence test
In processor $j,\ j = 1, 2, \ldots, p$
$\vartheta_j = \langle s_j^{(i+1)}, r_j^{(i+1)} \rangle$
$\sigma_j = \langle s_j^{(i)}, r_j^{(i)} \rangle$
Processor 1 computes and broadcasts $\beta_{i+1} = -\sum_{j=1}^{p} \vartheta_j / \sum_{j=1}^{p} \sigma_j$
In processor $j,\ j = 1, 2, \ldots, p$
Compute and perform an allgather $p_j^{(i+1)} = r_j^{(i+1)} - \beta_{i+1} p_j^{(i)}$

4 PyPANCG basic tools

This section analyzes the basic tools used in the developed library. The language used for the development of the basic routines and on which the final library will be based was Fortran. The desired objective is to unite the development features offered by Python in a single platform and to approach the execution features offered by, in this case, Fortran. To do this, equivalent routines were developed in both languages. In addition, mixed routines which work with both languages at different levels were developed.

In order to access the routines developed in Fortran from Python, the F2PY tool (cens.ioc.ee/projects/f2py2e) was used. To increase the possible parallel environments, the library has been developed to enable work with two of the most common tools, *mpipython*, which forms part of Scientific Python [4], and *pyMPI*.

Another very important aspect, both for communication between Python and Fortran and for performance, is the use and handling of arrays or vectors; here too, two equivalent options can be used. This is important with regard to the performance of the Python codes and is indispensable when it comes to communication between languages. For the manipulation of vectors, we can use *Numeric* or the *numarray* module included in *NumPy*. The use of one tool or the other is directly related to the tool used to manage the parallel environment. If *mpipython* is used, *Numeric* must be used; if *pyMPI* is chosen, *numarray* must be used instead.

We have developed four specific routines for each functionality. These routines were developed in pure Fortran, or in pure Python, or using two different mixed models. The basic routines have been grouped into operations for sparse matrices (based on SPARSKIT, www-users.cs.umn.edu/~saad/software/SPARSKIT/sparskit.html), basic operations between vectors (based on BLAS, www.netlib.org/blas), and specific functions for the methods at hand, which are associated with different steps of the NLCG and NLPCG algorithms.

5 PyPANCG parameters

This section deals with the parameters which have to be passed to the Python functions which solve a sparse nonlinear system using the NLCG or NLPCG method. The only indispensable parameters are the parameters of the system to be solved ($Ax = \phi(x)$),

 ISBN 13: 978-84-613-5510-5

which are the size of the system, the matrix A stored in CSR (Compressed Sparse Row) format, and the nonlinear mapping $\phi(x)$. In addition the derivative of $\phi(x)$ ($\phi'(x)$) is required for computing δ according to Algorithm 2. However, there is a series of parameters that permits the modification of these algorithms. If values are not specified, default values are used. The optional parameters used in both algorithms and their default values are as follows:

- *initial_vector*: Initial iterate equal to zero.
- *global_stopping_error* $\xi = 10^{-7}$: Global stopping criterion evaluated using the euclidean norm of the residual vector ($\|r\|_2$).
- *alfa_stopping_error* $\zeta = 10^{-7}$: Stopping criterion for computing α evaluated using the absolute value of δ.
- *iter_alfa* $= 0$: By setting this parameter to a value higher than 1, we can limit the number of iterations performed to calculate α.
- *For_or_Py* = "Python_full": It selects one of the four different sets of routines to be used during the algorithm execution. As it has been mentioned above, these routines differ in the coding language.
- *trash_int*: Integer vector (see Section 6).
- *trash_double*: Double precision vector (see Section 6).

The NLPCG specific parameters and their default values are:

- *level* $= 1$: Level of fill-in of the incomplete LU factorization used in Algorithm 4 in order to obtain the inner splittings (5) (see Section 8).
- *niter_2e* $= 3$: Number of steps m performed by Algorithm 4 to approximate the corresponding linear system in Algorithm 5.
- *val_q* $= 3$: Number of inner iterations $q(j), 1 \leq j \leq p$ performed in Algorithm 4.

Another important parameter that the system can calculate -if the matrix is available in the *root* processor- is the size of the problem assigned to each processor; this is given by the parameter *block_dimensions*. This parameter is an integer vector whose dimension corresponds to the number of processors and which stores the block size assigned to each processor. In the examples provided by PyPANCG, the parameter is internally calculated, such that a load balancing is achieved. If the matrix is distributed among processors, this parameter must specify the portion available at each processor.

The parameter *For_or_Py* selects the set of routines to be used. The following options can be chosen with regard to this parameter:

1. *Python_full*: All of the routines used are codified in Python.

 ISBN 13: 978-84-613-5510-5

2. *Python*: The routines used are codified in Python but the functions that come from SPARSKIT and BLAS are in Fortran.

3. *Fortran*: All of the routines used are codified in Fortran. Moreover ϕ and ϕ' are codified independently.

4. *Fortran_full*: All of the routines used are codified in Fortran but ϕ and ϕ' are not codified independently.

The options are listed in performance order from poorest to best and in usability and development speed order from best to poorest. It is worth pointing out that the *Python* option is a mixed option whereas the rest of the options use either Python or Fortran for the basic routines. The difference between *Fortran* and *Fortran_full* is that in the first option the user must only codify the functions ϕ and ϕ' in Fortran, whereas in the latter option all routines implementing these functions must be codified in Fortran, and thus the user must understand the internal development of the method in great depth.

6 Nonlinearity implementation

One of the major obstacles to develop libraries for solving nonlinear systems is the implementation of the nonlinearity of the problem to be solved. One important aspect is that the ith component ϕ_i of ϕ only depends on the ith component of x. Thus, ϕ and ϕ' can be developed at vector level or at vector component level. For performance reasons, development will take place at vector level if the development is realized in Python and, for usability reasons, it will take place at component level when Fortran language is used. The example below shows the Python code for the function $\phi(x)$ used in the examples of PyPANCG.

```
def Fi_x(vector,trash_int,trash_double):
    sc = trash_double[0]
    x = -sc*numpy.exp(vector)
    return x
```

The same function developed in Fortran is:

```
double precision function phi(input,trash_int,trash_double)
    implicit none
    real*8 input,trash_double(*),sc
    integer trash_int(*)
    sc = trash_double(1)
    phi = -sc*exp(input)
    return
```

In addition to observing that the Python code works using vectors whilst Fortran works using a single component, it is important to note that both functions require a parameter transfer (sc) for the computation of ϕ. To realize this transfer -both real values and integer values if needed- we use two vectors, one integer vector *trash_int* and

 ISBN 13: 978-84-613-5510-5

one double precision real vector *trash_double*. These vectors are dynamic and thus all parameters required for the computation can be passed to functions ϕ and ϕ'. Naturally, these functions must always be implemented in order to adapt to the problem to be solved. If they are implemented in Python, the option *Python* or *Python_full* must be used. If they are implemented in Fortran, the option *Fortran* or *Fortran_full* must be used. Moreover, in the latter case, the module must be installed and compiled again following the development of the functions.

The options *Python* and *Fortran* are very similar; both use basic functions in Fortran but differ in their implementation of the functions ϕ and ϕ'. The option *Fortran_full* does not use these functions except for integrating them in the routines that use these functions. Thus, its adaptation is more complicated and laborious. However, it is the option that provides the best performance. On the other hand, *Python_full* option does not use any Fortran code, which enables much faster development but an excessively poor performance.

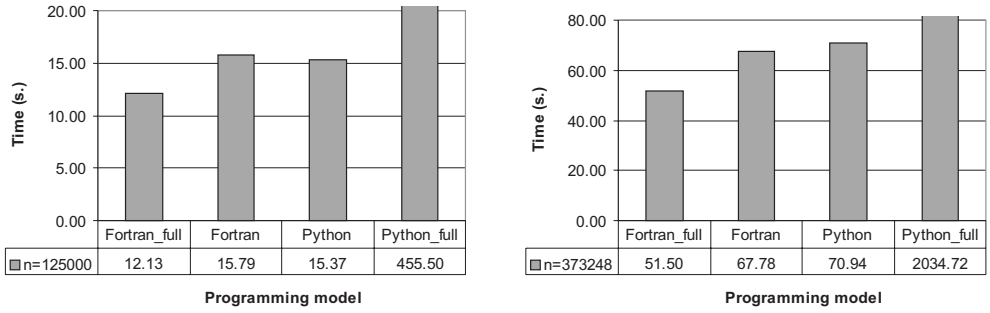
7 Using PySParNLCG and PySParNLPCG

As already mentioned, in order to use the library the size of the system (nrow), the matrix A in CSR format (tcol, trow, tval), the block size assigned to each processor (block_dimensions), and the nonlinear functions (ϕ and ϕ') must be passed at the very least. However, if we wish to pass additional parameters we will use the variables *trash_int* and *trash_double*. The following code shows the most simple NLCG function call, in which we assume that the functions ϕ and ϕ' were implemented in Python beforehand.

```
1  from math import exp
2  import numpy
3  import PyPANCG
4  import PyPANCG.PySParNLCG as PySparNLCG

5  iam = PySParNLCG.iam
6  trash_double = numpy.zeros(((1),),float)
7  trash_double[0] = 6/(float(49)**3)
8  nrow = 125000

9  nrow,block_dimensions,bls = _
       PyPANCG.MakeBlockStructure(nrow=nrow)
10 nnz,tcol,trow,tval = PyPANCG.PartialMatrixA _
       (Mx=Mx,s=bls[iam],d=block_dimensions[iam])

11 x,error,time,iter = PySParNLCG.nlcg(nrow=nrow, _
                tcol=tcol,trow=trow,tval=tval, _
                block_dimensions = block_dimensions, _
                Fi_x=Fi_x,Fi_prime_x=Fi_prime_x, _
                trash_double = trash_double)
```

The matrix A is obtained in lines 9 and 10; this code is enclosed with the library but can only be used as an example or test. It is important to point out that each

 ISBN 13: 978-84-613-5510-5



(a) $n = 125000$ (b) $n = 373248$

Figure 1: PySParNLCG using 2 processors, pyMPI, SULLI.

processor only contains the portion of the matrix that it requires. In line 11, the actual call to the NLCG method takes place, whereby we assume that *Fi_x* (ϕ) and *Fi_prime_x* (ϕ') were declared in Python and the vector *trash_double* is passed, in this case of a single component.

The most simple NLPCG function call is similar to the NLCG example above showed. In this case it must import PySParNLPCG module instead of PySParNLCG module in line 4,

```
4  import PyPANCG.PySParNLPCG as PySparNLPCG
```

and it must call NLPCG method in line 11,

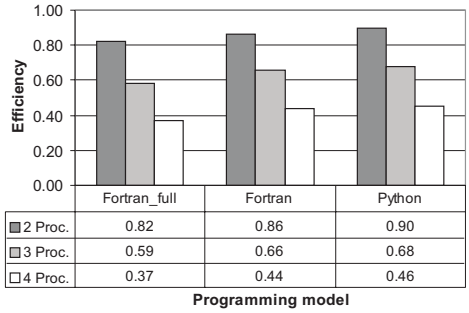
```
11 x,error,time,iter = PySParNLPCG.nlpcg(nrow=nrow, _
              tcol=tcol,trow=trow,tval=tval, _
              block_dimensions = block_dimensions, _
              Fi_x=Fi_x,Fi_prime_x=Fi_prime_x, _
              trash_double = trash_double)
```

8 Numerical experiments

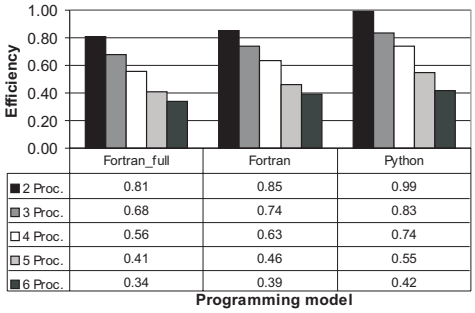
In order to illustrate the behavior of PyPANCG, we have tested the algorithms provided by this library on two multicore computers. The first platform, Bi-Quad, is a DELL PowerEdge 2900 with two Quad-Core Intel Xeon 5320 sequence processors at up to 1.86 GHz, with 8 GB of RAM. The second platform, SULLI, is an Intel Core 2 Quad Q6600, 2.4 GHz, with 4 GB of RAM.

As our illustrative example we have considered a nonlinear elliptic partial differential equation, known as the Bratu problem [2]. To solve this problem using the finite difference method, we consider a grid in Ω of d^3 nodes, where Ω is a 3D cube domain of unit length. The discretization of this problem yields a nonlinear system of the form

 ISBN 13: 978-84-613-5510-5



(a) SULLI



(b) Bi-Quad

Figure 2: Efficiency of PySParNLCG, $n = 373248$, pyMPI.

$Ax = \Phi(x)$, where $\Phi : \Re^n \to \Re^n$ is a nonlinear diagonal mapping. We present here results obtained with $d = 50$, $d = 72$ and $d = 84$, that lead to nonlinear systems of size 125000, 373248 and 592704, respectively. The convergence test used was $\|r\|_2 < 10^{-7}$ and the stopping criterion for α was $|\delta| < 10^{-7}$. Concretely, these are the default values for *global_stopping_error* and *alfa_stopping_error* in PyPANCG.

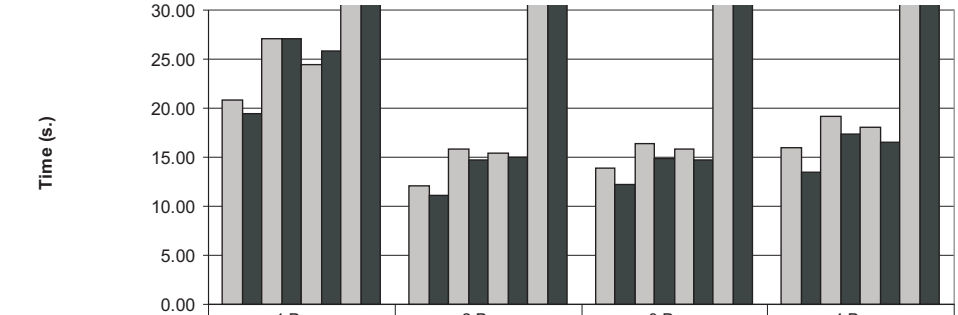
First, we analyze the behavior of PyPANCG.PySParNLCG depending on different values of the parameter *For_or_Py*. Figure 1 shows that the best results are obtained using routines fully developed in Fortran such that the computation of ϕ and ϕ' is performed inside these routines. The worst results are obtained with the option *For_or_Py='Python_full'*. Note that this option uses pure Python routines and it should only be used in the development process. On the other hand, the options that combine Fortran and Python code get similar performance.

Figure 2 analyzes the influence of the number of processors, on the two multicore platforms above mentioned. As it can be seen, the best efficiencies are obtained using 2 or 3 processors in SULLI, or a maximum of 5 processors in Bi-Quad.

Figure 3 compares the use of *mpipython* and *Numeric* with the use of *pyMPI* and *numpy* in the behavior of PyPANCG.PySParNLCG. As it can be seen, *Numeric* offers better performance than *numpy*, specially when the option *For_or_Py='Python_full'* is used. Therefore, a calling to a module with a single processor always uses *Numeric*.

In order to analyze the PyPANCG.PySParNLPCG module we consider, in our experiments, the outer splitting $A = P - Q$ determined by $P = \text{diag}(A_{11}, \ldots, A_{pp})$. Let us further consider an incomplete LU factorization of each matrix A_{jj}, $j = 1, 2, \ldots, p$, that is $A_{jj} = L_j U_j - R_j$, and at each *lth* step perform, for each j, $q(j)$ inner iterations of the iterative procedure defined by this splitting. Let us denote by ILU(S) the incomplete LU factorization associated with the zero pattern subset S of $S_n = \{(i, j) : i \neq j,\ 1 \leq i, j \leq n\}$. In particular, when $S = \{(i, j) :\ a_{ij} = 0\}$, the incomplete factorization with zero fill-in, known as ILU(0), is obtained. To improve the quality

 ISBN 13: 978-84-613-5510-5



	1 Proc.	2 Proc.	3 Proc.	4 Proc.
Fortran_full - pyMPI	20.85	12.13	13.90	16.01
Fortran_full - mpipython	19.48	11.16	12.27	13.45
Fortran - pyMPI	27.05	15.79	16.35	19.18
Fortran - mpipython	27.04	14.76	14.83	17.34
Python - pyMPI	24.42	15.37	15.84	18.10
Python - mpipython	25.85	15.05	14.75	16.55
Python_full - pyMPI	897.59	455.50	293.39	229.87
Python_full - mpipython	276.85	143.93	104.37	86.54

Number of processors

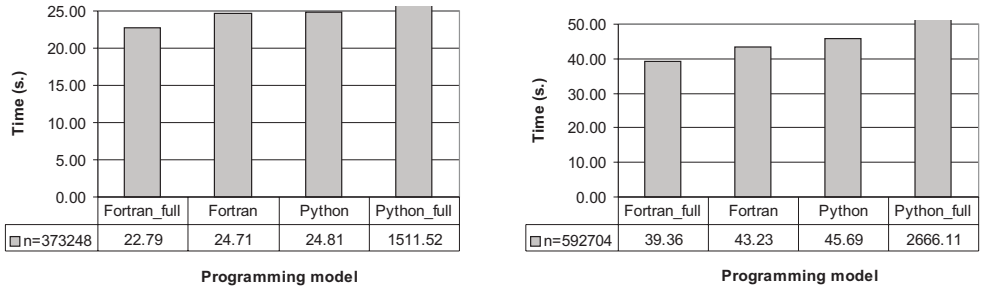
Figure 3: PySParNLCG: *mpipython* versus *pyMPI*, $n = 125000$, SULLI.

of the factorization, many strategies for altering the pattern have been proposed. In the experiments reported here, we have used the "level of fill-in" factorizations [5], ILU(κ), $\kappa \geq 0$. Figure 4 illustrates the behavior of PySParNLPCG depending on the different options of *For_or_Py*. Similar performances to those for PySParNLCG module are obtained. That is, the best results are obtained setting *For_or_Py='Fortran_full'* and the worst results using *For_or_Py='Python_full'*. The other two options present similar performance.

9 Conclusion

In this paper we have presented PyPANCG, a Python library-interface that implements both the conjugate gradient method and the preconditioned conjugate gradient method for solving nonlinear systems. We have described the use of the library and its advantages in order to get fast development. The aim of this library is to develop high performance scientific codes for high-end computers hiding many of the underlying low-level programming complexities from users with the use of a high-level Python interface. The library has been designed for adapting to different stages of the design process, depending on whether the purpose is computational performance or fast development.

 ISBN 13: 978-84-613-5510-5



(a) $n = 373248$ (b) $n = 592704$

Figure 4: PySParNLPCG using 2 processors, $\kappa = 1$, mpipython, SULLI.

Acknowledgements

This research was supported by the Spanish Ministry of Science and Innovation under grant number TIN2008-06570-C04-04.

References

[1] L. Adams, *M-step preconditioned conjugate gradient methods*, SIAM Journal on Scientific and Statistical Computing **6** (1985) 452–462.

[2] B. M. Averick, R. G. Carter, J. J. More and G. Xue, *The MINPACK-2 Test Problem Collection*, Technical Report MCS-P153-0692, Mathematics and Computer Science Division, Argonne, 1992.

[3] R. Fletcher and C. Reeves, *Function Minimization by Conjugate Gradients*, The Computer Journal **7** (1964) 149–154.

[4] K. Hinsen, *Scientific Python User's Guide*, Centre de Biophysique Moleculaire CNRS, Grenoble, France, 2002.

[5] H. P. Langtangen, *Conjugate gradient methods and ILU preconditioning of non-symmetric matrix systems with arbitrary sparsity patterns*, International Journal for Numerical Methods in Fluids **9** (1089) 213–233.

[6] V. Migallón and J. Penadés, *Convergence of two-stage iterative methods for hermitian positive definite matrices*, Applied Mathematics Letters **10(3)** (1997) 79–83.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A front-end for theorem proving with modal logics

Angel Mora[1], Emilio Muñoz-Velasco[1], Joanna Golińska-Pilarek[2] and Sergio Martín[1]

[1] *E.T.S.I. Informática, University of Málaga. Spain*

[2] *National Institute of Telecommunications, Warsaw, Poland.*
Institute of Philosophy. University of Warsaw . Poland

emails: amora@ctima.uma.es, emilio@ctima.uma.es, j.golinska@uw.edu.pl, animasergio@gmail.com

Abstract

We present a front-end for theorem provers developed in Prolog. The framework is checked on three modal logics: the standard modal logic K and two logics for order-or-magnitude qualitative reasoning. A general weakness of automated theorem provers, specially for modal logics, is the difficulty to communicate and interact with the user. We try to give a first step in this line, by providing a user-friendly environment which could be very useful both for research and educational applications, that is, as a tool for researchers and for teaching and learning proof theory.

Key words: theorem proving, implementation algorithms, modal logic, qualitative reasoning

1 Introduction

Automated reasoning is concerned with providing an algorithmic description to a formal calculus so that it can be implemented on a computer to prove theorems in an efficient manner. The problem of determining satisfiability and validity of logic formulas has received much attention by the automated reasoning community due to its important applicability in industry [18]. In particular, modal logics have many applications in Computer Science, for example in artificial intelligence, database theory, distributed systems, program verification, cryptography theory. We focus in this work on the description of a user-friendly front-end for theorem provers in modal logics. To begin with, we consider the standard modal logic K, together with two modal logics for order-of-magnitude qualitative reasoning. This kind of reasoning tries to deal with situations where the quantitative information is not available or, as humans do in many situations,

 ISBN 13: 978-84-613-5510-5

it is better to reason in a qualitative way. A form of qualitative reasoning is order-of-magnitude-reasoning, where the quantitative information is substituted by a finite number of qualitative classes and some relations between the qualitative classes, such as *negligibility, closeness,...* among others, are defined [19, 21]. Recent applications of order of magnitude reasoning can be seen, for example, in [4, 16], and multimodal logics for order-of-magnitude-reasoning have been presented in [6, 7].

The system presented in this paper is based on relational dual tableaux which are validity checkers [10, 11]. They are extensions of Rasiowa-Sikorski diagrams for first-order logic [20]. Relational dual tableaux are powerful tools for performing the major reasoning tasks and have many advantages as their modularity and their easy way to be implemented.

The details of the implementation of the theorem provers tested with the front-end can be seen in [5, 9, 15]. A natural improvement to these previous works is to enhance the interaction with the user during the proof process and specifically, the improvement of the graphical aspect of the interface. The aim is to develop a user-friendly interface with enough flexibility to be able to deal with this type of logics.

Many provers have been designed which can deal with modal logics. Very optimised ones like MSPASS [14] and FaCT [13]; generic logical frameworks like Isabelle [17]; and other approaches offer users the possibility to create a new prover, like LWB [12], LoTReC [8] LeanTAP [3] and TWB [1]. As far as we know, the provers presented in the literature are powerful but the mechanism of interaction with the user is poor. As stated in [2], although efficiency is an important aspect, depending on the intended application, other qualities can be important, such as portability, construction of counter-models, user-friendliness, or small size. In this line, our provers have been developed in Prolog and try to take advantage of the powerful capabilities of this language: fast prototyped, modular, and extensible to other modal logics.

In this paper, we present a front-end proving environment for three theorem provers we have developed in Prolog. These three logics are the modal logic K and two multi-modal logics for order-of-magnitude reasoning. This front-end can be easily extended with other theorem provers designed in Prolog. We believe this system could be useful for both research and educational applications, that is, as a tool for researchers and for teaching and learning proof theory. For example, by using its trace mode, it explains step by step the full process of the proof, indeed, it is very intuitive to see in every step which rule has been applied and how it works. We emphasize that the user can introduce friendly a formula for the logic considered using the virtual keyboard. Then, the prover tries to prove the formula and renders the result to the front-end. The environment collects the information and the user can analyze the proof process.

2 The front-end for modal logics

We now present a front-end for the provers cited above. All these provers are much easier to be understood and adapted, because they have been developed in Prolog and they take advantage of the powerful capabilities of this language.

 ISBN 13: 978-84-613-5510-5

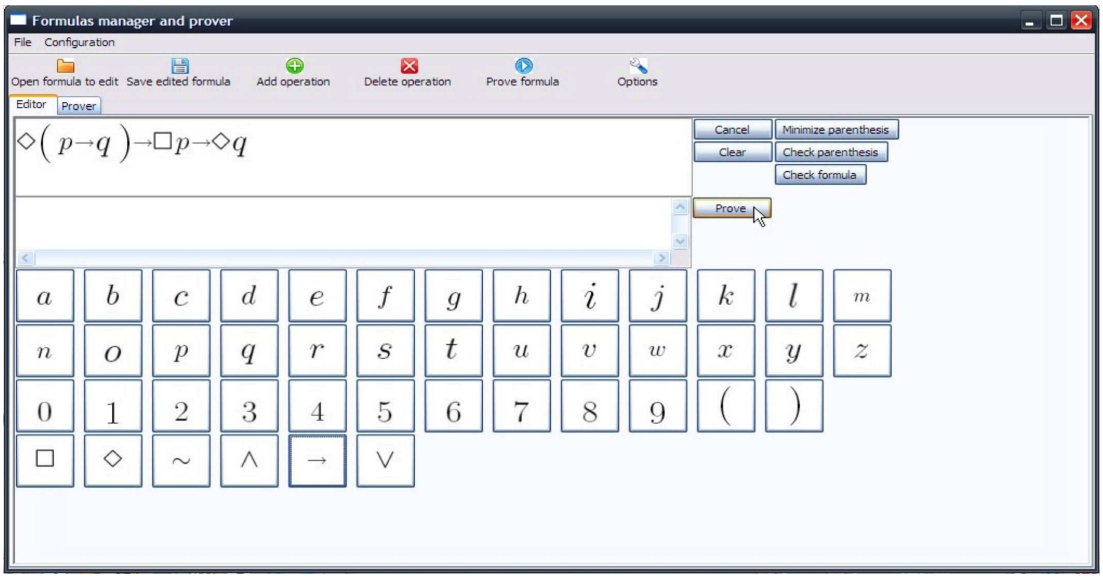


Figure 1: The front-end for modal logic K

Firstly, we select the prover and configure the path in the *Configuration menu*. Figure 1 shows the interface for proving formulas using the prover developed for the modal logic K.

The front-end has the capability of open-edit-save formulas using this intuitive user interface and can be extended easily to other languages. This keyboard can be modified by the user: adding-removing buttons and connecting the new button with the corresponding Prolog predicate used in the prover.

Formulas are represented by trees and the translation to other formats is possible by reading the tree in different ways. For instance, a translation to or from Latex is possible due to the easy representation used for the front-end.

The front-end connects with Prolog and the prover tries to satisfy the formula edited using the keyboard or opened from a file. Finally, it collects and summarizes the results obtained by the prover.

In Figure 2, the prover renders the results of the automatic application of the rules described on the specific prover.

In the left area, the front-end presents the time used in the proof. Below the time, the variables used for the proof are shown. This information is important for the researcher. We can see that this formula is *valid* (the prover has closed all the leaves of the tree). Finally, on this area, the user can see the order of the rules applied in order to prove the formula.

The front-end summarizes the work of the prover and presents in the `used_rules` predicate of Prolog a trace of the proof process. This mechanism of explanation may provide an important educational function. In Figure 2, the user can see the order of application of the rules: *union* rule on leaf *[1]* applied to a formula that appears later,

 ISBN 13: 978-84-613-5510-5

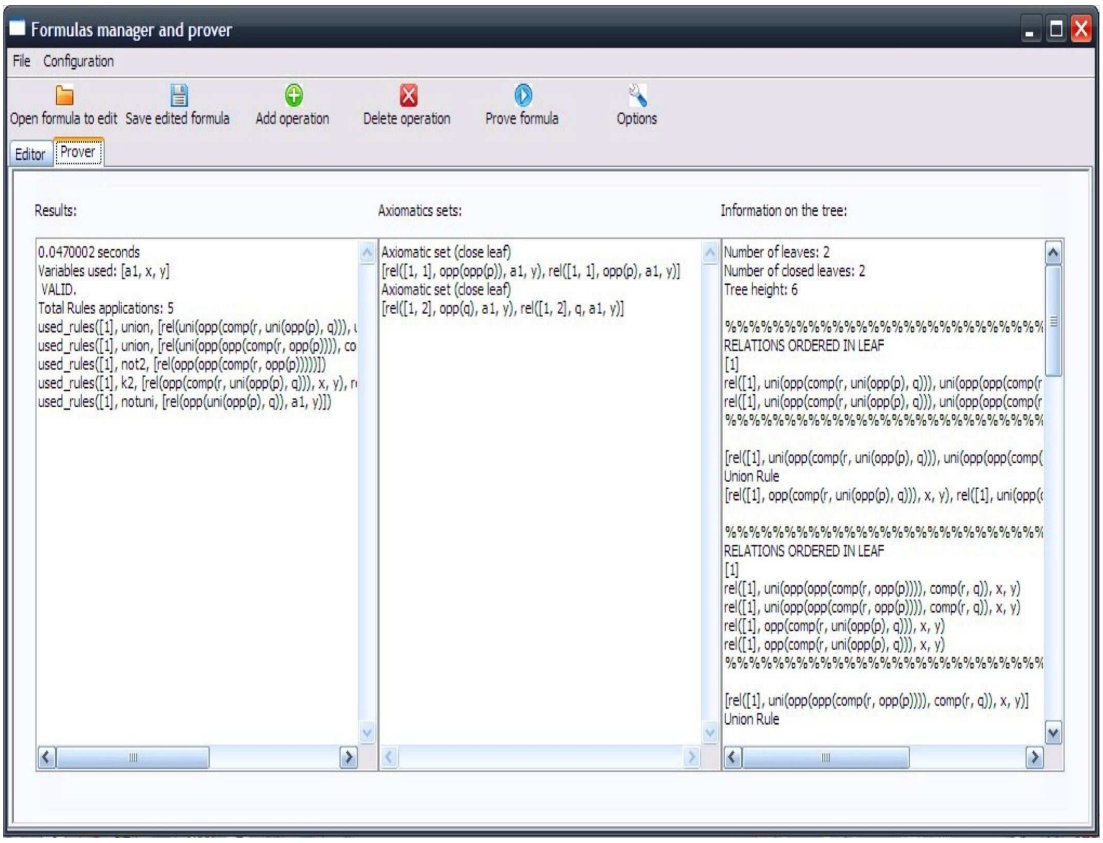


Figure 2: The front-end presents the results

then *union* rule on leaf *[1]* applied to other formula, then *not2* rule on leaf *[1]* applied to other formula, etc.

To improve the explanation, we can see in the central area the axiomatic sets found in order to close the leaves. For instance, in Figure 2 we can see that the prover has applied the *k2 rule* that divides a node in two sub-leaves. In the leaves `[1,1]` and `[1,2]` (successors from `[1]`) the prover detects two axiomatic sets and then these leaves are closed.

Moreover, in the right area, the front-end summarizes the information about the number of leaves of the tree, the number of closed leaves, the tree height and the rules applied. Below this information, the prover shows step by step how the rules have been applied, and presents the formulas involved in each application. This is the part of the front-end that we consider more useful for educational applications.

Finally, the result of the proof process can be saved in a log file from the front-end. The tool can be used for other similar provers developed in Prolog. We have tested the front-end those provers presented in [5, 9, 15]. We have extended the virtual keyboard in order to be used for the three provers in an easy way.

 ISBN 13: 978-84-613-5510-5

The front-end is extensible and we can add a new prover to the front-end indicating, in the configuration menu, the path of the prover. Moreover, the virtual keyboard can be adapted to manage the new logic and new symbols could be added.

3 Conclusions and future work

We have presented a front-end for some implementations of modal logic provers developed in Prolog. A general weakness of automated theorem provers, specially for modal logics, is the difficulty to communicate and interact with the user. We show a flexible front-end with capabilities of adaption to other similar provers developed in Prolog. The front-end allows us to introduce the formulas either from a virtual keyboard or from a file. In the interface we have the typical functions to manage the edited formulas: open, save, etc. Finally, the results of the proof process are shown and the user is informed about the applied rules, the axiomatic sets found in order to close the leaves of the tree, the number of leaves of the tree, the height of the tree, etc.

As extension of this work, we are improving the front-end with graphic capabilities for a better interaction with the user. The proof process will be showed in a tree and the user could analyze the application of the rules in a friendly way. Moreover, an automatic generator of formulas and a translation mechanism of our formulas to the typical format used for other provers is being implemented.

Acknowledgements

This work has been partially supported by Spanish projects TIN2009-14562-C05-01 and P09-FQM-5233.

References

[1] P. Abate and R Goré. The tableau workbench. *Electronic Notes in Theoretical Computer Science*, 231(55–67), 2009.

[2] Peter Balsiger, Alain Heuerding, and Stefan Schwendimann. A benchmark method for the propositional modal logics K, KT, S4. *Journal of Automated Reasoning*, 24(3):297–317, 2000.

[3] B. Beckert and J. Posegga. leanTAP: Lean, tableau-based deduction. *Journal of Automated Reasoning*, 15(3):339–358, 1995.

[4] B. Bredeweg and P. Struss. Current topics in qualitative reasoning. *AI Magazine*, 24(4):13, 2003.

[5] A. Burrieza, A. Mora, M. Ojeda-Aciego, and E Orłowska. An implementation of a dual tableaux system for order-of-magnitude qualitative reasoning. *International Journal on Computer Mathematics*, 86:1852–1866, 2009.

[6] A. Burrieza, E. Muñoz Velasco, and M Ojeda-Aciego. A logic for order of magnitude reasoning with negligibility, non-closeness and distance. *Lecture Notes in Artificial Intelligence*, 4788:210–219, 2007.

 ISBN 13: 978-84-613-5510-5

[7] A. Burrieza and M. Ojeda-Aciego. A multimodal logic approach to order of magnitude qualitative reasoning with comparability and negligibility relations. *Fundamenta Informaticae*, 68:21–46, 2005.

[8] Olivier Gasquet, Andreas Herzig, Dominique Longin, and Mohamad Sahade. Lotrec: Logical tableaux research engineering companion. *Lecture Notes in Artificial Intelligence*, 3702:318–322, 2005.

[9] J. Golińska-Pilarek, A. Mora, and E Muñoz Velasco. An ATP of a relational proof system for order of magnitude reasoning with negligibility, non-closeness and distance. *Lecture Notes in Artificial Intelligence*, 5351:128–139, 2008.

[10] J. Golińska-Pilarek and E. Muñoz Velasco. Relational approach for a logic for order of magnitude qualitative reasoning with negligibility, non-closeness and distance. *Logic Journal of IGPL*, 17(4):375–394, 2009.

[11] J. Golińska-Pilarek and E. Orłowska. Relational logics and their applications. *Lecture Notes in Artificial Intelligence*, 4342:125–161, 2006.

[12] A. Heuerding. LWBtheory: information about some propositional logics via the WWW. *Logic Journal of IGPL*, 5(1):169, 1997.

[13] I. Horrocks. The fact system. *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 307–312, 1998.

[14] U. Hustadt and R. Schmidt. MSPASS: Modal reasoning by translation and first-order resolution. *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 67–71.

[15] A. Mora, E. Muñoz Velasco, and J. Golińska-Pilarek. Implementing a relational theorem prover for modal logic K. *International Journal on Computer Mathematics*, to appear 2010.

[16] S. Parsons. Qualitative probability and order of magnitude reasoning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 11(3):373–390, 2003.

[17] L. C. Paulson. Isabelle: The next 700 theorem provers. *ArXiv Computer Science e-prints,*, October 1993.

[18] F. Portoraro. Automated reasoning. *Stanford Encyclopedia of Philosophy*, `http://plato.stanford.edu/entries/reasoning-automated/`, 2010.

[19] O. Raiman. Order of magnitude reasoning. *Artificial Intelligence*, 51:11–38, 1991.

[20] H. Rasiowa and R. Sikorski. *The Mathematics of Metamathematics*. Polish Scientific Publishers, 1963.

[21] L. Travé-Massuyès, F. Prats, M. Sánchez, and N. Agell. Relative and absolute order-of-magnitude models unified. *Annals of Math. and AI*, 45:323–341, 2005.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
26–30 June 2010.

Current-voltage characteristics for unipolar organic diodes with simultaneous carrier density and electric field dependent mobility

Luís Morgado[1], Luís Alcácer[2] and Jorge Morgado[2]

[1] *Dep. Física-ECT, Univ. de Trás-os-Montes e Alto Douro, Apartado 1013, 5001-801 Vila Real, Portugal*

[2] *Intituto de Telecomunicações, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*

emails: lmorgado@utad, alcacer@lx.it.pt, jorge.morgado@lx.it.pt

Abstract

Expressions for unipolar charge transport in single layer organic diode with field and carrier density dependent mobility are obtained. These relations, when combined, give the exact $J-V$ characteristic curve, parametrically in the value of the electric field at the ejecting electrode.

Key words: organic semiconductor devices, mobility

1 Introduction

The charge transport in doped polymers has been attracting physicists attention for many years now. For instance H. Bassler has intensely addressed the charge transport in "inert" polymer matrices doped with active molecules. Of particular relevance was the report in 1990 [J.H. Burroughes, D.D.C. Bradley, A.R. Brown, R.N. Marks, K. Mackay, R.H. Friend, P.L. Burn and A.B. Holmes, Nature 347 (1990), p. 539.] of electroluminescence observation in conjugated luminescent polymers. Besides bipolar charge transport, required to create the luminescent excited states, charge injection is the other phenomenon determining the current flow across such polymeric light-emitting devices. In view of the many parameters affecting charge injection and transport in such devices a strong effort has been put in developing models taking into account as many of such parameters as possible. In view of the huge task, and the complexity of these systems, the modelling of unipolar charge injection and transport is a very helpful approach. In this communication, we address the problem of unipolar charge transport in an organic diode, which, in the simplest single-layer type of structure, consists on an organic semiconductor, either a low molecular weight material or a polymer, in between a transparent anode, usually indium-tin oxide, ITO, and a metallic cathode.

 ISBN 13: 978-84-613-5510-5

2 Model

The starting point for this study is the drift equation for the constant hole density current (which means that under applied voltage, the electron current is negligible)[1]

$$J = qp\left(x\right)\mu\left(x\right)E\left(x\right), \tag{1}$$

and the Poisson equation

$$\frac{dE\left(x\right)}{dx} = \frac{q}{\varepsilon}p\left(x\right), \tag{2}$$

where $p\left(x\right)$ is the carrier density at x distance from the injecting contact, $E\left(x\right)$ is the electric field, $\mathfrak{e} = \varepsilon_r\varepsilon_o$ the permittivity of the organic semiconductor, and q the positive elementary charge. Here the mobility, $\mu\left(x\right)$, is position dependent through its electric field dependence [1] and also, as suggested by Monte Carlo simulations [2], through its local charge carrier density dependence: $\mu\left(x\right) = \mu\left(p\left(x\right), E\left(x\right)\right)$. Considering simultaneous dependence of the mobility on the electric field of Poole-Frenkel type and dependence on the charge density as in [3] we take

$$\mu\left(x\right) = a\,p\left(x\right)^b \exp\left(\gamma\sqrt{E\left(x\right)}\right), \tag{3}$$

where a and b are positive constants. With this expression for the mobility, and dropping the spatial dependence, equation (1) takes the form

$$J = qa\,p^{b+1}E\,\exp\left(\gamma\sqrt{E}\right), \tag{4}$$

and from (2) we arrive at

$$dx = \frac{\varepsilon}{q}\left(\frac{J}{qa}\right)^{-\frac{1}{b+1}} E^{\frac{1}{b+1}}\,\exp\left(\frac{\gamma}{b+1}\sqrt{E}\right)dE. \tag{5}$$

Integrating (5) across the organic material we obtain

$$L = \frac{\varepsilon}{q}\left(\frac{J}{qa}\right)^{-\frac{1}{b+1}} \int\limits_{E(0)}^{E(L)} E^{\frac{1}{b+1}}\,\exp\left(\frac{\gamma}{b+1}\sqrt{E}\right)dE, \tag{6}$$

where L is the film thickness, $x = 0$ and $x = L$ are the positions of the injecting (anode) and collecting (catode) electrodes, respectively. The electric potential, given by

$$V = \int\limits_0^L E dx, \tag{7}$$

can be calculated, using (5), by

$$V = \frac{\varepsilon}{q}\left(\frac{J}{qa}\right)^{-\frac{1}{b+1}} \int\limits_{E(0)}^{E(L)} E^{1+\frac{1}{b+1}}\,\exp\left(\frac{\gamma}{b+1}\sqrt{E}\right)dE. \tag{8}$$

 ISBN 13: 978-84-613-5510-5

The integrals in equations (6) and (8) can be expressed in terms of incomplete gamma functions Γ [4], since

$$\int E^{\alpha} \exp\left(\beta\sqrt{E}\right) dE = 2(-\beta)^{-2(1+\alpha)}\Gamma\left[2+2\alpha, -\sqrt{E}\beta\right],$$

making possible to obtain the exact solution for the $J-V$ characteristic, parametrically in $E(L)$. For some particular values of b, γ and for some limit values of the electric field it is possible to obtain other models in use [5] and explicit expressions for $J(V)$.

Acknowledgements

We thank FCT for financial support under the contract POCI/CTM/58767/2004.

References

[1] Martin Pope and Charles E. Swenberg, *Electronic Processes in Organic Crystals and Polymers*, Oxford University Press, 1999.

[2] W. F. Pasveer, J. Cottar, C. Tanase, R. Coehoorn, P. A. Bobbert, P. W. M. Blom, D. M. de Leeuw, and C. J. Michels, Phys. Rev. Lett. **94** (2005) 206601.

[3] M. C. J. M. Vissenberg and M. Matters, Phys. Rev. B **57** (1998) 12964.

[4] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions* , Dover Publications, New York, 1964.

[5] Juan Bisquert, José M. Montero, Henk J. Bolink, Eva M. Barea, and Germà Garcia-Belmonte, phys. stat. sol. (a) **203** (2006) 3762–3767.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Finite difference schemes for singular free boundary problems

Luísa Morgado[1] **and Pedro Lima**[2]

[1] *Cemat and Department of Mathematics, University of Trás-os-Montes e Alto Douro*

[2] *Cemat, Centro de Matemática e Aplicações, Instituto Superior Técnico*

emails: luisam@utad.pt, plima@math.ist.utl.pt

Abstract

In this paper a class of singular free boundary value problems arising in plasma physics is considered. Taking into account the behavior of the solution in the neighborhood of the singular points, a smothing variable substitution is proposed in order to avoid the decreasing of the convergence order of the finite difference scheme caused by the presence of singularities. Numerical results are presented and discussed.

Key words: m-Laplacian, singular Cauchy problem, singular boundary value problem, finite difference method.
MSC 2000: 65L05

1 Introduction

Here we consider the following nonlinear second order differential equation

$$\left(\left|y'\right|^{m-2}y'\right)' + \frac{N-1}{x}\left|y'\right|^{m-2}y' + f(y) = 0, \quad 0 < x < +\infty, \tag{1}$$

where we assume $N \geq 2$, $m > 1$ and

$$f(y) = ay^q - by^p, \tag{2}$$

$p < q$ and $a, b > 0$.

We look for a real $M > 0$ and a positive solution of this equation satisfying the following boundary conditions

$$y'(0) = 0, \quad y(M) = y'(M) = 0, \quad M > 0, \tag{3}$$

This kind of problems arise when we are looking for radial solutions of

$$\triangle_m y = f(y), \tag{4}$$

 ISBN 13: 978-84-613-5510-5

in a ball $B(0,M) \subset \mathbb{R}^N$, where $\triangle_m y = div(|Dy|^{m-2} Dy)$ is the degenerate m-Laplace operator.

When $m = 2$, equation (4) reduces to $\triangle y = f(y)$, where $\triangle$ is the classical Laplace operator. In this case, the existence of solutions, under different kinds of boundary conditions, was investigated by several authors. The authors of [9], [10], [13], [19], among others, have considered the regular case, when f is smooth in some sense; for results in the case of singular f , see [1], [3], [4], [5], [7], [18] and the references there. In [8], the authors have studied the case of the degenerate Laplacian ($m > 1$), discussing both the cases of boundary value problems on the half-line and free boundary problems.

According to Corollary 1 of [8], problem (1), (3) has a positive solution for some $M > 0$ provided either $N \leq m, -1 < p < m-1$ or $N > m, -1 < p < m-1, q < \sigma, \sigma = \frac{(m-1)N+m}{N-m}$.

When $m = 2$, $p = 1-\alpha$, $q = 1$, $a = 1$ and $b = \frac{1}{\alpha}$, $1 < \alpha < 2$, the solution of problem (1), (3) is related to the blow-up of self-similar solutions of a singular nonlinear parabolic problem arising in the study of force-free magnetic fields in a passive medium (see, for example [14], [15] and the references therein). The case $m = 2$, $q = \frac{1}{2}$ and $p = 0$, problem (1), (3) has been recently proposed as a simple model for the Tokamac equilibria with magnetic islands, [16]. This particular problem and problems of the type (1), (3) with $m = 2$ and $f(y) = ay^q - by^p$, $0 \leq p < q$, $a, b > 0$ which are singular only at $x = 0$, have been treated in [11], where the authors determined one-parameter family of solutions describing the behavior of the solution in the neighborhood of the singular point. Based on these families they constructed a shooting algorithm that allowed them to compute the solution accurately. In [17] the authors extended these results to the case where $m > 1$ and $-1 < p < q$ and implemented not only a shooting algorithm but also a finite difference method that took into account the behavior of the solution near the singular points. Even so, in some cases the convergence order of that finite difference scheme became very low.

Here, based on the asymptotic expansion of the solution in the neighborhood of the singularities, we propose a variable substitution and implement a finite difference method whose numerical results suggest quadratic convergence.

A similar approach was used in [12] where a singular boundary value problem, defined on an specified and limited domain, was analysed.

Since equation (1) may be written in the form

$$y'' = -\frac{1}{m-1}\left(\frac{N-1}{x}y' - \frac{f(y)}{|y'|^{m-2}}\right) \tag{5}$$

we can predict two singular points: the problem will be singular at the origin due to the division by x, and singular at both endpoints, when $m > 2$, due to the division by $|y'|^{m-2}$. On the other hand, the problem will be singular at $x = M$, whenever p or q are negative, taking (2) into account. This will be explained in detail in the next section.

 ISBN 13: 978-84-613-5510-5

2 Behavior of the solution near the singular points

In [17] the following theorems were proved:

Theorem 2.1 *Let $N \geq 2$, $m > 1$ and $-1 < p < q$. For each $y_0 > 0$, problem*

$$\begin{aligned} \left(\left|y'\right|^{m-2} y'\right)' + \frac{N-1}{x}\left|y'\right|^{m-2} y' + ay^q - by^p &= 0, \quad 0 < x < +\infty, \qquad (6) \\ y(0) = y_0, \quad y'(0) &= 0, \qquad (7) \end{aligned}$$

has, in the neighborhood of $x = 0$, a unique holomorphic solution that can be represented by

$$y(x, y_0) = y_0 - \frac{m-1}{m}\left(\frac{ay_0^q - by_0^p}{N}\right)^{\frac{1}{m-1}} x^{\frac{m}{m-1}} \left[1 + \sum_{l=0,k=0,l+k\geq 1}^{+\infty} g_{l,k} x^{l+k\frac{m}{m-1}}\right], \quad 0 \leq x \leq \delta(y_0), \delta(y_0) \geq 0,$$

where the coefficients $g_{l,k}$ depend on m, p, q, a, b, N and y_0.

Theorem 2.2 *Let $N \geq 2$, $m > 1$, $-1 < p < q$ and $m - 1 - p > 0$. For each $M > 0$, the problem*

$$\begin{aligned} \left(\left|y'\right|^{m-2} y'\right)' + \frac{N-1}{x}\left|y'\right|^{m-2} y' + ay^q - by^p &= 0, \quad 0 < x < +\infty, \\ y(M) = y'(M) &= 0, \end{aligned}$$

has, in the neighborhood of the possible singular point $x = M$, a unique holomorphic solution that can be represented by

$$y(x, M) = C_M (M - x)^{\frac{m}{m-1-p}} \left[1 + \sum_{l=0,j=0,l+j\geq 1}^{+\infty} G_{l,j} (M-x)^{l+j\frac{m(q-p)}{m-1-p}}\right], \quad 0 \leq x \leq \delta(M), \delta(M) \geq 0,$$

where $C_M = \left(\frac{b(m-1-p)^m}{m^{m-1}(m-1)(p+1)}\right)^{\frac{1}{m-1-p}}$ and the coefficients $G_{l,j}$ depend on m, p, q, a, b, N and M.

According to Theorem 2.1, in the neighborhood of $x = 0$, the solution behaves as the function

$$y_0 + C_0 x^{k_1},$$

where $k_1 = \frac{m}{m-1}$ and C_0 is a constant depending on y_0, m, N, p, q, a and b, and therefore it is easy to see, that near the origin the second derivative of the solution is bounded whenever $k_1 \geq 2$ which is equivalent to say that $m \leq 2$.

On the other hand, according to Theorem 2.2, in the neighborhood of $x = M$, the solution behaves as

$$C_M (M - x)^{k_2},$$

 ISBN 13: 978-84-613-5510-5

where $k_2 = \frac{m}{m-1-p}$ and C_M is a constant depending on m, p and b. Consequently, near the unknown boundary, the second derivative of the solution is bounded whenever $k_2 \geq 2$, or equivalently, whenever $m \leq 2(1+p)$.

In summary, problem (1), (3) has a singularity at $x = 0$ if $m > 2$ and a singular point at $x = M$ if $m > 2(p+1)$.

3 A finite difference scheme

In this section we focus on the case $m = 2$, that is the case where the m-Laplacian operator reduces to the classical one, and problem (1), (3) has known applicatons in physics. From the mathematical point of view, the case $m \neq 2$ is more interesting but much more complicated since the classical Laplacian, which is a linear operator is replaced by a nonlinear one. Theregore, we just leave here some ideas for the numerical treatment of problem (1), (3) for any $m > 1$.

First, let us recall that since we have a free boundary problem we have to approximate not only the solution y but also the endpoint M where certain conditions are imposed. Second, if we intend to implement a finite difference scheme, the natural first step is to perform the variable substitution $z = \frac{x}{M}$, for instance, in order to obtain a boundary value problem in the interval $[0, 1]$. Problem (1), (3) rewrites in the new variable z, considering $m = 2$, in the form

$$\ddot{y} + \frac{N-1}{z}\dot{y} + \lambda(ay^q - by^p) = 0, \quad 0 < z < 1 \tag{8}$$

$$\dot{y}(0) = 0 \tag{9}$$

$$y(1) = \dot{y}(1) = 0 \tag{10}$$

where $\lambda = M^2$ and $\dot{y}$ denotes the derivative of y with respect to z.

Taking into account the conclusions of the last section, since $m = 2$, we have at most one singular endpoint, which is now transfered to $z = 1$ and this happens if $p < 0$. In adition, as $z \to 1^-$, the solution behaves as $C(1-z)^{k_2}$ where C is a positive constant and $k_2 = \frac{2}{1-p}$. Note that if $p < 0$, the second derivative of the solution becomes unbounded near $z = 1$, but if we perform the variable substitution

$$t = 1 - (1-z)^{\frac{k_2}{2}}, \tag{11}$$

near $t = 1$ the solution behaves as $C(1-t)^2$, and therefore the second derivative of the solution remains bounded in the neighborhood of that point.

Let us remark that we could instead perform the variable substitution $\tilde{t} = (1-z)^{\frac{k_2}{2}}$ to achieve analogous conclusions, the difference is that the possible singular point $z = 1$ would be transfered to the origin.

In this new variable t, problem (8)-(10) rewrites

$$\frac{k_2}{2}(1-t)^{1-\frac{2}{k_2}}\left[y''\frac{k_2}{2}(1-t)^{1-\frac{2}{k_2}} + y'\left(\left(1-\frac{k_2}{2}\right)(1-t)^{-\frac{2}{k_2}} + \frac{N-1}{1-(1-t)^{\frac{2}{k_2}}}\right)\right] \tag{12}$$

 ISBN 13: 978-84-613-5510-5

$$\begin{aligned} +\lambda\left(ay^q - by^p\right) &= 0, \quad 0 < t < 1 \\ y'(0) &= 0 \qquad (13) \\ y(1) &= 0 \qquad (14) \\ y'(1) &= 0 \qquad (15) \end{aligned}$$

where, for simplicity, we use once again the notation y', but now to denote the derivative of y with respect to t.

In the interval $[0,1]$, we introduce an uniform grid of constant stepsize $h = \frac{1}{n}$, defined by the $(n+1)$ gridpoints $t_i = ih$, $i = 0, \ldots, n$. At each $i = 0, \ldots, n$, let y_i denote an approximation of $y(t_i)$. In order to approximate the first and second derivative of the solution at each gridpoint t_i, $i = 1, \ldots, n-1$, we use the second order formulas

$$\begin{aligned} y'(t_i) &\simeq \frac{y(t_{i+1}) - y(t_{i-1})}{2h}, \\ y''(t_i) &\simeq \frac{y(t_{i+1}) - 2y(t_i) + y(t_{i-1})}{h^2}, \end{aligned}$$

and

$$y'(0) = \frac{1}{2h}\left(-3y(0) + 4y(h) - y(2h)\right) + O(h^2)$$
$$y'(1) = \frac{1}{2h}\left(3y(1) - 4y(1-h) + y(1-2h)\right) + O(h^2).$$

to provide approximations for the first and second derivatives of the solution at $t = 0$ and $t = 1$, respectively.

Hence, we obtain the following discretization of problem (12)-(15):

$$\begin{aligned} -3y_0 + 4y_1 - y_2 &= 0 \\ \frac{k_2}{2}(1-t_i)^{1-\frac{2}{k_2}}\left[\left(y_{i+1} - 2y_i + y_{i-1}\right)\frac{k_2}{2}(1-t_i)^{1-\frac{2}{k_2}} + \frac{h}{2}\left(y_{i+1} - y_{i-1}\right)\times\right. \\ \left.\times\left(\left(1 - \frac{k_2}{2}\right)(1-t_i)^{-\frac{2}{k_2}} + \frac{N-1}{1-(1-t_i)^{\frac{2}{k_2}}}\right)\right] + \lambda h^2\left(ay_i^q - by_i^p\right) &= 0 \\ y_n &= 0 \\ 3y_n - 4y_{n-1} + y_{n-2} &= 0 \end{aligned}$$

We used the Newton's method to solve these $n+2$ equations on the $n+2$ unknowns $y_0, y_1, \ldots, y_n, \lambda$, using the parameters estimates for y_0 and $M = \sqrt{\lambda}$, determined in [11], to provide an initial approximation. Let us explain this with more detail. In [11] the authors proved that if

$$y_0 = y(0) > \beta_0 = \left(\frac{b(q+1)}{a(p+1)}\right)^{\frac{1}{q-p}},$$

a result that was easily deduced from Lemma 1.2.1 of [6]. Moreover they have shown that the quantity

$$M_{min} = \begin{cases} \sqrt{2N\frac{b(p-1)}{a(q-1)}}, & \text{if} \quad 0 \le p < q < 1; \\ \sqrt{2N/a}, & \text{if} \quad q = 1, p = 0. \end{cases}$$

 ISBN 13: 978-84-613-5510-5

is a lower bound for M.

Taking this into account we use as initial guesses for y_0, y_i, $i = 1, 2, \ldots, n-1$, and λ, the values β_0, $r(ih)$, $i = 1, 2, \ldots, n-1$, and M^2_{min}, respectively, where $r(t) = y_0(1-t)$.

Several numerical experiments were carried out considering different values of the parameters in the case where we know that the unknown boundary is a singular point and, for each case, we have determined estimates for the convergence order of the method (EOC). In what follows we give special attention to the application problem on force-free magnetic fields, mentioned in the introduction.

h	y_0	M
$\frac{1}{50}$	5.346587922	4.224300407
$\frac{1}{100}$	5.346011684	4.225902674
$\frac{1}{200}$	5.345865329	4.226293630
$\frac{1}{400}$	5.345828568	4.226388647
EOC	1.99	2.03

Table 1: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $\alpha = 1.1$, $b = \frac{1}{\alpha}$, $p = 1 - \alpha$ and $q = 1$

h	y_0	M
$\frac{1}{50}$	5.256595222	3.824463141
$\frac{1}{100}$	5.254723192	3.825765281
$\frac{1}{200}$	5.254252840	3.826077134
$\frac{1}{400}$	5.254134855	3.826152600
EOC	2.00	2.05

Table 2: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $\alpha = 1.3$, $b = \frac{1}{\alpha}$, $p = 1 - \alpha$ and $q = 1$

h	y_0	M
$\frac{1}{50}$	5.630343553	3.544298837
$\frac{1}{100}$	5.626305751	3.545518728
$\frac{1}{200}$	5.625303291	3.545813151
$\frac{1}{400}$	5.625055855	3.545885122
EOC	2.01	2.04

Table 3: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $\alpha = 1.5$, $b = \frac{1}{\alpha}$, $p = 1 - \alpha$ and $q = 1$

We also present some numerical results when both exponents p and q are negative.

 ISBN 13: 978-84-613-5510-5

h	y_0	M
$\frac{1}{50}$	6.710094248	3.342951097
$\frac{1}{100}$	6.700152866	3.344179828
$\frac{1}{200}$	6.697724591	3.344478527
$\frac{1}{400}$	6.697131784	3.344551663
EOC	2.03	2.04

Table 4: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $\alpha = 1.7$, $b = \frac{1}{\alpha}$, $p = 1 - \alpha$ and $q = 1$

h	y_0	M
$\frac{1}{50}$	12.41897365	13.09140957
$\frac{1}{100}$	12.37618599	13.07500708
$\frac{1}{200}$	12.41897365	13.09140957
$\frac{1}{400}$	12.43277829	13.09661188
EOC	1.60	1.63

Table 5: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $b = 1$, $p = -0.5$ and $q = -0.1$

4 Conclusions and future work

In this work a smothing variable substitution was proposed for problem (1), (3), in the particular case $m = 2$. The purpose of doing that was to avoid the decreasing of the convergence order of finite difference methods, in the presence of singularities. Comparing the results obtained here with those obtained in [17], in the cases were both algorithms, are applicable, we can observe an improvement of the convergence order. For example, in the case $p = -0.5, q = 1, N = 3, m = 2, a = b = 1$, the estimated convergence order is 2.01 when evaluating y_0 using the present algorithm, and only 0.7, without variable substitution. When evaluating M, the estimated convergence orders are 2.03 and 0.47, repectively. Moreover, the algorithm without variable substitution fails in many of the cases presented in the present paper, due to numerical instability. Concerning the case $m \neq 2$, and proceeding accordingly to what we have done in previous works, the natural way to proceed is to find suitable smoothing variable

h	y_0	M
$\frac{1}{50}$	13.34044610	16.47456975
$\frac{1}{100}$	13.58236913	16.59598170
$\frac{1}{200}$	13.67567476	16.64203302
$\frac{1}{400}$	13.70992868	16.65874200
EOC	1.41	1.43

Table 6: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $b = 1$, $p = -0.5$ and $q = -0.2$

 ISBN 13: 978-84-613-5510-5

h	y_0	M
$\frac{1}{50}$	14.65705132	22.29501386
$\frac{1}{100}$	15.10417565	22.59892593
$\frac{1}{200}$	15.30338410	22.73246382
$\frac{1}{400}$	15.38766255	22.78849085
EOC	1.20	1.22

Table 7: Approximate values of y_0 and M for $m = 2$, $N = 3$, $a = 1$, $b = 1$, $p = -0.5$ and $q = -0.3$

substitutions depending on the number of singular points. Let us be more precise: if $m < 2$, since we have no singularity at the origin, and if $m > 2(1 + p)$, we can use the same smoothing variable substitution (11). On the other hand, if $m > 2$ and $m \leq 2(1 + p)$, then the only singular point of problem (1), (3) is the origin, the result of Theorem 2.1 suggests the variable substitution $t = z^{\frac{k_1}{2}}$, where $k_1 = \frac{m}{m-1}$. Finaly, if both endpoints are singular, then we could use $t = \left(1 - (1 - z)^{\frac{k_2}{2}}\right)^{\frac{k_1}{2}}$.

We have experimented all these but we must say that the numerical results were not satisfactory. Althought we have obtained reasonable results in case $m < 2$, whenever we considered the case $m > 2$ the method became higly unstable and extremely sensitive to initial approximations. We must recall that in [17] we were not able to prove that the estimate M_{min} that we there have determined, is in fact a lower bound for M, as we have done in case $m = 2$. Hence, the case $m \neq 2$ deserves further investigation and will be the subject of future work.

Acknowledgements

L. Morgado acknowledges financial support from FCT, Fundação para a Ciência e Tecnologia, through grant SFRH/BPD/46530/2008.

References

[1] H. Chen, *On a singular nonlinear elliptic equation*, Nonlin. Anal., TMA **29** (1997), 337–345.

[2] H. Chen, *Analysis of blow-up for a nonlinear degenerate parabolic equation*, J. Math. Anal. Appl. **192** (1995), 180–193.

[3] M. G. Crandall, P. H. Rabinowitz and L. Tartar, *On a Dirichlet problem with a singular nonlinearity*, Comm. PDE **2** (1977), 193–222.

[4] J. I. Diaz, J. M. Morel and L. Oswald, *An elliptic equation with singular nonlinearity*, Comm. PDE **12** (1987), 1333–1344.

 ISBN 13: 978-84-613-5510-5

[5] A. M. Fink, J. A. Gatica, G. E. Hernandez and P. Waltman, *Approximation of solutions of singular second order boundary value problems*, SIAM J. Math. Anal. **22** (1991), 440–462.

[6] B. Franchi, Lanconelli E. and Serrin J., *Existence and uniqueness of nonnegative solutions of quasilinear equations in* $\mathbb{R}^n$, Advances in Math. **118** (1996),177–243.

[7] J. A. Gatica, V. Oliker and P. Waltman, *Singular nonlinear boundary value problems for second-order ordinary differential equations*, J. Diff. Eqns **79** (1990), 62–78.

[8] F. Gazzola, J. Serrin and M. Tang, *Existence of ground states and free boundary problems for quasilinear elliptic operators*, Advances in Differential Equations **5** (1-3) (2000), 1–30.

[9] B. Gidas, W. M. Ni and L. Nirenberg, *Symmetry and related properties via the maximum principle*, Comm. Math. Phys. **68** (1979), 209–243.

[10] H. G. Kaper and M. K. Kwong, *Free boundary problems for Emden-Fowler equations*, Diff. Int. Eqs **3** (1990), 353–362.

[11] P. Lima and L. Morgado, *Analysis and numerical approximation of a free boundary problem for a singular ordinary differential equation*, Tema Tend. Mat. Apl. Comput. **8** No 2 (2007), 259–268.

[12] P. Lima and L. Morgado, *Numerical Modelling of Oxygen Diffusion in Cells with Michaelis-Menten Uptake Kinetics*, J. Comp. Mathem. Chemistry, to appear, DOI: 10.1007/s10910-009-9646-x

[13] P. L. Lions, *On the existence of positive solutions of semilinear elliptic equations*, SIAM Rev. **24** (1982), 441–467.

[14] B. C. Low, *Resistive diffusion of force-free magnetic fiels in a passive medium-I*, Astrophys. J. **181** (1973), 209–226.

[15] B. C. Low, *Nonlinear classical diffusion in a contained plasma*, Phys. Fluids **25** (1973), 402–407.

[16] G. Miller, V. Faber, and A. B. White Jr., *Finding plasma equilibria with magnetic islands*, J. Computation Physics **79** (1998), 417–435.

[17] L. Morgado and P. Lima, *Numerical solution of a class of singular free boundary problems involving the m-Laplace operator*, J. Comp. Appl. Math. **229**, to appear, DOI: 10.1016/j.cam.2010.01.030

[18] D. O'Regan, *Existence of positive solutions to some singular and nonsingular secod order boundary value problems*, J. Diff. Eqns **84** (1990), 228–251.

 ISBN 13: 978-84-613-5510-5

[19] J. A. Smoller and G. Wasserman, *Existence, uniqueness, and nondegeneracy of positive solutions of semilinear elliptic equations*, Comm. Math. Phys. **95** (1984), 129–159.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Modularity and dynamical processes in a complex software network

Luis G. Moyano[1], Mary Luz Mouronte Lopez[1] and Maria Luisa Vargas[1]

[1] *Telefónica I+D, Emilio Vargas 6, 28043 Madrid, Spain*

emails: moyano@tid.es, mlml@tid.es, vargas@tid.es

Abstract

Complex networks have gathered enormous interest in the last two decades as researchers realise the major role they take in a multitude of areas [1, 2, 3, 4, 5, 6]. Complex technological networks represent a growing challenge to support and mantain [7, 8, 9, 10], as their number of elements become higher and their interdependencies more involved. On the other hand, for networks that grow in a descentralized manner, it is possible to observe certain patterns in their overall structure that may be taken into account for a more tractable analysis [11, 12, 13, 14]. An example of such a pattern is the spontaneous formation of communities or modules [15]. An important question regarding the detection of communities is if these are really representative of any internal network feature. In this work, we explore the community structure of a real telecommunication software complex network, and correlate the modularity information with the internal dynamical processes that the network is designed to support. Our results show that the correlation between community structure and internal dynamical processes is remarkable, supporting the fact that a community division of this complex network is helpful in the assessment of the underlying dynamical structure, and thus is a useful tool to achieve a simpler representation of the complexity of the network.

Key words: template, instructions
MSC 2000: AMS codes (optional)

References

[1] M. Newman, A. Barabási, D.J. Watts, The Structure and Dynamics of Networks: (Princeton Studies in Complexity), Princeton University Press, 2006.

[2] S. Wasserman, K. Faust, Social Network Analysis, Cambridge University Press, 1994.

 ISBN 13: 978-84-613-5510-5

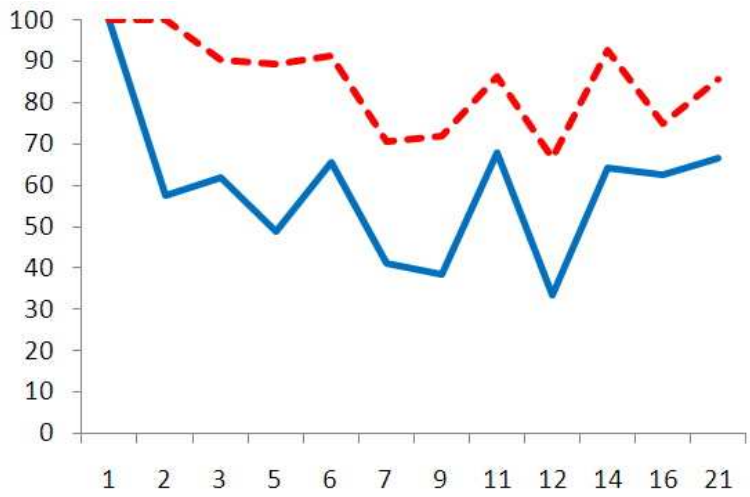


Figure 1: Average fraction of systems belonging to a process that correspond to only one community (solid line), or at most two communitites (dashed line).

[3] G. Caldarelli, Scale-Free Networks, Oxford University Press, 2007.

[4] M.E.J. Newman, SIAM Rev. 45 (2003) 167-256.

[5] D.J. Watts, S.H. Strogatz, Nature 393 (1998) 440-442.

[6] R. Albert, A. Barabási, Rev. Mod. Phys. 74 (2002) 47.

[7] S. Valverde, R.V. Solé, Phys. Rev. E 72 (2005) 026107, S. Valverde, R.V. Solé, Europhys. Lett. 72 (2005) 858-864.

[8] Q. Gunqun, Z. Lin, Z. Li, Proceedings of the 2008 International Conference on Computer Science and Software Engineering - Volume 02, IEEE Computer Society, 2008, pp. 310-316.

[9] K. He, R. Peng, J. Liu, F. He, P. Liang, B. Li, Journal of Systems Science and Complexity 19 (2006) 157181.

[10] C. Christensen, R. Albert, Int. J. Bifur. Chaos 17 (2007) 2201.

[11] D.J. Watts, Proceedings of the National Academy of Sciences 99 (2002) 5766-5771.

[12] A.E. Motter, Phys. Rev. Lett. 93 (2004) 098701.

[13] M.L. Sachtjen, B.A. Carreras, V.E. Lynch, Phys. Rev. E 61 (2000) 4877.

[14] H. Yang, Y. Nie, H. Zhang, Z. Di, Y. Fan, Computer Physics Communications 180 (2009) 1511-1515.

[15] S. Fortunato, Physics Reports 486 (2010) 75-174.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Optimum ratio estimators for the population proportion

Juan Francisco Muñoz[1], Encarnación Álvarez[1], Antonio Arcos[2], María del Mar Rueda[2] and Silvia González[3]

[1] *Department of Quantitative Methods for Economy and Enterprise, University of Granada*

[2] *Department of Statistics and Operational Research, University of Granada*

[3] *Department of Statistics and Operational Research, University of Jaén*

emails: jfmunoz@ugr.es, encarniav@ugr.es, arcos@ugr.es, mrueda@ugr.es, sgonza@ujaen.es

Abstract

The problem of the estimation of a population proportion using auxiliary information has been recently studied by Rueda et al. (2010), which proposed several ratio estimators of the population proportion and studied some theoretical properties. In this paper, we define a new ratio estimator based on a linear combination of two ratio estimators defined by Rueda et al. (2010). The variance of the new estimator is calculated and it is used to obtain the optimum value into the linear combination in the sense of minimal variance. Theoretical and empirical studies show that the suggested ratio estimator performs better than alternative estimators.

Key words: Auxiliary information, ratio type estimator
MSC 2000: 62D05

1 Introduction

In the presence of auxiliary information, there exist many design-based approaches (see [3], [5], [1]) to improve the precision of estimators in comparison to customary methods, which do not involve auxiliary information. However, techniques involving auxiliary information have been discussed for quantitative variables, and the extension to the estimation of a population proportion requires further investigation. For example, one should be aware of the risks when confidence intervals are constructed for a population proportion, since limits outside $[0, 1]$ could be achieved.

We consider the scenario of a finite population $U = \{1, \ldots, N\}$ containing N units. Let $A_1, \ldots, A_N$ denote the values of a attribute of interest A, where $A_i = 1$ if ith unit

 ISBN 13: 978-84-613-5510-5

possesses the attribute A and $A_i = 0$ otherwise. Let B denote an auxiliary attribute associated with A and values given by $B_1, \ldots, B_N$. We also assume that a sample s, of size n, is selected from U according to the well known simple random sampling without replacement (SRSWOR).

The aim is to estimate the population proportion of individuals that posses the attribute A, i.e. $P_A = N^{-1} \sum_{i=1}^{N} A_i$. Assuming a finite population, the naive estimator of P_A, which makes no use of the auxiliary information, is given by $\widehat{p}_A = n^{-1} \sum_{i \in s} A_i$.

We assume that the population proportion of individuals that posses the attribute B, $P_B = N^{-1} \sum_{i=1}^{N} B_i$, is known from a census or estimated without error.

Rueda et al (2010) defined the following ratio estimator for P_A:

$$\widehat{p}_r = \widehat{R} P_B, \tag{1}$$

where $\widehat{R} = \widehat{p}_A / \widehat{p}_B$ is an estimator of the population ratio $R = P_A / P_B$ and $\widehat{p}_B = n^{-1} \sum_{i \in s} B_i$ is the sample proportion of individuals that posses the auxiliary attribute B.

Let A^c and B^c denote the complementary attributes of A and B, and consider the population two-way table given by

	B	B^c	
A	N_{11}	N_{12}	$N_{1\cdot}$
A^c	N_{21}	N_{22}	$N_{2\cdot}$
	$N_{\cdot 1}$	$N_{\cdot 2}$	N

(2)

where $N_{1\cdot} = \sum_{i=1}^{N} A_i$ is the number of units in the population that posses the attribute A, $N_{2\cdot}$ is the number of units in the population that not to posses the attribute A, etc. Analogously, N_{11} is the number of units in the population that simultaneously posses the attributes A and B, N_{12} is the number of units in the population that simultaneously posses the attributes A and B^c, etc. Classification (2) can be also defined at the sample level as

	B	B^c	
A	n_{11}	n_{12}	$n_{1\cdot}$
A^c	n_{21}	n_{22}	$n_{2\cdot}$
	$n_{\cdot 1}$	$n_{\cdot 2}$	n

(3)

The estimator $\widehat{p}_r$ is a biased estimator of P_A and the asymptotic variance of $\widehat{p}_r$ is given by

$$AV(\widehat{p}_r) = \frac{N-n}{(N-1)n} \left(P_A Q_A + R^2 P_B Q_B - 2R\phi \sqrt{P_A Q_A P_B Q_B} \right), \tag{4}$$

where

$$\phi = \frac{N_{11} N_{22} - N_{12} N_{21}}{\sqrt{N_{1\cdot} N_{2\cdot} N_{\cdot 1} N_{\cdot 2}}}$$

is the Cramer's V coefficient based on the two-way classification (2).

 ISBN 13: 978-84-613-5510-5

We observe that the customary estimator $\widehat{p}_A$ can be also obtained as $\widehat{p}_A = 1 - \widehat{q}_A$, where $\widehat{q}_A = n^{-1}\sum_{i\in s} A_i^c$, hence $\widehat{p}_A$ has the same performance in the estimation of P_A than the performance of $\widehat{q}_A$ in the estimation of Q_A. However, this property is not satisfied by $\widehat{p}_r$, i.e. it can be easily seen that $\widehat{p}_r \neq 1 - \widehat{q}_r$, where $\widehat{q}_r = \widehat{R}^c Q_B$ and $\widehat{R}^c = (\widehat{q}_A/\widehat{q}_B)$. For this reason, Rueda et al. (2010) defined the ratio estimator $\widehat{p}_{r.q} = 1 - \widehat{q}_r$ for P_A and showed that $AV(\widehat{p}_r) < AV(\widehat{p}_{r.q})$ when $P_A < P_B$.

2 The optimum ratio estimator

In this section, we define a new ratio type estimator using a linear combination of the ratio estimators $\widehat{p}_r$ and $\widehat{p}_{r.q}$ previously defined. The choice of the optimum weight value into the linear combination is achieved by minimizing the variance. Finally, some interesting theoretical properties are also obtained.

The new ratio type estimator is

$$\widehat{p}_{r.w} = w\widehat{p}_r + (1-w)\widehat{p}_{r.q} \tag{5}$$

where $0 \leq w \leq 1$.

Theorem 1

The optimum value for w in the sense of minimum variance into the class of estimators $\widehat{p}_{r.w}$ is

$$w_{opt} = \frac{AV(\widehat{p}_{r.q}) - cov(\widehat{p}_r, \widehat{p}_{r.q})}{AV(\widehat{p}_r) + AV(\widehat{p}_{r.q}) - 2cov(\widehat{p}_r, \widehat{p}_{r.q})}. \tag{6}$$

Proof

Next, we determine the optimum value of w by minimizing the variance of $\widehat{p}_{r.w}$. The asymptotic variance of $\widehat{p}_{r.w}$ is given by

$$AV(\widehat{p}_{r.w}) = AV(w\widehat{p}_r + (1-w)\widehat{p}_{r.q}) =$$

$$= w^2 AV(\widehat{p}_r) + (1-w)^2 AV(\widehat{p}_{r.q}) + 2w(1-w)cov(\widehat{p}_r, \widehat{p}_{rq}).$$

By denoting $V_1 = AV(\widehat{p}_r)$, $V_2 = V(\widehat{p}_{r.q})$ and $C = cov(\widehat{p}_r, \widehat{p}_{r.q})$, the variance of $\widehat{p}_{r.w}$ can be expressed as

$$AV(\widehat{p}_{r.w}) = w^2 V_1 + (1-w)^2 V_2 + 2w(1-w)C.$$

The first derivative of $AV(\widehat{p}_{r.w})$ with respect to w is

$$\frac{\partial AV(\widehat{p}_{r.w})}{\partial w} = 2wV_1 - 2(1-w)V_2 + 2(1-2w)C = 0;$$

 ISBN 13: 978-84-613-5510-5

$$wV_1 - (1-w)V_2 + (1-2w)C = 0;$$
$$wV_1 - V_2 + wV_2 + C - 2wC = 0;$$
$$w(V_1 + V_2 - 2C) = V_2 - C;$$

$$w_{opt} = \frac{V_2 - C}{V_1 + V_2 - 2C}.$$

The second derivative is

$$\frac{\partial AV(\widehat{p}_{r.w})}{\partial^2 w} = 2V_1 + 2V_2 - 4C = 2(V_1 + V_2 - 2C) = 2AV(\widehat{p}_r + \widehat{p}_{r.q}) > 0,$$

and we conclude that w_{opt} really minimizes $AV(\widehat{p}_{r.w})$.

□

Therefore, the optimum ratio estimator in the sense of minimum variance into the class (5) is

$$\widehat{p}_{r.OPT} = w_{opt}\widehat{p}_r + (1 - w_{opt})\widehat{p}_{r.q}.$$

In practice, $\widehat{p}_{r.OPT}$ could be unknown, since w_{opt} depends on population variances, which are generally unknown. In this situation, we can use the estimator

$$\widehat{p}_{r.opt} = \widehat{w}_{opt}\widehat{p}_r + (1 - \widehat{w}_{opt})\widehat{p}_{r.q}, \tag{7}$$

where

$$\widehat{w}_{opt} = \frac{\widehat{V}(\widehat{p}_{r.q}) - \widehat{cov}(\widehat{p}_r, \widehat{p}_{r.q})}{\widehat{V}(\widehat{p}_r) + \widehat{V}(\widehat{p}_{r.q}) - 2\widehat{cov}(\widehat{p}_r, \widehat{p}_{r.q})}. \tag{8}$$

Following Särndal et al. (1992) pg 372, the variance of $\widehat{p}_{r.w}$ can be expressed as

$$AV(\widehat{p}_{r.w}) = (V_1 + V_2 - 2C)\left(w - \frac{V_2 - C}{V_1 + V_2 - 2C}\right)^2 + \frac{V_1V_2 - C^2}{V_1 + V_2 - 2C},$$

and we can deduce that the variance of the optimum estimator is

$$AV(\widehat{p}_{r.OPT}) = \frac{V_1V_2 - C^2}{V_1 + V_2 - 2C}.$$

An estimator of the variance of the optimum estimator can be obtained as

$$\widehat{V}(\widehat{p}_{r.opt}) = \frac{\widehat{V}(\widehat{p}_r)\widehat{V}(\widehat{p}_{r.q}) - \widehat{cov}^2(\widehat{p}_r, \widehat{p}_{r.q})}{\widehat{V}(\widehat{p}_r) + \widehat{V}(\widehat{p}_{r.q}) - 2\widehat{cov}(\widehat{p}_r, \widehat{p}_{r.q})}.$$

We observe that $AV(\widehat{p}_{r.OPT})$ depends on the covariance $C = cov(\widehat{p}_r, \widehat{p}_{r.q})$. Theorem 2 gives an expression for C.

Theorem 2

 ISBN 13: 978-84-613-5510-5

The covariance between the ratio estimators $\widehat{p}_r$ and $\widehat{p}_{r.q}$ is

$$cov(\widehat{p}_r, \widehat{p}_{r.q}) = \frac{N-n}{N-1}\frac{1}{n}\left(P_A Q_A + R R_c P_B Q_B - (R+R_c)\phi\sqrt{P_A Q_A P_B Q_B}\right),$$

where $R_c = Q_A/Q_B$ is the population ratio of the complementary proportions of the attributes A and B.

Proof

Using Taylor series (see Särndal et al. 1992, pg 178), $\widehat{R}$ can be expressed as

$$\widehat{R} \cong R + \frac{1}{nP_B}\sum_{i\in s}(A_i - RB_i) = R + \frac{1}{P_B}(\widehat{p}_A - R\widehat{p}_B),$$

and similarly

$$\widehat{R}_c \cong R_c + \frac{1}{Q_B}(\widehat{q}_A - R_c\widehat{q}_B).$$

Using the previous expressions we obtain

$$C = cov(\widehat{p}_r, 1-\widehat{q}_r) = -cov(\widehat{p}_r, \widehat{q}_r) = -cov(\widehat{R}P_B, \widehat{R}_c Q_B) =$$

$$= -P_B Q_B cov(\widehat{R}, \widehat{R}_c) = -P_B Q_B cov\left(R + \frac{1}{P_B}(\widehat{p}_A - R\widehat{P}_B), R_c + \frac{1}{Q_B}(\widehat{q}_A - R_c\widehat{q}_B)\right) =$$

$$= -cov(\widehat{p}_A - R\widehat{p}_B, \widehat{q}_A - R_c\widehat{q}_B) =$$

$$= -[cov(\widehat{p}_A, \widehat{q}_A) - R_c cov(\widehat{p}_A, \widehat{q}_B) - R cov(\widehat{p}_B, \widehat{q}_A) + R R_c cov(\widehat{p}_B, \widehat{q}_B)] =$$

$$= -cov(\widehat{p}_A, 1-\widehat{p}_A) + R_c cov(\widehat{p}_A, 1-\widehat{p}_B) + R cov(\widehat{p}_B, 1-\widehat{p}_A) - R R_c cov(\widehat{p}_B, 1-\widehat{p}_B) =$$

$$= V(\widehat{p}_A) - R_c cov(\widehat{p}_A, \widehat{p}_B) - R cov(\widehat{p}_A, \widehat{p}_B) + R R_c V(\widehat{p}_B) =$$

$$= V(\widehat{p}_A) + R R_c V(\widehat{p}_B) - (R+R_c) cov(\widehat{p}_A, \widehat{p}_B) =$$

$$= \frac{N-n}{N-1}\frac{1}{n}\left(P_A Q_A + R R_c P_B Q_B - (R+R_c)\phi\sqrt{P_A Q_A P_B Q_B}\right).$$

□

An estimator of the covariance $cov(\widehat{p}_r, \widehat{p}_{r.q})$ is

$$\widehat{cov}(\widehat{p}_r, \widehat{p}_{r.q}) = \frac{1-f}{n-1}\left(\widehat{p}_A\widehat{q}_A + \widehat{R}\widehat{R}_c\widehat{p}_B\widehat{q}_B - (\widehat{R}+\widehat{R}_c)\widehat{\phi}\sqrt{\widehat{p}_A\widehat{q}_A\widehat{p}_B\widehat{q}_B}\right),$$

 ISBN 13: 978-84-613-5510-5

where

$$\widehat{\phi} = \frac{n_{11}n_{22} - n_{12}n_{21}}{\sqrt{n_{1\cdot}n_{2\cdot}n_{\cdot 1}n_{\cdot 2}}}.$$

Theorem 3

The optimum weight w_{opt} in expression (6) can be expressed as

$$w_{opt} = \frac{R_c - \beta}{R_c - R},$$

where

$$\beta = \frac{cov(\widehat{p}_A, \widehat{p}_B)}{V(\widehat{p}_B)}.$$

Proof

Knowing that

$$V_1 = V(\widehat{p}_A) + R^2 V(\widehat{p}_B) - 2Rcov(\widehat{p}_A, \widehat{p}_B),$$
$$V_2 = V(\widehat{q}_A) + R_c^2 V(\widehat{q}_B) - 2R_c cov(\widehat{q}_A, \widehat{q}_B) =$$
$$= V(\widehat{p}_A) + R_c^2 V(\widehat{p}_B) - 2R_c cov(\widehat{p}_A, \widehat{p}_B)$$

and

$$C = V(\widehat{p}_A) + RR_c V(\widehat{p}_B) - (R + R_c)cov(\widehat{p}_A, \widehat{p}_B),$$

the numerator and the denominator of w_{opt} in (6) are given by

$$V_2 - C = V(\widehat{p}_B)(R_c^2 - RR_c) - cov(\widehat{p}_A, \widehat{p}_B)[2R_c - (R - R_c)] =$$

$$= V(\widehat{p}_B)R_c(R_c - R) - cov(\widehat{p}_A, \widehat{p}_B)(R_c - R) =$$

$$= (R_c - R)[V(\widehat{p}_B)R_c - cov(\widehat{p}_A, \widehat{p}_B)]$$

and

$$V_1 + V_2 - 2C = V(\widehat{p}_B)(R^2 + R_c^2 - 2RR_c) - cov(\widehat{p}_A, \widehat{p}_B)[2R + 2R_c - 2(R + R_c)] =$$

$$= V(\widehat{p}_B)(R_c - R)^2$$

By replacing these expressions in (6) we obtain

$$w_{opt} = \frac{V_2 - C}{V_1 + V_2 - 2C} = \frac{(R_c - R)[V(\widehat{p}_B)R_c - cov(\widehat{p}_A, \widehat{p}_B)]}{V(\widehat{p}_B)(R_c - R)^2} =$$

 ISBN 13: 978-84-613-5510-5

$$= \frac{V(\widehat{p}_B)R_c - cov(\widehat{p}_A, \widehat{p}_B)}{(R_c - R)V(\widehat{p}_B)} = \frac{R_c - \beta}{R_c - R}.$$

□

Following Theorem 2, the estimated optimum weight $\widehat{w}_{opt}$ given by (8) can be calculated as

$$\widehat{w}_{opt} = \frac{\widehat{R}_c - \widehat{\beta}}{\widehat{R}_c - \widehat{R}}, \tag{9}$$

where

$$\widehat{\beta} = \frac{\widehat{cov}(\widehat{p}_A, \widehat{p}_B)}{\widehat{V}(\widehat{p}_B)}.$$

From expression (9) we conclude that $\widehat{w}_{opt} = 1$, that is $\widehat{p}_{r.opt} = \widehat{p}_r$, if $\widehat{\beta} = \widehat{R}$, whereas $\widehat{w}_{opt} = 0$, that is $\widehat{p}_{r.opt} = \widehat{p}_{r.q}$, if $\widehat{\beta} = \widehat{R}_c$. In other words, the ratio estimator $\widehat{p}_r$ has a larger weight into the optimum estimator $\widehat{p}_{r.opt}$ as $\widehat{\beta}$ is closer to $\widehat{R}$. On the other hand, the ratio estimator $\widehat{p}_{r.q}$ has a larger weight into the optimum estimator $\widehat{p}_{r.opt}$ as $\widehat{\beta}$ is closer to $\widehat{R}_c$.

Theorem 4

The asymptotic variance of the optimum ratio estimator $\widehat{p}_{r.OPT}$ can be calculated as

$$AV(\widehat{p}_{r.OPT}) = V(\widehat{p}_A)(1 - \phi^2).$$

Proof

The asymptotic variance of $\widehat{p}_{r.OPT}$ is

$$AV(\widehat{p}_{r.OPT}) = \frac{V_1 V_2 - C}{V_1 + V_2 - 2C},$$

where the denominator, as seen in proof of Theorem 2, can be obtained as

$$V_1 + V_2 - 2C = V(\widehat{p}_B)(R - R_c)^2.$$

Next, we obtain the numerator of $AV(\widehat{p}_{r.OPT})$. For the sake of simplicity, we denote $V_A = V(\widehat{p}_A)$, $V_B = V(\widehat{p}_B)$ and $C_{AB} = cov(\widehat{p}_A, \widehat{p}_B)$. We had that

$$V_1 = V_A + R^2 V_B - 2RC_{AB},$$

$$V_2 = V_A + R_c^2 V_B - 2R_c C_{AB}$$

and

$$C = V_A + RR_c V_B - (R + R_c)C_{AB}.$$

 ISBN 13: 978-84-613-5510-5

First,

$$V_1V_2 = V_A^2 + R_c^2V_AV_B - 2R_cV_AC_{AB} + R^2V_AV_B + R^2R_c^2V_B^2$$

$$-2R^2R_cV_BC_{AB} - 2RV_AC_{AB} - 2RR_c^2V_BC_{AB} + 4RR_cC_{AB}^2.$$

The square of the covariance can be expressed as

$$C^2 = V_A^2 + R^2R_c^2V_B^2 + (R+R_c)^2C_{AB}^2 + 2V_ARR_cV_B$$

$$-2(R+R_c)V_AC_{AB} - 2RR_c(R+R_c)V_BC_{AB} =$$

$$= V_A^2 + R^2R_c^2V_B^2 + R^2C_{AB}^2 + R_c^2C_{AB}^2 + 2RR_cC_{AB}^2 + 2V_ARR_cV_B$$

$$-2RV_AC_{AB} - 2R_cV_AC_{AB} - 2R^2R_cV_BC_{AB} - 2RR_c^2V_BC_{AB}.$$

Then, the numerator of $AV(\widehat{p}_{r.OPT})$ is

$$V_1V_2 - C^2 = V_AV_B(R_c^2 - 2RR_c + R^2) - C_{AB}^2(R^2 + R_c^2 - 2RR_c) =$$

$$= (V_AV_B - C_{AB}^2)(R - R_c)^2.$$

The variance of $\widehat{p}_{r.OPT}$ can be also obtained as

$$AV(\widehat{p}_{r.OPT}) = \frac{V_AV_B - C_{AB}^2}{V_B} = \frac{V(\widehat{p}_A)V(\widehat{p}_B) - cov(\widehat{p}_A,\widehat{p}_B)^2}{V(\widehat{p}_B)}. \tag{10}$$

Replacing $V(\widehat{p}_A)$, $V(\widehat{p}_B)$ and $cov(\widehat{p}_A,\widehat{p}_B)$ in (10) by their respective expressions under SRSWOR we obtain

$$AV(\widehat{p}_{r.OPT}) = \frac{N-n}{N-1}\frac{1}{n}\left[\frac{P_AQ_AP_BQ_B - \phi^2P_AQ_AP_BQ_B}{P_BQ_B}\right] =$$

$$= \frac{N-n}{N-1}\frac{1}{n}P_AQ_A(1-\phi^2) = V(\widehat{p}_A)(1-\phi^2).$$

□

Theoretical comparison between the ratio estimator $\widehat{p}_{r.OPT}$ and the simple expansion estimator $\widehat{p}_A$ is fairly simple using Theorem 2. In fact, $\widehat{p}_{r.OPT}$ is more efficient than $\widehat{p}_A$, since $1-\phi^2 \leq 1$, and both estimators has the same performance when $\phi^2 = 0$.

Using Theorem 2, an estimator of the optimum ratio type estimator variance is

$$\widehat{V}(\widehat{p}_{r.opt}) = \widehat{V}(\widehat{p}_A)(1-\widehat{\phi}^2).$$

 ISBN 13: 978-84-613-5510-5

3 Simulation study

In this section, the proposed optimum ratio estimator $\widehat{p}_{r.opt}$ is compared numerically with alternative proportion estimators. Simulation studies are based on several simulated populations which cover a wide number of possible scenarios, including small and large proportions, small and large Cramer's V coefficients between the attribute of interest and the auxiliary attributes, etc. Simulated populations are briefly described as follows.

A total of 30 populations of $N = 1000$ units were generated to study the effect of different aspects on the estimators of a population proportion. Populations were generated as a random sample of 1000 units from a Bernoulli distribution with parameter $p = \{0.1, 0.25, 0.5, 0.75.0.9\}$, and the attributes of interest were thus achieved with the aforementioned population proportions. Auxiliary attributes were also generated by using the same distribution, but we randomly change a given proportion of values in order to the Cramer's V coefficient between the attribute of interest and the auxiliary attribute goes from 0.5 to 0.9. Since $P_A < P_B$ when $P_A = 0.25$, we also generated populations with $P_A = 0.25$ and $P_A > P_B$, which allow us to study the effect of the relation between P_A and P_B on the various estimators, specially on the estimator $\widehat{p}_r$.

For each of the 30 populations, $D = 10000$ samples were selected to compare the various estimators in terms of relative bias (RB) and relative efficiency (RE), where

$$RB = \frac{E[\widehat{p}] - P_A}{P_A} \quad ; \quad RE = \frac{MSE[\widehat{p}]}{MSE[\widehat{p}_A]},$$

$\widehat{p}$ is a given estimator and the empirical expectation ($E[\cdot]$) and the empirical mean square error ($MSE[\cdot]$) are given by

$$E[\widehat{p}] = \frac{1}{D}\sum_{d=1}^{D}\widehat{p}(d) \quad ; \quad MSE[\widehat{p}] = \frac{1}{D}\sum_{d=1}^{D}(\widehat{p}(d) - P_A)^2,$$

$\widehat{p}(d)$ denotes the estimator $\widehat{p}$ calculated at the dth simulation run. Values of RE less than 1 indicate that the estimator $\widehat{p}$ is more efficient than the customary estimator $\widehat{p}_A$, which is considered as the reference estimator in the efficiency studies.

We considered the proposed optimum ratio estimator $\widehat{p}_{r.opt}$, the ratio estimators $\widehat{p}_r$ and $\widehat{p}_{r.e}$ proposed by Rueda et al. (2010) and the difference estimator given by

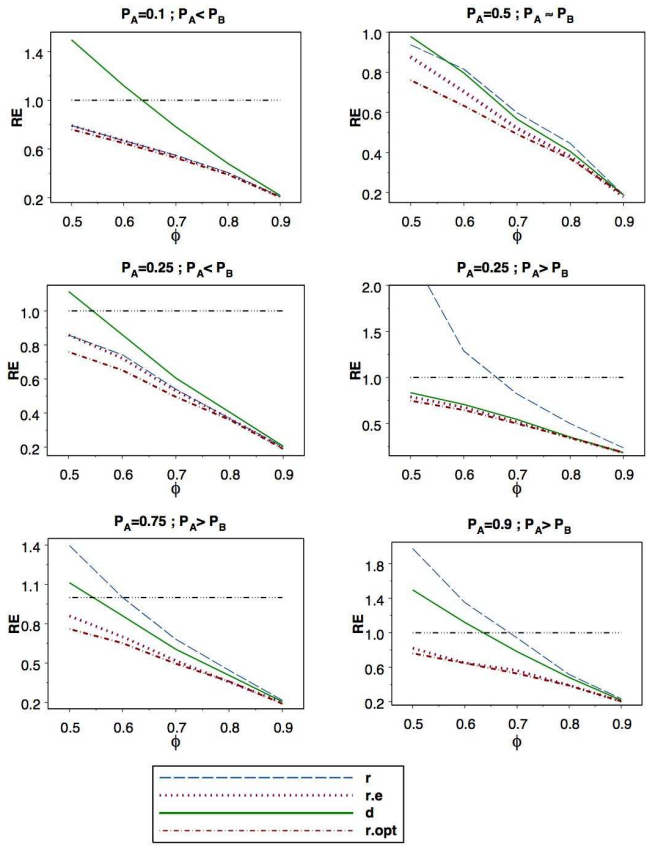
$$\widehat{p}_d = \widehat{p}_A + (P_B - \widehat{p}_B).$$

Values of RB in this simulation study are within a reasonable range, i.e, they are all less than 1% and are thus omitted. Figure 1 reports the values of RE for the various estimators and samples selected under SRSWOR. We observe that the proposed optimum estimator is more efficient than alternative estimators, whereas other estimators such as $\widehat{p}_r$ and $\widehat{p}_d$ can be less efficient than the customary estimator $\widehat{p}_A$.

 ISBN 13: 978-84-613-5510-5

4 Figures

Figure 1: Values of Relative Efficiency (RE) for the various estimators of P_A. ϕ goes from 0.5 to 0.9 and P_A goes from 0.1 to 0.9.



References

[1] S. Singh, *Advanced sampling theory with applications: How Michael "selected" Amy,*, The Netherlands, 2003.

[2] C.E. Särndal, B.Swensson and J. Wretman , *Model Assisted Survey sampling*, Springer Verlag, 1992.

 ISBN 13: 978-84-613-5510-5

[3] J.C. Deville, C.E. Särndal, *Calibration Estimators in Survey Sampling*, J. Am. Stat. Assoc. **87** (1992) 376-382.

[4] M.M. Rueda, J.F. Muñoz, A. Arcos, E. Álvarez, and S. Martínez, *Estimators and confidence intervals for the proportion using binary auxiliary information with applications to pharmaceutical studies* , Journal of Biopharmaceutical Statistics (2010)In press.

[5] C.E. Särndal, *The calibration approach in survey theory and practice*, Surv. Methodol. **33** (2007) 99-119.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A multicore pipelined algorithm for Image Sequence Analysis

A. Murli[1], D. Casaburi[2], L. D'Amore[1], A. Galletti[3] and L. Marcellino[3]

[1] *University of Naples Federico II, ITALY,*

[2] *SPACI, ITALY,*

[3] *University of Naples Parthenope, ITALY,*

emails: almerico.murli@dma.unina.it, daniela.casaburi@dma.unina.it, luisa.damore@unina.it, ardelio.galletti@uniparthenope.it, livia.marcellino@uniparthenope.it

Abstract

We describe a parallel algorithm designed for efficiently performing image sequence analysis on advanced multicore processors. The idea is to partition the sequence into ordered subsets of frames and to perform the operations on these subsets by *overlapping the execution of the tasks via pipelining*. As is usual, the effectiveness of any pipelined computation depends on the load balance between the involved tasks. In order to improve the performance gain of the pipelined algorithm tasks are distributed among multicore processors. Finally, by using the scale-space framework, each task is described in terms of non linear time dependent PDEs sharing common computational kernels. A case study is described: the segmentation of ultrasound sequences. Experiments on real data are carried out using a multicore-based parallel computer system.
Key words: Image Sequence, Pipelined computations, Multicore processors.

MSC 2000: 65J22 Inverse problems, 65Y05 Parallel computation, 68U10 Image processing, 68W10 Parallel algorithms.

1 Introduction

This paper is motivated by ongoing research activities where the authors are being involved during last years [2, 3, 4]. We describe a quite general computing methodology for an efficient image sequence analysis. Main idea is to combine the functional parallelism underlying pipelined computations and data parallelism underlying parallel computing algorithms [1, 5].
Given an image sequence, assume that we need to apply on each frame a certain number of transformations. For instance, we need to reduce the noise, then to compute

 ISBN 13: 978-84-613-5510-5

the motion field and, finally, to track the movement of some contours present on each frame. In the following we refer to these operations as *tasks*. We assume that there is a strong *dependence* between these tasks. More precisely, each task acts on the output produced by the *previous* one. As a consequence, the tasks do not act independently of each other, instead, they operate on each frame of the sequence following a prescribed order. This assumption is quite natural in sequence analysis because of the high temporal correlations that exist between frames. This occurs, for instance, in the space-time segmentation as well as in digital film restoration .
Dependence among tasks *synchronizes* tasks execution in such a way that the overall computation seems to be intrinsically sequential.
We introduce concurrency during the tasks execution by suitably *overlapping tasks execution via pipelining.* Problems with no dependencies run concurrently on the frames of the sequence. Moreover, in order to reduce synchronization overheads among tasks due to their load imbalance, each task is further parallelized and executed on multicore processors. The approach that we choose for introducing concurrency inside the pipelined algorithm takes into account both the computational cost of each task and the system architecture (a multicore multiprocessor). We consider three parallelization strategies: first strategy distributed the execution of each task among multiple cores employing a fine-grained parallelism, second strategy introduces concurrency inside task operations at a coarser level, and the last one combines the previous two.
Next section describes the way this is obtained and reports the theoretic performance analysis. Section 3 describes the case study that we consider in order to verify the feasibility of this approach. Numerical implementation is described in section 4, while section 5 concludes the paper.

2 The pipelined algorithm PA

Let us give the following:

Definition 1 [image sequence]: *Let $D \subset \Re$ be a bounded interval. Given $t \in D$, let $z(t) \equiv (x(t), y(t)) \in \Omega$, where $\Omega = \Omega_x \times \Omega_y \subset \Re^2$ is the image plane*[1]*. We define the image sequence on D as the piecewise smooth function*:

$$I_0 : t \in D \longrightarrow z\,(t) \in \Omega \longrightarrow I_0\,(z\,(t)\,, t) \equiv I_0(t) \in \Re$$

Let us assume that the interval D consists of FN distinct values, that is:

$$D = \{t_1 < t_2 < \ldots < t_{FN}\}$$

Given $t_i \in D$ and $r \in N - \{\infty\}$, we assume that on $I_0(t)$ we have to perform r *tasks*, let us say P_1, P_2, ..., P_r, to get $I_r(t)$, more precisely we have:
$P \equiv \{P_1, P_2, \ldots, P_r\}$, and:

[1]The image plane Ω should change with the acquisition time t. In practice, it is the same at each t because it refers to the rectangular plane of the image acquisition.

 ISBN 13: 978-84-613-5510-5

$$P : I_0(t) \rightarrow I_r(t)$$

Moreover, we assume that there is a strong *dependence* between these tasks: each task P_i, $i = 2, \ldots, r$ acts on the output provided by the *previous* one, i.e. P_{i-1}. This means that if we set:

$$P \equiv P_r \circ P_{r-1} \circ \ldots \circ P_1$$

then, it is:

$$\begin{array}{lll} P_1 : I_0(t) & \longrightarrow P_1(I_0(t)) & \equiv I_1(t), \\ P_2 : P_1(I_0(t)) & \longrightarrow P_2(P_1(I_0(t))) & \equiv I_2(t) \\ \ldots\ldots\ldots\ldots & \ldots\ldots\ldots & \ldots\ldots\ldots \\ P_i : P_{i-1}(I_s(t)) & \longrightarrow P_i(P_{i-1}(I_s(t)) & \equiv I_i(t) \quad i = 2, \ldots, r \quad , \quad s = 0 \ldots, i-2 \end{array}$$

In general, we have that:

$$I_s(t) = P_s(P_{s-1}(\ldots(P_1(I_0(t)))\ldots), \quad i = 2, \ldots, r \quad , \quad s = 0 \ldots, i-2$$

A straightforward approach to perform P is based on the execution of the r tasks $P_1, \ldots P_r$ on each frame of the sequence $I_0(t)$. Schematically, this is described by the following algorithm:

1. `for` $k = 1, \ldots, FN$
2. `to perform tasks` $P_1, P_2, \ldots, P_r$`, onto` $I_0(t_k)$`, that is:`
   - 2.1 `for` $j = 2, \ldots, r$
   - 2.2 `to compute` $I_j(t_k) = P_j(P_{j-1}(\ldots(P_1(I_0(t_k)))\ldots)$
   - 2.3 `endfor` j
3. `endfor` k

Image Sequence Algorithm A

Of course, in this case the computing time increases as the number of frames to process and the overall computation may be too expensive for $FN \rightarrow \infty$.
Consider the ordered subsets I_i^r made of $r < FN$ subsequent frames of the sequence:

$$I_i^r = \{I_{i-(r-1)}(t_{k-(r-1)}), \ldots, I_{i-1}(t_{k-1}), I_i(t_k)\}, \quad k = r, \ldots, FN, \quad i = 1, \ldots, r.$$

We describe a new approach to perform P which is based on a *pipelined computation*. At step k of the pipelined algorithm, the r tasks act *concurrently* onto the subset I_i^r made of r consecutive frames, as described in the following way:

1.`for` $k = r, \ldots, FN$
2. `to perform the pipelined computation on the subset` I_k^r:
   - 2.1. `for` $i = 1, \ldots, r$`,`
   - 2.2. P_i `acts onto` $I_{i-1}(t_{k-(i-1)})$
   - 2.3. `endfor` i
3. `endfor` k.

Pipelined Image Sequence Algorithm PA (middle part)

 ISBN 13: 978-84-613-5510-5

The main difference between algorithm A and the pipelined algorithm PA relies on the step 2, i.e. on the execution of the r tasks on a single frame and on the subset I_k^r, respectively. While in algorithm A the tasks are necessarily performed in a sequential way, one after the previous, because they act on the same frame, in the pipelined algorithm PA) the tasks P_i act *concurrently* on the ordered subset I_k^r, because they act on different frames. As we will describe later, this approach reduces the overall execution time by a factor depending on the size of I_k^r.
Let us now describe the first $r-1$ steps of the pipelined algorithm PA. As expected, due to the fact that for $k = 1, \ldots, r-1$ the number of available frames is less than the number of tasks to perform on these frames ($k < r$), only a part of the r tasks P_i $i = 1, \ldots, r$, may be performed. Indeed, first $r-1$ steps represent the **start-up** time of the pipelined algorithm. More precisely, it holds:

1. `for` $k = 1, \ldots, r-1$
   1.1 `for` $i = 1, \ldots, k$
   1.2 P_i `acts onto` $I_{i-1}(t_{k-i})$
2. `endfor` k.

Start up of Algorithm PA

At step $k = FN$ of the pipelined algorithm PA, P_r operates onto the frame corresponding to the acquisition time $t_{FN-(r-1)}$. This means that it remains to still process $r-1$ frames. This is done within further $r-1$ steps, as described in the following:

1.`for` $k = 1, \ldots, r-1$,
   P_r `acts onto` $I_{r-1}(t_{FN-(r-(k+1))})$.
2. `endfor` k.

Final stage of Algorithm PA

Hence, the pipelined computation consists of three main stages: the start-up, made of $r-1$ steps, the central computation, made of $FN - r + 1$ steps, and the final stage, made of $r-1$ steps.

2.1 The Parallel Pipelined Algorithm PPA

We consider three parallelization strategies:

1. first strategy distributes the execution of each task of PA among multiple cores employing a fine-grained parallelism of PA operations. We refer to this strategy as: **data parallel pipelined algorithm (data PPA)**,

2. second strategy decomposes the execution of PA among multiple processors introducing concurrency inside PA at a coarser level. We refer to this algorithm as: **functional parallel pipelined algorithm (functional PPA)**,

 ISBN 13: 978-84-613-5510-5

3. the last one combines the previous two. The pipelined algorithm PA is first decomposed among multi processors (as for the second strategy). Then each task of PA is decomposed among multi cores (as for the second strategy). This approach balances the computational load of each task as for the data parallelism improving performance gain of the functional PPA. We refer to this strategy as: **hybrid parallel pipelined algorithm (hybrid PPA)**.

2.1.1 Performance analysis

We introduce the following quantities:

- T_{P_i}, $i = 1, \ldots, r$: computing execution time needed to perform P_i on one frame,
- $T = max_{i=1,\ldots,r}\{T_{P_i}\}, i = 1, \ldots, r$: the maximum of T_{P_i}, $i = 1, \ldots, r$ on one frame
- T_{PA}: execution time of the middle part of PA,
- T_{PA}^{fin}: execution time of the final stage of PA algorithm,
- T_{PA}^{in}: execution time of *start up* of PA algorithm.
- $T_{PPA}^{fun} = T_{PA}^{in} + T_{PA}^{fin} + \sum_{k=r}^{FN} T_{max}$: computing time of PPA algorithm (without I/O),
- $T_{PPA}^{data}(nproc)$: computing time of PPA algorithm (without I/O) on *nproc* core,
- $T_{seq}^{FN} = FN \cdot \sum_{i=1}^{r} T_{P_i}$: computing time required by a serial image sequence computation.
- $S_{nproc}^{data} = \frac{T_{seq}^{FN}}{T_{PPA}^{data}(nproc)}$: speed up of data PPA algorithm on *nproc* core.
- $S_{nproc}^{fun} = \frac{T_{seq}^{FN}}{T_{PPA}^{fun}(nproc)}$: speed up of functional PPA algorithm on *nproc* core.
- $T_{Pi}(nproc_i)$: computing time of task P_i on $nproc_i$ core.
- $S_{nproc_i}^{P_i}$: speed up of task P_i on $nproc_i$ core.

The following results can be proved:

Proposition 1 (data PPA) It holds that:

$$S_{nproc}^{data} \leq S_{nproc}^{max} = max\{S_{nproc}^{P_i} \ , \quad i = 1, \ldots, r\}$$

Proof:

$$S_{nproc}^{data} = \frac{T_{seq}^{FN}}{T_{PPA}^{data}} = \frac{FN \cdot \sum_{i=1}^{r} T_{P_i}}{FN \cdot \sum_{i=1}^{r} T_{P_i}(nprocs)} = \frac{FN \cdot \sum_{i=1}^{r} T_{P_i}}{FN \cdot \sum_{i=1}^{r} \frac{T_{P_i}}{S_{nproc}^{P_i}}} \leq$$

 ISBN 13: 978-84-613-5510-5

$$\leq \frac{FN \cdot \sum_{i=1}^{r} T_{P_i}}{FN \cdot \sum_{i=1}^{r} \frac{T_{P_i}}{S_{nproc}^{max}}} = S_{nproc}^{max} \qquad \square$$

Proposition 1 states that the performance of data PPA is at most equals to the highest performance of the r tasks P_i. Then, the highest performance gain we expect from data PPA equals the maximum number of computing nodes employed for the parallel execution of each task. Moreover, as it is usual for parallel algorithms implementing data parallelism, as the processor number grows, overheads predominate causing speed-up degradation. This means that, in case of a fixed-size application, due to the limited amount of available parallelism, there exists an *optimal* number of processing nodes and each additional processor contributes slightly less or do not.

Concerning the second strategy, it can be proved that:

Proposition 2 (functional PPA): *It is*:

$$\lim_{FN\to\infty} S_{nproc}^{fun} = \lim_{FN\to\infty} \frac{T_{seq}^{FN}}{T_{PPA}^{fun}} = \frac{\sum_{i=1}^{r} T_{P_i}}{T_{max}}$$

Proof:

$$T_{seq}^{FN} = FN \sum_{i=1}^{r} T_{P_i}$$

and

$$T_{PPA}^{fun} = T_{PA}^{in} + T_{PA}^{fin} + \sum_{k=r}^{FN} T_{max} = T_{PA}^{in} + T_{PA}^{fin} + (FN + r - 1) \cdot T_{max}$$

It holds:

$$\lim_{FN\to\infty} \frac{T_{seq}^{FN}}{T_{PPA}^{fun}} = \lim_{FN\to\infty} \frac{FN \cdot \sum_{i=1}^{r} T_{P_i}}{T_{PA}^{in} + T_{PA}^{fin} + (FN + r - 1) \cdot T_{max}} = \frac{\sum_{i=1}^{r} T_{F_i}}{T_{max}} \quad \square$$

Proposition 2 states that performance of functional PPA depends on the computational cost of tasks P_i. In particular, it depends on a suitable load balance among tasks. Indeed, it holds:

Corollary 1(functional PPA): *Assume that* $\forall\ i = 1, \ldots, r$, $T_{P_i} = T_{max}$, *that is execution time of tasks* P_i *is the same, then:*

$$\lim_{FN\to\infty} S_{nproc}^{fun} = r$$

Proof:

$$T_{seq}^{FN} = r \cdot FN \cdot T_{max}$$

then

$$\lim_{FN\to\infty} S_{fun} = \lim_{FN\to\infty} \frac{T_{seq}^{FN}}{T_{PPA}^{fun}} = \lim_{FN\to\infty} \frac{r \cdot FN \cdot T_{max}}{T_{PA}^{in} + T_{PA}^{fin} + (FN + r - 1) \cdot T_{max}} = \frac{r \cdot T_{max}}{T_{max}} = r \quad \square$$

 ISBN 13: 978-84-613-5510-5

This means that the highest performance gain that we may expect for functional PPA is equal to r, the number of tasks, and this occurs if all tasks require about the same computing time. On the contrary, it is:

Corollary 2 (functional PPA): *Assume that there exist only one among the r tasks, let us denote* $P_{\widetilde{i}}$*, whose time execution is*

$$T_{P\widetilde{i}} \geq \sum_{i \neq \widetilde{i}} T_{Pi}$$

then

$$\lim_{FN \to \infty} S^{fun}_{nproc} \leq 2$$

Proof:

$$\sum_{i=1}^{r} T_{P_i} = T_{P\widetilde{i}} + \sum_{i \neq \widetilde{i}} T_{Pi} \leq T_{P\widetilde{i}} + T_{P\widetilde{i}} = 2 \cdot T_{P\widetilde{i}}$$

It follows that:

$$\lim_{FN \to \infty} \frac{T^{FN}_{seq}}{T^{fun}_{PPA}} = \lim_{FN \to \infty} \frac{FN \cdot \sum_{i=1}^{r} T_{P_i}}{T^{in}_{PA} + T^{fin}_{PA} + (FN + r - 1) \cdot T_{P\widetilde{i}}} \leq \frac{2T_{P\widetilde{i}}}{T_{P_{\widetilde{i}}}} \leq 2 \quad \square$$

This means that the lowest performance we may expect for functional PPA occurs when one task is more time consuming than the others.

Such results suggest us to combine data PPA and functional PPA in order to take advantages of both the first approach and the second one. Indeed, using functional parallelism we overlap the execution of tasks P_i, while introducing data parallelism inside their computations, we balance their computing time.
Indeed, it can be proved that:

Proposition 3 (hybrid PPA): *Let* $1 \leq \widehat{i} \leq r$ be such that $T_{P\widehat{i}}(nproc_{\widehat{i}})$ is the maximum execution time of the r parallel tasks P_i, i.e.

$$T_{P\widehat{i}}(nproc_{\widehat{i}}) = \max_{i=1,\ldots,r} T_{P_i}(nproc_i) = \frac{T_{P\widehat{i}}}{S^{P\widehat{i}}_{nproc_{\widehat{i}}}}$$

then it is:

$$S^{hyb}_{nproc} = \frac{\sum_{i=1}^{r} T_{P_i}}{T_{P\widehat{i}}(nproc_{\widehat{i}})} \leq nproc_{\widehat{i}} \cdot S^{fun}_{nproc}$$

Proof:

$$S^{hyb}_{nproc} = \frac{\sum_{i=1}^{r} T_{P_i}}{T_{P\widehat{i}}(nproc_{\widehat{i}})} \leq \frac{\sum_{i=1}^{r} T_{P_i}}{\frac{T_{P\widehat{i}}}{nproc_{\widehat{i}}}} = nproc_{\widehat{i}} \frac{\sum_{i=1}^{r} T_{P_i}}{T_{P\widehat{i}}} = nproc_{\widehat{i}} \cdot S^{fun}_{nproc} \quad \square$$

 ISBN 13: 978-84-613-5510-5

The following result is a straightforward consequence of **Corollary 2** and **Proposition 3**:

Corollary 3 (hybrid PPA): *Assume that* $\forall\ i = 1, \ldots, r$, *the execution time of parallel tasks* P_i *on nproc cores is the same, that is*

$$T_{Pi}(nproc_i) \equiv T_{max}(nproc) \quad , \quad i = 1, \ldots, r$$

then:

$$\lim_{FN \to \infty} S^{hyb}_{nproc} = nproc \cdot r$$

Corollary 3 states that, using the hybrid PPA, the highest performance gain we expect with respect to Algorithm A is *nproc* times the highest performance gain we expect when using the functional PPA. In other words, the hybrid PPA actually may provide a significant improvement with respect to the other two strategies. Such results are confirmed by the experiments we carried out and described in Section 4.

3 A case study

As case study we consider the segmentation of medical structures from degraded ultrasound images. We focus on detection and delineation of the expansion of the ventricle chamber at each frame of ultrasound image sequences. Besides the presence of the speckle noise that affects ultrasound images, the problem is to detect and delineate the expansion of the left ventricle (LV) chamber at each frame of the sequence. A major limitation of most segmentation models in ultrasound imaging is to detect the ventricle contour in the vicinity of the cardiac valve, mainly in those frames where it is open and its position is almost confused. Indeed, in these frames we are faced with contours *with missing parts*. This application consists of $r = 3$ tasks P_i, $i = 1, 2, 3$. These are: P_1 the despeckle plus contrast enhancement, P_2 is the recovery of missing edges via the optic flow computation, and P_3 which is the LV segmentation. Finally, $I_3 \equiv I_3(C(t))$, that is the image brightness of the LV contour $C(t)$.
Starting from multiscale analysis provided by scale-space theory and variational methods, despeckling, segmentation, and optic flow computation can be described by the following non linear time dependent PDE (see [2, 3, 4] and their references):

The scale space image analysis: *We consider the following PDE*:

$$\frac{\partial I}{\partial \tau} = |\nabla I| \nabla . \left(g(|\nabla I|) \frac{\nabla I}{|\nabla I|} \right) - \alpha K^* \left(KI - I^0 \right) \quad \tau \geq 0 \tag{1}$$

where initial and boundary conditions change according to the task (usually they are Dirichlet or Neumann conditions).
The term $g(v)$ *is a non increasing real function such that* $g(v) \to 0$ *while* $v \to \infty$ *and it is used for the enhancement of the edges. In general the Perona-Malik function is considered:* $g(v) = 1/(1+v^2/\beta)$, $\beta > 0$. *The function* $g(|\nabla I|)$ *is replaced by its smoothed*

 ISBN 13: 978-84-613-5510-5

version $g(|\nabla G_\sigma * I|)$, where G is a smoothing kernel, e.g. the Gauss function, and $*$ is the convolution operator. Parameter α is the so-called regularization parameter. K is the blurring operator and K^* is the transpose.

We perform, for each $t \in D$, the following tasks:

$$P_1 : I_0(t) \rightarrow I_1 \rightarrow P_2(I_1) = I_2(t) \rightarrow P_3(I_2(t)) = I_3(t)$$

The PDE was discretized using the *semi implicit* scheme respect to the scale derivatives. This choice leads to inconditionally stable numerical schemes. For the spatial discretization, we use finite differences for despeckling and optic flow models and the (complementary) finite volumes for the segmentation problem. At each scale step we have to solve a linear system. We use the Additive Operator Scheme (AOS) for depeckling. Regarding optic flow and segmentation models, we use the GMRES iterative method equipped with the algebraic recursive multilevel preconditioner (ARMS).

4 Experimental results

Experiments were performed using 16 blade Dell PowerEdge M6000 each made of 2 processor quad core Intel Xeon E5410@2.33GHz (64 bit) connected by the high performance network InfiniBand. Data movement between tasks is implemented trough read/write operations. We carried out many experiments aimed at monitoring both the performance of each task and of the pipelined algorithm. We report here main results. The sequence is made of $FN = 26$ frames of size 300×300 (1 cardiac cycle). Tasks are not load balanced. In particular despeckle task is the cheapest (it needs about *40 secs* on 1 core and the highest performance is reached using 8 cores with *24.6 secs*) and the segmentation is the most time consuming (it requires more than *400 secs* on 1 core and *78 secs* with *28* cores). Regarding the optic flow, it needs about *100 secs* on 1 core and *4 secs* on 42 cores) (see Figures 1-2). We get the following results:

1. **Algorithm A**: $T_{seq}^{FN} = 773.92$ *sec.*, frame rate/sec $\frac{FN}{T_{sec}} = 0.0336$.

2. **Functional PPA**: in agreement with **Corollary 2** we expect a low speed up, indeed $T_{PPA}^{fun} = 705.724$ *sec* and the frame rate is 0.037. $S_{fun} = 1.096$. Performance gain with respect to Algorithm A is of $8,80\%$.

3. **Data PPA**: because each task has a fixed size, the *optimal* configuration for data PPA is reached using 16 cores for segmentation, 6 for the optic flow computation and 2 for despeckle task. We have: $T_{PPA}^{data} = 122.4472$ *sec* and $S_{24}^{data} = 6.3$. The frame rate is 0.212, while the performance gain with respect to Algorithm A is of $84,17\%$.

4. **Hybrid PPA** (see Figure 3), gets the greatest performance using $nproc = 30$ cores distributed as follows: 2 for despeckle and contrast stretching, 6 for the optic flow computation and 24 for the segmentation tasks. We get $T_{PPA}^{hyb} = 82.654$ *sec*

 ISBN 13: 978-84-613-5510-5

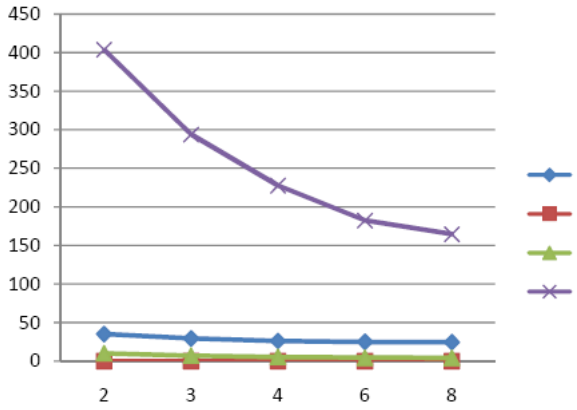


Figure 1: *Execution Time (in seconds) of tasks P_i versus the core number. Circle denotes task P_1 (despeckle), Box denotes task $P_{1.2}$ (contrast enhancement), Triangle denotes task P_3 (optic flow computation) and Star denotes the task P_4 (Segmentation).*

with a frame rate of 0.314 frame/sec. In this case the speed up $S_{hyb} = 9.3$ and performance gain with respect to Algorithm A is of 89.29.%. Observe that, in agreement with **Corollary 3**, we have $T_{P\widehat{i}}(nproc_{\widehat{i}}) = \max_{i=1,\ldots,r} T_{P_i}(nproc_i) = T_{P3}(24)$, and $S_{hyb} \leq 3 \cdot 24$.

Performance analysis needs to also take into account *data movements* between tasks (data I/O). We do not go into details of this issue, because it can be accomplished in different ways accordingly to the computing architecture. As a consequence, the *sustained* performance of PPA algorithms depends on the computing environment and on the overheads introduced by the communication network.

5 Conclusions

We describe an algorithm for image sequence analysis. Main idea underlying this approach is to perform the operations on each frame of the sequence in the same way the Arithmetic Logic Unit (ALU) concurrently performs floating point operations on large data sets. By *overlapping the execution of the tasks via pipelining, along the frame sequence and, by concurrently performing operations of each task,* the overall execution time significantly reduces. As is known, the performance of any pipelined computations depends on a suitable synchronization among the execution of each task and improves as the length of the array sequence increases. By introducing data parallelism inside the execution of each task we get a suitable load balance among tasks and a significant time reduction with respect to any sequential computation. Finally, we validate the performance also using the number of frame per second (throughput) that measures the rate at which the frames are processed by the algorithm. Using this approach, the throughput scales of about 90% with respect to that of any sequential computation.

 ISBN 13: 978-84-613-5510-5

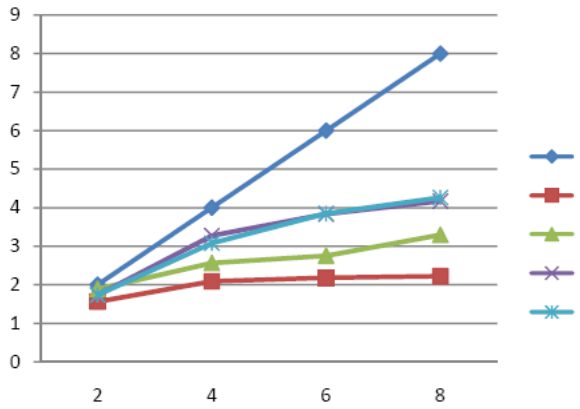


Figure 2: *Speed Up tasks P_i versus the core number. Circle denotes task P_1 (despeckle), Box denotes task $P_{1.2}$ (contrast enhancement), Triangle denotes task P_3 (optic flow computation) and Star denotes the task P_4 (Segmentation).*

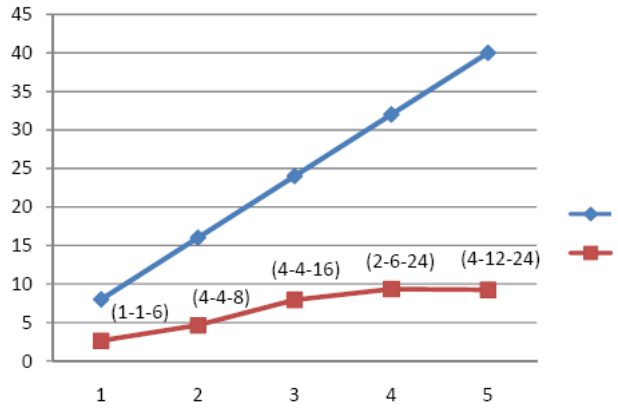


Figure 3: *Speed up of Hybrid PPA versus the number of nodes(8 core). Box refers to teh speed up line and Circle denotes the Ideal Speed up. At each box of the speed up line between parenthesis is reported the number of cores employed for Despeckle+Contrast Enhancement-Optic Flow Computation-Segmentation, respectively.*

References

[1] F. Buschmann, K. Henney, D. C. Schmidt - *Pattern-Oriented Software Architecture Volume 4: A Pattern Language for Distributed Computing* , 2007 Kindle Edition.

[2] D. Casaburi, L. D'Amore, L. Marcellino, A. Murli - *Real Time ultrasound image sequence segmentation on multicores*, Parallel Computing: From Multicores and GPU's to Petascale , Advances in Parallel Computing, Vol. 19, Eds. B. Chapman, F. Desprez, G.R. Joubert, A. Lichnewsky, F. Peters and T. Priol, 2010

 ISBN 13: 978-84-613-5510-5

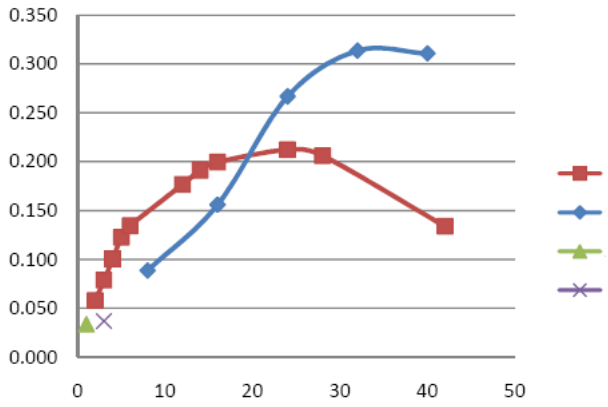


Figure 4: *Throughput versus the core number. Triangle denote throughput of Algorithm A, Box denote the throughput of data PPA, Star denotes the throughput of functional PPA and circle denotes the throughput of hybrid PPA.*

[3] D. Casaburi, L. D'Amore, L. Marcellino, A. Murli - *Ultrasound Image Segmentation via Motion Estimation*, proceedings of ENUMATH 2009, June 29-July 3, Uppsala, Sweden, LNCS, Springer-Verlag, in press.

[4] L. D'Amore, L. Marcellino, A. Murli - *Image sequence inpainting: a numerical approach for blotch detection and removal via motion estimation*, JCAM,Vol. 198, N. 2, pp. 396-413, 2007.

[5] C. Isaacson - *Software Pipelines and SOA: Releasing the Power of Multi-Core Processing*, 2008 Kindle Edition.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Applications of the Extended Euclidean Algorithm to Privacy and Secure Communications

J.A.M. Naranjo[1], J.A. López-Ramos[2] and L.G. Casado[1]

[1] *Department of Computer Architecture and Electronics, University of Almería*

[2] *Department of Algebra and Mathematical Analysis, University of Almería*

emails: jmn843@ual.es, jlopez@ual.es, leo@ual.es

Abstract

The Extended Euclidean algorithm provides a fast solution to the problem of finding the greatest common divisor of two numbers. In this paper, we present three applications of the algorithm to the security and privacy field. The first one is a method for controlling the disclosure of discrete logarithm-based public keys. It can be used to privately deliver a public key to a set of recipients with only one multicast communication. The second one is an authentication mechanism to be used in scenarios in which a public-key infrastructure is not available. Finally, the third application of the Extended Euclidean algorithm is a zero-knowledge proof that reduces the number of messages between the two parts involved, with the aid of a central server.

Key words: secure group communication, authentication, zero-knowledge proofs
MSC 2000: 11-04, 11Z05

1 Introduction

Multicast communications allow a host to simultaneously send information to a set of other hosts, avoiding the establishment of point-to-point connections with all of them. IP multicast technologies (which use routing techniques at a low level over a network, such as the IGMP protocol) have not achieved the expected success due to several reasons (need for compatible routers, implantation costs, lack of support from Internet providers, etc.). As a recent alternative, application level multicast has taken over, since it offers the same functionality at a lower cost and easier deployment. Instead of requiring physical deployment a logical network is built, and hosts resend messages themselves.

Multicast communications can be either *one-to-many* if the source of the transmitted data is one entity only over time (such as IPTV or P2PTV services) or *many-to-many*, if several clients, or all, act as a source of data. Multiconferences are an example of this (strictly, each data source establishes a one-to-many multicast communication).

 ISBN 13: 978-84-613-5510-5

There are services that take advantage of multicast but need to keep communications private. Those technologies that make it possible are known as *secure multicast*. Applications of secure multicast are, among others, pay-per-view IPTV or P2PTV, private multiconferences (oriented to business, politics or even military affairs), or any private service that involves several participants or clients.

The typical approach to establish secure multicast communications is to agree on one or several symmetric encryption keys to encrypt messages (depending on the topology and size of the network). However, the key, or keys, must be renewed periodically to prevent attacks from outsiders, or even insiders.

Depending on how key distribution and management are carried out, secure multicast schemes are divided into centralized and distributed. Centralized schemes depend directly on a single entity to distribute every cryptographic key. In a distributed approach, key distribution is more complex, usually involving entities that act as local sub-servers and manage subgroups of users. Full or partial message re-encryption is needed in some cases. The following paragraphs introduce some well known solutions.

The *Secure Lock* centralized solution is proposed in [1]. It is based on the Chinese Remainder Theorem. A drawback are the inefficient computations required at the Key Server side on each key refreshment: the computation time needed quickly becomes excessive when the number of members grows [2].

RFC 2627 [3] presents some approaches to the problem. Among all, the *Hierarchical Tree Approach* (HTA) is the recommended option. It uses a logical tree arrangement of the members in order to facilitate key distribution. The benefit of this idea is that the storage requirement for each client and the number of transmissions required for key renewal are both logarithmic in the number of members.

In [4], a divide-and-conquer extension to Secure Lock is proposed. It combines the Hierarchical Tree Approach and the Secure Lock: members are arranged in a HTA fashion, but Secure Lock is used to refresh keys on each tree level. Therefore, the number of computations required by Secure Lock is reduced.

IOLUS [5] is a well known framework designed for the secure multicast problem. Nodes are physically distributed in subgroups, which are organized on a tree fashion. Some special trusted nodes handle the subgroups and serve as gateways among them. It supports huge sets of members, due to its distributed nature. Since IOLUS is a framework, not a protocol, the key refreshment scheme used within subgroups is not stated. Any scheme can be used.

An IETF Working Group, MSEC [6], is currently working in a set of protocols to standardize secure multicast. They are focusing, in an initial stage, in IP-layer centralized multicast, assuming the presence of groups and a single trusted entity in each one.

These technologies make a good job assuring privacy and (in most cases) efficient key refreshment. However, they do not cover other aspects such as authentication or trust among peers. This paper presents a secure multicast solution for centralized scenarios that provides:

1. private communications and efficient key refreshment,

 ISBN 13: 978-84-613-5510-5

2. key server messages authentication, and
3. validation among peers.

Three different and complementary schemes are proposed in order to achieve the proposed goals. Depending on the scenario and its necessities the schemes can be implemented along with the others or on their own.

The paper is organized as follows. Section 2 describes the scenario conditions that are assumed for our solution. Section 3 presents the key refreshment scheme. Section 4 introduces the scheme for authentication of the key server. The authentication among hosts scheme is proposed in Section 5. Finally, the conclusions of the paper are presented in Section 6.

2 Scenario

The target scenario is the following: private communications must be established within a restricted group. There is a central server that manages the key management issues. From now on, we will refer to the server as *Key Server*, and to the clients as *members*. Depending on the nature of the service, communications can be either one-to-many or many-to-many.

In any case, *forward secrecy* must be maintained. This requirement implies that a member which leaves the network (i.e. her membership expires) should not be able to decrypt any ciphered information transmitted after her exit, and forces to refresh the encryption keys whenever a member leaves the network. Some services may require *backward secrecy*: an arriving member should not be able to decrypt any ciphered information transmitted before her arrival. This imposes, again, a refreshment of the keys when a member enters the system. These two restrictions may become an efficiency problem if the churn rate (joins and leaves) is too high. The scheme proposed here is efficient enough to cope with high churn rates, as will be shown next.

Obviously, the security and privacy features of an application level secure multicast solution should not only be restricted to private communications. Authentication is a key issue, too. Members should have a way to check that the source of a message is a trusted entity, either if the source is the Key Server or other member.

3 Controlled disclosure of public keys within closed groups

The first scheme we present allows to establish private group communications. The proposed approach: the Key Server owns an asymmetric key pair (of an encryption algorithm based on the discrete logarithm problem) and discloses the public key only to the members of the restricted group. Communications from the Key Server are encrypted with its private key. It is clear that only the members of the group will be able to decrypt the messages. We have named this solution *controlled disclosure of a public key*.

The usual method to publish public keys is the use of public key certificates. This is extremely useful when the disclosure process involves two participants: the owner of

 ISBN 13: 978-84-613-5510-5

the public key and the recipient. For more participants the process can be repeated, obviously. What the present scheme tries to solve is how to simultaneously disclose a public key to a selected audience only, while preserving security and efficiency. What's more, the process should be lightweight enough to be repeated as many times in time as required for key refreshment purposes. This problem appears in services such as pay-per-view IPTV or P2PTV and multiconferences.

The most relevant features of the scheme are:

- Only one message is generated per key refreshment.
- Suitable for all topologies. No need for node hierarchies, though they can be supported.
- No need for message re-encryption.
- Only one secret piece of info is held by each client. We call this pieces *member tickets*.
- Cost-effective and easy to deploy.

The scheme is described next. Let us assume there are n members at a given time in the group, and that the Key Server generates an asymmetric key pair of the form:

$$K_{pub} : g, m, g^k \ mod \ m$$
$$K_{priv} : k$$

which is an Elgamal key [7]. K_{pub} is the key to be disclosed.

When a member i joins, the Key Server assigns it a member ticket, x_i. Every ticket is a large prime[1] and is communicated to the corresponding member under a secure channel: SSL/TLS, for example. This communication is made once per member only, so it does not affect global efficiency. All tickets must be different from each other, at least during a relatively wide period of time. Note that x_i is known only by its owner and the Key Server, and K_{pub} is shared by all members and the Key Server.

Generation of (K_{pub}, K_{priv}) and distribution of K_{pub} are done as follows.

1. The Key Server selects:
   - m and p, large prime numbers, such that $m - 1 = p \cdot q$.
   - k and δ, such that $\delta = k + p$ and $\delta < x_i$, for every $i = 1 \dots n$.
   - g that verifies $g^p = 1 \ mod \ m$ (such a value is easy to calculate[2]).
2. The Key Server calculates $L = \prod_{i=1}^{n} x_i$. L is kept private in the Key Server.
3. The Key Server finds u, v, by means of the Extended Euclidean Algorithm [8], such that
$$u \cdot \delta + v \cdot L = 1 \tag{1}$$

[1]Strictly, it is sufficient that all x_i are coprime and greater than δ. In that case, however, it would be necessary that every x_i has a large prime factor in order to make the factorization of L harder (L will be introduced shortly).

[2]Once the Key Server has chosen $m = p \cdot q + 1$, a value a is chosen satisfying that $m - 1$ is the least integer such that $a^{m-1} \ mod \ m = 1$ (that is, a is a primitive value from $\mathbb{Z}_m$). Then $g = a^q \ mod \ m$.

 ISBN 13: 978-84-613-5510-5

4. The Key Server multicasts (makes public) $g,\ m$ and u on plain text.
5. Each member i calculates $u^{-1}\ mod\ x_i = \delta$ and $g^{\delta}\ mod\ m = g^k\ mod\ m = K_{pub}$. The length of K_{pub}, by definition, can not exceed that of m.

New values for $m,\ g,\ p$ and/or k must be chosen for each refreshment of (K_{pub}, K_{priv}). Note that $\delta,\ u$ and v depend on them and will change as they do.

Fortunately it is not necessary, for successive refreshments, to recompute L from scratch if there were joins or leaves: L should be multiplied by the incoming members' tickets, and divided by those of the leaving members. That speeds the process up. Finally, for security reasons, the Key Server might decide to refresh (K_{pub}, K_{priv}) after a long period of time with no members joining or leaving.

Since the use of asymmetric key encryption along with large pieces of data may be inefficient the *key hierarchy* solution can be adopted. A symmetric key is encrypted by the Key Server with its private key, and delivered to the set of members. Data messages are encrypted using the symmetric key. Members can decrypt data messages after receiving the Key Server's public key and the symmetric key. For security and efficiency reasons, the symmetric key may be refreshed at a higher frequency than the asymmetric key pair. Examples of this technique are shown in [9]. An additional benefit of using a key hierarchy is the possibility to establish both one-to-many and many-to-many communications: both the Key Server and every member know the symmetric key and may use it to encrypt its own messages.

3.1 Proof of correctness for the disclosure scheme

Given that $\delta < x_i,\ i = 1 \dots n$ and with every x_i prime (or coprime at least), it is clear that:

$$gcd(\delta, x_i) = 1,\ \text{for every } i = 1, \dots, n \tag{2}$$

and hence,

$$gcd(\delta, L) = 1 \tag{3}$$

Equation (3) ensures, by the Extended Euclidean Algorithm, the existence of $u, v \in \mathbb{Z}$ such that $\delta \cdot u + v \cdot L = 1$, from where it is deduced that $\delta \cdot u \equiv_{x_i} 1$ and so $u^{-1} \equiv_{x_i} \delta$, for every $i = 1, \dots, n$. The Chinese Remainder Theorem guarantees that the solution for $u^{-1}\ mod\ x_i = \delta$ and $\delta < x_i$, for every $i = 1, \dots, n$ is unique.

The value $K_{pub} = g^k\ mod\ m$ is obtained as shown next:

$$\begin{aligned} g^{\delta} &\equiv_m g^{k+p} \\ &\equiv_m g^k \cdot 1 \\ &\equiv_m g^k \end{aligned} \tag{4}$$

g is public, but the use of δ assures that an outsider will not be able to guess k and, therefore, K_{pub}.

 ISBN 13: 978-84-613-5510-5

3.2 Security and scalability considerations

Security in the distribution of K_{pub} relies on the unfeasibility of calculating the right δ in a reasonable time if a valid x_i is not known by the attacker (recall that values for Eq.(1) are unique). The privacy of k and p is guaranteed if:

- a sufficiently large value is chosen for m,
- p and q have a similar bitlength (recall that $m - 1 = p \cdot q$).

In that case factorizing $m - 1$ will be more difficult. Additionally, a strong prime can be chosen for m.

Note that the product L is not public in order to make attackers' work more difficult. In case L was discovered and factorized by an attacker, she would gain access to every member ticket. But such a factorization is impractical by means of a brute force attack. A legal member, say i, might be tempted to factorize $\frac{u \cdot \delta - 1}{x_i}$. If she was successful, she would obtain, again, every other ticket included in L. The problem of factorizing such a value, however, is equivalent to that of factorizing L.

Nevertheless, there is a security measure that must be taken when a new member joins: she should not be assigned a previously used ticket (at least recently). This is done to prevent the old owner to keep intercepting refreshment messages and using her old ticket to discover the secret.

Regarding scalability, we can observe that L will be large, given that $L = \prod_{i=1}^{n} x_i$. So will be u (recall Eq. (1)). For n members and b-bits tickets the maximum length of L is then $n \cdot b$ bits. That is also the maximum length of u. As an example, for $b = 64$ and $n = 1000$, u will be 64000 bits long at most, i.e. $\approx$ 8 KBs. Though that is an affordable message length for many devices (requirement 4), a shorter message would be desirable.

The solution that allows to overcome these problems consists of dividing the set of members into subgroups and delivering the same K_{pub} to all of them. Assume there are s disjoint subgroups, each one with a similar number of members. Still, the join and leave operations require the whole set of members to obtain a new key, therefore s refreshment messages (g, m and the corresponding u) must be computed and multicasted now; each one for a different subgroup. The final bandwidth requirement does not change, but adopting this approach brings many benefits which are discussed next.

First, for a fixed number of members the length of u values decreases linearly as the number of subgroups increases. In the previous example, arranging the same audience in 20 groups of 50 members would yield 20 messages of 3200 bits = 400 Bs maximum, each one shorter than a typical X.509 certificate. Shorter messages will be handled more easily and quickly by the recipients. This means less hardware requirements.

Second, the message generation process that takes place at the Key Server can be sped up. Every different u can now be computed by a separate process, which may run concurrently with the others. This is specially appropriate for nowadays multi-core computers. The whole process can be sped up by nearly s times if the software is properly tuned.

 ISBN 13: 978-84-613-5510-5

This subgroup approach provides a better scalability, allowing to increase the maximum number of clients that can be handled. As a remark, users should be assigned to subgroups in a balanced way, in order to keep refreshment messages as short as possible. This raises other issues, such as the problem of rebalancing subgroups after a leave avalanche, for example.

3.3 Communication with different groups

There are scenarios that require separate communications with different groups of members. Examples are different pay-per-view channels in the same TV platform and different private multiconferences managed by the same Key Server. Handling this situation is easy: the Key Server only needs to maintain a different key pair for each group. Every join or leave event will imply the refreshment of the affected group's key pair only. Figure 1 depicts this situation. It can happen that a member is enlisted in two or more

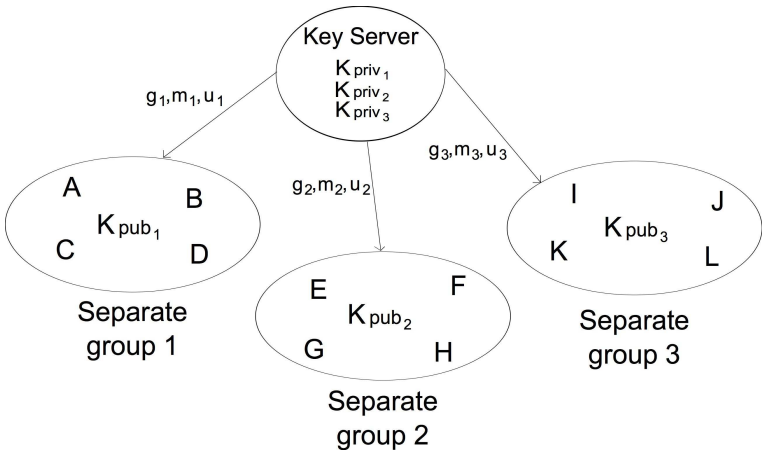


Figure 1: Managing different groups. Capital letters denote members.

groups at the same time (that is usually the case of pay-per-view channel packages, for example). It is clear that a join or leave of that member would require a refreshment in every group she belongs to.

3.4 Simulation

We have developed a Java implementation of the scheme in order to perform simulations and obtain execution times. The BigInteger Java class was used for handling large numbers, and the Miller-Rabin test was employed for primality tests. Figures 2 and 3 show execution times for the algorithm in Section 3, both in the Key Server and in a member, for different group sizes and ticket lengths. They were obtained in a Intel Core 2 Duo processor at 2,26 GHz with 3 MB of L2 cache and 2 GB of RAM.

Two main conclusions can be extracted from the Key Server times. First, key pair refreshment messages are computed very fast, excepting in the case of 2048 bits. This means that the scheme can be applied to a wide variety of scenarios. Second, execution

 ISBN 13: 978-84-613-5510-5

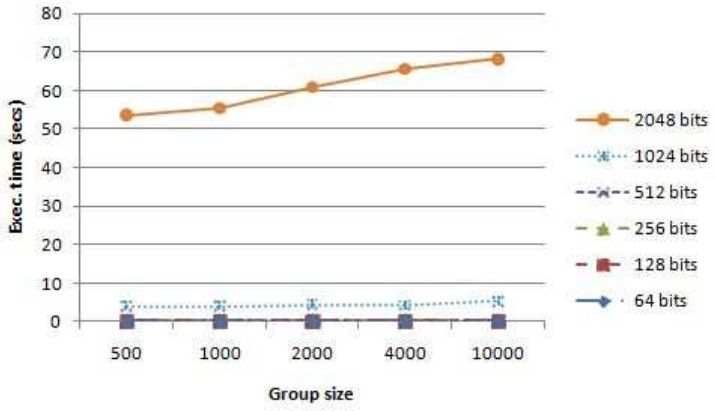


Figure 2: Key Server execution times for different ticket lengths and network sizes.

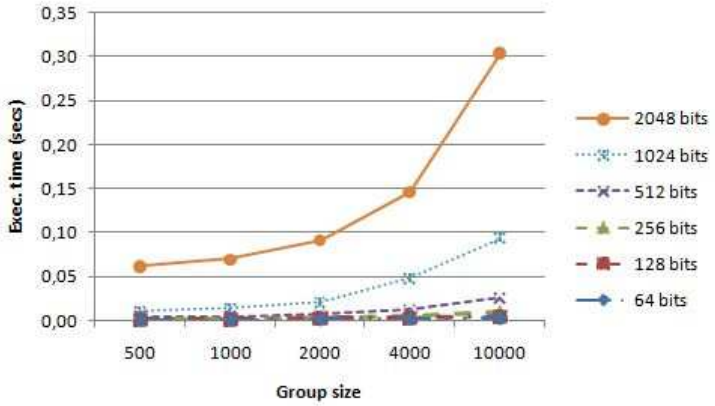


Figure 3: Member execution times for different ticket lengths and network sizes.

times are mainly affected by ticket length and not by the number of members considered. That is good news when large audiences are addressed. However, remember that the length of the refreshment message might force the audience to be split into several subgroups (see Section 3.2).

Member times show that retrieving the secret is a very fast process. The main problem at the server side is, again, handling a long message.

4 Key refreshment message authentication

At this point we have achieved privacy in multicast communications. This section presents a mechanism that authenticates the refreshment messages from the Key Server: that is required in order to protect the system against forged refreshment messages. The usual technology for message authentication is digital signature: a hash of the

 ISBN 13: 978-84-613-5510-5

message is encrypted with the sender's private key. The receiver can then decrypt the hash and compare it with its own result of a hash operation on the received information.

That solution is not applicable in our case, because refreshment messages disclose a public key: members would need to use the public key they are receiving to verify the message that contains it. A digital signature of that nature would not assure authentication at all, since the message might be forged and the signature still be valid. An alternative is to use the key pair used before the refreshment, but just-arrived members would not be able to verify the signature, since they would not know the previous key pair. Obviously, the simplest solution involves having a different, invariable key pair for authentication purposes.

We propose, instead, an approach which is not based in the use of public key cryptography. Our solution proves that the sender knows the recipient's ticket. The two only entities in the system that know any given ticket are its owner member and the Key Sever. Assuming the ticket has not been stolen, any message received by a member that passes the verification scheme can only come from the Key Server.

The scheme is described next. We assume the Key Server is performing a refreshment of its key pair, and therefore the authentication process is complementary to that described in Section 3. We assume, too, that members receive the refreshment message.

1. The Key server:
   (a) computes $s = (g^k)^{-1}\ mod\ L$ by means of the Extended Euclidean Algorithm,
   (b) chooses a random number a, such that $a < x_i$, for every x_i, and
   (c) multicasts $\{a \cdot s, h(a)\}$. $h(a)$ is the output of a hash operation on a. The hash algorithm is not specified here.
2. Every member i receives the authentication message and computes $h(a{\cdot}s{\cdot}K_{pub}\ mod\ x_i)$, which should be equal to the value $h(a)$ received if x_i is a factor of L.

It is convenient that the authentication message is attached to the refreshment message so authenticity can be verified upon reception.

4.1 Security and efficiency considerations

Regarding security, the key point is that $a \cdot s \cdot r\ mod\ \alpha$ is only equal to a if $\alpha = L$ or $\alpha = x_i\ \forall\ x_i$. An attacker willing to forge an authenticated key refreshment message must know either L or at least one x_i. In the first case the forged message will pass the verification test in every client, while in the second case only the owner of x_i will be fooled. However, both L and every x_i are kept secret, and stealing them is equivalent to stealing a private key. We can therefore state that in terms of security, and for the scenario described in Section 2, the authentication scheme proposed here is a valid substitute for digital signature.

Regarding efficiency, the arbitrary-precision arithmetic additional operations required in the Key Server side are a modular inverse and a multiplication. Every client must compute a modular multiplication. Those operations have very little impact on the final runtime since they can be run very efficiently by any hardware with arbitrary-precision arithmetic capabilities.

 ISBN 13: 978-84-613-5510-5

The scheme poses a disadvantage, however: the authentication message can be as long as the key refreshment message. This should be taken into account in low bit rate scenarios.

5 Peer validation: a zero-knowledge proof

Once secure multicast and Key Server messages validation have been achieved, the last proposal in this paper deals with authentication among peers. The aim is to verify that a given peer j holds a valid ticket x_j: this means that j is a legal peer, assuming no information leakage. Verification is carried out with no disclosure of any private nor sensible information. The scheme is presented next. Assume that peer i wants to verify whether peer j is a legal peer, prior to establishing communications with it. Assume too that the public key disclosure algorithm from Section 3 has been run previously. Recall the form of $(K_{pub},\ K_{priv})$:

$$K_{pub} : g, m, g^k\ mod\ m$$
$$K_{priv} : k$$

Authentication is performed as follows:

1. Peer i chooses a random integer r such that $1 < r < m$ and sends it to the Key Server.
2. The Key Server computes $inv = r^{-1}\ mod\ L$ and sends it to i.
3. Peer i sends $\{inv,\ g^{x_i}\ mod\ m\}$ to j.
4. Peer j calculates $r_j = inv^{-1}\ mod\ x_j$, $\beta_j = r_j \cdot (g^{x_i})^{x_j}$ and sends $\{\beta_j,\ g^{x_j}\}$ to i.
5. Peer i computes $\beta_i = r \cdot (g^{x_j})^{x_i}$, which should be equal to β_j.

If $\beta_i = \beta_j$ then it is clear that j owns a valid ticket x_j. Otherwise peer i should warn the Key Server so preventive measures can be taken against j. Modular inverses can be computed by means of the Extended Euclidean Algorithm.

In case this protocol is implemented in a standalone way and no public key disclosure algorithm is being run then the Key Server must choose the values values g and m as shown in Section 3 and communicate them to peers before any authentication is done.

5.1 Security and efficiency considerations

Security is assured by two facts:

1. peer j needs to know a valid ticket x_j in order to obtain a r_j equal to r, by means of a modular inverse calculation (step 4), and
2. the complexity of the discrete logarithm problem in a finite field [10].

We warn now against the possibility of performing Denial of Service (DoS) attacks against the Key Server and peer j, if a malicious entity sends verification requests at

 ISBN 13: 978-84-613-5510-5

an intentional very high rate. That same entity might arbitrarily warn the Key Server against legal peers, too.

Regarding efficiency and scalability, the protocol involves one communication with the Key Server and modular exponentiations. This makes the protocol applicable only to small centralized networks or distributed networks in which subgroups are managed by local entities, such as IOLUS [5]. However, the Key Server plays only a small role (modular inverses can be found very efficiently) and the main part of the work is carried out by peers. Having said this, member authentication by means of digital certificates is a more realistic approach for large groups. However, our scheme may be an alternative for small sets of members.

6 Conclusions

We have presented three different uses for the Extended Euclidean Algorithm, all of them focusing on privacy and security in multicast scenarios. The first one, a controlled disclosure mechanism, is suitable for scenarios in which a single entity (a Key Server) communicates its public key to a set of client hosts, so private communications can be held. The communication can be done in a single multicast message, and there is no need for encryption. The mechanism is secure, and simulation results were shown to prove its efficiency, both on the Key Server and on the client side.

The second application is an authentication mechanism which is not based on public-key cryptography. It can be used in situations in which public-key cryptography is not available (due to low capacity devices on the client side, for example). It can also be used along with the first scheme, though.

Finally, a zero-knowledge protocol was presented which can be used for for peer validation. By using this protocol peers can decide whether to trust a peer or not before establishing communications with it. It works by challenging peers to demonstrate that they own a valid ticket. No sensible information is disclosed.

The three mechanisms can be applied to the same scenario, say, a peer-to-peer television platform. Future lines of research include the implementation and test of a combination of them in a simulator (e.g. PeerSim [11]) or a real testbed, such as PlanetLab [12].

Acknowledgements

J. A. M. Naranjo and L. G. Casado are supported by the Spanish Ministry of Science and Innovation (TIN2008-01117). L. G. Casado is also supported by funds of Junta de Andalucía (P08-TIC-3518). J. A. López-Ramos is supported by the Spanish Ministry of Science and Innovation (TEC2009-13763-C02-02) and Junta de Andalucía (FQM 0211).

 ISBN 13: 978-84-613-5510-5

References

[1] Guang-huei Chiou and Wen-Tsuen Chen, *Secure broadcasting using the secure lock*, IEEE Trans. Softw. Eng. 15(8): 929–934, 1989.

[2] Peter S. Kruus and Joseph P. Macker, *Techniques and issues in multicast security*, In Proceedings of Military Communications Conference, MILCOM, 1998, pages 1028–1032.

[3] D. Wallner, E. Harder and R. Agee, *Key management for multicast: Issues and architectures*, RFC 2627, 1999.

[4] O. Scheikl, J. Lane, R. Boyer and M. Eltoweissy, *Multi-level secure multicast: the rethinking of secure locks*, In Parallel Processing Workshops, 2002. Proceedings. International Conference on, pages 17–24, 2002.

[5] Suvo Mittra, *Iolus: A framework for scalable secure multicasting*, In Proceedings of ACM SIGCOMM'97, pages 277–288, 1997.

[6] *Msec working group*, http://www.ietf.org/dyn/wg/charter/msec-charter.html.

[7] Elgamal, Taher, *A public key cryptosystem and a signature scheme based on discrete logarithms*, Proceedings of CRYPTO 84 on Advances in cryptology, pages 10–18, 1984.

[8] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, *Handbook of Applied Cryptograhpy*, CRC Press, 1996.

[9] J. A. M. Naranjo, J. A. López-Ramos and L. G. Casado *Key management schemes for peer-to-peer multimedia streaming overlay networks*, In WISTP '09: Proceedings of the 3rd IFIP WG 11.2 International Workshop on Information Security Theory and Practice. Smart devices, Pervasive Systems and Ubiquitous Networks, pages 128–142, 2009, Springer-Verlag.

[10] Whitfield Diffie, Martin E. Hellman, *New Directions in Cryptography*, IEEE Transactions on Information Theory, 22(6), pages 644-654, 1976.

[11] Márk Jelasity, Alberto Montresor, Gian Paolo Jesi, and Spyros Voulgaris, *The Peersim simulator*, `http://peersim.sf.net`.

[12] *The PlanetLab network*, `http://www.planet-lab.org`.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Conservative finite difference scheme for the Zakharov–Kuznetsov equation

Hirota Nishiyama[1], Takahiro Noi[1] and Shinnosuke Oharu[1]

[1] *Department of Mathematics, Chuo University, Tokyo*

emails: nisiyama@gug.math.chuo-u.ac.jp, s17004@gug.math.chuo-u.ac.jp, oharu@gug.math.chuo-u.ac.jp

Abstract

In this paper we discuss conservative finite difference schemes by means of discrete variational method and show the numerical solvability of a two dimensional nonlinear wave equation which is known as the Zakharov–Kuznetsov equation. We propose a finite difference scheme that inherit mass, energy conservation properties from the Zakharov–Kuznetsov equation. Our treatments refers to the procedure that Furihata has presented for real–valued nonlinear partial differential equations (PDEs). Numerical results are shown to illustrate the accuracy and validity of the numerical solutions obtained.

Key words: Zakharov-Kuznetsov equation, discrete variational method, conservative finite difference scheme

1 Introduction

We deal with the Zakharov–Kuznetsov(ZK) equation of the form

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} + \frac{\partial^3 u}{\partial x \partial y^2} = 0, \tag{1}$$

which is a two-dimensional generalization of the Korteweg–de Vries equation. This was first derived by V. E. Zakharov and E. A. Kuznetsov [1] in three–dimensional form to describe the motion of nonlinear ion-acoustic waves in a magnetized plasma [2, 3]. A variety of physical phenomena, in the purely dispersive limit, are governed by this type of equations; for exsample, the long waves on a thin liquid film [4], the Rossby waves in rotating atmosphere [5], and the isolated vortex of the drift waves in three-dimensional plasma [6]. In spite of such generality, detailed properties of solutions to eq (1) have not yet been fully explored. A cylindrically symmetric solitary wave solution (hereafter we call this solution the bell–shaped pulse) was obtained numerically and

 ISBN 13: 978-84-613-5510-5

its stability has been shown by means of a variational estimate [1]. In this paper, we perform numerical simulations by use of the finite difference scheme which is obtained through a discrete variatioal method and exhibit what roles the bell–shaped pulses play in the evolution processes governed by the ZK equation. In section 2, we are concerned with fundamental properties of the ZK equation. We formulate the finite difference scheme through a discrete variational method for the Zakharov–Kuznetsov equation in Section3. Some of the numerical results for the initial boundary value problems are presented in Section4.

2 Features of the ZK equation

As a preliminary to the numerical investigation, we employ typical solutions and conservation laws assosciated with eq (1). One of the exact solutions is a steady progressive wave solutions of the form

$$U = 3c\,\mathrm{sech}^2\left[\frac{1}{2}\sqrt{c}((\tilde{x}-x_0)\cos\theta + (y-y_0)\sin\theta)\right], \quad \tilde{x} = x - ct, \tag{2}$$

where c is the wave velocity. This solution represents an oblique one–dimensional solitary wave with an inclined angle θ with respect to the x–axis; which is a straightforward extension of the one dimensional K-dV soliton. For the cylindrically symmetric case, the solution U in (2) satisfies

$$r^{-1}\frac{d(rdU/dr)}{dr} = cU - \frac{1}{2}U^2 \tag{3}$$

where $r \equiv (x^2+y^2)^{1/2}$. Existence of bounded solutions to eq (3) and their stability have been discussed through a variational estimate[1]. However, a Painlev'e test asserts that eq. (3) is not integrable, and that no general analytical solutions exist. The ZK equation admits the following integrals of motion as shown in [2]:

$$M \equiv \int\int u(x,y,t)dxdy, \;\; P \equiv \int\int \frac{1}{2}u^2(x,y,t)dxdy,$$
$$H \equiv \int\int \left[\frac{1}{2}(\nabla\cdot u)^2 - \frac{1}{6}u^3(x,y,t)\right]dxdy.$$

Quantities M, P and H represent the mass, momentum and energy, which are all conservative quantities.

3 Design of scheme

In this section we formulate finite difference schemes for the target equation (1) with discrete variational derivatives.

 ISBN 13: 978-84-613-5510-5

3.1 Difference operators

Throughout this paper the following differece operators are employed. First, we write $U_{i,j}^n \sim u(n\Delta t, i\Delta x, j\Delta y)$, $(n \in \mathbf{Z}, i, j = 0, 1, 2, \cdots)$, where $\Delta t, \Delta x, \Delta y > 0$ denote the mesh sizes in t, x, y. We then use the following difference operators;

$$\delta_i^+ U_{i,j} = \frac{U_{i+1,j} - U_{i,j}}{\Delta x}, \quad \delta_j^+ U_{i,j} = \frac{U_{i,j+1} - U_{i,j}}{\Delta y}, \tag{4}$$

$$\delta_i^- U_{i,j} = \frac{U_{i,j} - U_{i-1,j}}{\Delta x}, \quad \delta_j^- U_{i,j} = \frac{U_{i,j} - U_{i,j-1}}{\Delta y}, \tag{5}$$

$$\delta_i^{(1)} U_{i,j} = \frac{U_{i+1,j} - U_{i-1,j}}{2\Delta x}, \quad \delta_j^{(1)} U_{i,j} = \frac{U_{i,j+1} - U_{i,j-1}}{2\Delta y}, \tag{6}$$

$$\delta_i^{(2)} U_{i,j} = \frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{(\Delta x)^2}, \quad \delta_j^{(2)} U_{i,j} = \frac{U_{i,j+1} - 2U_{i,j} + U_{i,j-1}}{(\Delta y)^2}. \tag{7}$$

3.2 Formulation of finite difference scheme

We define the discrete energy by

$$G_d(\mathbf{U})_{i,j} \equiv \frac{1}{6} U_{i,j}^3 - \frac{1}{2} \left\{ \frac{(\delta_i^+ U_{i,j})^2 + (\delta_i^- U_{i,j})^2}{2} + \frac{(\delta_j^+ U_{i,j})^2 + (\delta_j^- U_{i,j})^2}{2} \right\} \tag{8}$$

Then the discrete global energy is given by

$$H_d(\mathbf{U}) \equiv \sum_{i=0}^{N_x}{}'' \sum_{j=0}^{N_y}{}'' G_d(\mathbf{U})_{i,j} \Delta x \Delta y \tag{9}$$

where N_x and N_y are numbers of grid points in x and y, respectively. To define a discrete variational derivative, we consider the difference $H_d(\mathbf{U}) - H_d(\mathbf{V})$ defined by,

$$H_d(\mathbf{U}) - H_d(\mathbf{V}) = \sum_{i=0}^{N_x}{}'' \sum_{j=0}^{N_y} \left(\frac{\delta G_d}{\delta(\mathbf{U}, \mathbf{V})} \right)_{i,j} (U_{i,j} - V_{i,j}) \Delta x \Delta y, \tag{10}$$

$$\left(\frac{\delta G_d}{\delta(\mathbf{U}, \mathbf{V})} \right)_{i,j} = \frac{1}{6}(U_{i,j}^2 + U_{i,j}V_{i,j} + V_{i,j}^2) + \frac{1}{2} \left(\delta_i^{(2)}(U_{i,j} + V_{i,j}) + \delta_j^{(2)}(U_{i,j} + V_{i,j}) \right).$$

In the above identities we use the ***summation-by-parts formula*** in i and j directions separately. Then we have a finite difference scheme:

$$\frac{U_{i,j}^{n+1} - U_{i,j}^n}{\Delta t} = -\delta_i^{(1)} \left(\frac{\delta G_d}{\delta(\mathbf{U}, \mathbf{V})} \right)_{i,j}, \tag{11}$$

$$\begin{aligned} \left(\frac{\delta G_d}{\delta(U^{n+1}, U^n)} \right)_{i,j} &= \frac{1}{6}((U_{i,j}^{n+1})^2 + U_{i,j}^{n+1}U_{i,j}^n + (U_{i,j}^n)^2) \\ &+ \frac{1}{2} \left(\delta_i^{(2)}(U_{i,j}^{n+1} + U_{i,j}^n) + \delta_j^{(2)}(U_{i,j}^{n+1} + U_{i,j}^n) \right). \end{aligned}$$

 ISBN 13: 978-84-613-5510-5

4 Results of numerical simulations

4.1 Double collision of solitons Here we treat show the double collision of solitons to the ZK equation

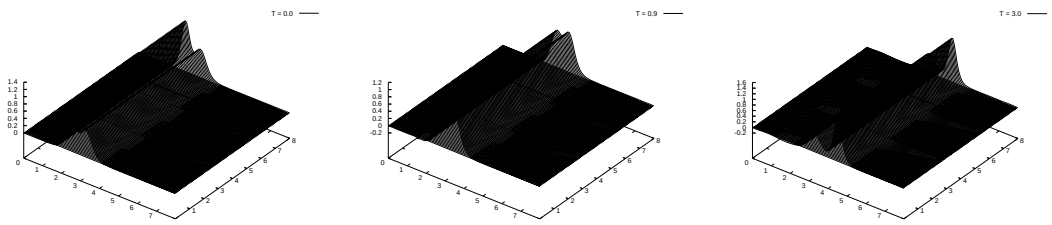
$$u_t + uu_x + \varepsilon(u_{xxx} + u_{yyx}) = 0. \tag{12}$$

In the case of double collision of solitons, the following initial condition

$$u(x, y, 0) = \sum_{j=1}^{2} 3c_j \mathrm{sech}^2 \left(0.5\sqrt{\frac{c}{\varepsilon}}((x - x_j)\cos\theta + (y - y_j)\sin\theta) \right) \tag{13}$$

is employed, where $c_1 = 0.45$, $c_2 = 0.25$, $\varepsilon = 0.01$, $\theta = 0.0$, $x_1 = 2.5$, $y_1 = 0.0$, $x_2 = 3.3$, $y_2 = 0$. The results under periodic boundary conditions in both $x-$ and $y-$directions in $[0, 8] \times [0, 8]$ with 100×100 uniform cells are depicted in the figures below :

Figure. Behavior of numerical solution in the time interval $t = 0.0 \sim 3.0$



References

[1] V.E., Zakharov and E.A. Kunznetsov, *Three-dimensional solitons*, Sov. Phys. JETP **39** (1974) 285-286.

[2] E.A. Kunznetsov, A.M. Rubenchik and V.E., Zakharov, *Soliton stability in plasmas and hydrodynamics*, Phys. Rep **142** (1986) 103.

[3] E.W. Laedke and K.H. Spatschek, *Nonlinear ion - acoustic waves in weak magnetic fields*, Phys. Fluid **25** (1982) 985.

[4] S. Melkonian and S.A. Maslowe, *Two-dimensional amplitude evolution equations for nonlinear dispersive waves on thin films*, Physica D **34** (1989) 255.

[5] V.I. Petviashvili, *Red Spot of Jupiter and drift soliton in a plasma*, JETP. Lett. **32** (1980) 619.

[6] K. Nozaki, *Vortex Solitons of Drift Waves and Anomalous Diffusion*, Phys. Rev. Lett **46** (1981) 184.

[7] Yan Xu and Chi–Wang Shu, *Local discontinuous Galerkin methods for two classes of two–dimensional nonlinear wave equations*, Physica D **208** (2005) 21.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Flow analysis around structures in slow fluids and its applications to environmental fluid phenomena

S. Oharu[1], Y. Matsuura[2] and T. Arima[3]

[1] *Department of Mathematics, Chuo University, Tokyo, Japan*

[2] *Department of Maritime Safety Technology, Japan Coast Guard Academy*

[3] *Wako Research Center, Honda R&D Co., Ltd. Saitama, Japan*

emails: oharu@math.chuo-u.ac.jp, matsuura@jcga.ac.jp,
t.arima@f.rd.honda.co.jp

Abstract

Importance and application of numerical flow analysis in environmental science and technology are outlined. Fluid phenomena in the ocean, rivers, atmosphere and the ground are investigated by means of numerical methods and in turn proposals for the control, restoration and counterplans against the so-called environmental disrupters which destroy natural environment as well as ecological systems in nature. All such environmental disrupters diffuse in and are transported by environmental fluids. Those disrupters sometimes react on some other chemicals to generate more poisonous materials. Environmental fluid dynamics is effective for the evaluation, prediction and restoration of the environmental damage. In this paper a mathematical model of environmental fluid is presented and results of numerical simulations based on the model are exhibited. *Key words: Environmental fluid, computational fluid dynamics, numerical simulation, environmental restoration technology, three dimensional visualization.*

MSC 2000: 39A12, 62P12, 65M06, 65M12, 76D05

1 Introduction

Fluid dynamical technologies are now important in the field of environmental science and technology. Evaluation of environmental fluid flow using numerical methods is particularly useful to understand the complex fluid motion and make it possible to control the flow fields from the point of view of environmental restoration. The environmental fluid problems may be classified by three types of applications. The first application is concerned with ultimate use of exergy. This is the most important subject for existing engine that use fossil fuel for combustion. Secondly, new energy sources such as wind

 ISBN 13: 978-84-613-5510-5

and wave power generation should be extensively studied. Thirdly, it is indispensable to develop not only efficient and harmless energy sources but also technologies to restore the environment which has already been polluted by exhaust gases through combustion of fossil fuel. It is also important to develop effective methods for protecting the environment fluids against pollutants. In this paper the field of studies in evaluation, control and prediction of transport phenomena which arise in a variety of environmental problems is called *environmental fluid dynamics*. Obviously, the environmental fluid dynamics is one of the key theories to invent efficient technologies for the preservation and restoration of the natural environment. Here we focus our attention on a dynamical analysis of diffusion and convective processes of pollutants in environmental fluids. A mathematical model of environmental fluids is presented and results of simulations are exhibited. It is then expected that new environmental restoration technology will be developed.

2 A mathematical model of environmental fluids

As a mathematical model describing the motion of environmental fluids, we employ the compressible Navier-Stokes system. Although it is known (see [9]) that the application of numerical methods for the compressible Navier-Stokes system to low-speed flows does not necessarily provide us with satisfactory results on the convergence. This fact implies that numerical simulations become inefficient and the associated computational results turn out to be inaccurate. These numerical difficulties are caused by the circumstances that the characteristic velocities in the compressible Navier-Stokes systems are convective and sound speeds, and so that their ratios become large and the so-called stiffness of the system may occur due to the disparity of eigenvalues of the system. One of efficient counter measures for those numerical difficulties is to employ the so-called Boussinesq approximation under the assumption that the ratio of the change in density to that of temperature is small. In fact, under the Boussinesq approximation numerical methods which have been developed for incompressible fluids can directly applied. On the other hand, in the case that the ratio of the change in density to that of temperature is considerably large, the low Mach number approximation is proposed in [5].

In this paper we apply the Boussinesq approximation to the Navier-Stokes system and formulate the following system of equations (1-3) as our mathematical model for describing the motion of environmental fluids:

$$\nabla \cdot \mathbf{v} = 0 \tag{1}$$

$$\rho\left[\mathbf{v}_t + (\mathbf{v}\cdot\nabla)\mathbf{v}\right] = -\nabla p + \mu\Delta\mathbf{v} - \rho\beta(T - T_0)\mathbf{g} \tag{2}$$

$$\rho Cp\left[T_t + (\mathbf{v}\cdot\nabla)T\right] = \nabla(\kappa\nabla T) + S_c \tag{3}$$

Here the parameters $\mathbf{v}$, ρ, p, μ, β, $\mathbf{g}$, T and T_0 represent the velocity vector, density, pressure, viscosity coefficient, rate of volume expansion, the acceleration of gravity, temperature and its reference temperature, respectively. Also, the coefficient κ means the thermal conductivity and S_c stands for the sum of heat sources in the fluid.

 ISBN 13: 978-84-613-5510-5

Our main objective here is to get numerical data describing the flow field around bodies in an environmental fluid. For this purpose we impose Dirichlet boundary conditions for $\mathbf{v}$ and T and homogeneous Neumann boundary conditions for p on the inflow boundary. On the outflow boundary we impose homogeneous Neumann conditions for $\mathbf{v}$, T and Dirichlet boundary conditions for p. On the surface of each body standing in the fluid, we impose the non-slip conditions for $\mathbf{v}$ and homogeneous Neumann conditions for T and employ an inhomogeneous Neumann boundary conditions for p which is obtained from equation (2). In this paper a new numerical scheme is proposed such that a fully implicit Euler scheme for the numerical velocity is introduced.

3 Numerical Model

Making discretization in time of (1) by use of the implicit Euler method, we obtain the following system of nonlinear equations:

$$\nabla \cdot \mathbf{v}^{n+1} = 0 \tag{4}$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = -(\mathbf{v}^{n+1} \cdot \nabla)\mathbf{v}^{n+1} - \frac{1}{\rho}\nabla p^{n+1} + \frac{\mu}{\rho}\Delta \mathbf{v}^{n+1} - \beta(T^{n+1} - T_0)\mathbf{g} \tag{5}$$

$$\frac{T^{n+1} - T^n}{\Delta t} = -(\mathbf{v}^{n+1} \cdot \nabla)T^{n+1} + \frac{1}{\rho C_p}\nabla(\kappa \nabla T^{n+1}) + Sc \tag{6}$$

Substituting Equation (5) into Equation (4), Poisson's equation for pressure is derived:

$$\mathbf{\Delta} p^{n+1} = -\rho \left[\nabla \cdot \left\{ (\mathbf{v}^{n+1} \cdot \nabla)\mathbf{v}^{n+1} \right\} - \frac{\nabla \cdot \mathbf{v}^n}{\Delta t} \right] - \rho\beta \nabla \cdot (T^{n+1}\mathbf{g}). \tag{7}$$

In what follows, we regard Equations (4-6) as the governing equations for the motion of numerical fluids. We here investigate the structure and behavior of numerical solutions of this basic model.

3.1 Implicit iterative scheme

We here present a new method such that given velocity field $\mathbf{v}^n$, pressure field p^n and temperature field T^n at the n-th time step, the velocity field $\mathbf{v}^{n+1}$, pressure field p^{n+1} and temperature field T^{n+1} are obtained at the $(n+1)$-th step through the iteration procedures, respectively. Our mathematical models of the numerical fluid as expressed by Equations (5), (6) and (7) are fully implicit in time and this implicit form guarantees numerical stability and robustness. We adopt a procedure of constructing iterative numerical solutions that is not only much more economical but also remains most of the stability and accuracy properties of the fully implicit scheme.

The iteration procedure employed in the present study is summarized as follows: In the following, the superscript n refers to the value which are known from the previous time step, the superscript k refers to the iteration cycle between the solutions at time step n and $n+1$, the superscript 0 is associated with an initial guess for the first iteration step $k = 0$.

 ISBN 13: 978-84-613-5510-5

Step1: Choose an inferred initial data for computing the values $\mathbf{v}^{n+1}$, p^{n+1}, and T^{n+1} at the next time step. The simplest choice is to use the solutions themselves at the current time step:

$$\mathbf{v}^0 = \mathbf{v}^n, \quad p^0 = p^n, \quad T^0 = T^n$$

Step2: Poisson's equation for the pressure (7) is solved by applying the successive over relaxation (SOR) method, in which the residual cutting method (RCM) developed in [7] is used to speed up the convergence, to get the pressure at the current iteration step, say k:

$$\mathbf{\Delta} p^k = -\rho \left[\nabla \cdot \left\{ (\mathbf{v}^k \cdot \nabla)\mathbf{v}^k \right\} - \frac{\nabla \cdot \mathbf{v}^n}{\Delta t} \right] - \rho\beta \nabla \cdot (T^k \mathbf{g}) \tag{8}$$

Step3: The following equation of the delta-form for $\delta\mathbf{v}^k$ $(= \mathbf{v}^{k+1} - \mathbf{v}^k)$ is solved.

$$\left[1 + \Delta t \left(\mathbf{v}^n \cdot \nabla - \frac{\mu}{\rho}\Delta \right) \right] \delta\mathbf{v}^k = rhs_m^k, \tag{9}$$

$$rhs_m^k = -(\mathbf{v}^k - \mathbf{v}^n) + \Delta t \left[-(\mathbf{v}^k \cdot \nabla)\mathbf{v}^k - \frac{1}{\rho}\nabla p^k + \frac{\mu}{\rho}\Delta \mathbf{v}^k - \beta(T^k - T_0)\mathbf{g} \right]. \tag{10}$$

Step4: Compute the velocity at the next iteration step $k+1$ by

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \delta\mathbf{v}^k.$$

Step5: The following equation of the delta-form for δT^k $(= T^{k+1} - T^k)$ is solved.

$$\left[1 + \Delta t \left(\mathbf{v}^{k+1} \cdot \nabla - \frac{\kappa}{C_p \rho}\Delta \right) \right] \delta T^k = rhs_T^k, \tag{11}$$

$$rhs_T^k = -(T^k - T^n) + \Delta t \left[-(\mathbf{v}^{k+1} \cdot \nabla)T^k + \frac{\kappa}{C_p\rho}\Delta T^k + \frac{Sc}{\rho C_p} \right]. \tag{12}$$

Step6: Compute the temperature at the next iteration step $k+1$ by

$$T^{k+1} = T^k + \delta T^k.$$

Step7: Check the convergence of Newton's iteration for the equations of the delta form for the velocity and the temperature as follows:

$$\sum_\Omega \left| \mathbf{v}^{k+1} - \mathbf{v}^k \right| < \epsilon_v, \quad \sum_\Omega \left| T^{k+1} - T^k \right| < \epsilon_T.$$

where $\sum_\Omega$ means the summation over the whole computational domain, ϵ_v and ϵ_T are small values prescribed as admissible error bounds which also stand for radii of convergence for the respective inequalities. This completes one cycle of the iterative process. If more iterations are required, the process should be continued from Step 2. In particular, experiences suggest that only 2 or 3 iterations are enough to get desired approximate numerical solutions. We find in this scheme that if $|\mathbf{v}^{k+1} - \mathbf{v}^k| \to 0$ and $|T^{k+1} - T^k| \to 0$ then $\mathbf{v}^k = \mathbf{v}^{k+1} = \mathbf{v}^{n+1}$, $T^k = T^{k+1} = T^{n+1}$ and $p^k = p^{n+1}$, because Equations (9) and (11) converge to Equations (5) and (6), respectively, for $\delta\mathbf{v}^k = 0$ and $\delta T^k = 0$; and then pressure equation (8) converges to Equation (7).

 ISBN 13: 978-84-613-5510-5

3.2 Spatial discretization with TVD property

Concerning the discretization in space, we employ a central difference scheme of the second order accuracy for discretization of terms except convective terms. For the discretization of the convective terms on the right-hand side of the Equations (9) and (11), we apply an upwind scheme of the third order accuracy which is proposed in [3]. A more favorable way for discretizing the advection terms would be the use of TVD schemes. To obtain TVD schemes of higher accuracy, the method of Monotone Upstream Centered Schemes for Conservation Laws (MUSCL) [4] can be used. For simplicity, we consider the following time-dependent Cauchy problem in one space dimension:

$$\frac{\partial \phi}{\partial t} + v\frac{\partial \phi}{\partial x} = 0, \quad -\infty < x < \infty,\ t \geq 0, \quad \phi(x,0) = \phi_0(x). \tag{13}$$

Here $\phi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ means velocity. We find that the solution of this equation has a TVD property because the solution of ϕ of Equation (13) is constant along curve $dx/dt = v$, which is known as the characteristics equation. Therefore, it is possible to construct a TVD scheme by starting with Equation (13). Thus, we consider the following equation similar to Equation (13).

$$\frac{\partial \phi}{\partial t} + \frac{\partial (v\phi)}{\partial x} - \phi\frac{\partial v}{\partial x} = 0. \tag{14}$$

Since the form of the second term in the above equation is of the form of derivative of flux, we incorporate a discretization with TVD property throught the MUSCL approach [4]. We discretize the $x - t$ plane by choosing a mesh width Δx and a time step Δt, and define the discrete mesh points (x_i, t_n) by

$$x_i = i\Delta x, \quad i = ..., -1, 0, 1, 2, ..., \quad t_n = n\Delta t, \quad n = 0, 1, 2, ...$$

For simplicity we take a uniform mesh, with Δx and Δt being constant. The finite difference methods provide approximations $u_i^n \in \mathbb{R}$ to solution $u(x_i, t_n)$ at the discrete grid points. Here we discretize Equation (14) as follows:

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = -\frac{1}{\Delta x}\left(\tilde{f}_{i+\frac{1}{2}} - \tilde{f}_{i-\frac{1}{2}}\right) \tag{15}$$

$\tilde{f}_{1\pm\frac{1}{2}}$ denotes a numerical flux function on the cell interface $x_i + \Delta x/2$. This can be evaluated as the sum of discretizations of the last term of Equation (14) and the discretized flux of the second term by the MUSCL method with *minmod* limiter function [1]. Since the last term can be discretized as,

$$\phi\frac{\partial v}{\partial x} \Rightarrow \left[(a_{i+\frac{1}{2}} - a_{i-\frac{1}{2}})/\Delta x\right]\phi,$$

with $a = v^n$, the total numerical flux can be evaluated as follows:

$$\begin{aligned}\tilde{f}_{i+\frac{1}{2}} = -a_{i+\frac{1}{2}}\phi_i + f_{i+\frac{1}{2}}^{(upw)} &+ a_{i+\frac{1}{2}}^{+} \cdot \frac{1}{4}\left[(1+\kappa)\Phi_{i+\frac{1}{2}}^{+C} + (1-\kappa)\Phi_{i+\frac{1}{2}}^{+U}\right] \\ &- a_{i+\frac{1}{2}}^{-} \cdot \frac{1}{4}\left[(1+\kappa)\Phi_{i+\frac{1}{2}}^{-C} + (1-\kappa)\Phi_{i+\frac{1}{2}}^{-U}\right]\end{aligned} \tag{16}$$

 ISBN 13: 978-84-613-5510-5

The first term of Equation (16) corresponds to the last term of Equation (14). The second term of Equation (17), $f_{i+\frac{1}{2}}^{(upw)}$ corresponds to the first-order accurate upwind difference of the second term of Equation (14) and the other terms are corrections to make the scheme of higher order accuracy. These can be written as follows:

$$f_{i+\frac{1}{2}}^{(upw)} = a_{i+\frac{1}{2}}^{+}\phi_i + a_{i+\frac{1}{2}}^{-}\phi_{i+1}.$$

Here

$$a = v^n, \quad a^{\pm} = v^{\pm} = \frac{1}{2}\left(v^n \pm |v^n|\right),$$

and Φ is defined as follows:

$$\begin{aligned}
\Phi_{i+\frac{1}{2}}^{+C} &= minmod\left[\phi_{i+1} - \phi_i, \beta(\phi_i - \phi_{i-1})\right] \\
\Phi_{i+\frac{1}{2}}^{+U} &= minmod\left[\phi_i - \phi_{i-1}, \beta(\phi_{i+1} - \phi_i)\right] \\
\Phi_{i+\frac{1}{2}}^{-C} &= minmod\left[\phi_{i+1} - \phi_i, \beta(\phi_{i+2} - \phi_{i+1})\right] \\
\Phi_{i+\frac{1}{2}}^{-U} &= minmod\left[\phi_{i+2} - \phi_{i+1}, \beta(\phi_{i+1} - \phi_i)\right],
\end{aligned}$$

$$minmod(x, y) = \frac{1}{2}\left[sgn(x) + sgn(y)\right] min(|x|, |y|).$$

The parameter β is called a compression parameter in [1] and must satisfy $\beta \geq 1$, and its upper bound is determined by the TVD condition. The parameter κ is one for discretization accuracy, e.g., the second-order accuracy for $\kappa = -1$ and $\kappa = 1/3$ for the third-order accuracy, we have $-1 \leq \kappa \leq 1$. The numerical flux $\tilde{f}_{i-\frac{1}{2}}$ is obtained by replacing subscript $i + \frac{1}{2}$ by $i - \frac{1}{2}$. In this replacement of subscripts, we should note that the first term with the replaced subscript is not $-a_{i-\frac{1}{2}}\phi_{i-1}$ but $-a_{i-\frac{1}{2}}\phi_i$. Using $a = a^+ + a^-$, Equation (17) can be rewritten as follows:

$$\begin{aligned}
\tilde{f}_{i+\frac{1}{2}} = a_{i+\frac{1}{2}}^{-}(\phi_{i+1} - \phi_i) + a_{i+\frac{1}{2}}^{+} \cdot \frac{1}{4}\left[(1+\kappa)\Phi_{i+\frac{1}{2}}^{+C} + (1-\kappa)\Phi_{i+\frac{1}{2}}^{+U}\right] \\
- a_{i+\frac{1}{2}}^{-} \cdot \frac{1}{4}\left[(1+\kappa)\Phi_{i+\frac{1}{2}}^{-C} + (1-\kappa)\Phi_{i+\frac{1}{2}}^{-U}\right]
\end{aligned} \tag{17}$$

When this scheme is written as

$$u_i^{n+1} = u_i^n - C_{i-\frac{1}{2}}(u_i - u_{i-1}) + D_{i+\frac{1}{2}}(u_{i+1} - u_i),$$

the conditions for this scheme to be Total Variation Diminishing (TVD) are:

$$C_{i+\frac{1}{2}} \geq 0, \quad D_{i+\frac{1}{2}} \geq 0, \quad C_{i+\frac{1}{2}} + D_{i+\frac{1}{2}} \leq 1.$$

From the conditions $C_{i+\frac{1}{2}} \geq 0$ and $D_{i+\frac{1}{2}} \geq 0$, we obtain

$$(1 \leq)\, \beta \leq \frac{3-\kappa}{1-\kappa}.$$

 ISBN 13: 978-84-613-5510-5

From the conditions $C_{i+\frac{1}{2}} + D_{i+\frac{1}{2}} \leq 1$, we obtain

$$\Delta t \leq \frac{\Delta x}{|a_{i+\frac{1}{2}}| + \frac{1}{4}(a^+_{i+\frac{3}{2}} - a^+_{i-\frac{1}{2}})(\beta(1+\kappa)+1-\kappa)}$$

Under these conditions, the scheme becomes a TVD scheme [2] for the discretization of Equation(15). When the advection speed is constant ($a = const$), it becomes

$$\Delta t \leq \frac{4}{5-\kappa+\beta(1+\kappa)} \cdot \frac{\Delta x}{|a|}.$$

This scheme is of the third-order accuracy for the values $\kappa = 1/3$ and $\beta = 4$.

We can directly incorporate this formula with the convective terms on the right-hand side of Equations (10) and (12). If this formula is applied to the convective term on the right-hand side of Equation (10), the same formula may have to be incprporate with the first term of Equation (8) in terms of the consistency to the Equations (10). Thus, it is possible to employ the TVD discretization in our iterative implicit scheme.

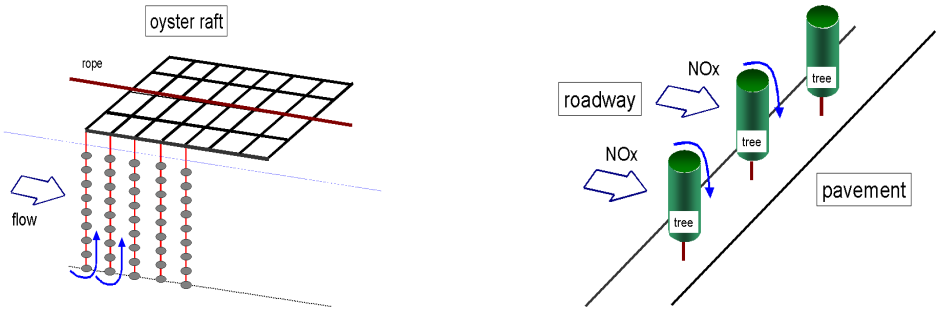
4 Settings of Numerical Simulation

We here discuss a flow analysis around cylinders with bottom ends standing in an environmental fluid. In the numerical simulations we have performed, flow analysis was made for a typical fluid flow. Our setting may be outlined as follows: We consider a parallelepiped region $\mathbf{R}$ in $\mathbb{R}^3$ and assume that one side is the inflow boundary and the opposite side is the outflow boundary. We then insert two circular cylinders with radius 1R and length 20R both of which have bottom ends in the region $\mathbf{R}$ in such a way that they are arranged in a row at an interval of 10R and perpendicular to the top side of $\mathbf{R}$, as illustrated in Figure 2. For convenience, we call the cylinder facing the inflow boundary the front cylinder and the cylinder facing the outflow boundary the rear cylinder. In this setting we performed numerical simulations and made detailed analysis around the two cylinders parallel to each other. One of the typical applications of this analysis is the simulation of the behavior of red tide plankton in a neighborhood of the farming part of an oyster raft floating on the ocean, as illustrated in Figure 1(a). Another application is the study in the effect and influence of convective diffusion phenomena of automobile exhaust fumes on roadside trees, as shown in Figure 1(b). Numerical conditions are put in the following way: The Reynolds number ($= \rho|\mathbf{v}|2R/\mu$) in accordance with the main flow velocity is assumed to be $Re = 2500$ and the temperature distribution is assumed to follow a linear distribution such that $T = 300K$ on the top of the front cylinder and $T = 290K$ under the bottom end.

5 Generation of Grid Point Systems

In order to obtain numerical solutions, we first define grid point system fitting with the surfaces of the bodies and compute solutions of discretized equations consistent with

 ISBN 13: 978-84-613-5510-5



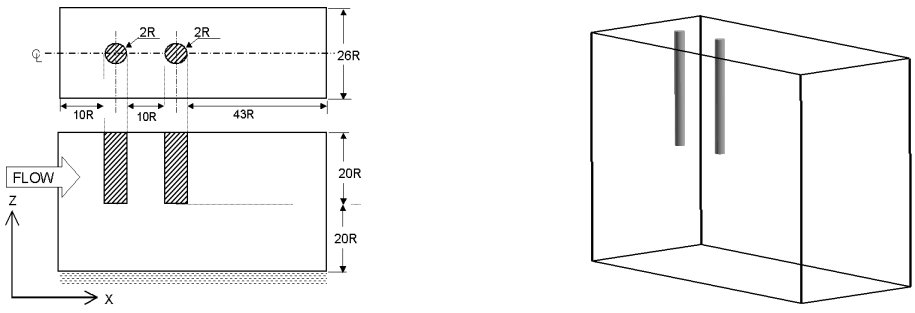
(a) Oyster raft (b) Street trees

Figure 1: Structures in the ocean and atmosphere

(1) - (3) on the grid points. Here an effective method for generating grid point systems which fit the surfaces of bodies in the fluid is proposed, and it is demonstrated that a simulation code can be developed on the grid point system. We generate grid point systems not only on the region of the fluid but also in the inside of the bodies so that those grid point systems are connected continuously on the boundaries of the bodies. Therefore the final grid point system covers a simply connected domain on which simulations are performed. A concrete procedure for generating such grid point system may be outlined as follows: Firstly, on each horizontal plane crossing the cylinders, we divide the computational domain into the inside and outside of the circles which are cross sections of the cylinders and then subdivide the outside of the circles into a finite number of simply connected subdomains. Secondly, we generate initial grid point systems on each subdomain by algebraic methods and then construct a spatial smooth grid point system by solving the associated elliptic system in a numerical sense.

Let $\xi = \xi(x, y)$ and $\eta = \eta(x, y)$ denote the mapping from the physical space to the computational space. The basis of this method is that, following Thompson et al.[8], the mapping functions are required to satisfy the system of Poisson's equations.

$$\xi_{xx} + \xi_{yy} = P, \quad \eta_{xx} + \eta_{yy} = Q \tag{18}$$



(a) Settings of computational domain (b) 3D view of computational domain

Figure 2: Settings for numerical simulation

 ISBN 13: 978-84-613-5510-5

Actual computation is to be done in the rectangular transformed field, where the curvilinear coordinates, ξ, η, are the independent variables, with the Cartesian coordinates, x, y, as dependent variables. The following relations are useful in transforming equations between computational space and physical space:

$$\xi_x = y_\eta/J, \quad \xi_y = -x_\eta/J, \quad \eta_x = -y_\xi/J, \quad \eta_y = x_\xi/J \tag{19}$$

where

$$J = x_\xi y_\eta - x_\eta y_\xi.$$

Applying Equation (19) to Equation (18) implies the transformed Poisson's equations

$$\begin{aligned} \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} &= -J^2 \left(P x_\xi + Q x_\eta\right) \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} &= -J^2 \left(P y_\xi + Q y_\eta\right), \end{aligned}$$

where

$$\alpha = x_\eta^2 + y_\eta^2, \quad \beta = x_\xi x_\eta + y_\xi y_\eta, \quad \gamma = x_\xi^2 + x_\xi^2.$$

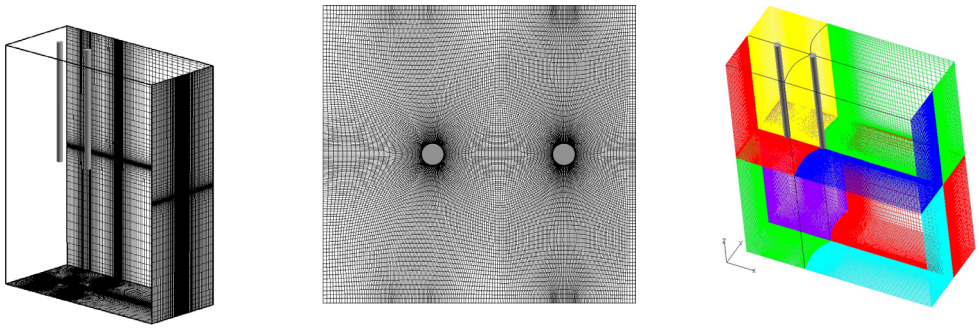
Using appropriate inhomogeneous terms P, Q in an iterative way based on an idea of Steger and Sorenson [6], we may generate a grid point system in such a way that the spacial distribution of grid points is concentrated near the boundaries of the bodies and lines connecting adjacent grid points are orthogonal to the boundaries.

Finally, we stack up the grid points constructed on each 2D cross section in the direction of the axes of the cylinders and generate the aimed 3D grid point system in the computational domain. The constructed grid point system used for our numerical simulations is depicted in Figure 3. Figure 3(a) shows a general view of the whole grid point system by plotting grid points on the boundaries of the computational domain. Figure 3(b) gives a closeup of the grid points generated around the cylinders. The number of grid points in the direction of the flow (the direction of x-axis) is 239, that in the direction of the axes of the cylinders (the direction of z-axis) is 119 and that in the direction perpendicular to the x and z-axes (the direction of y-axis) is 91, and so the total number of the grid points is 2,588,131. In order to shorten the time for computation, we use a parallel computer of the distributed memory type. In the parallel computing, the whole computational domain is divided into subdomains as shown in Figure 3(c) and computation on each subdomain is allocated to each CPU. In the case of parallel computing of the distributed memory type, the allocated boundary data are saved in separate memories. Hence we perform the parallel computing through the data communication using an Message Passing Interface Library.

6 Results of Numerical Simulations

Computation is started with a uniform initial data and qualitative features are investigated by analyzing the numerical results of the simulation at a time step at which the flow field is well developed and reaches a quasi-stationary state. Figure 4 depicts the velocity vector field and contours of the pressure on the cross section containing the axes of the two cylinders. In the velocity vector field upward flows along the back of

 ISBN 13: 978-84-613-5510-5



(a) 3D general view

(b) Grid points around the circular cylinders

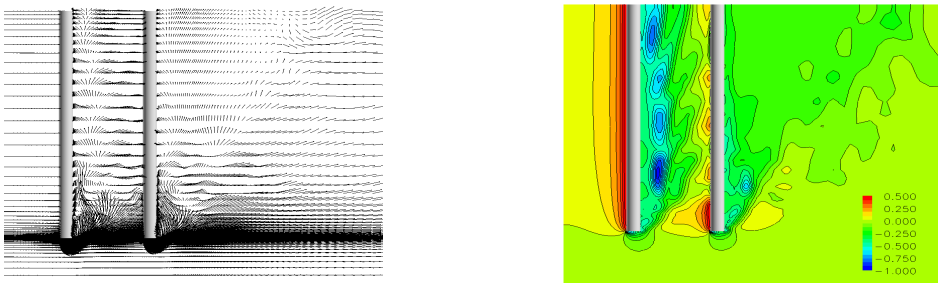
(c) Grid decomposition for parallel computation

Figure 3: Grid point system

the front cylinder are observed. These upward flows are formed when the horizontal uniform flow runs around the bottom end and are remarkable in a neighborhood of the bottom end and even reach the top part of the cylinder. Similar upward flows are also observed behind the rear cylinder. These flows are formed in such a way that they seem to roll the bottom part up and go up towards the top part. Moreover, such upward flows are observed in a wide range behind the rear cylinder since there are no obstacles. In the figure of contours of the pressure it is observed that a vertical sequence of separate regions like cells of negative pressure are formed. This is due to the presence of nonstationary vorticities of Karmann-type.

On the other hand, a vertical sequence of regions of positive pressure are observed in the front of the rear cylinder. This phenomenon suggests that the nonstationary vorticities generated by the front cylinder interact the regions of stagnation existing in the front of the rear cylinder and deteriorate the stagnation pressure. Figure 5 depicts the iso-surfaces of pressure and vorticity and illustrates the 3D structure of the pressure field and vorticity distribution. It is observed in the pressure field that there is regular variation from the top part of the cylinders until the downstream due to the formation of vorticity pairs of Karmann type, and that in the bottom parts of the cylinders the variation of pressure is restricted to the rear sides. On the other hand, concerning the vorticity distribution, regular variation due to the formation of vorticity pairs of Karmann type is observed in the top part of the rear cylinder in the same way as in the pressure field. Vorticity distribution is concentrated in the back of the middle part of the rear cylinder, although the vorticity distribution in the bottom part is comparatively diffusive. This suggests that the flows running around the bottom parts of the cylinders form longitudinal vortices. It is then inferred that these longitudinal vortices would motivate the upward flows behind the two cylinders. In Figure 6 the stream lines and trajectories of particles in the fluid are depicted. Trajectories of particles are drawn in the following way: We release the particles from the back of each cylinder and trace the trajectories forward and backward in time until the particles reach the boundaries of the computational domain and those of the

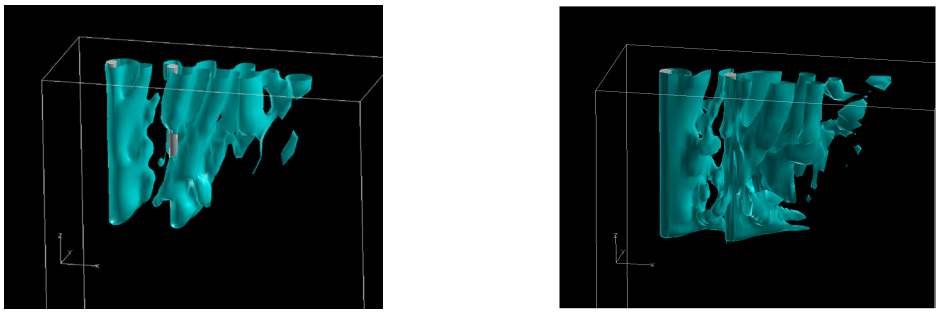
 ISBN 13: 978-84-613-5510-5



(a) Velocity vector (b) Pressure contour

Figure 4: Computed results on x-z plane across circular cylinders

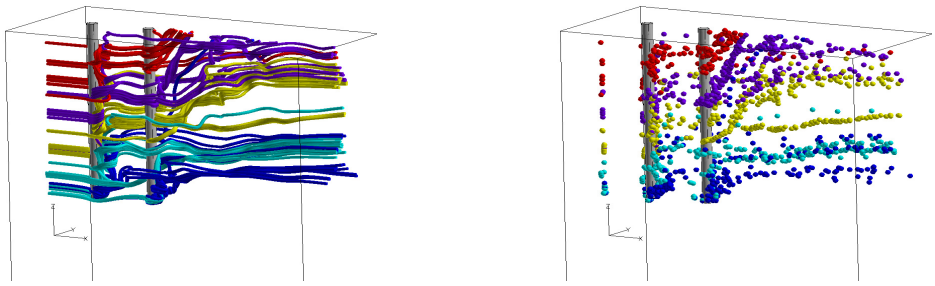
bodies in the fluid. It is seen from Figure 6(a) that upward flows behind the cylinders are rolling up towards the top. Furthermore, the motion of longitudinal vortices around the bottom sides can be observed as inferred from the iso-surfaces of vorticities. Figure 6(b) is obtained by arranging particles on the same trajectories as in Figure 6(a) at regular time intervals. From this it is seen that the particle distribution represents how long a particle stay in the flow, and that particles are concentrated in the back of the front cylinder. These results of numerical simulations may have applications to various environmental problems. The farming part of an oyster raft as illustrated in Figure 1(a) may be regarded as a regular arrangement of cylinders with bottom ends. Our results suggests that red tide plankton would stay in the upward vortices rolling up along the back of the cylinders provided that the oyster raft is moved or water currents flow against the raft. Another application is that street trees as illustrated in Figure 1(b) are regarded as a sequence of cylinders with top ends, and that automobile exhaust fumes may be caught in the back of each tree which purify those poisonous gases. It is then expected that new environmental restoration technology will be developed by applying the results of numerical simulations for environmental fluids.



(a) Pressure (b) Absolute vorticity

Figure 5: Computed iso-surfaces

 ISBN 13: 978-84-613-5510-5



(a) streamline (b) Particle trajectory

Figure 6: Upward flow motion observed behind two circular cylinders

References

[1] S. R. Chakravarthy and S. Osher, *A new class of high accuracy TVD scheme for hyperbolic conservation laws*, AIAA Paper 85-0363 (1985).

[2] A. Harten, *On a class of high resolution Total-Variation-Stable finite-difference schemes*, SIAM Journal of Numerical Analysis **21** (1), (1984),1–12.

[3] T. Kawamura and K. Kuwahara, *Computation of High Reynolds Number Flow around a Circular Cylinder with Surface Roughnes*, AIAA Paper 84-0340 (1984).

[4] B. van Leer, *Toward the ultimate conservative difference scheme. 4, A new approach to numerical convection*, J. of Comput. Phys. **23** (1977) 276–299.

[5] H. N. Najm, P. S. Wyckoff and O. M. Knio, *A Semi-implicit Numerical Scheme for Reacting Flow*, J. Comput. Phys. **143** (1998) 381-402.

[6] E. Witten, *Automatic mesh-point clustering near a boundary in grid generation with elliptic partial differential equations*, J. Comput. Phys. **33** (1979) 405–410.

[7] A. Tamura, K. Kikuchi and T. Takahashi, *Residual cutting method for elliptic boundary value problems: application to Poisson's equation*, J. Comput. Phys. **137** (1997) 247–264.

[8] J. Thompson, Z. U. A. Warsi and C. W. Mastin, *Numerical Grid Generation : Foundations and Applications*, McGraw-Hill/Appleton & Lange, 1985.

[9] E. Turkle, *Preconditioned Method for Solving the Incompressible and Low Speed Compressible Equations*, J. Comput. Phys. **72** (2) (1987) 277-298.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Hierarchical Radiosity on Hybrid Platforms

Emilio J. Padrón[1], Margarita Amor[1], Montserrat Bóo[2], Gabriel Rodríguez[1] and Ramón Doallo[1]

[1] *Computer Architecture Group, Universidade da Coruña*

[2] *Computer Architecture Group, Universidade de Santiago de Compostela*

emails: emilioj@udc.es, margamor@udc.es, montserrat.boo@usc.es, grodriguez@udc.es, ramon.doallo@udc.es

Abstract

Achieving an efficient realistic illumination is an important aim of research in computer graphics. In this paper a new parallel global illumination method for hybrid systems based on the hierarchical radiosity method is presented. Our solution allows the exploitation of systems that combine independent nodes with multiple cores per node. Thus, multiple nodes work in parallel in the computation of the global illumination for the same scene. Within each node, all the available computational cores are used through a shared-memory multithreading approach. The good results obtained in terms of speedup on several distributed-memory and shared-memory different configurations show the versatility of our hybrid proposal.

Key words: Hybrid Platforms, Global Illumination, Hierarchical Radiosity

1 Introduction

Radiosity [1] is one of the best solutions to get a physically-based illumination, essential key for realistic rendering. One of the key features of the radiosity approach is that it obtains view-independent global illumination results. Unfortunately, the radiosity method, like other global illumination alternatives, has high computational and memory requirements which justifies the use of parallel computing techniques to implement it.

In the last years, the progressive popularization of multicomputers and multicore-based systems makes the design and implementation of efficient parallel algorithms an appealing alternative for high demanding computer graphics techniques. The main challenge nowadays is to take advantage of all the different computational resources in a system, putting together shared-memory and distributed-memory concepts by means of versatile and efficient hybrid approaches [2].

There are multiple parallel approaches in the literature that have been proposed to speed up the radiosity calculation. However, most of the existing proposals for parallel

 ISBN 13: 978-84-613-5510-5

global illumination fall in one of these two categories: either purely shared-memory oriented or purely distributed-memory oriented. Shared-memory approaches [3, 4, 5] are simpler and achieve really good speedups, but they present the inherent scalability problems of this kind of systems. Furthermore, they are mostly fine-grain approaches, so a considerable overhead is introduced due to synchronization issues. On the other hand, distributed-memory approaches to parallel global illumination are notably more complex [6, 7, 8] and have typically obtained worse performance, mainly due to the important communication overhead, above all if the geometric data is distributed among the memories of the system nodes.

In [9] we find a hybrid distributed-shared proposal, based on what authors call *task pool teams*, basically an extension of the task pool approach commonly used in SMP (Symmetric Multiprocessing) computing. It is a generic non-specific alternative for irregular algorithms, using global illumination and HR as an example of application. Unfortunately, only results for quite small and unrealistic scenes are shown, so the method can not be considered as a valid solution for real global illumination.

Last trends in global illumination are mostly focussing on the exploitation of GPUs (Graphics Processing Unit) [10, 11, 12], taking the most data-parallel parts out of the CPUs to run on the GPU. However, important parts of global illumination methods are not suitable for this GPGPU (General-purpose computing on graphics processing units) kind of processing, so a parallel processing to take advantage of multi-processor systems must not be neglected.

In this paper we propose a novel parallel global illumination method, based on the hierarchical radiosity algorithm [13] (HR), the radiosity algorithm with a better quality/performance trade-off. The irregular and mostly unpredictable workload that is needed for the computation of the different parts of a scene with the hierarchical approach, based on an adaptive refinement, makes traditionally difficult to achieve a good parallel solution, especially in distributed-memory contexts. Our parallel design is focused on obtaining a versatile hybrid approach, exploiting systems with both distributed-memory and shared-memory resources by combining a message-passing paradigm with a multithreading approach.

The message-passing scheme we have implemented allows the parallel execution of HR in independent nodes of a distributed-memory cluster, minimizing the communication among nodes. As far as each node is concerned, a simple scheduling for the efficient processing of the input scene in a multithreaded environment has been implemented. This scheduling follows a coarse-grain approach, balancing the computational load within a node at a patch level and introducing a minimal overhead. A novel mutual exclusion protocol allows the concurrent subdivision of patches during each HR iteration effectively. This solution introduces minimal waiting times and needs few additional storage requirements.

The paper is organized as follows: Section 2 presents the HR method and Section 3 outlines the generic structure of our parallel proposal, with Subsections 3.1 and 3.2 describing the distributed-memory and the multithreading solutions, respectively. Experimental results and concluding remarks are presented in Section 4 and Section 5.

 ISBN 13: 978-84-613-5510-5

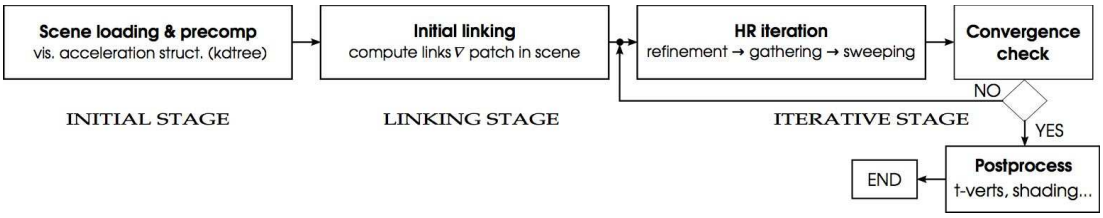


Figure 1: Sequential HR algorithm

2 Hierarchical Radiosity Algorithm

Radiosity [1] tackles the global illumination problem by applying a finite element approach to compute the transport of energy in an environment. Thus, the scene to be illuminated is discretized into a set of surface elements, usually called *patches* in the context of radiosity, and the light energy leaving each surface is computed, obtaining the classical discrete radiosity equation:

$$B_i = E_i + \rho_i \sum_{j=1}^{n} B_j F_{ij},\ 1 \leq i \leq n; \tag{1}$$

where B_i is the radiosity value (light energy per unit time per unit area leaving the surface) of a patch P_i, computed as the emittance of that patch (E_i, light energy produced by the surface itself, i.e. in case of light sources) plus the energy coming from the rest of the scene and reflected by it. Thus, the term ρ_i is the diffuse reflectance index of P_i, and the summation represents the energy reaching the patch from the other patches of the scene. The interaction or link between two patches in the scene is based on a geometric term called *form factor*, F_{ij}, that represents the proportion of radiosity leaving P_i that is received by the patch P_j.

The hierarchical approach to radiosity [13] is based on the application of a basic idea taken from the classic N-body problem: the importance of small details decreases with increasing distance. Thus, the input patches are subdivided into a hierarchy of surface elements with links with different level of refinement between them that simulate the light transport in the scene.

In general terms, the sequential HR method consists of three main stages (see Figure 1): *Initial Stage*, *Linking Stage* and *Iterative Stage*, this last one being the core of the HR process. The *Initial Stage* includes preprocessing work such as loading the input scene and building auxiliary data structures to accelerate the visibility determination between patches during the radiosity computation.

In the *Linking Stage* the starting interactions between pairs of patches in the scene are computed, building a list of initial links for each patch in the scene. Basically, two patches are interacting when they are (at least partially) visible to each other, and the corresponding form factor (a geometric term that represents the proportion of radiosity transported between two surface elements) is computed and stored for each link. Therefore, visibility determination and form factor computation are the main tasks performed in this stage.

 ISBN 13: 978-84-613-5510-5

The main phase in HR computation is the *Iterative Stage*, when the global illumination of the scene is calculated. This is an iterative process, usually some sort of Gauss-Seidel strategy, which computes the energy being transported through all the links in the scene, refining those links when necessary. One common approach is to apply a three-step process to every patch of the scene in each iteration:

1. *Refinement.* In this step each link of the target patch is analyzed and adaptively refined when the energy transported through this link exceeds a threshold value. The element with the largest area is subdivided into four new elements, building a quadtree hierarchy.

2. *Gathering.* Once the refinement step for a patch is completed, the energy received from the rest of elements in the scene is computed.

3. *Sweeping.* After gathering the light energy coming from the rest of the scene, the radiosity values of the patch are coherently updated along the hierarchical structure resulting from the two previous steps. Each element adds its energy to its children and weights the radiosity received from its parent.

Each iteration is completed once all the patches of the scene have been processed. Then, the convergence is checked (see Figure 1), comparing with a certain threshold the difference in the total energy transported between two consecutive iterations. If the convergence criterion is not fulfilled a new iteration begins.

3 Parallel Hierarchical Radiosity

Our parallel approach to HR targets systems that combine several independent nodes with multiple cores per node. The scene is partitioned into non-overlapping sub-scenes, and the computation of each sub-scene is carried out independently in a different node. Our method lies in a SPMD paradigm, with a unique process per node and message passing for communicating updated illumination values among nodes. Within a node, the HR algorithm is applied to a sub-scene concurrently by multiple tasks that exploit the patch-level parallelism in the *Linking* and *Iterative Stages*. Our implementation for this shared-memory scenario is based on POSIX threads, lightweight processes that allows us to take advantage of the different computational resources in the system, such as multiple cores per node and SMT capabilities per core.

3.1 Distributed-Memory Solution for HR

The irregular behavior of the hierarchical approach to radiosity, based on an adaptive refinement, makes difficult to achieve a good parallel solution, above all in a distributed-memory context. One of the first decisions to be taken is whether the geometric data of the input scene should be completely distributed among the different nodes or not. By distributing also data in addition to only computation, larger scenes could be theoretically processed, as the total capacity of the system memory is exploited. However,

 ISBN 13: 978-84-613-5510-5

this kind of solution dramatically increases the communication —in number and size— among the nodes in the system, especially for solving visibility queries. Furthermore, this would mean a more coupled execution in all the system, since a greater number of synchronization points would be needed, hindering performance.

Our approach is based on the minimization of communications and on avoiding to establish an excessive number of synchronization points among the different processes, yet without renouncing to process high complex scenes. With that aim in mind, only the input patches (coarse geometric data) has been replicated in the memory of every node. This allows the resolution of visibility queries with no communication at all, and the pay-back is not too high anyway, given the large amounts of memory per node and the low storage requirements of the coarse patches.

In Figure 2, an outline of the parallel algorithm executed in each node is depicted. The three stages seen in the sequential method are carried out in parallel by all the processes in the distributed-memory system, with a unique process running on each node. Communication among nodes is implemented by message passing, using the MPI standard. Specifically, asynchronous non-blocking functions are used. The steps that involve communication with other nodes are shadowed in the diagram. Within a node, the *Linking* and the *Iterative Stages* can be executed concurrently, spawning multiple threads within the process, as will be described in Subsection 3.2.

The preliminary work regarding the loading of the input scene and the construction of auxiliary structures for accelerating visibility determination is performed concurrently in every node, but is not parallelized (first step of the *Initial Stage* in Figure 2). As a result of this stage, all the nodes keep the initial patches (coarse geometric data) in its local memory. That will permit to avoid a lot of communication in the next two stages, as commented above.

The other main task carried out during the *Initial Stage* is to distribute the computation among the nodes by making a partition of the scene. Each node assigns itself one of the disjoint sub-scenes resulting from this partition. That sub-scene will be the local scene in which the global illumination will be computed by the process running in the node. Although a uniform geometric partition has been employed, specifically a regular 3D grid with a final volume optimization to obtain a tight-fitting bounding box of each sub-scene, this parallel proposal is independent of the kind of partition to be applied. Nevertheless, convex geometric partitions allow the exploitation of spatial locality of the objects in a scene.

With regards to the *Linking Stage*, all the patches in the local sub-scene are processed and two lists of initial links are computed, one with interactions with other patches in the local sub-scene (local links), and the other one with links to patches in the rest of the scene (remote links). Since all the initial geometry is accessible in the local memory, no communication among nodes is needed in this stage.

In the *Iterative Stage*, the three steps involved in the sequential HR iteration are computed for the local sub-scene using the local and remote links of each patch as a starting point. This stage entails communication among nodes, since data from remote nodes should be refreshed for each new iteration. Specifically, two different kinds of remote data need to be updated between iterations: radiosity values of patches, for the

 ISBN 13: 978-84-613-5510-5

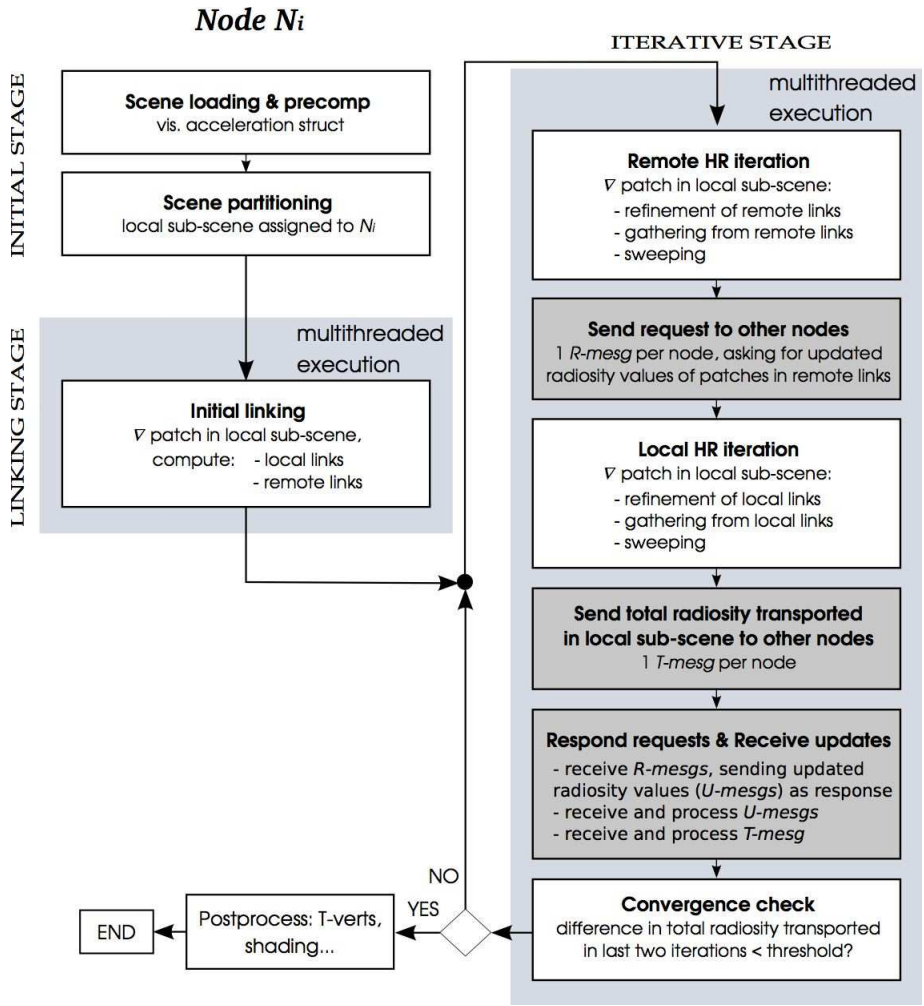


Figure 2: Outline of the distributed-memory scheduling

refinement and gathering of remote links, and the total radiosity transported in each sub-scene, for the convergence checking.

As it was commented above, one of the objectives behind this parallel approach for distributed-memory platforms is to keep as low as possible the number of messages to exchange between nodes, so as to favor an asynchronous and independent execution with few synchronization points among nodes. Keeping this in mind a scheduling with 6 steps for the *Iterative Stage*, as depicted in Figure 2, has been proposed.

The first thing a node would do in our scheme should be the processing of the remote links associated with all the patches in the local sub-scene (*Remote HR iteration* in Figure 2). Thus, the three steps —refinement, gathering and sweeping– of a HR iteration are applied to these links. A decision on splitting up the processing of local and remote links has been taken based on two reasons: firstly, processing the remote energy earlier assures the presence of energy to be transported in every sub-scene, even

 ISBN 13: 978-84-613-5510-5

though some of them have no light sources; on the other hand, the radiosity values from remote patches that interact with the local patches must be updated after each iteration. This update is done by means of message passing, and it means a first message requesting the remote radiosity values needed to the rest of nodes. Our scheduling tries to overlap this communication phase with the processing of the local links.

Therefore, the second step after finishing with the remote links is to send a message to each of the rest of nodes (*R-mesg* in *Send request to other nodes* in Figure 2), asking for the updated radiosity values that will be needed in the next iteration. The information that need to be sent for this request is only the *id* of the elements whose radiosity value is needed.

In the step three, *Local HR iteration* in Figure 2, the local links are processed. Once all the energy has been transported in the local sub-scene for an iteration, the total radiosity value obtained is sent to the rest of nodes in step four, *Send total radiosity transported* (*T-mesg*).

The fifth step, *Respond requests & Receive updates* in Figure 2, deals with the exchange of messages among nodes. Each node receives request messages, *R-mesg*, that must be properly responded by sending messages with the updated radiosity values requested by each node (*U-mesg*). The *U-messages* sent by the rest of nodes are also received in this step, together with the *T-messages* with the total radiosity transported in each sub-scene.

The last step, *Convergence check*, does not involve communication and is carried out in every node like the sequential version: the total energy transported within the whole scene between two consecutive iterations is compared with a threshold. If the convergence criterion is not fulfilled a new iteration begins.

3.2 HR on SMT Multi-core Processors

At this point, HR computation on shared-memory parallel environments is addressed by applying a multithreading approach to exploit all potential computational resources available in each node: multiple computational cores, all of them with local access to a common memory, as well as potential SMT capabilities.

Initially, only one thread (*Main thread*) is being executed in the node. The preliminary work regarding the loading of the input scene and the construction of visibility acceleration structures is performed by this thread and is not parallelized within a node (*Initial Stage* in Figure 2).

After this stage, multiple threads are spawned to carry out the radiosity computation. Thus, there is only one process running on the physical node, but it consists of multiple threads sharing the same virtual address space and exploiting the multiple cores and SMT capabilities available in the node. These threads will work concurrently until convergence is achieved in the *Iterative Stage*. The number of threads to be spawned, t, can be different on each node and can be either the total number of processing cores in the node or not.

This shared-memory approach applies a coarse grain parallelism, both during the *Linking* and *Iterative Stages*. Thus, once a patch is assigned on demand to a thread,

 ISBN 13: 978-84-613-5510-5

all the computation associated with that patch during the stage is carried out by this thread. Other approximations to HR on SMP systems are based on task pools, using a finer and more specific task division. Our solution simplifies thread synchronization and minimizes the overhead needed in finer grain implementations, leading to good results, as will be shown in Section 4.

Of course, every piece of the code performed by the threads must be thread-safe, since they are running simultaneously in a shared address space. Therefore, multiple access to shared data must be satisfied and protected, avoiding race conditions and deadlocks among the threads. All these issues are managed by setting critical sections in the code by means of mutual exclusion (*mutex*) algorithms and operations. Essentially, the scheduling can be summarized in the following stages:

1. At the beginning of the *Linking/Iterative Stage* the first patch to be processed by each thread is pre-assigned. Simply, Thr_1 will be in charge of P_1, Thr_2 of P_2 and so on.

2. The rest of patches are assigned on demand. An index variable, *nextFree* points to the next unprocessed patch. Thus, every time a thread finishes with a patch it checks whether the *nextFree* variable is indexing a valid patch (its value is not higher than N, the total number of patches in the local scene). If so, it takes the patch and updates the variable, just increasing its value.

Since *nextFree* is a variable shared by all the threads in a node and, its access must be protected with a mutex variable. On one side, the update of this variable is a critical section, and on the other, we must guarantee that each patch is only assigned to a thread.

During the *Iterative Stage*, specific actions must be taken due to the refinement process performed during the HR computation. Thus, multiple threads could try to subdivide the same element while refining different links. A link refinement means that either the source or the interacting destination element is subdivided, so different threads may try to subdivide the same element at the same time. Therefore, element subdivision is an important critical section in this scenario.

To obtain an efficient multithreaded HR computation, a simple yet effective mutual exclusion protocol to deal with the refinement process has been implemented. This protocol, together with a little modification in the sequential version of the refinement code, allows the management of the patch subdivision tasks during the refinement process, providing mutual exclusion without any deadlock efficiently and with a minimal storage cost. The proposed protocol needs a maximum of $t+1$ mutex variables, being t the total number of threads spawned in the node, and it permits a thread-safe adaptive refinement of the scene. Specific details about this mutual exclusion protocol can be read in [14].

Additionally, the convergence test at the end of each iteration must be considered in the scheduling of the *Iterative Stage*:

1. Each thread uses a local variable, *localRad*, to accumulate all the radiosity being gathered by the patches processed during an iteration.

 ISBN 13: 978-84-613-5510-5

(a) (b)

Figure 3: Test scenes: (a) *Building* (b) *Livingroom*

2. A new shared variable, *totalRad*, is needed to add up the contribution of the energy gathered by all the threads at the end of each iteration. Again, the access to this variable is protected with a mutex variable.

3. A barrier before starting each new iteration of the *Iterative Stage* acts as a synchronization point avoiding race conditions. It also allows each thread to check the convergence of the algorithm with the updated *totalRad* variable.

4 Experimental Results

The HR parallel solution presented in this paper has been tested on a system with eight nodes with 8 GB RAM and two Intel Xeon E5520 2.26 GHz quad core processors per node, with a 2-context SMT configuration enabled on each core (Intel HyperThreading), resulting in a total of 16 virtual processing units per node (though with only 8 physical cores per node). All nodes are equipped with IB 4X DDR cards (Qlogic IBA7220), so they communicate each other through a low latency InfiniBand network with 16 Gb/s of effective bandwidth.

Our parallel implementation was coded using the C programming language. The compiler used for these experiments was GNU gcc version 4.1.2. The POSIX threads library (*Pthreads*)is used to implement all the thread-related issues. POSIX threads are the best alternative when writing portable multithreaded code, as it offers a system-level standard library, much more flexible and versatile than higher-level libraries like OpenMP. Message-passing is managed through the MVAPICH2 library, a free implementation of the MPI API especially designed for InfiniBand and other low latency networks.

Two input scenes have been used for our tests (see Figure 3): *Building* and *Livingroom*, with respectively 2880 and 6960 input triangles, and 135480 and 34314 output triangles after the HR refinement. The scene *Building* has multiple identical rooms

 ISBN 13: 978-84-613-5510-5

Table 1: Performance results: Time (s) and Speedup

		Building		Livingroom	
Node	**Threads/node**	**Time**	**Speedup**	**Time**	**Speedup**
1	1	42.91	1	95.93	1
	2	21.76	1.97	48.72	1.97
	4	11.95	3.59	25.69	3.73
	8	6.52	6.58	13.45	7.13
	16	5.25	8.17	10.87	8.83
2	1	24.16	1.78	61.11	1.57
	2	12.88	3.33	32.16	2.98
	4	6.90	6.22	17.98	5.34
	8	3.86	11.12	9.77	9.82
	16	3.05	14.07	8.43	11.37
4	1	12.86	3.34	49.07	1.95
	2	6.84	6.27	27.26	3.52
	4	3.70	11.60	16.04	5.98
	8	2.11	20.34	9.02	10.64
	16	1.78	24.11	7.92	12.11
8	1	6.60	6.50	36.78	2.61
	2	3.45	12.44	20.44	4.69
	4	1.89	22.70	11.99	8.00
	8	1.25	34.33	6.88	13.94
	16	1.15	37.31	6.03	15.91

communicated by doors, so radiosity is transported between contiguous spaces. In contrast, *Livingroom* has a irregular distribution of objects with different complexity within a unique room.

In Table 1 the execution time and the corresponding speedup achieved for the HR computation of the two test scenes are shown. Different configurations of distributed and shared-memory resources have been checked: the first column shows the number of distributed-memory nodes used for the computation, whereas the second column indicates the number of threads spawned per node. Since each node of the target platform has really 8 physical cores, running 16 threads per node allow us to effectively exploit the SMT support available in Xeon processors.

In order to analyze the table, it should be noticed that the different configurations with only one node show a pure shared-memory scenario, allowing us to confirm the good performance of our multithreading approach. Thus, speedups of 8.17 and 8.83 have been obtained for the two scenes, significant values considering that there are only 8 physical cores within a node.

On the other hand, the shadowed rows in the table correspond to a pure distributed-

 ISBN 13: 978-84-613-5510-5

memory configuration, with a unique thread running on each node. Looking at the execution times, we can appreciate the drastic reduction achieved by our parallel approach for the *Building* scene in all cases (up to 6.5 for 8 nodes with only one thread per node, and a maximum of 37.31 for 8 nodes with 16 threads per node). The *Livingroom* scene achieves good results with regards to the multithreaded shared-memory part, but gets a poorer distributed-memory performance, probably due to the work imbalance across the different nodes produced by a more irregular geometric data distribution. Since our parallel HR approach is independent of the scene partitioning method, we expect to improve the results for irregular scenes through spatially adaptive, non-uniform partitions.

5 Conclusions and Future Work

This work approaches the parallelization of the HR method, a reference model in global illumination, in a hybrid context, exploiting distributed and shared memory architectures. The proliferation of multi-core processors with SMT capabilities is causing scenarios with clusters of SMP machines to become increasingly frequent. Our approach is based on: a workload distribution among nodes through a convex partition of the scene; a minimum number of message-passing communications among nodes thanks to the replication of the coarse geometric data in the different nodes; an efficient multithreaded scheduling within a node based on kernel-level threads, taking advantage of the multiple computing resources with access to a shared memory; and a low-cost mutual exclusion algorithm for the concurrent refinement of the scene.

In this paper we describe the first steps towards a complete hybrid parallel solution for HR. First results have been obtained using a uniform space partition, but we expect to improve the performance by means of a non-uniform partition that prevent load imbalance among nodes. Besides, the full hybrid system will be enhanced in a future version with an extension to heterogeneous multi-core systems, using GPGPU.

Acknowledgements

This work was partially supported by the Ministry of Education and Science of Spain under the contract MEC TIN 2007-67537-C03 and also supported by the Xunta de Galicia under the contracts 08TIC001206PR, INCITE08PXIB105161PR.

References

[1] M. F. Cohen, J. R. Wallace, Radiosity and realistic image synthesis, Academic Press Professional, 1993.

[2] L. M. Liebrock, S. P. Goudy, Methodology for modelling SPMD hybrid parallel computation, Concurrency and Computation: Practice & Experience 20 (8) (2008) 903–940.

 ISBN 13: 978-84-613-5510-5

[3] R. E. M. Ramírez, Adaptive and depth buffer solutions with bundles of parallel rays for global line monte carlo radiosity, Ph.D. thesis, Universitat Politècnica de Catalunya (2004).

[4] F. Sillion, J.-M. Hasenfratz, Efficient parallel refinement for hierarchical radiosity on a DSM computer, in: Proc. Third Eurographics Workshop on Parallel Graphics and Visualization, 2000, pp. 61–74.

[5] J. P. Singh, A. Gupta, M. Levoy, Parallel visualization algorithms: Performance and architectural implications, IEEE Computer Graphics and Applications 27 (7) (1994) 45–55.

[6] F. Baiardi, P. Mori, L. Ricci, Parallel hierarchical radiosity: the PIT approach, Applied Parallel Computing (Lecture Notes in Computer Science) Volume 3732/2006 (2006) 1031–1040.

[7] C.-C. Feng, S.-N. Yang, A parallel hierarchical radiosity algorithm for complex scenes, in: Proc. Third Parallel Rendering Symposium (IEEE-ACM/SIGGRAPH) (PRS'97), 1997, pp. 71–78.

[8] R. Garmann, On the partitionability of hierarchical radiosity, in: 1999 IEEE Symp. on Parallel Visualization and Graphics, 1999, pp. 69–78.

[9] J. Hippold, G. Rünger, Task pool teams for implementing irregular algorithms on clusters of SMPs, in: Proc. International Parallel and Distributed Processing Symposium (IPDPS'03), 2003, p. 54.2.

[10] C. Dachsbacher, M. Stamminger, G. Drettakis, F. Durand, Implicit visibility and antiradiance for interactive global illumination, ACM Transactions on Graphics 26 (3) (2007) 61:1–61:10.

[11] Z. Dong, J. Kautz, C. Theobalt, H.-P. Seidel, Interactive global illumination using implicit visibility, in: Pacific Conference on Computer Graphics and Applications, IEEE Computer Society, Washington, DC, USA, 2007.

[12] T. Ritschel, T. Engelhardt, T. Grosch, H.-P. Seidel, J. Kautz, C. Dachsbacher, Micro-rendering for scalable, parallel final gathering, ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2009) 28 (5).

[13] P. Hanrahan, D. Saltzman, L. Aupperle, A rapid hierarchical radiosity algorithm, in: Proc. SIGGRAPH'91, Vol. 25, 1991, pp. 197–206.

[14] E. J. Padrón, M. Amor, M. Bóo, R. Doallo, High performance global illumination on multi-core architectures, in: Proc. of the 17th Euromicro Conference on Parallel, Distributed and Network Based Processing (PDP 2009), 2009, pp. 93–100.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A New Parallel Implementation of the RX Algorithm for Anomaly Detection in Hyperspectral Images

A. Paz[2], J. M. Molero[1], E. M. Garzón[1], J. A. Martínez[1] and A. Plaza[2]

[1] *Department of Computer Architecture and Electronics, University of Almería*

[2] *Department of Technology of Computers and Communications, University of Extremadura*

emails: apazgal@unex.es, jmolero@ace.ual.es, gmartin@ual.es, jamartine@ual.es, aplaza@unex.es

Abstract

Remotely sensed hyperspectral sensors provide image data containing rich information in both the spatial and the spectral domain, and this information can be used to address detection tasks in many applications. In many surveillance applications, the size of the objects (targets) searched for constitutes a very small fraction of the total search area and the spectral signatures associated to the targets are generally different from those of the background, hence the targets can be seen as anomalies. One of the most widely used and successful algorithms for anomaly detection in hyperspectral images is the one proposed by Reed and Xiaoli, commonly known as RX algorithm. Despite its wide acceptance and high computational complexity when applied to real hyperspectral scenes, few approaches have been developed for parallel implementation of this algorithm due to the complex calculation of the inverse of the sample covariance matrix in parallel. In this paper, we evaluate the suitability o using a local approach for the calculation of the inverse of the sample covariance matrix of a high-dimensional hyperspectral scene in parallel. The considered approach is quantitatively evaluated using hyperspectral data collected by the NASA's Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) system over the World Trade Center (WTC) in New York, five days after the terrorist attacks that collapsed the two main towers in the WTC complex. The precision of the algorithms is evaluated by quantitatively substantiating their capacity to automatically detect the thermal hot spot of fires (anomalies) in the WTC area.

Key words: Hyperspectral imaging, anomaly detection, Reed-Xiaoli algorithm (RX).

 ISBN 13: 978-84-613-5510-5

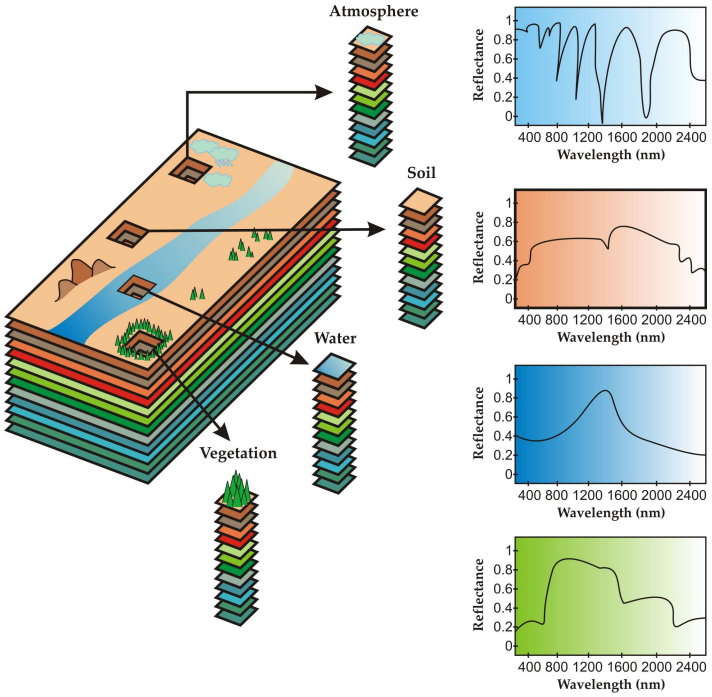


Figure 1: Concept of hyperspectral imaging.

1 Introduction

Hyperspectral imaging [1] is concerned with the measurement, analysis, and interpretation of spectra acquired from a given scene (or specific object) at a short, medium or long distance by an airborne or satellite sensor [2]. Hyperspectral imaging instruments such as the NASA Jet Propulsion Laboratory's Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) [3] are now able to record the visible and near-infrared spectrum (wavelength region from 0.4 to 2.5 micrometers) of the reflected light of an area 2 to 12 kilometers wide and several kilometers long using 224 spectral bands. The resulting "image cube" (see Fig. 1) is a stack of images in which each pixel (vector) has an associated spectral signature or *fingerprint* that uniquely characterizes the underlying objects [4]. The resulting data volume typically comprises several GBs per flight [5].

Anomaly detection is an important task for hyperspectral data exploitation. An anomaly detector enables one to detect spectral signatures which are spectrally distinct from their surroundings with no *a priori* knowledge. In general, such anomalous signatures are relatively small compared to the image background, and only occur in the image with low probabilities. A well-known approach for anomaly detection was developed by Reed and Yu, and is referred to as the RX algorithm, which has shown success in anomaly detection for multispectral and hyperspectral images [4]. The RX uses the pixel currently being processed as the matched signal. Since the RX uses the sample covariance matrix to take into account the sample spectral correlation, it performs the

 ISBN 13: 978-84-613-5510-5

same task as the Mahalanobis distance, which has been widely used in hyperspectral imaging applications [6]. A variation of the algorithm consists in applying the same concept in local neighborhoods centered around each image pixel, this is known as the kernel version of the RX algorithm [7].

Despite its wide acceptance and high computational complexity when applied to real hyperspectral scenes, few approaches have been developed for parallel implementation of this algorithm due to complexity of calculating the sample covariance matrix (and its inverse) in parallel. The success of the kernel version of the algorithm in [7] led us to believe that a standard data partitioning framework for parallel implementation may provide different (or even better) results if the sample covariance matrices are calculated independently for small data portions rather than computing the sample covariance matrix for the entire hyperspectral image, which requires additional inter-processor communications that may reduce parallel performance. This aspect is crucial for the RX implementation since the consideration of a local or global strategy for the computation of the sample covariance matrix is expected to show an important impact in the scalability of the parallel solution but there is a trade-off between the increase in parallel efficiency and the quality of the final solution in terms of anomaly detection accuracy.

In this paper, we investigate the use of a local approach for the calculation of the inverse of the sample covariance matrix, in which each processing node calculates the covariance matrix of its local partition in parallel and inter-processor communications are significantly reduced. The remainder of the paper is structured as follows. Section 2 briefly describes the classic RX algorithm. Section 3 describes the parallel implementation adopted in this work. Section 4 describes the hyperspectral data set considered in experiments, which comprises a data set collected by the NASA's Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) system over the World Trade Center (WTC) in New York, five days after the terrorist attacks that collapsed the two main towers in the WTC complex. Section 5 conducts a detailed experimental assessment of the precision and scalability of the proposed parallel implementations using the aforementioned scene as a relevant case study. Finally, section 6 concludes with some remarks and hints at plausible future research.

2 RX algorithm

The RX algorithm has been widely used in signal and image processing [8]. The filter implemented by this algorithm is referred to as RX filter and defined by the following expression:

$$\delta^{\mathrm{RX}}(\mathbf{x}) = (\mathbf{x} - \mu)^T \mathbf{K}^{-1} (\mathbf{x} - \mu), \tag{1}$$

where $\mathbf{x} = \left[x^{(0)}, x^{(1)}, \cdots, x^{(n)}\right]$ is a sample, n-dimensional hyperspectral pixel (vector), μ is the sample mean of the hyperspectral image and $\mathbf{K}$ is the sample data covariance matrix. As we can see, the form of δ^{RX} is actually the well-known Mahalanobis distance [6]. It is important to note that the images generated by the RX algorithm are generally

 ISBN 13: 978-84-613-5510-5

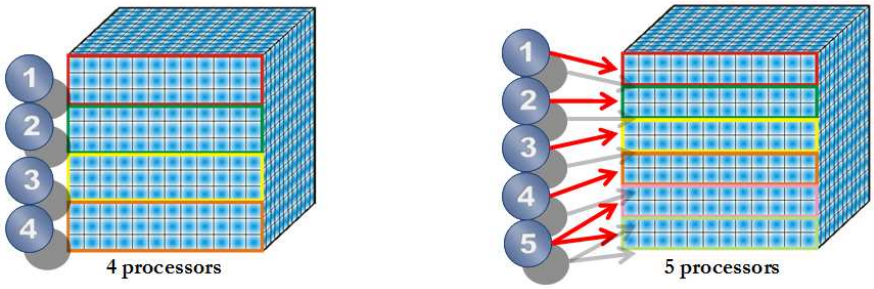


Figure 2: Spatial-domain decomposition of a hyperspectral data set.

gray scale images. In this case, the anomalies can be categorized in terms of the value returned by RX, so that the pixel with higher value of $\delta^{\mathrm{RX}}(\mathbf{x})$ can be considered the first anomaly, and so on.

3 Parallel implementation

3.1 Data partitioning strategies

In the considered parallel algorithms, a data-driven partitioning strategy has been adopted as a baseline for algorithm parallelization. Specifically, two approaches for data partitioning have been tested [9]:

- *Spectral-domain partitioning*. This approach subdivides the multi-channel remotely sensed image into small cells or sub-volumes made up of contiguous spectral wavelengths for parallel processing.

- *Spatial-domain partitioning*. This approach breaks the multi-channel image into slices made up of one or several contiguous spectral bands for parallel processing. In this case, the same pixel vector is always entirely assigned to a single processor, and slabs of spatially adjacent pixel vectors are distributed among the processing nodes (CPUs) of the parallel system. Fig. 2 shows two examples of spatial-domain partitioning over 4 processors and over 5 processors, respectively.

Previous experimentation with the above-mentioned strategies indicated that spatial-domain partitioning can significantly reduce inter-processor communication, resulting from the fact that a single pixel vector is never partitioned and communications are not needed at the pixel level [9]. In the following, we assume that spatial-domain decomposition is always used when partitioning the hyperspectral data cube.

3.2 Parallel algorithm

Our parallel version of the RX algorithm for anomaly detection adopts the spatial-domain decomposition strategy depicted in Fig. 2 for dividing the hyperspectral data

 ISBN 13: 978-84-613-5510-5

cube in master-slave fashion. The approach considered in this work represents a variation of the one presented in [10, 11], in which a global approach is adopted for the covariance matrix calculation. In this work, we use a local approach for the calculation of the covariance matrix instead, i.e. each node calculates the covariance matrix of its local partition. The parallel algorithm is given by the following steps:

1. The master processor divides the original image cube into P spatial-domain partitions and distributes them among the workers.

2. The master calculates the n-dimensional mean vector $\mathbf{m}$ concurrently, where each component is the average of the pixel values of each spectral band of the unique set. This vector is formed once all the processors finish their parts.

3. Each worker calculates the covariance matrix of its local partition, and applies (locally) the RX filter given by the Mahalanobis distance to all the pixel vectors in the local partition as follows: $\delta^{(\mathrm{RX})}(\mathbf{x}) = (\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}(\mathbf{x}-\mathbf{m})$, and returns the local result to the master.

4. The master now selects the t pixel vectors with higher associated value of $\delta^{(\mathrm{RX})}$, and uses them to form a final set of targets $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_t\}$.

4 Hyperspectral data set

The image scene used for experiments in this work was collected by the AVIRIS instrument, which was flown by NASA's Jet Propulsion Laboratory over the World Trade Center (WTC) area in New York City on September 16, 2001, just five days after the terrorist attacks that collapsed the two main towers and other buildings in the WTC complex. The full data set selected for experiments consists of 614×512 pixels, 224 spectral bands and a total size of (approximately) 140 MB. The spatial resolution is 1.7 meters per pixel. The leftmost part of Fig. 3 shows a false color composite of the data set selected for experiments using the 1682, 1107 and 655 nm channels, displayed as red, green and blue, respectively. Vegetated areas appear green in the leftmost part of Fig. 3, while burned areas appear dark gray. Smoke coming from the WTC area (in the red rectangle) and going down to south Manhattan appears bright blue due to high spectral reflectance in the 655 nm channel.

Extensive reference information, collected by U.S. Geological Survey (USGS), is available for the WTC scene[1]. In this work, we use a U.S. Geological Survey thermal map[2] which shows the locations of the thermal hot spots (which can be seen as anomalies) at the WTC area, displayed as bright red, orange and yellow spots at the rightmost part of Fig. 3. The map is centered at the region where the towers collapsed, and the temperatures of the targets range from 700F to 1300F. Further information available from USGS about the thermal hot spots (including location and temperature)

[1] http://speclab.cr.usgs.gov/wtc

[2] http://pubs.usgs.gov/of/2001/ofr-01-0429/hotspot.key.tgif.gif

 ISBN 13: 978-84-613-5510-5

Figure 3: False color composition of an AVIRIS hyperspectral image collected by NASA's Jet Propulsion Laboratory over lower Manhattan on Sept. 16, 2001 (left). Location of thermal hot spots in the fires observed in World Trade Center area, available online: http://pubs.usgs.gov/of/2001/ofr-01-0429/hotspot.key.tgif.gif (right).

is reported on Table 1. The thermal map displayed in the rightmost part of Fig. 3 will be used in this work as ground-truth to validate the target detection accuracy of the proposed parallel algorithms and their respective serial versions.

5 Evaluation

The parallel computing platform used in this experiments is the Sun Fire x4600, it is composed of 8 quad 2.3 GHz AMD Opteron 8356 (32 cores), with 64 Gb of main memory. The operating system used at the time of experiments was Debian, and Open MPI was used as parallel interface programming. Although the selected parallel

Table 1: Properties of the thermal hot spots reported in the rightmost part of Fig. 3.

Hot spot	**Latitude (North)**	**Longitude (West)**	**Temperature (Kelvin)**
'A'	$40^o42'47.18"$	$74^o00'41.43"$	1000
'B'	$40^o42'47.14"$	$74^o00'43.53"$	830
'C'	$40^o42'42.89"$	$74^o00'48.88"$	900
'D'	$40^o42'41.99"$	$74^o00'46.94"$	790
'E'	$40^o42'40.58"$	$74^o00'50.15"$	710
'F'	$40^o42'38.74"$	$74^o00'46.70"$	700
'G'	$40^o42'39.94"$	$74^o00'45.37"$	1020
'H'	$40^o42'38.60"$	$74^o00'43.51"$	820

 ISBN 13: 978-84-613-5510-5

platform is based on a shared memory architecture, according to our experience MPI has shown how is able to exploit this architecture and adapts to the characteristic of our problem [13].

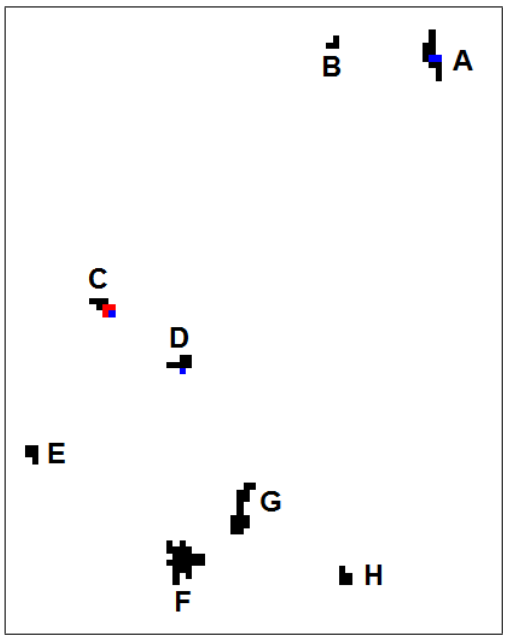


Figure 4: Detection results for the different implementations of RX, where black pixels represent the ground-truth, red pixels represent the targets detected by the serial version, and blue pixels represent the targets detected by the parallel version.

The evaluation of RX capability to automatically detect the anomalies is based on the results showed in Fig. 4. For illustrative purposes, Fig. 4 shows the detection results obtained by the serial and the parallel implementation of RX using different number of processors ($P = 16$ and $P = 32$). In all cases, the number of target pixels to be detected was set to $t = 30$ after calculating the virtual dimensionality of the data [4]. The pixels labeled in black color in Fig. 4 represent the original targets. The pixels labeled in red color in Fig. 4 represent the pixels detected by the serial version. Finally, the pixels labeled in blue color in Fig. 4 represent the pixels detected by the parallel version (the results obtained using $P = 16$ and $P = 32$ processors were overlapped). As shown by Fig. 4, the same targets were detected by the parallel version regardless of the number of partitions. In this case, a local approach is used for calculating the covariance matrix, while the serial version uses a global approach. The global strategy introduces additional inter-processor communications which negatively affect parallel performance as indicated in [10, 11], while the local strategy exhibits results which are almost identical to those obtained by the global strategy. In all cases, only three out of eight targets (i.e. those labeled as 'A', 'C' and 'D') were detected. However, increasing the number of targets to be detected t increases the detection results. In this work, however, we have decided to adopt the value $t = 30$ based on the calculation of the

 ISBN 13: 978-84-613-5510-5

virtual dimensionality of the data, which provides an objective criterion for setting the number of targets.

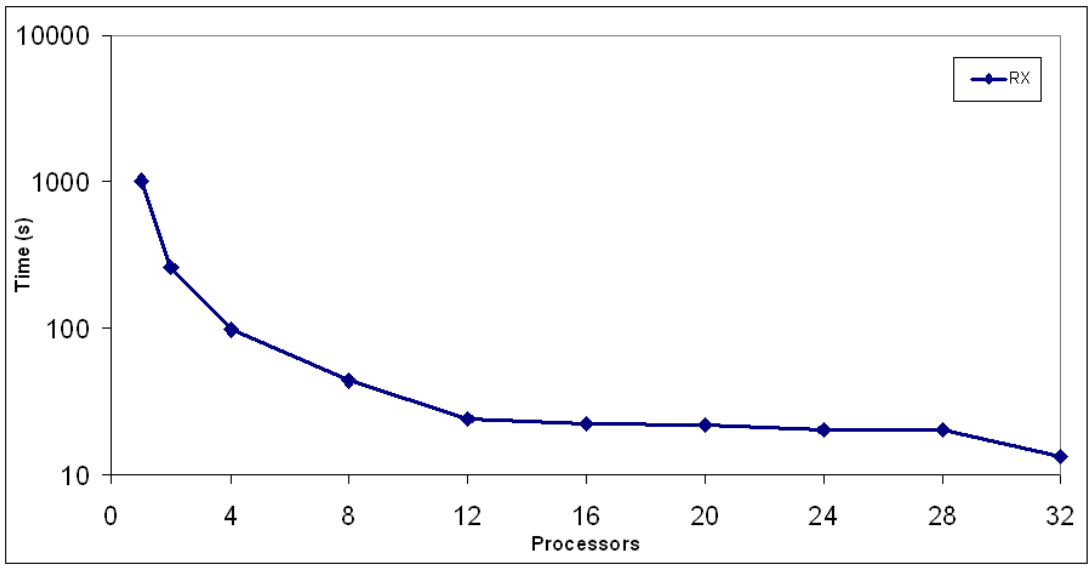


Figure 5: Run-Time of parallel RX.

Fig. 5 shows the performance of parallel RX for different number of processors P. It shows a super-linear speed-up because $T(1)/T(P) > P$ for all values of P. Experiences with super-linear speed-up in parallel computing have been described in the literature [12, 14]. The super-linear speed-up is achieved when the computational power increases as the load of every processor decreases. Frequently, this behavior is related to applications which need large memory requirements, since the memory management is improved as P increases. The memory requirements to store the hyperspectral images are large. Consequently, the parallel RX, based on the partition of hyperspectral image, shows superlinear speed-up due to the improvements of the memory management when P increases.

Hence, the speed-up is not an appropriate parameter to evaluate the scalability of parallel RX, since the sequential RX is penalized by a hard memory management. With this aim, the parameter called Incremental Speed-up ($IncSpUp$) has been proposed as an alternative to the speed-up [12]. It provides information about how much the computing time diminishes when the number of PEs increases. $IncSpUp$ is defined as follows:

$$IncSpUp(2^k) = \frac{T(P = 2^{k-1})}{T(P = 2^k)}, \tag{2}$$

where T(P) is the run time of the execution with P PEs. Values of the Incremental Speed-up equal to 2 are equivalent to lineal speed-ups.

Figure 6 shows $IncSpUp$ obtained by the parallel RX algorithm using different number of processors on the considered parallel system. As shown by Figure 6, the

 ISBN 13: 978-84-613-5510-5

proposed parallel implementation provides good scalability results when compared to the implementation in [10, 11]. These results are a consequence of a key of parallel RX, that is, there are not communications among the processors which penalize the performance when P increases. However, the proposed parallel implementation of the RX algorithm can still be enhanced by further exploring different strategies for the parallel calculation of the covariance matrix.

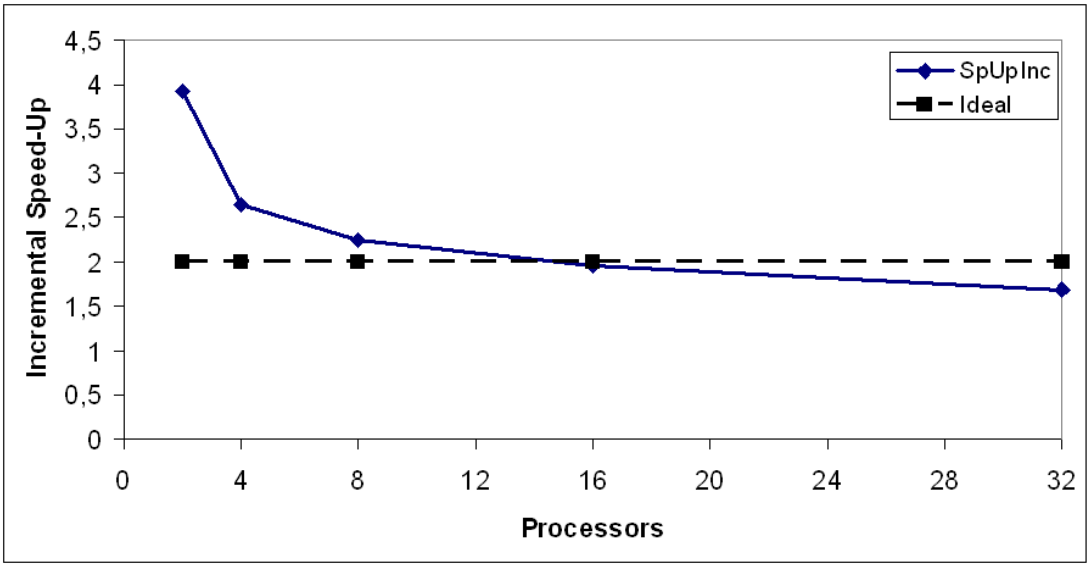


Figure 6: Incremental Speedup for the RX algorithm.

6 Conclusions and future research lines

In this paper, we have analyzed the accuracy and scalability of a new parallel implementation of the RX algorithm which uses a local approach for the calculation of the covariance matrix in parallel. The proposed approach has some competitive advantages (in terms of scalability) with regards to the commonly used (global) strategies adopted for calculating the covariance matrix in parallel when implementing this algorithm. The parallel version has been validated in the context of a real hyperspectral imaging application, focused on detecting the thermal hot spots (anomalies) of the fires in the World Trade Center area in New York City, just few days after the terrorist attacks of September 11th, 2001. Our experimental results indicate that the proposed (local) strategy provides similar accuracy results to those reported for the (global) one adopted in previous work, and presents potential for improving the scalability of the algorithm due to the fact that the proposed strategy reduces significantly the amount of inter-processor communications. Although the results reported in this work are encouraging, further experiments should be conducted in order to increase the scalability of the proposed parallel algorithms to a higher number of processors by resolving memory issues

 ISBN 13: 978-84-613-5510-5

and optimizing the parallel design of such algorithms. Experiments with additional scenes under different target/anomaly detection scenarios are also highly desirable.

Acknowledgements

This work has been supported by the European Communitys Marie Curie Research Training Networks Programme under reference MRTN-CT-2006-035927 (HYPER-I-NET) and by Spanish Ministry of Science and Innovation TIN2008-01117. Funding from the Spanish Ministry of Science and Innovation (HYPERCOMP/EODIX project, reference AYA2008-05965-C04-02) is gratefully acknowledged.

References

[1] A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging spectrometry for Earth remote sensing," *Science*, vol. 228, pp. 1147–1153, 1985.

[2] A. Plaza, J. A. Benediktsson, J. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, J. Gualtieri, M. Marconcini, J. C. Tilton, and G. Trianni, "Recent advances in techniques for hyperspectral image processing," *Remote Sensing of Environment*, vol. 113, pp. 110–122, 2009.

[3] R. O. Green, M. L. Eastwood, C. M. Sarture, T. G. Chrien, M. Aronsson, B. J. Chippendale, J. A. Faust, B. E. Pavri, C. J. Chovit, M. Solis *et al.*, "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sensing of Environment*, vol. 65, no. 3, pp. 227–248, 1998.

[4] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification.* Norwell, MA: Kluwer, 2003.

[5] A. Plaza and C.-I. Chang, *High performance computing in remote sensing.* Boca Raton: CRC Press, 2007.

[6] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction.* Springer, 2006.

[7] H. Kwon and N. M. Nasrabadi, "Hyperspectral anomaly detection using kernel rx-algorithm." in *International Conference on Image Processing (ICIP)*, 2004.

[8] I. Reed and X.Yu, "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distrihution." *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 38, pp. 1760–1770, 1990.

[9] A. Plaza, D. Valencia, J. Plaza, and P. Martinez, "Commodity cluster-based parallel processing of hyperspectral imagery," *Journal of Parallel and Distributed Computing*, vol. 66, pp. 345–358, 2006.

 ISBN 13: 978-84-613-5510-5

[10] A. Paz, A. Plaza, and S. Blazquez, "Parallel implementation of target and anomaly detection algorithms for hyperspectral imagery," *Proc. IEEE Geosci. Remote Sens. Symp.*, vol. 2, pp. 589–592, 2008.

[11] A. Paz, A. Plaza, and J. Plaza, "Comparative analysis of different implementations of a parallel algorithm for automatic target detection and classification of hyperspectral images," *Proc. SPIE*, vol. 7455, pp. 1–12, 2009.

[12] Garzón, E.M. and García, I. "A parallel implementation of the eigenproblem for large, symmetric and sparse matrices," *Recent advances in PVM and MPI, LNCS 1697*, Springer-Verlag, 1999, 380–387.

[13] S.Tabik, E.M. Garzón, I. García, and J.J. Fernández. High Performance Noise Reduction for Biomedical Multidimensional Data.*Digital Signal Processing*, Vol. 17, n. 4, pp. 724–736. 2007.

[14] Gustafson, J.L. Reevaluating Amdahl's Law, *Communications of the ACM*, 1988, 532–533

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Modeling artificial immunity against mammary carcinoma

Marzio Pennisi[1], Carlo Bianca[2], Francesco Pappalardo[1] and Santo Motta[1]

[1] *Department of Mathematics and Computer Science, University of Catania, Italy*

[2] *Politecnico di Torino, Torino, Italy*

emails: mpennisi@dmi.unict.it, carlo.bianca@polito.it, francesco@dmi.unict.it, motta@dmi.unict.it

Abstract

A typical, and unfortunately very spread, endogenous cancer is represented by the mammary carcinoma. Lollini et al.[1] prevented mammary carcinogenesis in HER-2/neu transgenic mice using the Triplex cellular vaccine under a Chronic schedule (vaccine cycles started at 6 weeks of age and continued up to the end of the experiment). When the vaccine is administered, immune system (IS) cells are stimulated to duplicate in order to eliminate tumor cells, going back to normal levels after cancer eradication. However, in endogeneous tumors, newborn cancer cells will be formed and then vaccine administrations are necessary to stabilize the cancer – immune system competition. Thus the system is unstable and will be stabilized by the external action of the vaccine.

The question whether the Chronic protocol is the minimal vaccination protocol yielding complete protection from tumor onset, or whether a lower number of vaccination cycles would provide a similar degree of protection, is still an open question. In order to answer to this question we presented in [2] an agent-based model of IS responses to vaccination. However computational models do not allow a qualitative and asymptotic analysis, neither an easy investigation of parameters' space. For this reason we are actually working on a ODE model that is realized upon the same conceptual scheme used for the computational model. The model equations are shown as follows:

$$\frac{dVC}{dt} = k_{in}(t) - (\mu_4 + \alpha_1 TC + \alpha_2 Ab + \alpha_3 NK) \cdot VC \tag{1}$$

$$\begin{aligned}\frac{dTAAv}{dt} &= \alpha_{10}(\mu_4 + \alpha_1 TC + \alpha_2 Ab + \alpha_3 NK) \cdot VC + \\ &\quad -(\mu_{10} + \mu_{11} B + \alpha_{14} Ab + \mu_{15} MP + \mu_{16} DC) \cdot TAAv\end{aligned} \tag{2}$$

$$\frac{dMP}{dt} = \alpha_{15} TAAv + \alpha_{16} TAAc - \mu_{17} MP \tag{3}$$

 ISBN 13: 978-84-613-5510-5

$$\frac{dDC}{dt} = \alpha_{17}TAAv + \alpha_{18}TAAc - \mu_{18}DC \tag{4}$$

$$\frac{dB}{dt} = \alpha_{20}TH + (\alpha_{21}IL2 - \mu_{20}) \cdot B \tag{5}$$

$$\frac{dTH}{dt} = \alpha_{22}TAAv + \alpha_{23}TAAc + \alpha_{24}VC + (-\mu_{21} + \alpha_{25}IL2 + \alpha_{26}IL12) \cdot TH \tag{6}$$

$$\frac{dIL12}{dt} = \alpha_{75} \cdot k_{in}(t) - (\mu_5 + \alpha_{27}TH + \alpha_{28}TC + \alpha_{29}NK) \cdot IL12 \tag{7}$$

$$\frac{dIL2}{dt} = \alpha_{30}TH - (\mu_{30} + \alpha_{31}B + \alpha_{32}TC) \cdot IL2 \tag{8}$$

$$\frac{dCC}{dt} = \left[\left(1 - \frac{CC}{c_{max}}\right)\right] \cdot [k_1\, CC] + p_1 - (k_2NK + k_3TC + k_4Ab) \cdot CC \tag{9}$$

$$\frac{dTC}{dt} = \alpha_{40} \cdot VC + (\alpha_{41} - \mu_{40}) \cdot TC \tag{10}$$

$$\frac{dTAAc}{dt} = \alpha_{50}(k_2NK + k_3TC + k_4Ab) \cdot CC - (\mu 50 + \\ +\mu_{61}B + \alpha_{62}Ab + \mu_{63}MP + \mu_{64}DC) \cdot TAAc \tag{11}$$

$$\frac{dAB}{dt} = \alpha_{70}B - (\mu_{70} + \alpha_{71}CC + \alpha_{72}VC + \alpha_{73}TAAc + \alpha_{74}TAAv) \cdot AB \tag{12}$$

Vaccine cells (VC) are administered through intraperitoneal injections following a predefined dosage. Inoculation is modeled by a function $k_{in}(t)$ which returns the number of vaccine cells inoculated into the host at time t if a that time an injection is scheduled. As the vaccine cells come from the external, this is the only source element in equation (1). Vaccine cells die for natural death (μ_4), killed by Cytotoxic T cells (TC), Natural killer cells (NK) or by specific antibodies (AB). Antigens released by vaccine cells ($TAAv$) are proportional to the number of vaccine cells that die. This is the source element of the first part of equation (2). $TAAv$ are subjected to degradation and phagocytosis by antigen presenting cells, i.e. by B cells, macrophages (MP), dendritic cells (DC). Moreover AB can bind to free antigens producing immune complexes. MP and DC activation (eqns. 3 and 4) depends mainly by tumor associated antigens released by VC and cancer cells (CC). MP and DC can die and can undergo to resting status ($-\mu_{17}MP$ and $-\mu_{18}DC$ terms). Antigen activated B (eqn. 5) can be stimulated to duplicate by helper T cells (TH) positive feedback. Interleukin 2 ($IL2$) plays an adjuvant role in this stimulation process. Death is modeled by $\mu_{20}B$ term.

Equation (6) models the priming of TH which can be primed through interactions with specialized antigen presenting cells, by major histocompatibility class II / peptide complex presentation. Presentation is not directly modeled, so the number of activated TH is estimated from the number of $TAAv$, $TAAc$ and VC present in the system. $IL2$ and $IL12$ also contribute to this priming. The death factor is modeled by $-\mu_{21}TH$ term. $IL12$ (eqn. 7) is introduced through vaccine administration, so it depends on the dosage. It is subjected to normal degradation ($-\mu_5 IL12$) and it is partially absorbed for mitotic and stimulation signals by TC and TH priming and NK activation. $IL2$

 ISBN 13: 978-84-613-5510-5

stimulates TH priming and primed TH produce further IL2. It is subjected to normal degradation ($-\mu_{30} IL2$) and it is partially absorbed for mitotic and stimulation signals in TC priming and B duplication. Equation (9) describes CC dynamics. The term $\left[\left(1 - \frac{CC}{c_{max}}\right)\right] \cdot [k_1\, CC]$ models CC growth. p_1 models the continuous production of newborn cancer cells due to transgenic nature of the host. The other terms describe CC death mainly due by NK, AB, TC actions. TC priming (eqn. 10) depends mainly by VC allogeneic major histocompatibility class I complex. Duplication factor is modeled by $\alpha_{41} TC$ term whereas the normal death factor is modeled by $-\mu_{40} TC$ term. $TAAc$ (eqn. 11) are proportional to the number of CC that die. This is the source element of the first part of the equation. $TAAc$ are subjected to degradation and phagocytosis by B, MP and DC. AB can also bind to free antigens producing immune complexes. Equation (12) describes AB dynamics. AB are released by B cells that differentiate into plasma B cells and are subjected to normal degradation (modeled by $-\mu_{70} AB$ term). Moreover they disappear in absolving their functions: binding to specific targets, i.e. CC, VC, $TAAv$ and $TAAc$. NK are constant an do not vary during the experiment (e.g. $d/dt\ NK = 0$). Initial Cauchy conditions are set to 0 for all the equations except for NK whose initial value is set according the leukocyte formula observed "in vivo".

Using known data from literature and data coming directly from the "in vivo" experiment we were able to find a tentative tuning for the model, capable to show a reasonable IS response that reflects the one observed in "in vivo" experiments and in the computational model. With this tuning we can try to describe the state of the "immune system – cancer competition" using three fundamental variables: the number of CC (representing the foreign pathogen), the number of TC representing the cellular response and the number of AB, principal outcome of the humoral response. Thus it is possible to represent the evolution of the system in a 3-D states space using the variables' curves, with the time representing the curves parameter, as shown in [3]. We firstly analyze the untreated scenario. In this scenario the number of cancer cells grows with no control up to the saturation threshold (Figure 1(a)). Figure 1(b) shows that the immune system is unable to engage in fight against cancer cells. The second scenario is represented by the administration of the Early vaccination schedule, composed by three vaccine cycles starting at the beginning of the experiment. A vaccine cycle consists of two vaccine administrations over two weeks followed by two weeks of rest [1]. The effect of the schedule is to contrast the initial growth of tumor cells (Figure 1(b)). This effect is presented in Figure 1(d) as a large loop in which the cancer cells growth is reduced by TC and AB action. After this initial phase, cancer cells start to grow again with no constraints and the straight line is similar to the plot of the untreated case. Finally we consider the Chronic schedule. This schedule consists on repeated Early cycle administrations for the entire life of the host. Looking at Figure 1(e) one can see that after an initial burst, cancer cells are eradicated and their level is kept near to 0. Figure 1(f) shows that the system (immune system - cancer) is stabilized and an equilibrium region is reached.

This preliminary analysis shows that the vaccine, when administered with a Chronic

 ISBN 13: 978-84-613-5510-5

schedule, is able to stabilize the immune system – cancer competition around values that are safe for the host. Asymptotic and sensitivity analyses, as well as analytical study of a simplified model is in progress and results will be presented in due course.

Figures

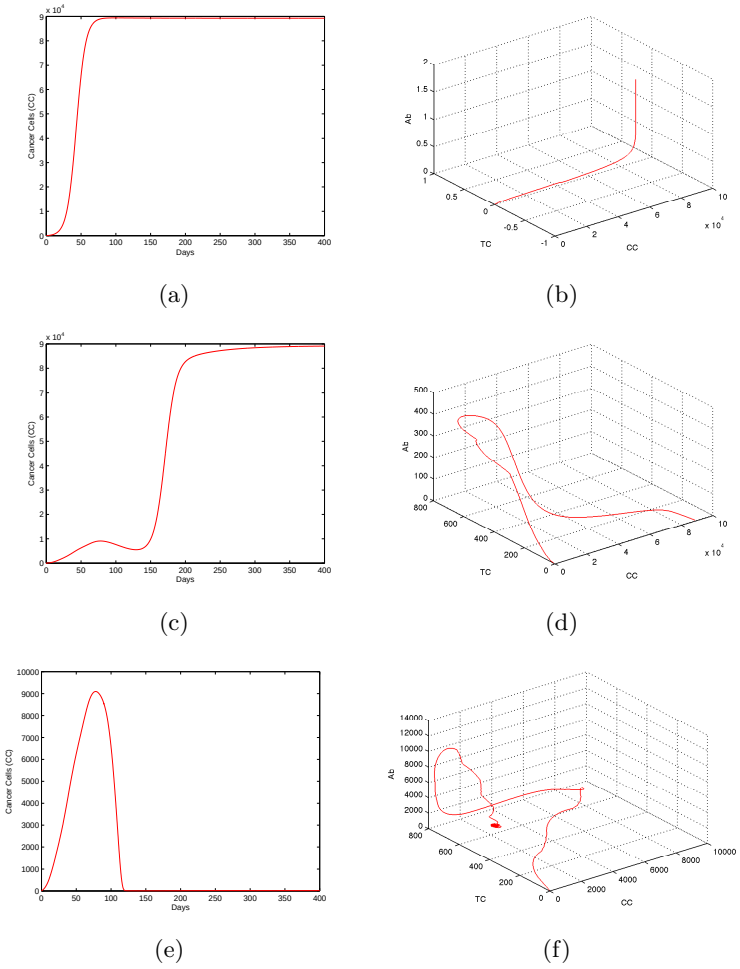


Figure 1: 3-D States' space for the untreated, Early and Chronic scenario

References

[1] De Giovanni C., *et. al.*: Immunoprevention of HER-2/neu transgenic mammary carcinoma through an interleukin 12-engineered allogeneic cell vaccine. Cancer Res. 64(11) (2004) 4001-4009.

[2] Pappalardo F., Castiglione,F., Lollini,P.-L., Motta,S.: Modelling and Simulation of Cancer Immunoprevention vaccine. Bioinformatics 21(12) (2005) 2891-2897.

[3] Pappalardo F., Motta S., Lollini P.-L., Mastriani E., Pennisi M. The Stabilization Effect of the Triplex Vaccine, Proc. of the 7th Int. FLINS Conference, Applied Artificial Intelligence, World Scientific, (2006) 587-592.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Improving the Growing Neural Gas Algorithm with Ensembles

Santiago Porras[1], Álvaro Alonso[1], Bruno Baruque[1], Hujun Yin[2], Emilio Corchado[3] and Jordi Rovira[4]

[1] *Dept. Civil Engineering, University of Burgos.*

[2] *School of Electrical and Electronic Engineering, University of Manchester.*

[3] *Dept. of Informatics and Automatics, University of Salamanca.*

[4] *Department of Biotechnology and Food Science, University of Burgos.*

emails: spa0001@alu.ubu.es, aad0038@alu.ubu.es, bbaruque@ubu.es, h.yin@manchester.ac.uk, escorchado@usal.es, jrovira@ubu.es

Abstract

This study presents a novel version of the Growing Neural Gas algorithm that is based on the use of ensemble techniques. Its aim is to improve the stability and classification capabilities of these types of topology preserving networks. This is accomplished through with the use of a semi-supervised algorithm that is able to incorporate unlabeled samples into the regular training of these networks thet use labeled samples; and through the use of ensemble training and a final fusion algorithm to generate a single final network. A comparative study was made of all these model combinations to provide a complete study of their capabilities. They are applied here to an interesting case of study: the classification of cured ham samples. Data sets were collected using an "electronic nose" to analyze the ham samples and the models proposed were used for the quality determination, based on that chemical measurements.

Key words: Artificial Neural Networks, Topology Preserving Networks, Automatic Learning, Ensemble Learning, Data Classification

1 Introduction

Artificial Neural Networks (ANNs) are very useful mathematical models owing to their ability to perform tasks such as data classification, data clustering and pattern matching without any need for a very complex model of the problem to be classified. Belonging to the Automated Learning family of algorithms, these networks are able to adapt to

 ISBN 13: 978-84-613-5510-5

the data set under study by "learning" from the examples extracted from a similar data set.

A combination of an electronic device for the analysis of volatile compounds (the electronic nose or "e-nose") with different combinations of an algorithm for incorporating unclassified samples into a machine learning algorithm and an ensemble summarization algorithm for topology preserving networks are tested in the classification of a wide variety of "Serrano" Hams samples. The pourpose is to establish whether this procedure is able to discriminate, in an easy and reliable way, between ham quality on the basis of different olfactory characteristics.

Consumer trust is a very important factor, when a product is being introduced into a new market or consolidated in an existing one. Dry-cured "Serrano" ham is a widely consumed traditional product in Spain that has also found a foreign market and is increasingly exported abroad. It is important to find quick and easy, low-cost techniques that apply simple parameters to evaluate the quality of these products prior to their purchase and consumption by the consumer. In this case, automated learning techniques are considered as a possible solution this problem.

Several devices have recently appeared that perform analytical techniques in the food industry to support the subjective decisions of professional human testers. One disadvantage of these alternative tests is that whatever humans interpret as tastes and smells, machines will inevitably interpret as complex, numeric measurements. Thus, the aim of this multidisciplinary research is to devise an artificial intelligence system capable of interpreting the analysis made by an e-nose and to present the results in an easily understandable way to human experts.

The rest of this paper is organized as follows: Section 2 describes the Growing Neural Gas base algorithm; while Section 3 and Section 4 describe the proposed modifications to the algorithm that is discussed on the present work. Section 5 briefly presents the way in which the test data was gathered. Section 6 includes descriptions of the tests and their results. Finally, Section 7 summarizes the final conclusions and future lines of work.

2 The Growing Neural Gas Algorithm

The Growing Neural Gas (GNG) algorithm is a clustering and classification algorithm proposed by Fritzke [5]. It is based on the Neural Gas (NG) algorithm previously proposed by Martinetz *et al.* [10] for finding optimal data representations based on feature vectors, which is in turn a modification of the widely known Self-Organizing Map by Kohonen [8]. The main characteristic of the NG algorithm is that instead of expanding through the data input space as a fixed grid of units, like the SOM algorithm; it allows the neighboring relationships of its units to change, expanding more like a gas over the data space.

The GNG method is different from the previous algorithms in that it is an incremental algorithm, so there is no need to determine *a priori* the number of nodes. Network shape and size are determined during the training, while the SOM and NG are often

 ISBN 13: 978-84-613-5510-5

trained on a fixed network size throughout.

The GNG is a combination of Fritzke's Growing Cell Structures (GCS) [6] and Martinetz's Competitive Hebbian learning (CHL) [9]. The network topology of the GNG is generated incrementally by the CHL algorithm, which successively inserts topological connections or edges. The main principle of the CHL is: for each input x connect the two closest centers (measured by Euclidean distance) by an edge.

Algorithm 1 Growing Neural Gas Algorithm

1: **procedure** GNG(set of samples $\in \mathbf{R}^n$)
2: Start with two units i and j at random positions in the input space.
3: Present an input vector $x \in \mathbf{R}^n$ from the input set or according to input distribution.
4: Find the nearest unit s_1 and the second nearest unit s_2.
5: Increment the age of all edges emanating from s_1.
6: Update the local error variable by calculating:

$$\triangle error(s_1) = \|w_{s_1} - x\|^2$$

7: Move s_1 and all its topological neighbors (i.e. all the nodes connected to s_1 by an edge) towards x by fractions of e_b and e_n of the distance:

$$w_{s_1} = e_b(x - w_{s_1})$$
$$w_n = e_n(x - w_n) \text{ for all direct neighbours of } s_1$$

8: If s_1 and s_2 are connected by an edge, set the age of the edge to 0 (refresh). If there is no such edge, create one.
9: Remove edges with their age larger than a_{max}. If this results in nodes having no emanating edges, remove them as well.
10: If the number of input vectors presented or generated so far is an integer or multiple of a parameter λ, insert a new node r as follows:

- Determine unit q with the largest error.
- Among the neighbors of q, find node f with the largest error.
- Insert a new node r halfway between q and f as follows: $w_r = \frac{w_q + w_f}{2}$
- Create edges between r and q, and r and f. Remove the edge between q and f.
- Decrease the error variable of q and f by multiplying them with a constant α. Set the error r with the new error variable of q.

11: Decrease all error variables of all nodes i by a factor β.
12: If the stopping criterion is not met, go back to step(2).
13: **end procedure**

 ISBN 13: 978-84-613-5510-5

3 The Semi-Supervised Learning Meta-Algorithm

As several recent studies point out [4, 3], the use of unlabeled data has proven to be a useful way of improving the classification of previously labeled data sets. This is an specially useful procedure when new information added to previous data sets is expensive or difficult to classify. In this case, due to the use to which the data will be put, which is the human consumption of the ham, the categorization always requires a human expert with a specific knowledge that is very difficult to formulate for classical approaches such as classification rules. This idea of the inclusion of unlabeled samples in the training is perfect for our purpose, as the final system will in all likelihood have been trained in a representative database of ham samples, but the addition of new samples obtained along its future use would be a desirable feature. Thus, characteristics of new unclassified samples to be tested may easily be incorporated into the network in this way.

This study will therefore make use of the semi-supervised algorithm proposed in [13]. Its implementation pseudo-code is detailed in Algorithm 2.

Algorithm 2 Semi-Supervised Training Algorithm

Input : L and U, let $L' \neq \{\phi\}$ represent an initial empty set of newly labeled data.
Output : Classifier Algorithm (K)

1: **procedure** SEMI-SUPERVISED LEARNING ($L = (x, y)\, U = x'$: $x, x' \in \mathbf{R}^n$,$y \in C$)
2: Present L to the classification algorithm K and train the classifier only with L.
3: **for all** input x_j from U **do**
4: Label x_j according to the class label of the winning node using classifier K.
5: Remove x_j from the current unlabeled data set, U.
6: Add x_j into the newly labeled data set L'.
7: **end for**
8: Present L and L' together to the classifier (K) and retrain with $L + L'$. Evaluate new classification performance.
9: Check the labels of U; if they become stable during successive iterations, stop. That is, if classification performance varies less than a threshold in two different iterations.
Otherwise go back to Step (2).
10: **end procedure**

This is a two-stage method where labeled data are firstly used to train a classifier and then unlabeled data are labeled according to the classifier trained with the original labeled data. The second stage involves classifying unlabeled data and re-training the classifier from the classified unlabeled data as well as the original labeled data. The two stages are iterated until the training process converges, in other words, until the training errors stabilize.

This procedure can be applied to many different classifiers. In this study will be applied to the GNG and its ensemble variation to improve its classification results.

 ISBN 13: 978-84-613-5510-5

4 Voronoi Polygons Similarity Ensemble Fusion

The ultimate goal of constructing an ensemble is to improve the performance obtained by a single working unit. With regard to classification, it is generally accepted that the sets of patterns misclassified by the different classifiers would not necessarily overlap. This suggests that different classifier designs have the potential offer complementary information about the patterns to be classified and could be harnessed to improve the performance of the selected classifier [12]. In the present study, the central idea is to verify the improvements that an ensemble technique can provide in the multi-dimensional data classification field over an unsupervised learning process such as Competitive Learning.

As the initial aim of the ensemble architecture was to improve supervised classification, few models attempt to deal with ensembles in the field of unsupervised learning, and very few try to deal with the topology preserving map family of algorithms [7, 2]. In this study, an algorithm for topology preserving networks summarization proposed by Saavedra *et al.* [11] is used as a potential means of improving the single model's performance.

This algorithm uses the Voronoi polygons related to the units of different networks as a means of comparing them and deciding on the structure for the final network. Each unit in a topology preserving map can be associated with a portion of the input data space called the Voronoi polygon [1]. That portion of the multi-dimensional input space, contains data for which that precise unit is the Best Matching Unit (BMU) of the whole network. Thus, a logical conclusion is to consider that the units related to similar Voronoi polygons can be considered similar between themselves, as they should be situated relatively close in the input data space. A record may be kept of which data entries activated each unit as the BMU, to calculate the dissimilarity between the Voronoi polygons of two units. This can easily be done by associating a binary vector with the unit, the length of which is the size of the data set and contains zeros in the positions where the unit was the BMU for that sample and zeros in all other positions. The dissimilarity (i.e. the distance) between units can therefore be calculated, as shown in Eq. 1:

$$ds(b_r, b_q) = \frac{\sum XOR(b_r, b_q)}{\sum OR(b_r, b_q)} \tag{1}$$

where, r and q are the units whose dissimilarity will be determined and b_r and b_q the binary vector relating each unit with the data sample that it recognizes.

The main issue with this proximity criterion is that it depends on data recognition of by the network. To avoid problems with "dead" units, all units with a reacting rate lower than a set threshold are removed before calculating the similarities between them.

Units that are sufficiently similar are grouped together, in order to form a single unit in the final map. Eq. 2 is used to determine the units that will be part of the same group -or cluster- .

$$\begin{cases} ds(b_r, b_q) < \theta_f & \forall r, q \in Ws_n \\ ds(b_r, b_q) > \theta_f & \forall r, q \notin Ws_n \end{cases} \tag{2}$$

 ISBN 13: 978-84-613-5510-5

where $ds(b_r, b_q)$ is the dissimilarity between units b_r and b_q, (see Eq. 1) and θ_f is the fusion threshold.

Having determined the units that will be fused together, the final unit will be obtained by calculating the centroid of the fused units (Eq. 3):

$$w_c = \frac{1}{|W_k|} \sum_{w_i \in W_k} w_i \tag{3}$$

Finally, the similarity criteria must be used again to keep a notion of neighboring relations between the units of the fused network. Units with dissimilarity below a certain threshold will be considered as neighbors in the fused network. Units with dissimilarity below a certain threshold among their initial composing units will be considered as neighbors in the fused network (Eq. 4):

$$\min_{b_r \in Ws_k, b_q \in Ws_l} ds(b_r, b_q) < \theta_c \tag{4}$$

where θ_c is the connection threshold of the algorithm.

A detailed description is shown in Algorithm 3.

This whole procedure implies that the final network will approximate very well to the data set, enhancing the vector quantization feature of the SOM. Its drawbacks are that the number of units in the final network is an unknown factor before its fusion (and will almost certainly differ from the size of the composing networks) and that the neighborhood relationships of the composing networks will be ignored in the final one, as the latter will create a new neighborhood for each unit based on its dissimilarity with the others.

5 A Food Industry Case of Study

Several Spanish hams of differing quality and various origins were used in this research. The data sets consisted of measurements taken from seven types of Spanish dry-cured ham from among the various brands available on the Spanish market. The samples also included some that were tainted and/or that had a rancid/acidic taste. The tainted samples were randomly selected from among all the different quality types and origins of hams. The commercial brands of the hams in the samples were not taken into account in this study. In this case the e-nose was used to measure the odor of the ham samples. The data was collected and treated by the different models, in order to find the simplest and most reliable method for testing and analyzing their olfactory properties.

5.1 E-Nose Odor Recognition

The odor recognition process may be summarized as follows:

1. The sample is heated for a given time to generate volatile compounds in the head-space of the vial containing the sample.

 ISBN 13: 978-84-613-5510-5

Algorithm 3 Network Fusion by Voronoi Polygon Similarity

Input: Set of trained topology-preserving maps: $M_1 ... M_n$,
usage threshold: θ_u , fusion threshold: θ_f, connection threshold: θ_c
Output: A final fused map: M_{fus}

Select a training set $S = \langle (x_1, y_1) \dots (x_m, y_m) \rangle$
train several networks by using the bagging (re-sampling with replacement) meta-algorithm : M_n
let θ_u, θ_f and θ_c be the usage, fusion and connection thresholds respectively
procedure FUSION($M_1 ... M_n$)
    for all $M_i \in M_n$ **do** ▷ for all networks in the ensemble
        for all $w_j \in W_i$ **do** ▷ for all units in each network
            ▷ accept units with a recognition rate higher than a given threshold
            $W_{fus} \leftarrow w_i \; if \; \sum_i b_r(i) > \theta_u$
        end for
    end for
    for all $w_i \in W_{fus}$ **do**
        calculate dissimilarity between w_i and ALL units in W_{fus} (Eq. 1)
        $D_i \leftarrow ds(w_i, w_k) \forall w_k \in W_{fus}$
    end for
    group into different sub-sets (Ws_n) the units that satisfy Eq. 2
    for all Ws_n **do**
        calculate the centroid (w_c) of the set.
        add the centroid to the set of nodes of the final network (W^*_{fus})
    end for
    for all $w_r \in W^*_{fus}$ **do** ▷ for all units in the fused network
        Connect w_r with any other unit in W^*_{fus} , if they satisfy Eq. 4
    end for
end procedure

 ISBN 13: 978-84-613-5510-5

2. The gas phase is transferred to a detection device which reacts to the presence of molecules.

3. The differences in sensor reactions are recorded using statistical calculation techniques to classify the odors.

The readings taken by each sensor are separated and stored in a simple database for further study. In this study, the analyses are performed using an E-Nose α-FOX 4000 (Alpha M.O.S., Toulouse, France) with a sensor array of 18 metal oxide sensors. The e-nose takes readings every 0.5 seconds, and has an acquisition time of 120 seconds and an acquisition delay of 600 seconds. Only the highest reading from each sensor is stored in the database for further analysis.

6 Experiments and Results

Experiments to test the different combinations of the techniques described in this work were performed. The combinations under comparison were: a simple GNG network trained in a single run, a simple GNG network trained using the semi-supervised technique in section 3, an ensemble of GNG networks trained as a single run and summarized as explained in section 4; and an ensemble of GNG networks that were trained using the semi-supervised algorithm. The procedure for this latter model was to train all networks individually with the single run procedure and then summarize them using the Voronoi Similarity algorithm and employ the semi-supervised algorithm over this last network.

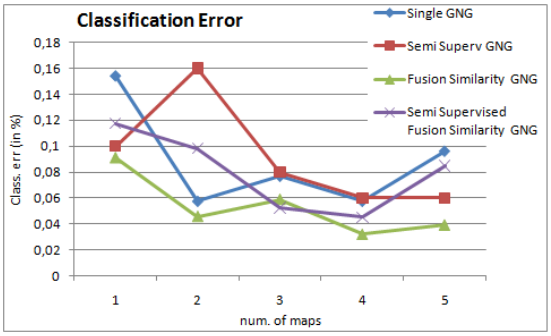
Regarding the use of the data set, two experiments were designed. The first one uses all the content of the data set and studies how the variations in the number of networks in the ensemble models affected the final performance of the model. The second one consisted of reducing the size of the data set from its original size to 1/5 of that size, in order to analyze how the reduction of data affected the performance of the different models.

All the results presented here were performed over a 5-fold cross-validation, which means that 1/5 of the data set was always reserved to obtain the performance measures, while the remaining 4/5 were used in the experiments. The size associated with each test in the second experiment is the total of training + validation data set. At all times, 3/4 of the training data set was used as the labeled part, while the remaining 1/4 of the training set was used as the unlabeled part in the semi-supervised models; while the single models were only trained with the labeled part.

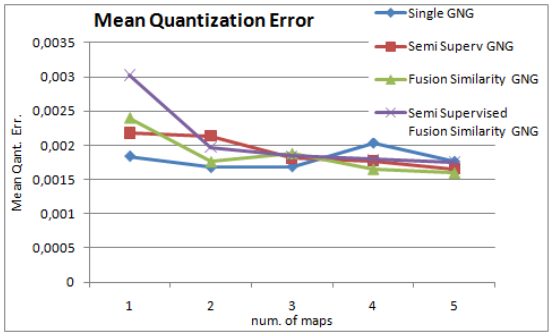
Fig. 1 and Fig. 2 respectively, show the results of the first and second experiment.

From the results of the first experiment, it is easy to conclude that Fusion by Similarity trained in the unsupervised way shows the best performance for classification tasks (Fig. 1a), repeatedly yielding the lowest error. It is only outperformed once by the Fusion by Similarity trained in a Semi-Supervised manner. As expected, the simple GNG is the most unstable of all, alternating low error values with much higher values even though the values represented are the mean of the cross-validation. This

 ISBN 13: 978-84-613-5510-5

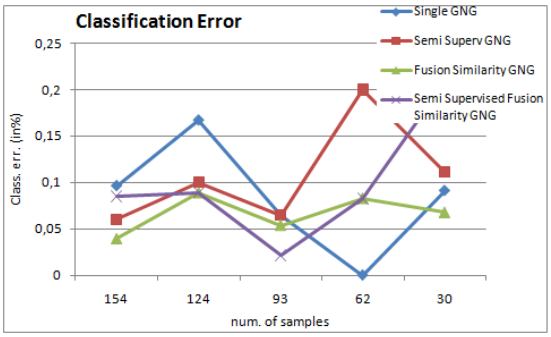


(a) Classification Error Measure (in %)

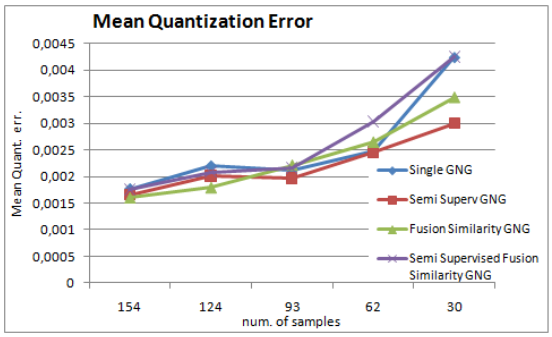


(b) Mean Quantization Error Measure

Figure 1: Experiment over complete data set, varying the number of networks.



(a) Classification Error Measure (in %)



(b) Mean Quantization Error Measures

Figure 2: Experiment using 5 networks in the ensembles, varying the number of data samples.

instability is somehow reduced by the use of the ensembles; the ensemble-based models showing less variability in their results in the different tests, with a more graduated behavior. Moreover as expected, classification error is reduced for the ensemble-based models when the number of networks increases. The more networks are included, the more the diversity of the ensemble increases, leading to better results.

The mean quantization error results for the four models under comparison (Fig. 1b) are very similar for all models. In general, they are all very low, but still the minimizing effect of the inclusion of more maps in the ensembles can be appreciated for the ensemble-based models.

In the second experiment (in Fig. 2), the added instability that arises from decreasing the number of samples is more than evident, as it leads to more inconsistent behaviour than in the first experiment. Nevertheless, the most consistent continues to be Unsupervised Fusion by Voronoi polygon Similarity. Even the Semi-Supervised fusion model, which appears comparable -or even better- in the range from 124 samples to 62 samples, becomes the worst model in the final step, with only 30 samples.

 ISBN 13: 978-84-613-5510-5

Also in this case, mean quantization error is a much less distinguishing characteristics with which to compare the models. All models tend to increase their errors when proportionally decreasing the number of samples.

It is interesting to note that the more complex models among those tested, Semi-supervised Fusion by Voronoi polygons Similarly, does not clearly outperform Unsupervised Fusion by Voronoi polygons Similarity. This situation probably points to some kind of over-fitting or interaction between the Semi-Supervised and Ensemble algorithms, one of which cancels out the improvement effect of the other.

7 Conclusions and Future Work

This research has presented a study of several solutions to the problem of ham sample quality and its automatic classification. The aim was to generate an algorithm capable of classifying new ham samples, so as to assess ham quality in an objective and simple way. Due to the specific characteristics of the problem, the algorithms are all based on competitive learning, a subset of the unsupervised learning. Two different potential ways of improving the classification have been tested in combination with the well-known Growing Neural Gas Algorithm: an ensemble fusion algorithm specifically designed for topology preserving networks and a semi-supervised algorithm that uses still unlabeled samples in the training. The most useful technique seems to be the ensemble algorithm. Although both are able to improve the performance of the original model, the ensemble is able to induce greater stability and the model therefore yields more reliable results.

A number of tests remain that are of interest as future work. For instance, observing the differences in performance between semi-supervised training before adding the networks to the ensemble; or training the formed ensemble in a semi-supervised way; or even experimenting with a wider range of base classifiers with those techniques to search for a more effective model. Moreover, the measures used in this work are two of the most immediate in terms of their calculation, although many others appearing in literature could provide additional information on how these models interact with the ensemble and semi-supervised algorithms. This should lead to more comprehensive studies of the different options with which to solve the presented problem.

Acknowledgments

This research has been partially supported through projects CIT-020000-2008-2 and CIT-020000-2009-12 of the Spanish Ministry of Education and Innovation and BU006A08 of the Junta of Castilla and León. The authors would also like to thank the manufacturer of components for vehicle interiors, Grupo Antolín Ingeniería, S.A. in the framework of the project MAGNO 2008-1028 CENIT Project funded by the Spanish Ministry of Science and Innovation.

References

[1] F. Aurenhammer and R. Klein. Voronoi diagrams. In *Handbook of Computational Geometry*, number 5, pages 201–290. North-Holland, Amsterdam, Netherlands, 2000.

[2] Bruno Baruque and Emilio Corchado. A weighted voting summarization of SOM ensembles. *Data Mining and Knowledge Discovery*, on-line, January 2010.

[3] A. Bouchachia. Learning with partly labeled data. *Neural Computing Applications*, 16:267–293, 2006.

[4] R. Dara, S. C. Kremer, and D. A. Stacey. Clustering unlabelled data with SOMs improves classification of labelled real-world data. In *Proc. IEEE World Congress on Computational Intelligence*, pages 2237–2242, May 2002.

[5] B. Fritzke. Unsupervised clustering with growing cell structures. In *IJCNN'91 - Seattle International Joint Conference on Neural Networks*, volume 2, pages 531–536, Jul 1991.

[6] B. Fritzke. Growing cell structures - a self-organising network for unsupervised and supervised learning. *Neural Networks*, 7:1441–1460, 1994.

[7] Apostolos Georgakis, Haibo Li, and Michaela Gordan. An ensemble of SOM networks for document organization and retrieval. In *International Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, pages 6–141, 2005.

[8] Teuvo Kohonen. *Self-Organizing Maps*, volume 30. Springer, Berlin, Germany, 1995.

[9] T. Martinetz. Competitive hebbian learning rule forms perfectly topology preserving maps. In *Proceeding of the Int. Conf. on Artificial Neural Networks (ICANN 93)*, pages 426–438, 1993.

[10] T. M. Martinetz and K. J. Schulten. A neural-gas network learns topologies. *Artificial Neural Networks*, pages 397–402, 1991.

[11] Carolina Saavedra, Rodrigo Salas, Sebastián Moreno, and Héctor Allende. Fusion of self organizing maps. In *9th International Work-Conference on Artificial Neural Networks (IWANN 2007)*, pages 227–234, 2007.

[12] A.J.C. Sharkey and N.E. Sharkey. Combining diverse neural nets. *Knowledge Engineering Review*, 12, 3:1–17, 1997.

[13] Shireen M. Zaki and Hujun Yin. A semi-supervised learning algorithm for growing neural gas in face recognition. *Journal of Mathematical Modelling and Algorithms*, 7:425–435, 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Power saving-aware solution for SSD-based systems

Laura Prada[1], Jose Daniel Garcia[1], Jesus Carretero[1] and Javier Garcia[1]

[1] *Computer Science Department, Universidad Carlos III de Madrid, Leganes, Madrid, Spain*

emails: laura.prada@uc3m.es, josedaniel.garcia@uc3m.es, jesus.carretero@uc3m.es, fjblas@inf.uc3m.es

Abstract

This article considers the problem of saving energy in the disk drive taking advantage of SSD drives. SSD and disk devices offer different power characteristics, being SSD drives much less power consuming than conventional disk drives. We propose the design and implementation of a sequential prefetching algorithm that uses a SSD device as a cache for a single disk device. We implemented a simulator composed of disk and SSD devices. We evaluated the proposed approach with the help of realistic workloads.

Key words: Power saving, Solid State Disk, Write-Buffering, Prefetching

1 Introduction

Saving energy in the computing system context has become lately an important and worrying need. Energy has turned into an augmenting demand in many systems, specially in data centers and supercomputers. As stated by the Green500 [1] list, which provides a ranking of the most energy-efficient supercomputers in the world, even in the supercomputers scope, energy is as relevant as performance. Moreover, the performance-per-watt is established as a new metric to compare supercomputers.

In current computers, disk drives are one of the most power consuming elements, and it is important to notice that they contribute about 86% of the energy consumption in some systems [2]. In order to reduce disk consumption, numerous disks provide energy efficient transition modes. However, disk idle periods have to be large enough to excuse both performance and power overhead, due to the spinning up/down disk scheme. The desirable goal is to increase disk inter-access times by means of gathering accesses to disks in fragments with expanded idle times in the middle.

SSDs (Solid State Disks) show contrasting features with disk drives. On the one hand, random-access writes in SSD devices can take longer than magnetic disk drives

 ISBN 13: 978-84-613-5510-5

while reads perform faster. On the other hand, SSDs have lower power consumption ratios. This characteristic makes them optimal to behave as big caches for saving energy in disk drives. For example, if SSD devices are treated as write-back logs, every disk write request will be absorbed by the SSD device, and only when that SSD is full, the write requests will be flushed to the disk in background. This mechanism clearly will increase disk idleness.

Prefetching has been identified as a useful method for augmenting burstiness in disk drive accesses [3]. When a SSD miss occurs due to a disk read access, and those data are not stored in the SSDs, data can be pre-loaded in advance in the SSD devices. Thus, future disk accesses can be concentrated in the same sequence of I/O fragments, making idle times bigger.

In this paper we explore the usage of prefetching between SSD and disk drives to save energy. We propose a novel power saving solution based on SSD devices, namely, SBPS (SSD-Based Power Saving). SBPS includes two power saving mechanisms, write-buffering and prefetching. We make the following contributions: we present a prefetching method to provide burstiness in read accesses;

an evaluation of saving gains, that may be strongly dependent on the workload behavior perceived at the storage system; the conviction that significant energy saving in a hybrid system is possible, as stated through a validation of results from a simulation-based prototype.

2 Preliminaries

In this section we present four realistic workloads which are common in data-intensive I/O systems. Financial1 is the I/O core of an OLTP application gathered at a huge financial organization. It performs about 5 million requests over 24 disks. WebSearch2 is a famous search engine server [4], which executes about 4 million read requests over 6 disks. Openmail6 is an OpenMail e-mail server at the HP Labs [5], and runs about 1 million requests over 21 disks. Cello99 [5] is a shared compute/mail server from HP Labs. It performs about 6 million requests over 25 disks.

Figure 1 illustrates the read access patterns for Financial1 and Cello99 at disk 1, and for WebSearch2 and Openmail6 at disk 0. We initially measured the sequentiality ratio as the percentage of read requests that are sequential in a certain period. The Financial1 is highly sequential at disk1, i.e, in a 79% of the total read requests, the last block requested corresponds with the first block of the following request. In WebSearch2 and Openmail6, sequentiality is 38% and 15%, respectively. For Cello99 sequentiality turns out to 5%.

It is remarkable that in WebSearch2 and Openmail6 the range of accesed LBAs is highly concentrated in certain bands, and this access pattern remains stable during the execution. If we could detect those specific bands, and from that point only read blocks in that range, we would save space in SSD device to host the prefetched blocks that actually will be accessed.

 ISBN 13: 978-84-613-5510-5

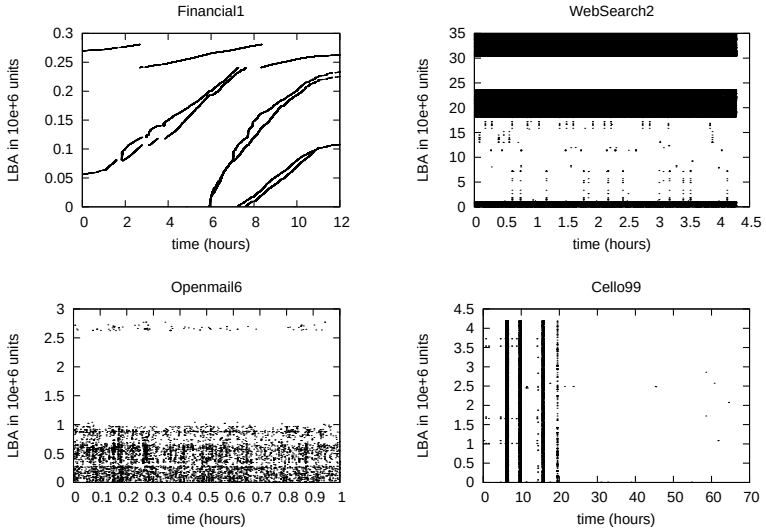


Figure 1: Read access patterns

3 SSD-Based Power Saving Approach

The SSD-based power saving approach lies on an hybrid system, consisting of SSD and conventional disk devices. Our general procedure aims to use the SSD device as a block cache for the disk drive. Both devices appear like an unique one to file systems, i.e., we handle the diverse devices under the file system transparently. SSD drives perform faster random-access reads and sequential-access writes than disk drives [6]. Moreover, the number of erases is dramatically reduced because applications do not write many times in the same page of a SSD device. This results in less invalid pages in blocks and less overhead.

For write operations, the following steps are performed. Write requests are first committed to the SSD device. The SSD drive is treated in a log-manner, writing requests sequentially. This mechanism takes advantage of the asymmetric performance of SSD devices. When the SSD device is full, dirty blocks are committed to the disk. Meanwhile, subsequent writes can be sent to the disk device. This approach can postpone many write requests in order to write all of them in a clustered manner (write-buffering). This increases idle intervals in the disk drive, allowing it more opportunities to spin-down, and consequently, save energy.

Read request are sent to the current SSD device if the indirection map indicates it. This reading process presents a new problem to handle. Whenever the requested block is in the SSD drive, it can be served by it. When a read miss SSD occurs, that request is sent to the disk drive. That read miss implies breaking a possible long idle interval, and consequently, waking up the disk drive. In order to avoid that situation as possible, whenever a read miss occurs, we would need to bring from disk to the SSD the following read blocks to be requested in the next future, namely, we need to prefetch

 ISBN 13: 978-84-613-5510-5

the following read blocks to be accessed.

Our solution lies in an indirection map, which permits blocks to be allocated in different devices, mapping SSD and disk logical bock addresses (LBAs). Every read and write request is sent to the corresponding device after mapping the indirection using the actual physical location. Data consistency is enforced in the system by not allowing more than one copy of a file block at SSD. Our approach is based on a one copy consistency model. Either the data is in one device or in both, there is a need to have the most accurate information in the remapping of the block addresses.

4 Power-saving prefetching method

The main goal of conventional prefetching methods is to reduce request service times, improving I/O accesses in terms of performance. However, power-saving aware prefetching focus on reading in advance an enough number of blocks in order to avoid break long idle intervals. If a workload is sequential enough, reading successive blocks into the SSD device using a single I/O operation, can considerably satisfy almost the totality of read requests during a long idle period of time, and take advantage of shorter service demands that comes from the not extreme head displacements at disk drives.

We propose a sequential prefetching method as an energy-efficient strategy for SSD-based hybrid system. This method consists in read in advance consecutive blocks when a read miss occurs. We employ this approach by using adaptive window sizes. Depending on the workload access pattern, the algorithm chooses a compromise window size in order to get the highest possible hit ratio.

Our prefetching algorithm calculates dynamically the maximum sequential data flow with the arrived requests, and uses it as the window size in logical blocks. So we constantly derive spatially adjacent requests from the data flow, targeting each disk. Multiple simultaneous data flows or strided data flow make difficult the derivation and modeling of sequentiality. We identify merged strided data flows by comparing each request block not only against the nearly previous request block and length, but also to the n previous requests. We introduce a history-based log of the last n I/O accesses. This can be considered of as augmenting the read-ahead window for this calculation from 1 to n. For every I/O access, if necessary, we linearly look through this record and find out if we have a higher sequentiality.

The calculation of the degree of sequential read-ahead is described in Algorithm 1. Every time a $reqBlock$ is requested, the distance ($size$) with respect to the previous block accessed ($prevBlock$) is determined (line 4). If blocks are not sequential, the system provides a second chance to find sequentiality. Lines 6-13 deal with the case of comparing $reqBlock$ with the previous $windowSize$ requests hosted in the $separation$ history list. If $size$ results 1 in this case, last accessed block in the current request ($currBlock$) is saved into the $separation$ history list, and the least recently added block is removed, if the list is full (lines 14-20). Lines 21-25 record $currBlock$ as the last block accessed in its data flow and the length of the data flow is compared with the length of the longest data flow ($maxSeq$) and updated if applicable. Line 27 records the

 ISBN 13: 978-84-613-5510-5

Algorithm 1 Calculation of the degree of sequential read-ahead

```
Block reqBlock is requested:
1: if time = timeThreshold then
2:     bandDetection()
3: end if
4: size ⇐ reqBlock - prevBlock
5: currBlock ⇐ reqBlock + (rwSize)/blockSize − 1
6: if size ≠ 1 then
7:     for i ⇐ separation.begin() to separation.end() do
8:         if reqBlock - i = 1 then
9:             prevBlock ⇐ i
10:            size ⇐ 1
11:        end if
12:    end for
13: end if
14: if size = 1 then
15:    separation.add(currBlock)
16:    if separation.size() > windowSize then
17:        separation.remove(min(separation))
18:    end if
19:    dataflows[currBlock] ⇐ dataflows[prevBlock]
20:    dataflows.erase(prevBlock)
21:    separ ⇐ currBlock - dataflows[currBlock]
22:    if maxSeq < separ then
23:        maxSeq ⇐ separ
24:    end if
25: end if
26: prevBlock ⇐ currBlock
```

last accessed block in the current request ($currBlock$) as the previous accessed block ($prevBlock$) for the next request.

When a certain $timeThreshold$ has elapsed, the $bandDetection$ method is activated (lines 1-3). This mechanism, described in Algorithm 2, scans every LBA that belongs to a disk block which is inside the SSD device (lines 3-16). A detection of a new band takes place when the difference with respect to the minimum LBA that belongs to the current band (min) is greater than a maximun established band size ($bandSize$), or a large gap (bigger than $maxGap$) exists between the current LBA and the previous one ($prevLBA$), and the number of blocks inside the band exceed a certain threshold ($LBAsThreshold$).

Algorithm 2 Band detection method

```
bandDetection()
1: min ⇐ CACHE.begin() , max ⇐ CACHE.end()
2: prevLBA ⇐ min
3: for i ⇐ CACHE.begin() to CACHE.end() do
4:     difference ⇐ i − min
5:     if (difference < bandSize)&(i − prevLBA < maxGap) then
6:         max ⇐ i , LBAs + +
7:     else
8:         if LBAs ≥ LBAsThreshold then
9:             band[j].min ⇐ min, band[j].max ⇐ max
10:            j + +
11:        end if
12:        min ⇐ i , max ⇐ i
13:        LBAs ⇐ 0
14:    end if
15:    prevLBA ⇐ i
16: end for
```

Only when a read miss occurs and the distance with respect to the previous accessed request ($size$) turns out to be 1, the system prefetches a total of $seqMax$ blocks (shown in Algorithm 3). If any band detection has been taken place, only blocks inside the detected bands are fetched into the SSD (lines 9-10 and 15-16).

 ISBN 13: 978-84-613-5510-5

Algorithm 3 Algorithm for sequential prefetching

Block $reqBlock$ is requested:

```
1: miss ⇐ 0
2: for i ⇐ reqBlock to (reqBlock + rwSize/blockSize − 1) do
3:     if i ∉ CACHE then
4:         miss ⇐ 1
5:     end if
6: end for
7: if miss = 1 AND size = 1 then
8:     for i ⇐ reqBlock to (reqBlock + rwSize/blockSize − 1 + maxSeq) do
9:         if bandInside(i) then
10:            putInCache(i)
11:        end if
12:    end for
13: else
14:    for i ⇐ reqBlock to (reqBlock + rwSize/blockSize − 1) do
15:        if bandInside(i) then
16:            putInCache(i)
17:        end if
18:    end for
19: end if
```

5 Evaluation

In order to evaluate our solution, we have implemented an hybrid architecture simulator in a widely used general purpose simulation platform, namely OMNeT++ [7]. We compare two scenarios, a baseline variant which does not employ any power saving scheme, and SSD-Based Power Saving (SBPS), our proposed energy saving solution. We use the well-know workloads presented in Section 3, which represent different scenarios.

5.1 Hybrid Architecture Simulator

The implemented simulator has three main modules. The first module represents the disk device. This module is directly connected to an instance of the Disksim simulator[8]. The second module represents a SSD device. This module is also connected to DiskSim Simulation Environment [9] through an instance of the SSD extension. We have employed a third module which symbolizes an indirection map, containing mappings between disk and SSD physical addresses. This last module maps a specific block in the disk, which has been cached in the SSD previously.

Table 5.1 resumes the main features of both simulated disk and SSD devices. We have included a power model in both disk and SSD modules. The disk power model employs an extension of the 2-Parameter Model described in [10]. This model applies the next formula to calculate an energy consumption estimation of each disk:

$$E_{disk} = E_{activeDisk} + E_{idleDisk} + E_{standbyDisk} \quad (1)$$

Where $E_{activeDisk}$ is calculated as $P_{activeDisk} x T_{activeDisk}$, $P_{activeDisk}$ is the power consumption when the disk is active, and $T_{activeDisk}$ is the time spent by the disk while satisfying disk requests. $E_{idleDisk}$ is calculated as $P_{idleDisk} x T_{idleDisk}$, $P_{idleDisk}$ is the power consumption in the idle mode, and $T_{idleDisk}$ is the length of the idle period. $E_{standbyDisk}$ is calculated as $P_{standbyDisk} x T_{standbyDisk}$, $P_{standbyDisk}$ is the power consumption in the standby mode and $T_{standbyDisk}$ is the length of the standby period.

 ISBN 13: 978-84-613-5510-5

Specification	Value	
Model	Seagate Cheetah 15K.4	Generic
Block Erase Latency	-	1.5 ms
Power consumption (idle)	12 W	0.1 W
Power consumption (active)	17 W	0.5 W
Power consumption (standby)	2.6 W	-

Table 1: Simulated disk and SSD specifications. The minus character means that specification is not applicable.

For the SSD power model we used the following formula to calculate an energy consumption estimation for a SSD:

$$E_{SSD} = E_{activeSSD} + E_{idleSDD} \tag{2}$$

Where E_{active} is calculated as $P_{activeSSD}xT_{actoveSSD}$, $P_{activeSSD}$ is the power consumption when the SSD is active, and $T_{activeSSD}$ is the time spend by the SSD while satisfying SSD requests. $E_{idleSSD}$ is calculated as $P_{idleSDD}xT_{idleSSD}$, $P_{idleSSD}$ is the power consumption in the idle mode, and $T_{idleSDD}$ is the length of the idle period.

We employed a third module which provides an indirection map, containing mappings between disk physical addresses and SSD physical addresses. Note that this last module knows how to map into the disk a block which is cached in the SSD, and vice-versa.

5.2 Financial1

Figure 2 shows misses at disk 0 for the accessed LBAs in the course of the Financial1 workload. The left side of the figure depicts the resultant misses without applying any prefetching scheme. In right side our prefetching algorithm has been applied and that is why density of misses is smaller. Figure 3 shows the reads and writes distribution in disk 0 for LBA accesses. In the left side we plot the clustered-in-time distribution of reads, and in right side we plot the writes distribution along the workload. Financial1 achieves nearly 51% energy savings. They are mainly because of the high percentage of writes which were redirected to some of the disks, as shown in Figure 3. Having a lot of writes makes easy to save energy because they all are redirected to the SSDs, and the disks only wake up when a few clustered-in-time read requests are not found in the SSDs. On the other hand, benefits were also obtained because of the highly sequential read access patterns at some of the disks and the effective work of our prefetching algorithm to detect them, as shown in Figure 2.

5.3 WebSearch2

Figure 4 depicts misses at disk 0 for the accessed LBAs in the course of the WebSearch2 workload. The left side of the figure shows the resultant misses without applying any

 ISBN 13: 978-84-613-5510-5

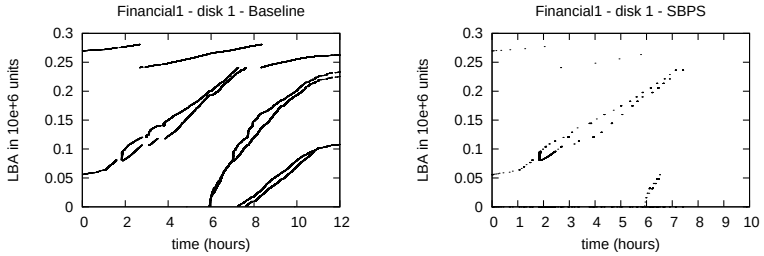


Figure 2: Misses at disk 0 for the accessed LBAs in the course of the Financial1 workload.

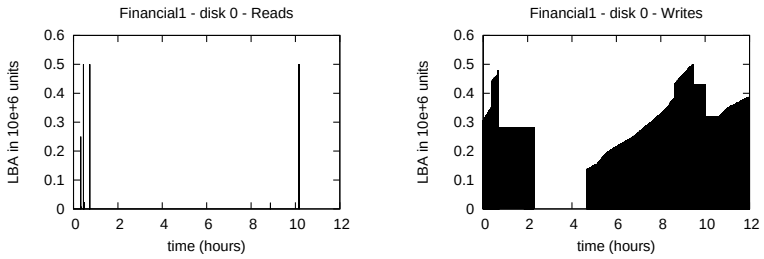


Figure 3: Distribution of read and write requests in the course of the Financial1 workload.

prefetching scheme. In right side our prefetching algorithm has been applied and that is why density of misses is smaller. In comparison, WebSearch, which is only performs read requests, achieves nearly 2% energy savings, and they are due to the effective work of our prefetching algorithm and its band detection method. Figure 4 demonstrates that along time, hits in SSD cache are further increased in disk 0. The reason is that the majority of the requests are concentrated in three clearly defined bands, and our prefetching algorithm is able to detect them.

5.4 Openmail6

Figure 5 depicts the time period distribution at disk 0 along the Openmail workload duration. The left side of the figure shows that the majority of the time the disk is idle, but idle periods are not long enough to put the disk in a standby mode. In right side it can be seen that the use of the SSD and our prefetching algorithm help increasing idle times, but again, they keep being short to put the disk in a lower power mode.

Openmail does not save energy for the majority of the disks. We observed that inter-arrival times between requests were not very long and the brief duration of the trace hardly let us appreciate the effectiveness of our prefetching algorithm. So in this specific case the shorter inter-arrival times stopped many disks from spinning down and consequently, from saving enough energy to overcome the energy wasted by using SSD devices, as shown in Figure 5. One of the advantages of avoiding saving energy in

 ISBN 13: 978-84-613-5510-5

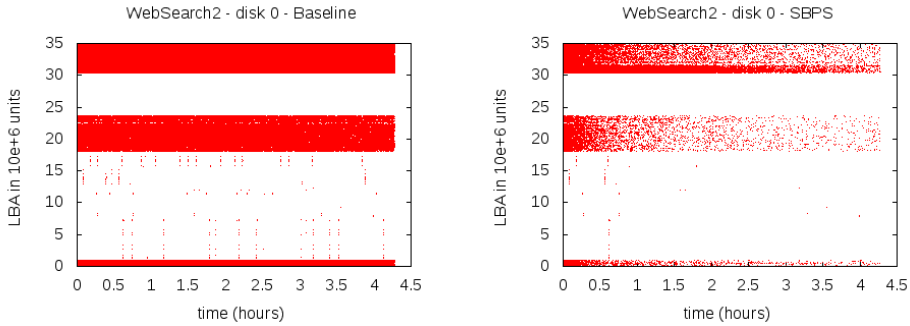


Figure 4: Misses at disk 0 for the accessed LBAs in the course of the WebSearch2 workload.

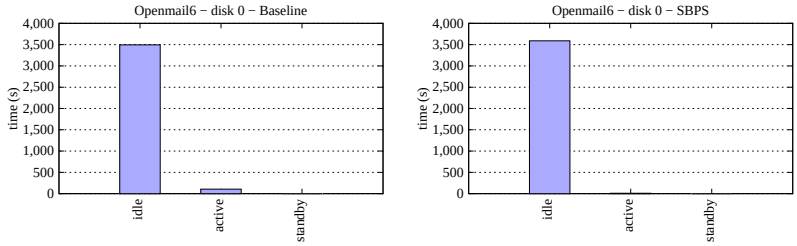


Figure 5: Distribution of the time periods in disk 0 along the Openmail6 workload duration.

this case was that the average response times were improved and this denotes another benefit of using our prefetching algorithm.

5.5 Cello99

Cello99 offers a limit save energy improvement for similar reasons to Openmail, but in this case, the execution time of the application is longer. Cello99 has long periods of inactivity combined with very bursty periods of activity. During the periods of activity the gain obtained by our prefetching algorithm is outshone by the effect of the short inter-arrival times. Moreover, during the long periods of inactivity the use of the SSD devices are counterproductive.

Figure 6 resumes the energy consumption of our approach and the baseline approach under the different workloads. Our approach saves energy in Financial and WebSearch workloads. Financial workload brings about 51% energy savings. They are mainly because of the highly sequential access patterns from the read requests of disks.

 ISBN 13: 978-84-613-5510-5

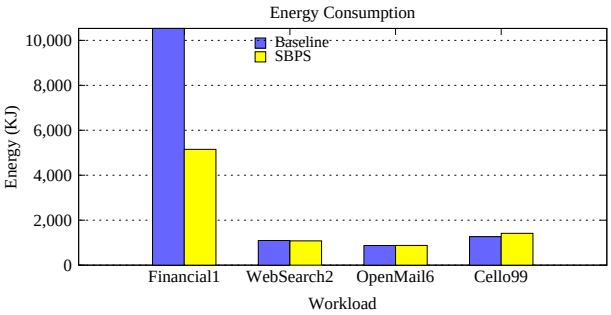


Figure 6: Energy consumptions with both approaches.

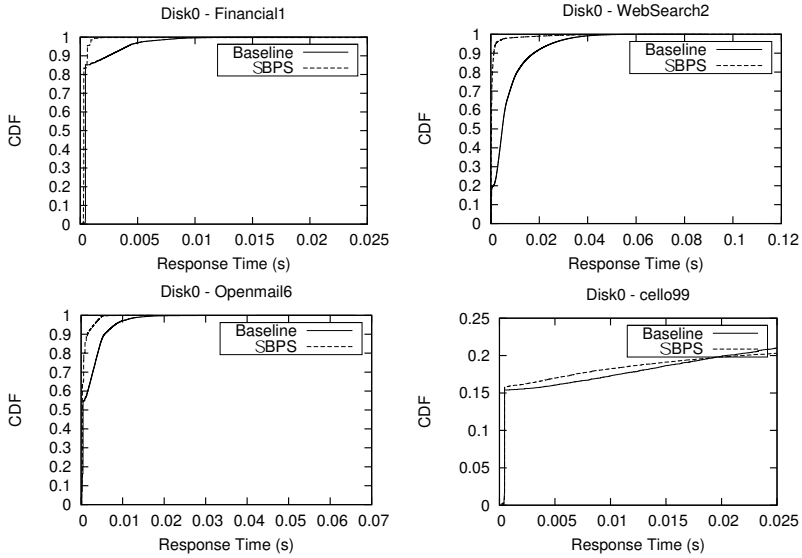


Figure 7: CDF (cumulative distribution function) curves for response times in all workloads at disk 0.

5.6 Performance

Figure 7 plots the cumulative distribution function curves (CDF) for response times in all workloads at disk 0. A particular (x, y) point on a function specifies that $y * 100\%$ of the response times are lower or equal to x (seconds). For a certain value of x in every workload, SBPS approach presents greater percentages in which response times are lower or equal to x than in the baseline approach. That is because we write on the SSD in a log-manner and this results in more performance gains. That is also because of the read hits on the SSD, which provides better performance gains. For a small fraction of the requests, we saw that in some cases we got service times of over 15 seconds. Those requests are the ones which have to wait for a disk to spin-up.

 ISBN 13: 978-84-613-5510-5

6 Related Work

Some approaches use multi-speed disks [11, 12], which are able to change their speed while spinning. However those disks are still inaccessible in the typical market. Other approaches [13, 14] commit data across drives, i.e., move blocks from most idle disks to less idle disks, such that the inactive disks can be put in low power mode longer. Moreover, big caches [15] at data centers are proposed to create greater inactive periods in some disks. Saving power at file system level is also exploited in [16, 17].

Power saving-aware prefeching and caching techniques aims to maximize disks idle times by using caching techniques [3]. The idea is to keep the disk in a low power mode as long as possible, taking advantage of in-memory cache. Our proposal differs from use main memory to cache data in two ways. First, memory is smaller than SSD devices, and second, RAM memory is volatile and SSD is persistent.

Other research deal with data migration across drives [13, 14]. Some techniques create enough inactivity in a number of disks by migrating their data to another disks. Some of these approaches just only manage write requests to save power. On the contrary, our method manages both write and read accesses. Second, their basic idea of data migration is to commit blocks from most idle disks to less idle disks, such that the inactive disks can be put in low power mode longer. Our method contrasts with these solutions in that these solutions do not consider individual disks to save power.

7 Conclusions

In this paper we have studied a power saving aware solution useful for hybrid storage systems based on SSD and disk drives. We have proposed a caching strategy for saving energy in this systems by exploiting the write/read performance asymmetry of these devices. Our approach writes/reads as much as possible from the SSD devices, letting the disks go to sleep for longer periods. We showed that the proposed approach is effective, based on realistic experiments on a simulation environment. Our study also provided a number of insights into a prefetching approach to increase the read hit ratios in the SSD devices in such a hybrid system. Our results showed that the write/read characteristics of the workload have a critical impact in the energy savings of such a hybrid storage system. In our approach, write requests from a single disk drive are written, as it is possible, in the SSD device in a log-based manner. Similarly, read requests are prefetched in the SSD drive.

References

[1] "The Green 500," 2010, http://www.green500.org.

[2] "Symmetrix 3000 and 5000 Enterprise Storage Systems product description guide," 1999, http://www.emc.com/.

[3] A. E. Papathanasiou and M. L. Scott, "Energy efficient prefetching and caching," in *In Proc. of the USENIX 2004 Annual Technical Conference*, 2004, pp. 255–268.

 ISBN 13: 978-84-613-5510-5

[4] "UMass Trace Repository," 2010, http://traces.cs.umass.edu.

[5] "The Cello99 and Openmail Traces," 2009, http://tesla.hpl.hp.com/private software.

[6] "Flash disk opportunity for server applications," 2008, http://queue.acm.org/detail.cfm?id=1413261.

[7] "OMNeT++ Community Site," 2010, http://www.omnetpp.org.

[8] G. R. Ganger, B. L. Worthington, B. L. Worthington, Y. N. Patt, and Y. N. Patt, "The disksim simulation environment version 2.0 reference manual," Tech. Rep., 1999.

[9] "SSD Extension for DiskSim Simulation Environment," 2010, http://research.microsoft.com/en-us/downloads/b41019e2-1d2b-44d8-b512-ba35ab814cd4/.

[10] J. Zedlewski, S. Sobti, N. Garg, F. Zheng, A. Krishnamurthy, and R. Wang, "Modeling hard-disk power consumption," in *FAST '03: Proceedings of the 2nd USENIX Conference on File and Storage Technologies*. Berkeley, CA, USA: USENIX Association, 2003, pp. 217–230.

[11] E. V. Carrera, E. Pinheiro, and R. Bianchini, "Conserving disk energy in network servers," in *In Proceedings of the 17th International Conference on Supercomputing*, 2003, pp. 86–97.

[12] S. Gurumurthi, A. Sivasubramaniam, M. T. Kandemir, and H. Franke, "Drpm: Dynamic speed control for power mangagement in server class disks," in *ISCA*, 2003, pp. 169–179.

[13] D. Colarelli and D. Grunwald, "Massive arrays of idle disks for storage archives," in *Supercomputing '02: Proceedings of the 2002 ACM/IEEE conference on Supercomputing*. Los Alamitos, CA, USA: IEEE Computer Society Press, 2002, pp. 1–11.

[14] D. Narayanan, A. Donnelly, and A. I. T. Rowstron, "Write off-loading: Practical power management for enterprise storage," in *FAST*, 2008, pp. 253–267.

[15] Q. Zhu, A. Shankar, and Y. Zhou, "Pb-lru: A self-tuning power aware storage cache replacement algorithm for conserving disk energy," in *In Proceedings of the 18th International Conference on Supercomputing*. ACM Press, 2004, pp. 79–88.

[16] E. B. Nightingale and J. Flinn, "Energy-efficiency and storage flexibility in the blue file system," in *OSDI'04: Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation*. Berkeley, CA, USA: USENIX Association, 2004, pp. 25–25.

[17] X. R., W. A., K. G., R. P., and P. G., "Conquest: Combining battery-backed ram and threshold-based storage scheme to conserve power," *19th Symposium on Operating Systems Principles (SOSP)*, 2003.

[18] L. Useche, J. Guerra, M. Bhadkamkar, M. Alarcon, and R. Rangaswami, "Exces: External caching in energy saving storage systems," *High Performance Computer Architecture, 2008. HPCA 2008. IEEE 14th International Symposium on*, Feb. 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Error analysis on the implementation of implicit Falkner methods for the special second-order I.V.P.

Higinio Ramos[1] **and Cesáreo Lorenzo**[1]

[1] *Department of Applied Mathematics, University of Salamanca*

emails: higra@usal.es, cesareo@usal.es

Abstract

Recently we have made an analysis of the implementation of explicit Falkner methods for solving special second-order initial-value problems [1], showing that the so-called unusual implementation increases the order of the method. In this paper we made an analysis of the accumulated errors considering different implementations of the implicit Falkner methods in predictor-corrector mode. A numerical example is included to show the performance of the different implementations. The numerical results agree with the theoretical ones, concluding that the best performance occurs when the predicted value is directly used in the implicit formulas to obtain respectively the approximations for the solution and the derivative.

Key words: error analysis, implicit Falkner methods, special second-order initial-value problems
MSC 2000: 65L70, 65L06

1 Introduction

Second-order differential equations appear frequently in applied sciences. Examples of that are the mass movement under the action of a force, problems of orbital dynamics, or in general, any problem involving Newton's law.

Among the general procedures for direct integration of the so-called *special second-order* initial value problem (I.V.P.)

$$y''(x) = f(x, y(x)), \quad y(x_0) = y_0, \quad y'(x_0) = y'_0\,, \tag{1}$$

the Falkner methods [5] is a well-known class of schemes of this type.

Although it is possible to integrate a second-order I.V.P. by reducing it to a first-order system and applying one of the methods available for such systems, it seems more natural to provide numerical methods in order to integrate the problem directly. The advantage of this procedure lies in the fact that they are able to exploit special

 ISBN 13: 978-84-613-5510-5

information about ODES, and this results in an increase in efficiency. For instance, it is well-known that Runge-Kutta-Nyström methods for (1) involve a real improvement as compared to standard Runge-Kutta methods for a given number of stages ([7], p. 285), although the computational cost remains high because of the number of function evaluations. On the other hand, a linear k-step method for first-order ODEs becomes a $2k$-step method for (1), ([7], p. 461), increasing the computational work.

Falkner methods can be written in the form [4]

$$y_{n+1} = y_n + h\, y'_n + h^2 \sum_{j=0}^{k-1} \beta_j \bigtriangledown^j f_n \,, \tag{2}$$

$$y'_{n+1} = y'_n + h \sum_{j=0}^{k-1} \gamma_j \bigtriangledown^j f_n \,, \tag{3}$$

where h is the stepsize, y_n and y'_n are approximations to the values of the solution and its derivative at $x_n = x_0 + n\,h$, $f_n = f(x_n, y_n)$ and $\bigtriangledown^j f_n$ is the standard notation for the backward differences. The coefficients β_j and γ_j can be obtained using the generating functions

$$G_\beta(t) = \sum_{j=0}^{\infty} \beta_j\, t^j = \frac{t + (1-t)\, Log(1-t)}{(1-t) Log^2(1-t)} \,,$$

$$G_\gamma(t) = \sum_{j=0}^{\infty} \gamma_j\, t^j = \frac{-t}{(1-t) Log(1-t)} \,.$$

The implicit Falkner formulas read [4]

$$y_{n+1} = y_n + h\, y'_n + h^2 \sum_{j=0}^{k} \beta^*_j \bigtriangledown^j f_{n+1} \,, \tag{4}$$

$$y'_{n+1} = y'_n + h \sum_{j=0}^{k} \gamma^*_j \bigtriangledown^j f_{n+1} \,, \tag{5}$$

with generating functions for the coefficients given by

$$G_{\beta^*}(t) = \sum_{j=0}^{\infty} \beta^*_j\, t^j = \frac{t + (1-t)\, Log(1-t)}{Log^2(1-t)} \,,$$

$$G_{\gamma^*}(t) = \sum_{j=0}^{\infty} \gamma^*_j\, t^j = \frac{-t}{Log(1-t)} \,.$$

Note that the formulas in (3) and (5) are respectively the Adams-Bashforth and Adams-Moulton schemes for the problem $(y')' = f(x, y)$, which are used to follow the values of the derivative. All the above formulas are of multistep type, specifically k-step formulas, and so k initial values must be provided in order to proceed with the methods. In this

 ISBN 13: 978-84-613-5510-5

paper, a rigorous analysis of the errors involved on the implementation of the implicit procedures in predictor-corrector (P-C) mode is made.

In the following section different implementations of the explicit and implicit Falkner methods are presented. Section 3 is devoted to the analysis of the local truncation errors and accumulated truncation errors. In Section 4 an example is presented to show the performance of the different implementations. In the final section some conclusions put an end to the article.

2 Implementations of Falkner methods

In the application of P-C modes, P indicates the application of the explicit method, in our case the predictor given by the first of the formulas in (2), and C indicates the application of the implicit method, that is, the corrector given by the first of the formulas in (4). For the derivative, we will use P' to indicate the application of the explicit second formula in (2), and C' to indicate the application of the implicit second formula in (4). And finally, E refers to the evaluation of the function f. Two different implementations for the explicit method were considered in [1], and for the implicit method three implementations will be considered here. They are summarized in what follows.

2.1 Explicit methods

2.1.1 PP'E mode

The usual implementation of the explicit Falkner method on each step for solving the problem in (1) is

1. Evaluate y_{n+1} using the formula in (2)
2. Evaluate y'_{n+1} using the formula in (3)
3. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$

2.1.2 PEC' mode

The unusual implementation of Falkner method on each step for solving the problem in (1) reads

1. Evaluate y_{n+1} using the formula in (2)
2. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$
3. Evaluate y'_{n+1} using the formula in (5)

which can be accomplished due to the absence of the derivative on the function f. Thus, having obtained the value y_{n+1} it is straightforward to obtain f_{n+1} to be used

 ISBN 13: 978-84-613-5510-5

in the formula in (5). Note that in this way the "implicit formula" in (5) is no longer implicit, resulting in an explicit formulation of the method.

For the PP'E mode the accumulated truncation error is of order $\mathcal{O}(h^k)$ while for the PEC' mode is of order $\mathcal{O}(h^{k+1})$ [1]. Thus, the unusual implementation of the explicit Falkner method provides a better performance.

2.2 Implicit methods

2.2.1 P'PECE mode

The first choice for the implementation of the implicit Falkner method is given by

1. Evaluate y'_{n+1} using the formula in (3)
2. Evaluate y_{n+1} using the formula in (2)
3. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$
4. Evaluate y_{n+1} using the formula in (4)
5. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$

2.2.2 PECC'E mode

The second possibility is given by

1. Evaluate y_{n+1} using the formula in (2)
2. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$
3. Evaluate y'_{n+1} using the formula in (5)
4. Evaluate y_{n+1} using the formula in (4)
5. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$

2.2.3 PECEC' mode

The last implementation consist in

1. Evaluate y_{n+1} using the formula in (2)
2. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$
3. Evaluate y_{n+1} using the formula in (4)
4. Evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$
5. Evaluate y'_{n+1} using the formula in (5)

Note that in the above formulations after the last value obtained for y_{n+1} we have to evaluate $f_{n+1} = f(x_{n+1}, y_{n+1})$, which will be used at the next step.

 ISBN 13: 978-84-613-5510-5

3 Error analysis

3.1 Local truncation errors

Consider the formula resulting after approximating the function f on the exact formula

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \int_{x_n}^{x_{n+1}} f(x, y(x), y'(x))(x_{n+1} - x)\, dx \tag{6}$$

by the interpolating polynomial on the grid points $x_{n-(k-1)}, \ldots, x_n$, and also consider the formula in (2). Using the localization hypothesis that $y(x_j) = y_j$, $j = n - (k - 1), \ldots, n$, and following a similar procedure as for the Adams formulas [9] we obtain that the local truncation error for the method in (2) is given by

$$\mathfrak{L}_P[y(x_n); h] = h^{k+2} y^{(k+2)}(\xi)\beta_k, \tag{7}$$

where ξ is an internal point of the smallest interval containing $x_{n-(k-1)}, \ldots, x_n$.

Similarly, for the formula in (3) the local truncation error reads

$$\mathfrak{L}_{P'}[y'(x_n); h] = h^{k+1} y^{(k+2)}(\psi)\gamma_k, \tag{8}$$

where ψ as before refers to an internal point.

And for the formula in (4) the local truncation error is given by

$$\mathfrak{L}_C[y(x_n); h] = h^{k+3} y^{(k+3)}(\xi)\beta^*_{k+1}, \tag{9}$$

and similarly for the formula in (5) the local truncation error is

$$\mathfrak{L}_{C'}[y'(x_n); h] = h^{k+2} y^{(k+3)}(\psi)\gamma^*_{k+1} \tag{10}$$

where we denote by ξ or ψ the internal points, or if they have to be different, by ξ_j or ψ_j, as in the following section.

3.2 Accumulated truncation errors for the implicit Falkner method in P-C modes

3.2.1 P'PECE mode

Assuming that we have to integrate the problem (1) on the interval $[x_0, x_N]$ and that we know in advance the starting values needed to apply the numerical scheme, we proceed by analyzing the accumulated errors on each successive application of the method along the grid points on the integration interval. We use a superscript P to denote the application of the explicit method and a superscript C to denote the application of the implicit one.

Assuming the localization hypothesis and using the formulas for the local truncation errors in (7) and (8), for the first step ($n = 1$) we have that the differences between the

 ISBN 13: 978-84-613-5510-5

true values, $y(x_1)\,, y'(x_1)$, and the approximated ones, $y_1^P\,, y_1'^P$, are given respectively by the local truncation errors, that is,

$$y'(x_1) - y_1'^P = h^{k+1} y^{(k+2)}(\psi_1)\gamma_k\,, \tag{11}$$

$$y(x_1) - y_1^P = h^{k+2} y^{(k+2)}(\xi_1)\beta_k\,. \tag{12}$$

where, for convenience, in the sequel the ξ_j and ψ_j will denote appropriate internal points.

After evaluating $f(x_1, y_1^P)$, using the Mean Value Theorem we can put that

$$f(x_1, y(x_1)) - f_1^P = \frac{\partial f}{\partial y}(x_1, \xi_1)\ \left(y(x_1) - y_1^P\right)$$

where $f_1^P = f(x_1, y_1^P)$. Assuming the localization hypothesis, we have

$$\begin{aligned} y(x_1) - y_1^C &= y(x_0) + h\, y'(x_0) + h^2 \sum_{j=0}^{k} \beta_j^* \bigtriangledown^j f(x_1, y(x_1)) \\ &\quad + h^{k+3} y^{(k+3)}(\xi_1)\beta_{k+1}^* - \left(y_0 + h\, y_0' + h^2 \sum_{j=0}^{k} \beta_j^* \bigtriangledown^j f_1^P \right) \\ &= h^{k+3} y^{(k+3)}(\xi_1)\beta_{k+1}^* + \mathcal{O}(h^{k+4})\,. \end{aligned} \tag{13}$$

For the next step ($n = 2$) we have that

$$y'(x_2) - y_2'^P = y'(x_1) + h \sum_{j=0}^{k-1} \gamma_j \bigtriangledown^j f(x_1, y(x_1)) \tag{14}$$

$$+ h^{k+1} y^{(k+2)}(\psi_2)\gamma_k - \left(y_1'^P + h \sum_{j=0}^{k-1} \gamma_j \bigtriangledown^j f_1^C \right) \tag{15}$$

where we have used the values of the step before, $y_1'^P$ and y_1^C, and have used the notation $f_1^C = f(x_1, y_1^C)$. After some calculations, using the formulas in (11) and (13), it results that

$$y'(x_2) - y_2'^P = h^{k+1}\gamma_k \left(y^{(k+2)}(\psi_1) + y^{(k+2)}(\psi_2) \right) + \mathcal{O}(h^{k+2})\,. \tag{16}$$

Similarly, for the predictor we have

$$y(x_2) - y_2^P = h^{k+2}\beta_k y^{(k+2)}(\xi_2) + h^{k+2} y^{(k+2)}(\xi_1)\gamma_k + \mathcal{O}(h^{k+3})\,. \tag{17}$$

And finally, for the corrector we get that

$$y(x_2) - y_2^C = y(x_1) + h\, y'(x_1) + h^2 \sum_{j=0}^{k} \beta_j^* \bigtriangledown^j f(x_1, y(x_1))$$

 ISBN 13: 978-84-613-5510-5

$$
\begin{aligned}
& + h^{k+3} y^{(k+3)}(\xi_2)\beta^*_{k+1} - \left(y_1^C + h\, y_1'^P + h^2 \sum_{j=0}^{k} \beta_j^* \bigtriangledown^j f_2^P \right) \\
= \;& h^{k+3}\beta^*_{k+1}\left(y^{(k+3)}(\xi_1) + y^{(k+3)}(\xi_2) \right) + h^{k+2} y^{(k+2)}(\psi_1)\gamma_k \\
& + \mathcal{O}(h^{k+4})\,. \qquad (18)
\end{aligned}
$$

Repeating the procedure along the nodes on the integration interval we determine that the accumulated error at the final point x_N is given by

$$
\begin{aligned}
y(x_N) - y_N^C \;=\; & h^{k+2}\gamma_k \sum_{j=1}^{N-1} (N-j) y^{(k+2)}(\psi_j) + h^{k+3}\beta^*_{k+1} \sum_{j=1}^{N} y^{(k+3)}(\xi_j) \\
& + \mathcal{O}(h^{k+4})\,, \qquad (19)
\end{aligned}
$$

and for the derivative we have

$$
y'(x_N) - y_N'^P \;=\; h^{k+1}\gamma_k \sum_{j=0}^{N-1} y^{(k+2)}(\psi_j) + \mathcal{O}(h^{k+2})\,. \qquad (20)
$$

Assuming that the derivatives $y^{(k+2)}(x)$ and $y^{(k+3)}(x)$ are continuous, after using the Mean Value Theorem the above formulas may be rewritten as follows:

$$
\begin{aligned}
y(x_N) - y_N^C \;=\; & \frac{1}{2} h^k \gamma_k y^{(k+2)}(\psi)(x_N - x_0)(x_N - x_1) \\
& + h^{k+1}\beta^*_{k+1} y^{(k+3)}(\xi)(x_N - x_0) + \mathcal{O}(h^{k+4}) \qquad (21)
\end{aligned}
$$

and

$$
y'(x_N) - y_N'^P \;=\; h^k \gamma_k (x_N - x_0) y^{(k+2)}(\psi) + \mathcal{O}(h^{k+2})\,. \qquad (22)
$$

3.2.2 PECC 'E mode

In this mode, for the first step ($n = 1$), after the application of the predictor P and the corrector C we have the same results as before, that is,

$$
y(x_1) - y_1^P \;=\; h^{k+2} y^{(k+2)}(\xi_1)\beta_k\,,
$$

and

$$
y(x_1) - y_1^C = h^{k+3} y^{(k+3)}(\xi_1)\beta^*_{k+1} + \mathcal{O}(h^{k+4})\,.
$$

Now the difference with the mode before results in the application of the corrector C' to obtain the approximation for the derivative. We have

$$
y'(x_1) - y_1'^{C'} \;=\; y'(x_0) + h \sum_{j=0}^{k} \gamma_j^* \bigtriangledown^j f(x_1, y(x_1))
$$

 ISBN 13: 978-84-613-5510-5

$$+ h^{k+2} y^{(k+3)}(\psi_1)\gamma^*_{k+1} - \left(y_0'^{C'} + h \sum_{j=0}^{k} \gamma^*_j \bigtriangledown^j f_1^P \right)$$

$$= h^{k+2} y^{(k+3)}(\psi_1)\gamma^*_{k+1} + \mathcal{O}(h^{k+3}) . \tag{23}$$

After the application of the predictor P to obtain an estimate y_2^P for $y(x_2)$, the application of the corrector C results in

$$y(x_2) - y_2^C = h^{k+3}\beta^*_{k+1}\left(y^{(k+3)}(\xi_1) + y^{(k+3)}(\xi_2) \right) + h^{k+3} y^{(k+3)}(\psi_1)\gamma^*_{k+1}$$

$$+ \mathcal{O}(h^{k+4}) , \tag{24}$$

and the application of the corrector C' produces

$$y'(x_2) - y_2'^{C'} = h^{k+2}\gamma^*_{k+1}\left(y^{(k+3)}(\psi_1) + y^{(k+3)}(\psi_2) \right) + \mathcal{O}(h^{k+3}) . \tag{25}$$

Repeating the procedure along the nodes on the integration interval, and assuming that $y^{(k+3)}(x)$ is continuous, we obtain that the accumulated errors at the final point x_N are given respectively by

$$y(x_N) - y_N^C = \frac{1}{2} h^{k+1}\gamma^*_{k+1} y^{(k+3)}(\psi)(x_N - x_0)(x_N - x_1)$$

$$+ h^{k+2}\beta^*_{k+1} y^{(k+3)}(\xi)(x_N - x_0) + \mathcal{O}(h^{k+3}) \tag{26}$$

for the solution, and

$$y'(x_N) - y_N'^{C'} = h^{k+1}\gamma^*_{k+1}(x_N - x_0) y^{(k+3)}(\psi) + \mathcal{O}(h^{k+3}) \tag{27}$$

for the derivative.

3.2.3 PECEC' mode

Now the difference with the mode before results in the application of the corrector C' to obtain the values $y_n'^{C'}$, after evaluating $f(x_1, y_n^C)$. The first difference to the previous mode is observed in the first step ($n = 1$) and results to be

$$y'(x_1) - y_1'^{C'} = y'(x_0) + h \sum_{j=0}^{k} \gamma^*_j \bigtriangledown^j f(x_1, y(x_1))$$

$$+ h^{k+2} y^{(k+3)}(\psi_1)\gamma^*_{k+1} - \left(y_0'^{C'} + h \sum_{j=0}^{k} \gamma^*_j \bigtriangledown^j f_1^C \right)$$

$$= h^{k+2} y^{(k+3)}(\psi_1)\gamma^*_{k+1} + \mathcal{O}(h^{k+3}) . \tag{28}$$

Note that the only difference between the formulas in (23) and in (28) is that f_1^P has been substituted by f_1^C, but this does not change the principal term of the errors. This means that the principal terms in the errors for the final formulas are the same as in the previous mode, and so the formulas in (26) and (27) remain valid in this mode.

 ISBN 13: 978-84-613-5510-5

4 Numerical example

In this section, we have considered the same example as in [1] to illustrate the performance of the implementations of the implicit Falkner method taking $k = 6$. Table 1 shows the numerical results for different number of steps, where we have included the maximum of the absolute errors at the nodal points in the integration interval for the solution

$$MaxErr(y) = \max_{j\in 0,\ldots,N} |y(x_j) - y_j| \,,$$

and the maximum of the absolute errors at the nodal points for the derivative

$$MaxErr(y') = \max_{j\in 0,\ldots,N} |y'(x_j) - y'_j| \,.$$

Further, we have also included the CPU time needed for each implementation. The implementations appear in the column Method, where 1 refers to the P'PECE mode, 2 refers to the PECC'E mode, and 3 refers to the PECEC' mode.

The I.V.P. considered is given by

$$y''(x) = -y(x) + \sin(x) \,, \quad y(0) = 1 \,, \quad y'(0) = 0 \tag{29}$$

whose exact solution is

$$y(x) = \frac{1}{2}(\sin(x) + (2 - x)\cos(x)) \,.$$

The problem has been integrated on the interval $[0, 20\pi]$ using the MATHEMATICA program. We observe that the implementations 1 and 2 have a similar behaviour in what concerns the CPU time, while the implementation 3 is more time consuming. On the other hand, implementations 2 and 3 have a similar accuracy, which is better than that on implementation 1. These considerations lead to choose as the best implementation the PECC'E mode.

Figure 1 depicts the propagation of the absolute global errors for the solution using the implementations 1,2 and 3 (from top to bottom). We can see that the best results correspond to the PECC'E implementation of the implicit Falkner method. For the absolute global errors on the derivatives similar results were obtained, as shown in Figure 2.

5 Conclusions

Falkner methods are commonly used for solving the special second-order differential initial-value problem (particularizations of these methods are the well-known Velocity-Verlet, Beeman method or Wilson method). On these approaches two formulas are needed for advancing the solution $y(x)$ and the derivative $y'(x)$. When the formulas are implicit, they may be implemented in predictor-corrector mode.

We have consider three different modes to implement the implicit Falkner formulas and have made an analysis of the accumulated truncation errors. Considering the

 ISBN 13: 978-84-613-5510-5

N steps	Method	Time (s.)	$MaxErr(y)$	$MaxErr(y')$
500	1	0.172	5.6090×10^{-4}	5.3804×10^{-4}
	2	0.156	3.2023×10^{-6}	3.3933×10^{-6}
	3	0.203	2.5646×10^{-6}	2.7175×10^{-6}
1000	1	0.297	8.8186×10^{-6}	8.4363×10^{-6}
	2	0.312	2.1011×10^{-8}	2.2248×10^{-8}
	3	0.422	1.9343×10^{-8}	2.0483×10^{-8}
1500	1	0.453	7.7476×10^{-7}	7.5348×10^{-7}
	2	0.453	1.1535×10^{-9}	1.2197×10^{-9}
	3	0.657	1.1577×10^{-9}	1.1755×10^{-9}
2000	1	0.625	1.3790×10^{-7}	1.3406×10^{-7}
	2	0.625	1.7914×10^{-10}	1.9040×10^{-10}
	3	0.890	1.6783×10^{-10}	1.7861×10^{-10}
2500	1	0.781	3.6212×10^{-8}	3.5576×10^{-8}
	2	0.781	4.7879×10^{-11}	4.4064×10^{-11}
	3	1.125	5.3216×10^{-11}	4.9219×10^{-11}
3000	1	0.922	1.2161×10^{-8}	1.2120×10^{-8}
	2	0.953	4.7984×10^{-11}	4.9226×10^{-11}
	3	1.359	4.9348×10^{-11}	5.0579×10^{-11}

Table 1: *Errors using different implementations for the implicit Falkner method in P-C mode with $k = 6$ for solving the problem in (29)*

expressions of these errors for the solution and the derivative we observe that the mode PECC'E shows the best performance. The resulting errors in this case are both of order $\mathcal{O}(h^{k+1})$. A numerical example is provided to make a comparison of the numerical performance of the different implementations. The numerical results agree with the theoretical analysis.

Acknowledgements

The authors wish to thank JCYL project SA050A08 and MICYT project MTM2008/05489 for financial support.

 ISBN 13: 978-84-613-5510-5

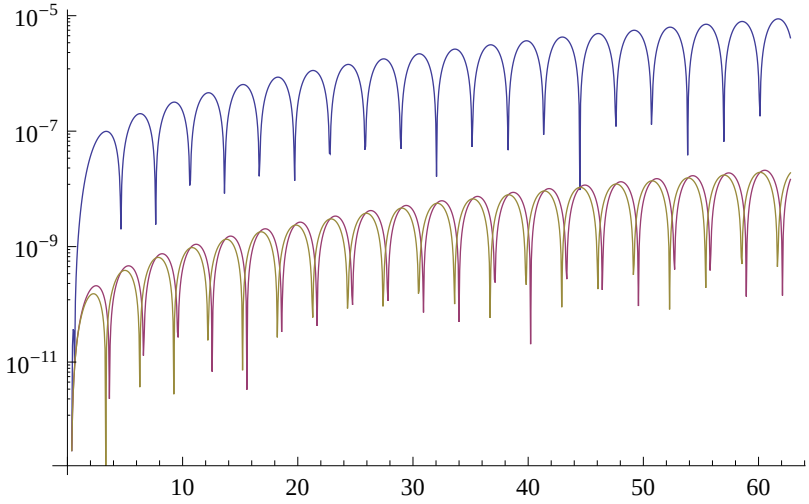


Figure 1: Absolute global errors in logarithmic scale for the solution $y(x)$ of problem (29) using modes 1,2 and 3 (from top to bottom)

References

[1] H. Ramos and C. Lorenzo, *Review of explicit Falkner methods and its modifications for solving special second-order I.V.P.s*, Submitted in Comput. Phys. Comm. (2010) 1–26.

[2] U. M. Ascher, L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia, 1998.

[3] D. Beeman, *Some multistep methods for use in molecular dynamics calculations*, J. Comput. Phys. **20** (1976) 130–139.

[4] L. Collatz, *The Numerical Treatment of Differential Equations*, Springer, Berlin, 1966.

[5] V. M. Falkner, *A method of numerical solution of differential equations*, Phil. Mag. S. **7** (1936) 621–640.

[6] I. Gladwell, R. Thomas, *Stability properties of the Newmark, Houbolt and Wilson θ methods*, I. J. Numer. and Anal. Meth. in Geom. **4** (1980) 143–158.

[7] E. Hairer, S. P. Norsett and G. Wanner, *Solving Ordinary Differential Equations I* , Springer, Berlin, 1987.

[8] P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley, New York, 1962.

 ISBN 13: 978-84-613-5510-5

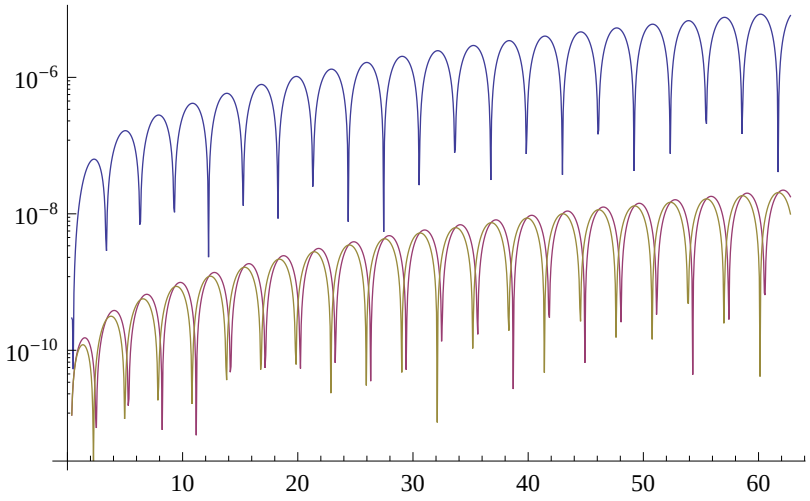


Figure 2: Absolute global errors in logarithmic scale for the derivative $y'(x)$ of problem (29) using modes 1,2 and 3 (from top to bottom)

[9] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley, England, 1991.

[10] L. F. Shampine and M. K. Gordon, *Computer solution of Ordinary Differential Equations. The initial Value Problem*, Freeman, San Francisco, CA, 1975.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Parallel algorithms for the facility location and design $(1 \mid 1)$-centroid problem on the plane*

J.L. Redondo[1], J. Fernández[2], I. García[1] and P.M. Ortigosa[1]

[1] *Dept. of Computer Architecture and Electronics, University of Almería*

[2] *Dept. of Statistics and Operational Research, University of Murcia*

emails: jlredondo@ual.es, josefdez@um.es, igarcia@ual.es, ortigosa@ual.es

Abstract

Several parallel strategies for solving a centroid problem are presented. In the competitive location problem considered in this paper, the aim is to maximize the profit obtained by a chain (the leader) knowing that a competitor (the follower) will react by locating another single facility after the leader locates its own facility. A global optimization memetic algorithm called UEGO_cent.SASS was proposed to cope with this hard-to-solve optimization problem. Now, five parallel implementations of the optimization algorithm have been developed. The use of several processors, and hence more computational resources, allows us to solve bigger problems and to implement new methods which increase the robustness of the algorithm at finding the global optimum. A computational study comparing the new parallel methods in terms of efficiency and effectiveness has been carried out.

Key words: Nonlinear bi-level programming problem, Centroid (or Stackelberg) problem, Continuous location, Competition, Evolutionary algorithm, Parallelism, Master-Slave, Coarse-grain.

1 Introduction

When a retail chain considers entering or extending its presence in a market by opening a new facility a key question to be investigated is 'where' to locate the new facility. If in the area there already exist other facilities offering the same goods, then the new facility will have to *compete* for the market. Many competitive location models are available in literature, see for instance the survey papers [6, 7, 10] and the references therein. Competitive location is nowadays a very active area of research within the field of 'Location Science' (for an introduction to this more general topic see [4, 5, 8]).

*† Corresponding author: J. L. Redondo. Email: jlredondo@ual.es. Tel: +34 950 014023; Fax: +34 950 015486

 ISBN 13: 978-84-613-5510-5

The scenario considered in this paper is that of a *duopoly*. A chain (the leader) wants to set up a single new facility in a region of the plane where similar facilities of a competitor (the follower), and possibly of its own chain, are already present. After the location of the leader's facility, the competitor will react by locating another new facility at the place that maximizes its own profit. The objective of the leader is to find the *location* and the *quality* of its facility that maximizes its profit, following the location of the facility of the follower. These type of problems are known as Stackelberg problems in economic literature and as Simpson's problems in voting theory. In Location literature they were introduced by Hakimi [9]. He introduced the terms *medianoid* for the follower problem, and *centroid* for the leader problem. Literature on centroid problems is scarce (see [6] for a review on the topic until 1996), and to our knowledge, among the existing papers only five of them [1, 2, 3, 12] deal with *continuous* problems (the set of feasible locations for the new facility is a region of the plane). In [11] four heuristics were proposed to cope with the problem considered in this paper, and among them an evolutionary algorithm called UEGO_cent.SASS was the one giving the best results: in the set of problems solved in [11], it always obtained the best objective value.

UEGO_cent.SASS belongs to the class of evolutionary optimization methods based on subpopulations (species). One of its main features is that the evaluation of a species requires intensive computational effort, since it involves the execution of another optimization algorithm. To reduce the computational burden and to be able to solve problems with reasonable sizes in a standard uniprocessor, UEGO_cent.SASS was designed to maintain few species, which may affect its ability to find the optimal solution. In spite of all, UEGO_cent.SASS can only solve instances with a moderate size. This clearly calls for a parallelization of the algorithm. In this work, a *master-slave* implementation and four *coarse-grain* algorithms, which differ in the migration policy, are proposed.

The objective of this paper is twofold. First, to parallelize UEGO_cent.SASS in such a way that, thanks to the use of more processing elements it can solve larger size problems. Also to study the efficiency of the parallelizations, to find out which of the parallel implementations has a better efficiency. As will be seen, the master-slave strategy reaches an almost ideal efficiency, and it allows to solve problems with up to 200 demand points when using 8 processing elements. Second, to increase the robustness of UEGO_cent.SASS at finding the global optimum. To this aim a new procedure was designed and executed with the parallel algorithm.

The rest of the paper is organized as follows: The problem is presented in the next section. In Section 3 a master-slave and four coarse-grain parallelizations of UEGO_cent.SASS are introduced. It is in Section 4 where the new proposal which allow us to solve the problem with more reliability is described. Computational studies to check the effectiveness of the proposal as well as the efficiency of the parallelizations are reported in Section 5. The paper ends with some conclusions.

2 The problem

Next the problem considered in this paper is briefly described. For a mathematical formulation and more details the interested reader is referred to [11].

A chain, the *leader*, wants to locate a single new facility $nf_1 = (x_1, y_1, \alpha_1)$ in a given area

 ISBN 13: 978-84-613-5510-5

of the plane, where there already exist m facilities offering the same goods or product. The first k of those m facilities belong to the chain, and the other $m-k$ to a competitor chain, the *follower*. The leader knows that the follower, as a reaction, will subsequently position a new facility $nf_2 = (x_2, y_2, \alpha_2)$, too. The demand, supposed to be inelastic, is concentrated at n demand points, whose locations and buying power are known. The location and quality of the existing facilities are also known. We consider that the patronizing behavior of customers is probabilistic, that is, demand points split their buying power among all the facilities proportionally to the attraction they feel for them, determined by the different perceived qualities of the facilities and the distances to them, through a gravitational or Huff-type model.

Given nf_1, the problem of the follower is the medianoid problem $(FP(nf_1))$, which aims at finding the best location (x_2, y_2) and quality α_2 for its facility so that it maximizes its profit $\Pi_2(nf_1, nf_2)$ (once the leader has set up its new facility at nf_1). The profit is to be understood as the difference between the revenues obtained from the captured market share minus the operating costs of the new facility (see [11]).

Let us denote by $nf_2^*(nf_1)$ an optimal solution of $(FP(nf_1))$. The problem for the leader is the centroid problem (LP), which aims at finding the best location (x_1, y_1) and quality α_1 for its facility so that it maximizes its profit $\Pi_1(nf_1, nf_2^*(nf_1))$, knowing that the follower will locate another single facility $nf_2^*(nf_1)$ at the place and with the quality that maximizes its own profit.

As can be seen, the leader problem (LP) is much more difficult to solve than the follower problem $(FP(nf_1))$. Notice, for instance, that to evaluate its objective function Π_1 at a given point nf_1 we have to first solve the corresponding medianoid problem $(FP(nf_1))$ to obtain $nf_2^*(nf_1)$. Furthermore, in order to compute the objective value of Π_1 at nf_1 accurately, the follower problem $(FP(nf_1))$ has to be solved exactly, otherwise, the error of the approximate value can be considerable.

3 High performance computing approaches for the leader problem

The leader problem was solved in [11] using the population-based method UEGO_cent.SASS. This algorithm works with a population of candidate solutions (species). It is important to mention that there is no relationship among species. This means that a single species can create a new offspring and evolve to the local or global optima without participation of the remaining ones. Therefore, there exists an intrinsic parallelism, which can be exploited by dividing the species among the available processing elements. This work explores and evaluates a master-slave and several coarse-grain strategies applied to UEGO_cent.SASS.

3.1 A master-slave strategy (MS)

A master-slave technique is a "global parallel model", since the management of the population is global, i.e. all the individuals in the population are considered when selection or creation procedures are carried out. Two kinds of processing elements can be distinguished, the *master* and the *slaves*. The master processor sees into making global decisions and delivering information

 ISBN 13: 978-84-613-5510-5

among the slaves, which execute different tasks in a concurrent way.

In our particular master-slave (MS) model, the master processor executes UEGO_cent.SASS sequentially. The parallelism comes from the simultaneous resolution of the medianoid problems to evaluate correctly the new leader's candidate solutions in the Create_species function, and from the concurrent execution of the Optimize_species procedure (see [11]). Therefore, new creation and optimization procedures have been designed to cope with the parallel model. These new procedures, called Create_species_paral and Optimize_species_ paral, are described now:

- *Create_species_paral*: At each level i, the master obtains a new offspring of candidate solutions for the leader. The computation of the corresponding follower's facility and the evaluation of the correct objective value for the leader's facility, are carried out in a parallel way. To this aim, the master processor divides the list of candidate solutions by the number of processors P and delivers the resulting sublists among all the processing elements (including itself). Each processing element receives a species sublist from the master processor and solve the medianoid problem to obtain the follower's location associated to every leader's facility. To do so, a processing element only needs to know the two features of the leader's facility nf_1, i.e. its location (x_1, y_1) and its quality α_1. Therefore, the amount of information involved in these communication procedures is quite small.

  The master processor does not receive information from the slaves until it has finished its work (first synchronization point). When it does, it passes to a reception state, where it picks up the follower sublists sent by the slaves. Once the master has received all the information from the slaves, it updates the candidate solutions list and includes it in the species list.

  Following with the general structure of UEGO_cent.SASS, the species list is now fused (see [11]). Notice that the fusion only implies the computation of distances between the centers of all pairs of species, then this process is not time consuming. If the list length is larger than the maximum allowed, the *Shorten_species_list* procedure is then performed (see [11]). These two procedures are carried out by the master, while the slaves stay idle (second synchronization point).

- *Optimize_species_paral*: To perform the optimization process, the master divides the species list among all the processors (again including itself). In this case, each slave executes the Optimize_species procedure to every species in its sublist, i.e. a local optimization process is applied to the leader's facility and the medianoid problem is solved to obtain the corresponding follower's center (see [11]). When the master finishes its work, it begins to receive the new species sublists from the slaves (third synchronization point), which will be fused (see [11]).

Note that the synchronization points are imposed by the need to know the correct objective function value at all the points of the leader before proceeding to execute the next stage of the optimization procedure, or because the master is managing the whole species list.

 ISBN 13: 978-84-613-5510-5

3.2 Coarse-grain strategies

In this section, UEGO_cent.SASS has been parallelized following a coarse-grain strategy. In a coarse-grain model, each processing element executes an algorithm (in our case UEGO_cent. SASS) independently of the remaining ones during most of the time. The idea is that different processing elements work with smaller and different subpopulations in such a way that, when merging all the subpopulations, a population similar to that of the sequential version can be obtained. Nevertheless, some information can migrate from a processing element to another one according to a migratory policy, which is controlled by the parameters: (i) *Interval of migration*, it establishes how often the migration of a certain amount of individuals will be conducted. (ii) *Rate of migration*, it indicates the number of individuals that have to communicate with other processing elements. (iii) *Selection criterion*, it determines the policy that will be applied for the selection of migratory individuals.

In this work, several migratory policies have been implemented. To evaluate all the coarse-grain strategies, an exhaustive computational study was carried out. Next, the four best parallel coarse-grain implementations, in terms of effectiveness and efficiency, are described. They are named *Collector*, *Ring-Fusion1*, *Ring-Fusion2* and *Collector+Ring-Fusion*.

3.2.1 Collector strategy (CO)

In this strategy, the exchange of information is carried out through a single processor. However, a hierarchy of processors does not exist, i.e. they all work in the same way. It is only during the migration process that we distinguish between two different types of processing elements: if P is the total number of processors, there will be $P-1$ *workers* and a single *collector*. The collector will be the communication channel among processors when the exchange of information is accomplished.

The *rate of migration* in this strategy is equal to the length of the species sublists. Therefore, in a migration, every worker sends its whole sublist to the collector.The collector processor fuses them (including its own species sublist) to decrease the possible redundant work. Once the fused list is obtained, the collector distributes it among all of the processors, including itself. After a previous analysis, the interval of migration has been carried out at every 3 iterations.

3.2.2 Ring-Fusion strategy (RF)

In this parallel technique, processors are connected following a ring topology in such a way that processor i can only communicate with processors $i-1$ and $i+1$. Moreover, a classification into workers and collectors exists, although now, half of the processors act as collectors and the other half as workers. It is important to highlight that the role of collector or worker is not fixed, i.e., processors interchange their role at every communication stage.

The rate of migration in this strategy is also equal to the length of the species sublists. Communications are established by pairs of processors as follows: In a migration, processor i is a worker and sends its species sublist to the next processor $i+1$ (collector). Processor $i+1$ fuses this sublist with its own sublist and distributes the resulting list between both processors. In the next communication stage, processor i will be a collector that will receive a sublist from

 ISBN 13: 978-84-613-5510-5

processor *i-1* and the processor *i+1* will be a worker that will send the sublist to the processor *i+2*.

Two different intervals of migration have been considered, giving place to the following *Ring-Fusion* versions: (i) RF1, the migration process is carried out at the first half of the levels of the algorithm. (ii) RF2, the migration process is carried out at every level of the algorithm.

3.2.3 Collector+Ring-Fusion strategy (CO+RF)

This parallel technique is a hybrid model from the previous ones. On the one hand, during the first half of the levels, $P-1$ processing elements will be considered as workers and a single one as collector. At each level (with $level < L/2$), the workers send their whole sublists to the collector. This processor fuses them, including its own species sublist and distributes the resulting list among all the processors, including itself. On the other hand, throughout the last half of the levels ($level \geq L/2$), the migratory policy RF is employed and therefore, global fusions are conducted by pairs of processing elements at each level.

4 Improving the quality of the solution. Obtaining a new creation procedure.

For the sequential version of UEGO_cent.SASS, and so as to solve as many problems as possible, some procedures were modified to reduce the computational load (see [11]). Nevertheless, if the search is not exhaustive enough, the algorithm may become trapped in local optima. Operating on the principle that large problems can almost always be divided into smaller ones, which can be solved concurrently ("in parallel"), this section is aimed at designing new alternative procedures that explore the search space deeper, although this additional effort requires the use of parallel architectures with more computing resources. In particular, the creation has been studied.

For every level i, UEGO_cent.SASS, has a radius value R_i, and two maxima on the number of function evaluations (f.e.), namely new_i (f.e. allowed when creating new species) and n_i (maximum f.e. allowed when optimizing individual species). The budget per species for the creation (bc_i) of a new offspring is given as $new_i/length(species\_list_i)$, while the budget per species for the optimization (bo_i) is n_i/max_spec_num (see [11]). This means that there is a remainder of $n_i - bo_i \cdot length(species\_list_i)$ f.e. in the optimization process, when the length of the $species\_list_i$ is not equal to the maximum allowed. The idea in this case is to use this number of evaluations for creating more candidate solutions and therefore, to explore the region more exhaustively. The new procedure forces the creation of more candidate solutions at each level of the algorithm. The budget per species in the level $i+1$ is then

$$bc_{i+1} = \frac{new_{i+1} + n_i - bo_i \cdot length(species\_list_i)}{length(species\_list_{i+1})}.$$

 ISBN 13: 978-84-613-5510-5

Table 1: Effectiveness evaluation of the $Create\_species_1$ procedure versus $Create\_species_2$ method.

Setting	$Create\_species_{11}$			$Create\_species_{21}$			
(n,m,k)	$Av(\Pi)$	$Av(T)$	$Av(Spc)$	$Av(\Pi)$	$Av(T)$	$Av(Spc)$	$DifObj$
(21,5,2)	228.381	540.0	52.8	230.675	783.2	63.8	-1.004
(21,5,3)	379.572	569.0	56.3	380.355	788.8	66.8	-0.206
(50,5,0)a	9.155	1015.0	45.8	9.156	1293.2	147.2	-0.009
(50,5,0)b	93.894	1089.0	42.0	94.123	1382.6	133.4	-0.244
(50,5,1)	143.487	1301.0	41.5	144.987	1624.6	128.2	-1.045
(50,5,2)a	189.583	1132.1	35.0	189.981	1266.2	110.6	-0.210
(50,5,2)b	111.171	1149.0	42.2	111.801	1398.6	143.4	-0.566
(50,6,3)a	292.530	1060.0	30.6	292.943	1355.0	83.0	-0.141
(50,6,3)b	212.170	1781.0	44.6	213.965	2260.8	140.4	-0.845
(50,6,3)c	230.256	1765.0	19.8	232.851	3153.4	45.60	-1.114
(50,8,4)	222.775	1860.0	37.4	223.993	2421.6	118.8	-0.546
(100,2,0)	169.696	1389.0	30.6	171.667	2080.0	117.0	-1.161
(100,2,1)	272.021	1705.0	35.2	273.342	2365.0	134.0	-0.485
(100,10,0)	40.943	3065.0	39.0	40.947	3993.0	150.0	-0.008

5 Computational studies

All the computational results in this paper have been carried out on a cluster of 32 nodes, each of them with 2 processors Xeon IV to 2.4GHz and 1 GByte RAM. The algorithms have been implemented in C++ and MPI (Message Passing Interface) and using only one processor in each node.

5.1 Effectiveness of the new creation procedures

First, this subsection studies whether the creation procedure described in Subsection 4 is able to improve the solutions provided by the original (i.e. $Create\_species_1$) procedure (see [11]). To this aim, the 14 problems solved by UEGO_cent.SASS in [11], have been solved again, but with this new mechanism. The problems were generated varying the number n of demand points, the number m of existing facilities and the number k of those facilities belonging to the chain. The settings (n,m,k) of such problems are detailed in the first column of Table 1 (for three of the settings more than one problem was generated, and we have added the letters a, b, and c at the end of the setting to highlight it). For every setting, the problems were generated by randomly choosing the parameters of the problems uniformly within given intervals [11].

The high computational requirements of the new creation procedure make the sequential UEGO_cent.SASS run out of memory most of the times. Thus, so as to check the new techniques, the master-slave parallel model has been selected. For the studies, the use of 2 processing elements has been enough.

As measurement of effectiveness we have computed the relative difference in objective value between the optimal value obtained by the sequential UEGO_cent.SASS in [11],$OptVal($

 ISBN 13: 978-84-613-5510-5

$UEGO\_cent.SASS$), and the solution obtained by the master-slave algorithm with the new $Creation\_species_2$ procedure when using 2 processing elements, $OptVal\_MS(2)$,

$$DifObj = \frac{OptVal(UEGO\_cent.SASS) - OptVal\_MS(2)}{OptVal(UEGO\_cent.SASS)}.$$

Negative values imply that the new creation mechanism improves the solution provided by UEGO_cent.SASS in [11], where the $Create\_species_1$ procedure is implemented. It is important to highlight that, due to the stochastic nature of the algorithms, all the experiments have been executed 5 times and average values have been considered when computing $DifObj$. However, the confidence intervals obtained for these average values were relatively narrow, which reveals the robustness of the algorithm's solutions.

Computational experiments were carried out and proved that $Create\_species_2$ provides better results. Table 1 compares the results obtained by the sequential UEGO_cent.SASS in [11] (see the three columns under $Create\_species_1$) to those obtained by $MS(2)$ with $Create\_species_2$. The corresponding average value of the objective function over the 5 runs ($Av(\Pi)$), the average running time ($Av(T)$), the average number of species in the final *species_list* ($Av(Spc)$), as well as $DifObj$, are given for each problem.

The new creation method $Creation\_species_2$ improves the profit more than 1% for some instances (see column $DifObj$). Nevertheless, the execution time is increased, even though double computing resources are employed (it is executed by the master-slave algorithm with 2 processing elements). The rise of CPU time is due to the additional effort the creation method makes to carry out a more exhaustive search (see columns $Av(Spc)$, the number of species in the final list, i.e., the number of local optimal points, is bigger with $Creation\_species_2$).

5.2 Efficiency results for small problems

One of the main goals in parallelism consists of increasing the performance of an application with respect to its execution on a uniprocessor. A commonly used metric to measure the performance of a parallel implementation on homogeneous processors is the efficiency, which estimates how well-utilized the processors are in solving the problem. The efficiency of a parallel version (run over P processors) with respect to the sequential one is computed as: $Eff(P) = \frac{T(1)}{P \cdot T(P)}$, where $T(P)$ is the CPU time employed by the algorithm when P processing elements are used. Nevertheless, since the sequential algorithm UEGO_cent.SASS cannot manage some difficult-to-solve problems, it is not possible to obtain the reference $T(1)$. An alternative measure is then the relative efficiency, defined by $Eff(Q,P) = \frac{Q \cdot T(Q)}{P \cdot T(P)}$, where Q is the minimum number of processors needed to solve the problem.

Table 2 shows the set of problems that each parallel strategy has been able to execute with Q processing elements. As can be seen, MS is the only method that was able to solve all the problems using only 2 processors. The reason for this fact is that, unlike coarse-grain strategies, MS does not do redundant work and balances the load in a good way. Concerning to the coarse-grain strategies, CO+RF is the one with the best behaviour. Its migration policy reduces the load unbalance and the redundant work more than the remaining ones.

It is important to highlight that an effectiveness analysis showed that all the parallel algorithms are able to provide the same optimal solution for the complete set of problems and for

 ISBN 13: 978-84-613-5510-5

Table 2: Problems solved by the parallel strategies with a minimum number of processors Q.

Q	MS	CO	RF1	RF2	CO+RF
2	(21,5,2)	(21,5,2)	(21,5,2)	(21,5,2)	(21,5,2)
	(21,5,3)	(21,5,3)	(21,5,3)	(50,5,0)a	(21,5,3)
	(50,5,0)a,b	(50,5,0)a,b	(50,5,0)a	(50,5,2)a	(50,5,0)a,b
	(50,5,1)	(50,5,2)a	(50,5,2)a	(50,6,3)a,c	(50,5,1)
	(50,5,2)a,b	(50,6,3)a,c	(50,6,3)a,c		(50,5,2)a
	(50,6,3)a,b,c				(50,6,3)a,c
	(50,8,4)				(50,8,4)
	(100,2,0)				
	(100,2,1)				
	(100,10,0)				
4		(50,5,1)	(50,5,0)b	(21,5,3)	(50,5,2)b
		(50,5,2)b	(50,5,1)	(50,5,0)b	(50,6,3)b
		(50,6,3)b	(50,5,2)b	(50,5,1)	(100,10,0)
		(50,8,4)	(50,6,3)b	(50,5,2)b	
		(100,10,0)	(50,8,4)	(50,6,3)b	
				(50,8,4)	
8		(100,2,0)	(100,2,0)	(100,2,0)	(100,2,0)
		(100,2,1)	(100,2,1)	(100,2,1)	(100,2,1)
			(100,10,0)	(100,10,0)	

all the values of $P \geq Q$.

Table 3 shows average values of efficiency $Eff(Q,P)$ for all the parallel strategies. Notice that averages have been computed considering the sets of problems in Table 2. For example, the values for $Q = 2$ and MS strategy correspond to the 14 problems solved by MS with two processing elements, one master and one slave processor (see Table 2).

Notice that the efficiency values obtained by MS are very close to the ideal case, and they decrease when the number of processors P increases. The decrease in the efficiency may be due to the increase in the communications and to the small number of species in the list, which may mean that the computational load is not enough for distributing it among many processing elements. Coarse-grain strategies are farther from the ideal efficiency than the master-slave algorithm (see Table 3). This may be due to the presence of redundant work and to the load unbalance which increase the waiting time of the processors when a migration stage has to be accomplished (synchronization point).

Table 4 shows the average computing time for the 14 problems when they are solved by the parallel strategies using $P = 8, 16, 32$. It can be seen that MS strategy is twice faster than the coarse-grain strategies.

Therefore, from the results presented in this subsection, we can conclude that the MS strategy is the parallel model with the best performance.

 ISBN 13: 978-84-613-5510-5

Table 3: Average efficiency results for MS and CGs strategies.

Q	P	MS	CO	RF1	RF2	CO+RF
2	4	1.02	0.72	0.74	0.71	0.77
	8	0.98	0.55	0.61	0.51	0.61
	16	0.85	0.41	0.56	0.39	0.44
	32	0.81	0.34	0.46	0.32	0.40
4	8		0.71	0.82	0.83	0.77
	16		0.51	0.63	0.62	0.52
	32		0.42	0.57	0.53	0.45
8	16		0.71	0.74	0.78	0.72
	32		0.65	0.65	0.68	0.65

Table 4: Average execution time for problems solved using $P = 8, 16, 32$.

P	MS	CO	RF1	RF2	CO+RF
8	475.53	904.60	846.11	860.07	819.44
16	262.64	626.56	565.39	575.21	571.70
32	137.73	376.27	323.55	338.52	324.20

5.3 Efficiency results of MS for large problems

In this subsection the behavior of MS when the size of the problem increases is analyzed. To this aim, a representative set of location problems has been generated, varying the number n of demand points, the number m of existing facilities and the number k of those facilities belonging to the chain. The settings (n,m,k) employed in this experiment can be seen in Table 5.

Table 5: Settings of the larger test problems

n	100			150			200		
m	1	2	5	1	3	7	2	5	10
k	0	0, 1	0, 2	0	0, 1	0, 3	0, 1	0, 2	0, 5

For every setting in Table 5, five problems were generated by randomly choosing the parameters of the problems uniformly within the intervals presented in [11]. Furthermore, remember that all the instances are solved 5 times and the average values are considered.

Table 6 shows average results (for all the values of m and k) for each value of n and P. In the column labeled $Av(Obj)$, the average objective function value is given, in $Av(T)$ the average computational time and in the last column $Eff(P,Q)$, efficiency values are given.

As can be seen, the computational requirements increase with the complexity of the problems at hand. In fact, to solve problems with 200 demands points, the master-slave strategy

 ISBN 13: 978-84-613-5510-5

requires, at least, 8 processing elements. Nevertheless, the behavior of the parallel algorithm is excellent, since its efficiency is very close to the ideal case for all the instances.

Table 6: Efficiency results for big problems

n	P	$Av(Obj)$	$Av(T)$	$Eff(P,Q)$
100	2	472.66	2512.24	-
	4	472.66	1218.48	1.03
	8	472.67	580.96	1.08
	16	472.66	271.28	1.06
	32	472.66	152.44	1.03
150	4	646.90	2271.08	-
	8	646.90	1161.28	0.99
	16	646.90	582.28	0.98
	32	646.90	295.71	0.96
200	8	850.70	964.53	-
	16	850.70	474.53	1.02
	32	850.70	238.74	1.01

6 Conclusions and future research

In this study, a centroid (Stackelberg or Simpson) problem introduced in [11] is considered. In that paper, the evolutionary algorithm UEGO_cent.SASS proved to be the best technique, among four heuristics, for handling the problem. UEGO_cent.SASS was designed to reduce the computational burden and to be able to solve problems with reasonable sizes in a standard uniprocessors. Even so, the algorithm was only able to solve small instances.

In this paper, a *master-slave* and four *coarse-grain* methods are analysed as parallel versions of UEGO_cent.SASS. Since the parallel algorithms can use more computational resources, they can incorporate new techniques able to improve the search, and hence, the solutions obtained. An exhaustive analysis has proved that the $Creation\_species_2$ method can improve the objective value more than 1% in some instances. This method generates new candidate solutions at every level of the algorithm, using the remaining evaluations of the previous level.

Concerning the efficiency of the parallel algorithms, the master-slave method has a good behaviour. It is able to solve more instances than the coarse-grain proposals using fewer processing elements and to obtain efficiencies close to or even greater than the ideal one.

It has also been shown that the MS strategy outperforms the coarse-grain strategies in terms of efficiency and capability of solving large problems. This is because, for the current problem, it is not admissible neither the presence of redundant work, since the evaluation of a subpopulation requires intensive computational effort, nor the load unbalance, because of the processors are idle longer.

 ISBN 13: 978-84-613-5510-5

Acknowledgements

This work has been funded by grants from the Spanish Ministry of Science and Innovation (TIN2008-01117,ECO2008-00667/ECON) and Junta de Andalucía (P06-TIC-01426, P08-TIC-3518), in part financed by the European Regional Development Fund (ERDF).

References

[1] BHADURY, J., EISELT, H., AND JARAMILLO, J. An alternating heuristic for medianoid and centroid problems in the plane. *Computers and Operations Research 30*, 4 (2003), 553–565.

[2] DREZNER, T., AND DREZNER, Z. Facility location in anticipation of future competition. *Location Science 6*, 1 (1998), 155–173.

[3] DREZNER, T., AND DREZNER, Z. Retail facility location under changing market conditions. *IMA Journal of Management Mathematics 13*, 4 (2002), 283–302.

[4] DREZNER, Z. *Facility Location: A Survey of Applications and Methods*. Springer Series in Operations Research and Financial Engineering. Berlin, 1995.

[5] DREZNER, Z., AND HAMACHER, H. *Facility location. Applications and theory*. Springer, Berlin, 2002.

[6] EISELT, H., AND LAPORTE, G. Sequential location problems. *European Journal of Operational Research 96*, 2 (1997), 217–231.

[7] EISELT, H., LAPORTE, G., AND THISSE, J. Competitive location models: a framework and bibliography. *Transportation Science 27*, 1 (1993), 44–54.

[8] FRANCIS, R., MCGINNIS, L., AND WHITE, J. *Facility layout and location: an analytical approach*. Prentice Hall, Englewood Cliffs, 1992.

[9] HAKIMI, S. On locating new facilities in a competitive environment. *European Journal of Operational Research 12*, 1 (1983), 29–35.

[10] PLASTRIA, F. Static competitive facility location: an overview of optimisation approaches. *European Journal of Operational Research 129*, 3 (2001), 461–470.

[11] REDONDO, J., FERNÁNDEZ, J., GARCÍA, I., AND ORTIGOSA, P. Heuristics for the facility location and design $(1 \mid 1)$-centroid problem on the plane. *Computational Optimizations and Applications* (2007). To appear, DOI: 10.1007/s10589-008-9170-0.

[12] SÁIZ, M., HENDRIX, E., FERNÁNDEZ, J., AND PELEGRÍN, B. On a branch-and-bound approach for a Huff-like Stackelberg location problem. *OR Spectrum 31* (2009), 679–705.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Automatic code generation for GPUs in `llc`

Ruymán Reyes[1] **and Francisco de Sande**[1]

[1] *Dept. de Estadística, I. O. y Computación,*
Universidad de La Laguna, 38271–La Laguna, Spain

emails: `rreyes@ull.es`, `fsande@ull.es`

Abstract

`llc` is a language based on C where parallelism is expressed using compiler directives. In this work we present the new backend of the `llc` compiler that produces code for GPUs. Additionally we have implemented a software architecture that eases the development of new backends. This design represents an intermediate layer between a high level parallel language and different hardware architectures.

We evaluate our development by comparing the OpenMP and `llc` parallelizations of three different algorithms. In all cases, it is clear that the probable performance loss with respect to a direct CUDA implementation is clearly compensated by a significantly smaller development effort.

Key words: GPGPU, CUDA, llc, OpenMP, compiler, automatic parallelization

1 Introduction

The use of graphic processors (GPUs) [8] in High Performance Computing (HPC) has bursted recently in the market as a new alternative to exploit parallelism in many applications [15]. For some problems, general-purpose computing on graphics processing units (GPGPU) [1] is a low cost alternative that in many cases delivers results similar or even better to those achieved with traditional hardware. The GPGPU technology has a wider market than HPC, and therefore it has arrived to the HPC market to perdure.

From our point of view, the greatest inconvenience for the adoption of the GPGPU technology by the end users of HPC is the lack of programmability of the GPUs. Although languages like CUDA [13] or OpenCL [17] contribute to diminish the impact of this inconvenience, we believe that hard work has to be done in this direction.

In this work we tackle the problem of automatic code generation for GPUs from high level languages. `llc` is a parallel language [5, 6] where parallelism is expressed through the use of compiler directives that follow the `OpenMP` syntax. `llc` mixes the

 ISBN 13: 978-84-613-5510-5

OpenMP simplicity of use with the MPI portability and performance. `llCoMP`, the `llc` compiler, is a source to source compiler that translates C code annotated with `llc` directives into high-level parallel code. At this moment, `llCoMP` has two different backends: one producing hybrid MPI+OpenMP code and the new backend generating CUDA code.

The performance of the MPI and hybrid MPI+OpenMP code generated by `llCoMP` has been studied in previous works [5, 6, 16], while in this paper we focus our attention in the study of the new `llCoMP` backend. With our approach, the performance loss with respect to a direct CUDA implementation is clearly compensated by a significantly smaller development effort.

The remainder of the paper is organized as follows. In Section 2 we expose the motivation of our work. The main ideas behind the translation performed by `llCoMP` are discussed in section 3. In section 4 we discuss some of the optimizations currently implemented in the new backend of the compiler. The experimental evaluation of the translation produced by the compiler is presented in Section 5. Finally, we summarize a few concluding remarks and future work in Section 6.

2 Motivation

At this moment, HPC technology is living a time of fast changes. The end of the Gigahertz race has broadened the computer architectures capable to achieve high performance. This deep changes in the hardware world are immediately followed by the corresponding movements in the software layer. New tools and languages are clearly needed if we want to take advantage of the new hardware capabilities.

In the last decade we have seen a proliferation of HPC specific languages and tools, promoted from the governments [12] and also from the academical and business environments [7]. They all try to offer the maximum performance with the less programming effort. New languages hide architectural constrains, and provide a highly expressive syntax, that allows to express parallelism accurately.

It is well known that the introduction of a new language has two main inconveniences: HPC users needs to learn a new programming language, and they cannot reuse their previous codes without some effort. An alternative approach consists in the extension of a widely known language (usually C or Fortran) adding a minimum amount of constructs to exploit parallelism. One of the most successful cases of this approach is `OpenMP`. It was designed as a shared memory programming standard and has shown great performance for these systems.

Another effect of the changes in the hardware layer has been the increase in heterogeneity in the HPC architectures [3]. This new situation has partially left behind `OpenMP` and most other languages. Almost none of the current `OpenMP` implementations have in consideration heterogeneous systems, or even support for specific computational devices.

Each one of this additional computational devices has its own programming interface and model. If we consider FPGA as example we observe that despite its increasing

 ISBN 13: 978-84-613-5510-5

popularity, there is no common programming API for them, and the programmer need to develop specific code for each device and for each function that she wants to implement.

The OpenCL [9] standard represents an effort to create a common programming interface for heterogeneous devices, which many manufacturers have joined. However, it is still immature, and its programming model is not simple.

```
1  void compute(int np, int nd, double *box, vnd_t *pos, vnd_t *vel,
2            double mass, vnd_t *f, double *pot_p, double *kin_p) {
3    double x, d, pot, kin;
4    int i, j, k;
5    vnd_t rij;

7    pot = kin = 0.0;
8    #pragma omp parallel for default(shared)
9                 private(i, j, k, rij, d) reduction(+ : pot, kin)
10   #pragma llc result(f[i], nd)
11     for (i = 0; i < np; i++) {     /* Pot. energy and forces */
12       for (j = 0; j < nd; j++)
13         f[i][j] = 0.0;
14       for (j = 0; j < np; j++) {
15         if (i != j) {
16           d = dist(nd, box, pos[i], pos[j], rij);
17           pot = pot + 0.5 * v(d);
18           for (k = 0; k < nd; k++) {
19             f[i][k] = f[i][k] - rij[k] * dv(d) /d;
20           }
21         }
22       }
23       kin = kin + dotr8(nd, vel[i], vel[i]); /* kin. energy */
24     }
25     kin = kin * 0.5 * mass;
26     *pot_p = pot;
27     *kin_p = kin;
28   }
```

Listing 1: Molecular Dynamic code simulation in `llc`

CUDA [13] is a more mature and extended approach (although currently only supports NVIDIA devices). CUDA offers a programming interface (mostly C with a small set of extensions). This framework allows HPC users to re-implement their codes using GPU devices. Despite of being partially simple to build a code using this framework, it is hard to achieve a good performance rate, requiring a huge coding and optimization effort to obtain the maximum performance of the architecture.

and the experts that design and develop the languages as, in general, the users do not have the skills necessary to exploit the tools involved in the development of the parallel applications. Any effort to narrow the gap between users and tools by

 ISBN 13: 978-84-613-5510-5

providing higher level programming languages and increasing their simplicity of use is thus welcome.

In the last years, we have been working on a project that tries to combine simplicity in the user side with reasonable performance and portability. We expose to the HPC programmer a simple and well known language that hides the hardware complexity. On the other side, we present templates, representing most common parallel patterns, where we can introduce optimized versions without too much effort. The bridge is a software architecture, conformed by a powerful transformation tool. We believe that simplicity in the user side is a key aspect in the success of any parallel language. Our language has a syntax compatible with `OpenMP` where it is possible. The `llc` compiler uses the information present in the directives to produce the parallel code.

Given positions, masses and velocities of `np` particles, the routine shown in listing 1 computes the energy of the system and the forces on each particle. The code is an implementation in `llc` of a Molecular Dynamics (MD) simulation. The `parallel for` at line 8 is an example of *parallel construct*. It indicates that the iterations of the for loop at line 11 represents independent tasks, and consequently can be split among the available computing units. Different directives have been designed in `llc` to support common parallel constructs: *forall*, *sections*, and *pipelines* [6, 4]. The `llc` code is compiled by `llCoMP`, the `llc` compiler-translator, which produces an efficient high level parallel code.

As all `OpenMP` directives and clauses are recognized by `llCoMP`, we have three versions with the same code: sequential, OpenMP and `llc`/MPI, and we only need to choose the proper compiler to obtain the appropriate binary. Figure 1 illustrates this process.

In the next sections we present preliminary results and discuss the implementation of the new `llCoMP` backend (the rightest path in Figure 1.

3 The translation process

Although we had a working compiler based on `bison` and `yacc` that translates OpenMP/`llc` code to MPI, we decided to go back to the drawing board. In order to increase the flexibility when dealing with different target languages, at the time to develop the new CUDA backend for `llCoMP`, we decided to adopt an object oriented approach. After considering different alternatives, we have chosen to use Python as development language. Some of the strengths of this language that motivate our choice are:

- Its friendly and readable syntax allow us to write high level code faster than our previous approach
- The strong introspection capabilities of Python enhances our debugging processes
- Modularity is a key concept to reach a flexible and robust design

Reusing the code from the pycparser project [2], we have been able to build a C frontend supporting `OpenMP` in a short time, and our software architecture design allowed us to

 ISBN 13: 978-84-613-5510-5

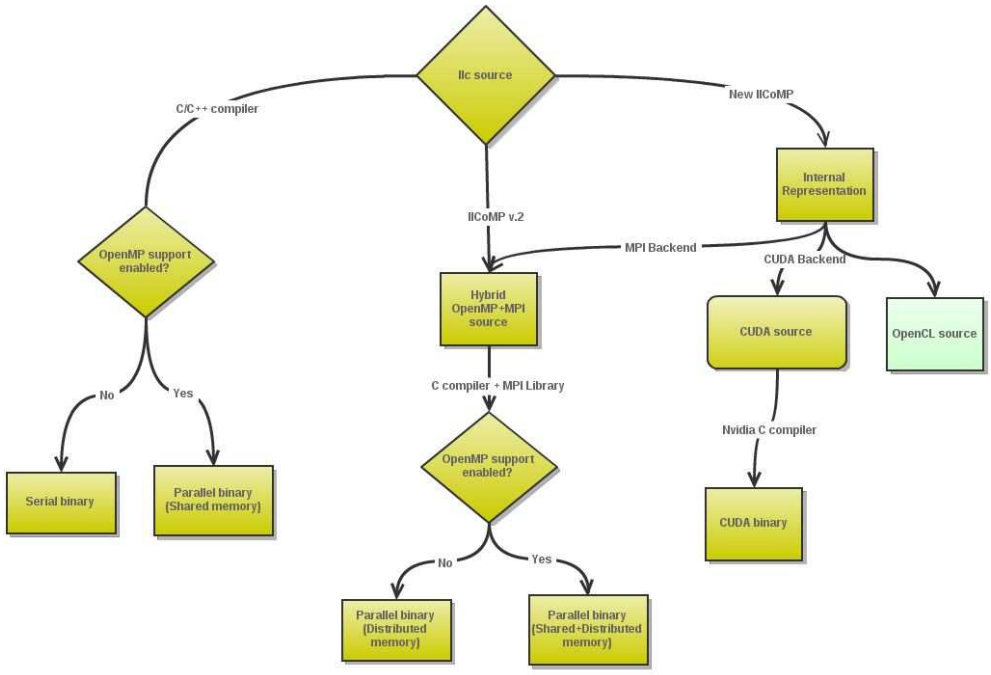


Figure 1: `llc` translation options.

write a translation system compound by a set of classes and methods, which encapsulate most of the work.

`llCoMP` starts translating the abstract syntax tree (AST) corresponding to the input source code to an internal representation (IR) based on a class hierarchy. Those parts of the IR corresponding to sequential code in the source are written in the target code with no previous transformation. The compiler searches in the AST for specific patterns using what we call a *Filter*. These patterns corresponds to different high-level parallel constructs. From the different parallel constructs available in the language, the CUDA back-end for now only supports parallel loops. The compiler has a filter class hierarchy that deals with this search task. Once a pattern is located in the AST, we can apply different mutators to achieve the desired translation. *Mutators* produce local modifications in the AST where they insert the high-level (CUDA) code corresponding to the desired translation. After all *Mutators* have been applied, the new AST is processed by the *CudaWriter* module to produce the target code. Figure 2 illustrate this process.

The code generation in `llCoMP` (like in the former version of the compiler) uses the *code pattern* concept. A *code pattern* is an abstraction that represents a specific task in the context of the translation. `llCoMP` uses two kind of code patterns: static and dynamic. The simplest code patterns are implemented using code templates, while the most complex cases require the implementation of a *Mutator*

 ISBN 13: 978-84-613-5510-5

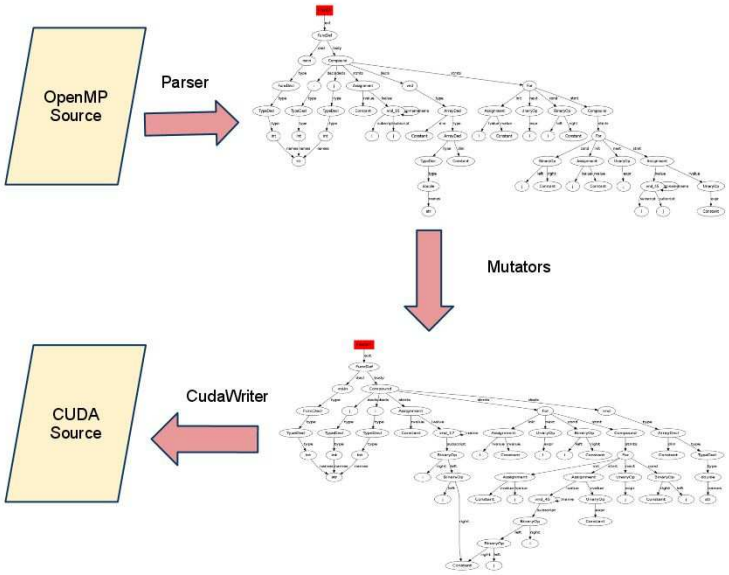


Figure 2: The translation process

A code template is a code fragment in the target language that will be modified accordingly to some input parameters. This code is interpreted and translated to the IR and afterwards it is grafted in the AST. The design of the backend using code templates will ease the implementation of new future backends.

Every time we need to use a device, we can identify several common tasks: initialization, local data allocation, device invocation, data retrieve and memory deallocation, among others. Each of these tasks identifies a pattern and each pattern is implemented through a code template. To manipulate these code templates and insert them in the IR `llCoMP` defines a set of operations that are collected in a library and exhibit a common facade.

4 Code optimizations

In our first approach to automatic code generation for CUDA, we have prevailed on simplicity, rather than focusing on code optimization. However, we have detected some situations where improvements in the target code will enhance the performance. We are currently working on the implementation of these improvements and some other complex optimizations that will be included in future releases of `llCoMP`.

One optimization already included in the current version of the translator is the use of a specialized kernel to perform reduction operations. With a small effort we have improved the performance of codes that make use of reductions.

Our first approach to make reductions in CUDA was based in the use of reduction

 ISBN 13: 978-84-613-5510-5

arrays that were communicated to the host to perform the operation. The specialized kernel implemented in the compiler [10] uses interleaved addressing and makes the first add during data fetching from global memory. This improvement benefits from using the device to perform the reduction and minimizes the size of the transfer between host and device. For the Mandelbrot set computation whose results are shown in section 5 we have increased the speedup in an average amount of 2%, and this figure will raise when combined with other planned optimizations.

Another key issue to enhance the performance in the CUDA architecture is the reduction of data transfer between host and device. In our PCI express $\times$ 16 bus this data transfer rate is 1.7 GB/s between CPU and GPU, and this constitutes a critical bottleneck.

In our strategy, at the end of each parallel loop we synchronize the memory of host and device (according to the OTOSP model underlying the `llc` implementation [5]). Lets consider the code in Listing 2 that is part of the implementation of the Jacobi iterative method both in `llc` and `OpenMP` taken from the `OpenMP` official website [14]. Without applying any optimization, the translation of the parallel loops in lines 6 and 10 to CUDA leads to communications between CPU and GPU at the beginning and at the end of each loop. However, this behavior introduces unnecessary communications since memory positions have not been modified between the end of the first loop and the beginning of the second.

```
1  while ((k < maxit) && (error > tol)) {
2    error = 0.0;
3    #pragma omp parallel shared(uold, u, ...)  private(i, j, resid)
4    {
5      #pragma omp for
6      for (i = 0; i < m; i++)
7        for (j = 0; j < n; j++)
8          uold[i][j] = u[i][j];
9      #pragma omp for reduction(+:error )
10     for (i = 0; i < (m - 2); i++) {
11       for (j = 0; j < (n - 2); j++) {
12         resid = ...
13        ...
14         error += resid * resid;
15       }
16     }
17   }
18   k++;
19   error = sqrt(error) / (double) (n * m);
20 }
```

Listing 2: Iterative loop in the Jacobi method implementation in `llc`/`OpenMP`

To avoid this overhead, our compiler injects the communications at the beginning and end of parallel regions. Inside the parallel region we assume that memory locations

 ISBN 13: 978-84-613-5510-5

allocated in the host remain unchanged. The programmer has to use the `OpenMP` `flush` construct in order to synchronize host and device in the case that access to variables computed in the device in a previous parallel loop is needed inside the parallel region. The insertion of the `flush` construct is not required in the case of function calls because they are automatically translated into device code.

Some of the optimizations that are currently under study or development are the following:

- To improve locality through a better use of the memory hierarchy
- To use the texture memory to store read-only data
- Enhance the translation of nested loops taking advantage of the architecture design
- An intelligent balance of load between host and device
- To implement interprocedural data flow analysis

5 Computational results

This work represents a preliminary evaluation of the results obtained with the new backend of the `llc` compiler. In order to establish the performance of the `llCoMP` translation we have used three algorithms: the Mandelbrot set computation, a Molecular Dynamic (MD) simulation and the solution of a finite difference equation using the Jacobi iterative method. All three source codes share the same implementation in `OpenMP` and `llc` (no specific `llc` annotations have been used). In all cases the Figures compare the speedup of the pure `OpenMP` implementation with 8 cores against the CUDA code generated by `llCoMP` using 64 threads. In all Figures, dark bars correspond to the CUDA code generated by `llCoMP` and light color correspond to the `OpenMP` implementation. The speedups are computed using exactly the same source code, for the `llCoMP` and `OpenMP` versions. The sequential code was obtained deactivating the `OpenMP` flags. The source code for all the examples is available at the `llc` project home page [11].

The computational experience has been carried out in a system build from two AMD Opteron QuadCore processors (8 cores) with 4 GB of RAM. This system has attached through a PCI-express 16x bus a Tesla C1060 card with 4 GB and 1 GPU with 240 cores.

Figure 3 shows the speedup obtained for three increasing problem sizes (number of points computed) in the Mandelbrot set area computation. While the `OpenMP` code do not increase the speedup when the problem size grows, the CUDA version benefits from it.

For the MD simulation code, the results are presented in Figure 4. In this case the size of the problem represents the number of particles involved in the simulation. For this test, the speedup of the CUDA code also grows with the problem size, while

 ISBN 13: 978-84-613-5510-5

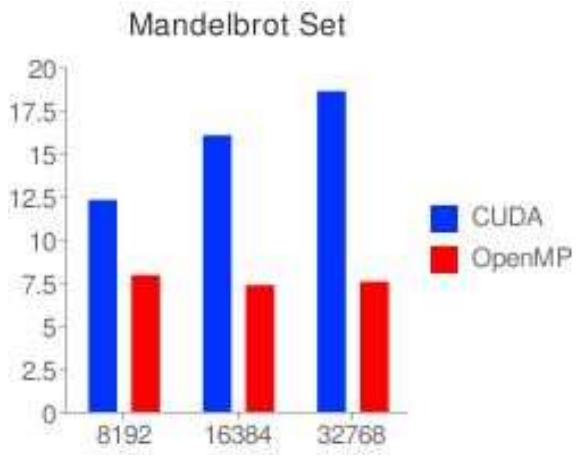


Figure 3: Speedup of the Mandelbrot set computation code

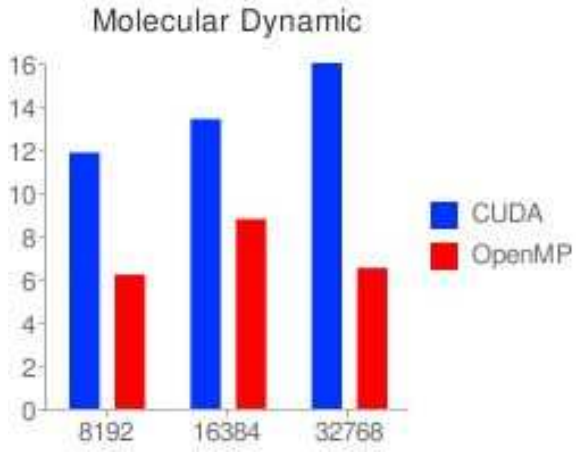


Figure 4: Speedup of the Molecular Dynamic simulation code

 ISBN 13: 978-84-613-5510-5

the OpenMP version do not show a regular behavior. This is probably due to memory constrains. The MD code has a predefined number of iterations (10 in our experiments) and each iteration involves calls to two different functions that have been parallelized independently. One of these two functions has a parallel loop similar to that seen in Listing 2, while the other function is a simple matrix update. This scenario is not the most beneficial for our approach since it produces an unnecessary amount of communications. Despite of this hindrance the CUDA results in Figure 4 are promising. An interprocedural data flow analysis of the source would enhance the performance of our strategy.

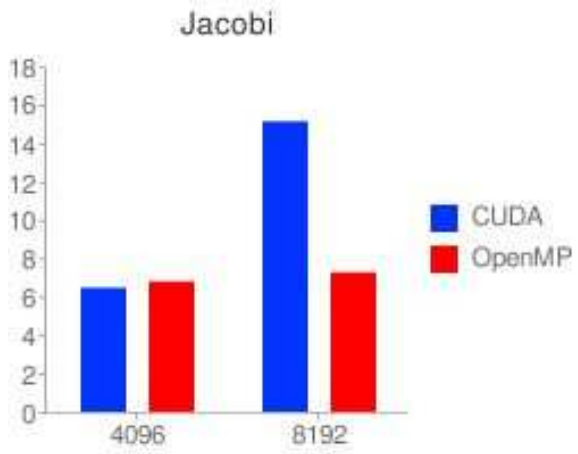


Figure 5: Speedup of the Jacobi iterative method

The iterative loop of the Jacobi method code was shown in Figure 2 and the corresponding results are presented in Figure 5. The size of the problem correspond to the dimension of the square matrices used in the computation. This code benefits from the optimization explained in Section 4 that minimizes the communications between host and device in the case of several parallel for loops inside the same parallel region. Again, while the OpenMP speedup remains almost constant when doubling the problem size, the CUDA implementation takes advantage of the largest amount of data involved.

6 Conclusions and future work

We have presented preliminary results obtained with the new implementation of the CUDA backend for the llc compiler. Taking into account the smaller effort to develop codes using llc compared with direct CUDA implementations, we conclude that llc is appropriate to implement some classes of parallel applications.

We have got the first version of a source to source compiler, written in a modern, flexible and portable language that represents a starting point for future works. We consider that the development of the new frontend of the compiler is a first milestone in our path to the future version of the llc language.

 ISBN 13: 978-84-613-5510-5

With the experience achieved in the elaboration of the CUDA backend, we believe that the incorporation of new target languages (OpenCL, for example) should not require an unaffordable effort. From now on, our goal is to continue the development of the language to increase its capabilities.

Work in progress concerning this topic includes the following:

- To increase the number of algorithms parallelized using our compiler, with particular attention to commercial applications
- To study and implement additional compiler optimizations that will enhance the performance of the target code
- To extend the `llc` syntax to capture additional information from the programmer for better adaption of the translation to the target architecture
- To study the generation of hybrid CUDA+OpenMP code

Acknowledgements

This work has been partially supported by the EU (FEDER), the Spanish MEC (Plan Nacional de I+D+I, contract TIN2008-06570-C04-03) and the Canary Islands Government (ACIISI, contract SolSubC200801000285)

References

[1] GPGPU, 2010. `http://www.gpgpu.org`.

[2] Eli Bendersky. Pycparse, 2010. `http://code.google.com/p/pycparser/`.

[3] André Rigland Brodtkorb, Christopher Dyken, Trond R. Hagen, Jon M. Hjelmervik, and Olaf O. Storaasli. State-of-the-art in heterogeneous computing. *Scientific Programming*, 18:1–33, 2010.

[4] A. J. Dorta, J. M. Badía, E. S. Quintana, and F. de Sande. Implementing OpenMP for clusters on top of MPI. In *Proc. of the 12th European PVM/MPI Users' Group Meeting*, volume 3666 of *LNCS*, pages 148–155, Sorrento, Italy, September 18–21 2005. Springer-Verlag.

[5] A. J. Dorta, J. A. González, C. Rodríguez, and F. de Sande. `llc`: A parallel skeletal language. *Parallel Processing Letters*, 13(3):437–448, September 2003.

[6] A. J. Dorta, P. López, and F. de Sande. Basic skeletons in `llc`. *Parallel Computing*, 32(7–8):491–506, September 2006.

[7] Tarek El-Ghazawi and Lauren Smith. UPC: unified parallel C. In *SC'06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, page 27, New York, NY, USA, 2006. ACM.

 ISBN 13: 978-84-613-5510-5

[8] Kayvon Fatahalian and Mike Houston. A closer look at GPUs. *Commun. ACM*, 51(10):50–57, 2008.

[9] Khronos Group. OpenCL the open standard for parallel programming of heterogeneous systems.

[10] M. Harris. Optimizing parallel reduction in cuda, 2007.

[11] `llc` Home Page. `http://llc.pcg.ull.es`.

[12] Ewing Lusk and Katherine Yelick. Languages for high-productivity computing: The DARPA HPCS language project. *Parallel Processing Letters*, 17(1):89–102, 2007. `doi:10.1142/S0129626407002892`.

[13] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with CUDA. *Queue*, 6(2):40–53, 2008.

[14] OpenMP Architecture Review Board. OpenMP official web site. `http://www.openmp.org/`.

[15] John D. Owens, David Luebke, Naga Govindaraju, Mark Harris, Jens Krüger, Aaron E. Lefohn, and Tim Purcell. A survey of general-purpose computation on graphics hardware. *Computer Graphics Forum*, 26(1):80–113, March 2007.

[16] R. Reyes, A. J. Dorta, F. Almeida, and F. Sande. Automatic hybrid MPI+OpenMP code generation with `llc`. In M. Ropo et al., editor, *Proc. of the 16th European PVM/MPI Users' Group Meeting*, volume 5759 of *Lecture Notes in Computer Science*, pages 185–195, Espoo, Finland, 2009. Springer-Verlag.

[17] Khronos Group Std. The OpenCL specification, version 1.0, online, 2009. `http://www.khronos.org/registry/cl/specs/opencl-1.0.48.pdf`.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Control of dengue disease: a case study in Cape Verde

Helena Sofia Rodrigues[1], M. Teresa T. Monteiro[2], Delfim F. M. Torres[3] and Alan Zinober[4]

[1] *School of Business Studies, Viana do Castelo Polytechnic Institute, Portugal*

[2] *Department of Production and Systems, University of Minho, Portugal*

[3] *Department of Mathematics, University of Aveiro, Portugal*

[4] *Department of Applied Mathematics, University of Sheffield, UK*

emails: sofiarodrigues@esce.ipvc.pt, tm@dps.uminho.pt, delfim@ua.pt, a.zinober@sheffield.ac.uk

Abstract

A model for the transmission of dengue disease is presented. It consists of eight mutually-exclusive compartments representing the human and vector dynamics. It also includes a control parameter (adulticide spray) in order to combat the mosquito. The model presents three possible equilibria: two disease-free equilibria (DFE) — where humans, with or without mosquitoes, live without the disease — and another endemic equilibrium (EE). In the literature it has been proved that a DFE is locally asymptotically stable, whenever a certain epidemiological threshold, known as the *basic reproduction number*, is less than one. We show that if a minimum level of insecticide is applied, then it is possible to maintain the basic reproduction number below unity. A case study, using data of the outbreak that occured in 2009 in Cape Verde, is presented.

Key words: dengue, basic reproduction number, stability, Cape Verde, control.
MSC 2000: 92B05, 93C95, 93D20.

1 Introduction

Dengue is a mosquito-borne infection that has become a major international public health concern. According to the World Health Organization, 50 to 100 million dengue infections occur yearly, including 500000 Dengue Haemorrhagic Fever cases and 22000 deaths, mostly among children [10]. Dengue is found in tropical and sub-tropical regions around the world, predominantly in urban and semi-urban areas.

 ISBN 13: 978-84-613-5510-5

There are two forms of dengue: Dengue Fever and Dengue Haemorrhagic Fever. The first one is characterized by a sudden fever without respiratory symptoms, accompanied by intense headaches and lasts between three and seven days. The second one has the previous symptoms but also nausea, vomiting, fainting due to low blood pressure and can lead to death in two or three days [3].

The spread of dengue is attributed to expanding geographic distribution of the four dengue viruses and their mosquito vectors, the most important of which is the predominantly urban species *Aedes aegypti*. The life cycle of a mosquito presents four distinct stages: egg, larva, pupa and adult. In the case of *Aedes aegypti* the first three stages take place in or near water while air is the medium for the adult stage [6]. The adult stage of the mosquito is considered to last an average of eleven days in the urban environment. Dengue is spread only by adult females, that require a blood meal for the development of eggs; male mosquitoes feed on nectar and other sources of sugar. In this process the female acquire the virus while feeding on the blood of an infected person. After virus incubation for eight to ten days, an infected mosquito is capable, during probing and blood feeding, of transmitting the virus for the rest of its life.

The organization of this paper is as follows. A mathematical model of the interaction between human and mosquito populations is presented in Section 2. Section 3 is concerned with the equilibria of the epidemiological model and their stability. In Section 4 the results obtained in the previous section are applied to a study case. Finally, some concluding notes are given in Section 5.

2 The mathematical model

Considering the work of [7], the relationship between humans and mosquitoes are now rather complex, taking into account the model presented in [4]. The novelty in this paper is the presence of the control parameter related to adult mosquito spray.

The notation used in our mathematical model includes four epidemiological states for humans:

$S_h(t)$	susceptible (individuals who can contract the disease)
$E_h(t)$	exposed (individuals who have been infected by the parasite but are not yet able to transmit to others)
$I_h(t)$	infected (individuals capable of transmitting the disease to others)
$R_h(t)$	resistant (individuals who have acquired immunity)

It is assumed that the total human population (N_h) is constant, so, $N_h = S_h + E_h + I_h + R_h$. There are also other four state variables related to the female mosquitoes (the male mosquitoes are not considered in this study because they do not bite humans and consequently they do not influence the dynamics of the disease):

$A_m(t)$	aquatic phase (that includes the egg, larva and pupa stages)
$S_m(t)$	susceptible (mosquitoes that are able to contract the disease)
$E_m(t)$	exposed (mosquitoes that are infected but are not yet able to transmit to humans)
$I_m(t)$	infected (mosquitoes capable of transmitting the disease to humans)

 ISBN 13: 978-84-613-5510-5

In order to analyze the effects of campaigns to combat the mosquito, there is also a control variable:

$c(t)$ level of insecticide campaigns

Some assumptions are made in this model:

- the total human population (N_h) is constant, which means that we do not consider births and deaths;
- there is no immigration of infected individuals to the human population;
- the population is homogeneous, which means that every individual of a compartment is homogenously mixed with the other individuals;
- the coefficient of transmission of the disease is fixed and do not vary seasonally;
- both human and mosquitoes are assumed to be born susceptible; there is no natural protection;
- for the mosquito there is no resistant phase, due to its short lifetime.

The parameters used in our model are:

N_h	total population
B	average daily biting (per day)
β_{mh}	transmission probability from I_m (per bite)
β_{hm}	transmission probability from I_h (per bite)
$1/\mu_h$	average lifespan of humans (in days)
$1/\eta_h$	mean viremic period (in days)
$1/\mu_m$	average lifespan of adult mosquitoes (in days)
μ_b	number of eggs at each deposit per capita (per day)
μ_A	natural mortality of larvae (per day)
η_A	maturation rate from larvae to adult (per day)
$1/\eta_m$	extrinsic incubation period (in days)
$1/\nu_h$	intrinsic incubation period (in days)
m	female mosquitoes per human
k	number of larvae per human
K	maximal capacity of larvae

The Dengue epidemic can be modelled by the following nonlinear time-varying state equations:

Human Population

$$\begin{cases} \frac{dS_h}{dt}(t) = \mu_h N_h - (B\beta_{mh}\frac{I_m}{N_h} + \mu_h)S_h \\ \frac{dE_h}{dt}(t) = B\beta_{mh}\frac{Im}{N_h}S_h - (\nu_h + \mu_h)E_h \\ \frac{dI_h}{dt}(t) = \nu_h E_h - (\eta_h + \mu_h)I_h \\ \frac{dR_h}{dt}(t) = \eta_h I_h - \mu_h R_h \end{cases} \quad (1)$$

 ISBN 13: 978-84-613-5510-5

and vector population

$$\begin{cases} \frac{dA_m}{dt}(t) = \mu_b(1-\frac{A_m}{K})(S_m+E_m+I_m) - (\eta_A+\mu_A)A_m \\ \frac{dS_m}{dt}(t) = -(B\beta_{hm}\frac{I_h}{N_h}+\mu_m)S_m + \eta_A A_m - cS_m \\ \frac{dE_m}{dt}(t) = B\beta_{hm}\frac{I_h}{N_h}S_m - (\mu_m+\eta_m)E_m - cE_m \\ \frac{dI_m}{dt}(t) = \eta_m E_m - \mu_m I_m - cI_m \end{cases} \tag{2}$$

with the initial conditions

$$\begin{array}{llll} S_h(0)=S_{h0}, & E_h(0)=E_{h0}, & I_h(0)=I_{h0}, & R_h(0)=R_{h0}, \\ A_m(0)=A_{m0}, & S_m(0)=S_{m0}, & E_m(0)=E_{m0}, & I_m(0)=I_{m0}. \end{array} \tag{3}$$

Notice that the equation related to the aquatic phase does not have the control variable c, because the adulticide does not produce effects in this stage of the life of the mosquito.

3 Equilibrium points and Stability

Let the set

$$\Omega = \{(S_h, E_h, I_h, A_m, S_m, E_m, I_m) \in \mathbb{R}^7_+ : S_h+E_h+I_h \leq N_h, A_m \leq kN_h, S_m+E_m+I_m \leq mN_h\}$$

be the region of biological interest, that is positively invariant under the flow induced by the differential system (1)–(2).

Proposition 1. *Let Ω be defined as above. Consider also*

$$\mathcal{M} = -\left(c(\eta_A+\mu_A)+\mu_A\mu_m+\eta_A(-\mu_b+\mu_m)\right).$$

The system (1)–(2) admits, at most, three equilibrium points:

- *if $\mathcal{M} \leq 0$, there is a Disease-Free Equilibrium (DFE), called Trivial Equilibrium, $E_1^* = (N_h, 0, 0, 0, 0, 0, 0)$;*
- *if $\mathcal{M} > 0$, there is a Biologically Realistic Disease-Free Equilibrium (BRDFE), $E_2^* = \left(N_h, 0, 0, \frac{kNh\mathcal{M}}{\eta_A\mu_b}, \frac{kNh\mathcal{M}}{\mu_b\mu_m}, 0, 0\right)$ or an Endemic Equilibrium (EE), $E_3^* = (S_h^*, E_h^*, I_h^*, A_m^*, S_m^*, E_m^*, I_m^*)$.*

It is necessary to determine the *basic reproduction number* of the disease, $\mathcal{R}_0$. This number is very important from the epidemiologistic point of view. It represents the expected number of secondary cases produced in a completed susceptible population, by a typical infected individual during its entire period of infectiousness [5]. Following [9], we prove:

Proposition 2. *If $\mathcal{M} > 0$, then the basic reproduction number associated to (1)–(2) is $\mathcal{R}_0^2 = \frac{B^2k\beta_{hm}\beta_{mh}\eta_m\nu_h\mathcal{M}}{\mu_b(\eta_h+\mu_h)(c+\mu_m)^2(c+\eta_m+\mu_m)(\mu_h+\nu_h)}$.*

BRDFE is locally asymptotically stable if $\mathcal{R}_0 < 1$ and unstable if $\mathcal{R}_0 > 1$.

 ISBN 13: 978-84-613-5510-5

From a biological point of view, it is desirable that humans and mosquitoes coexist without the disease reaching a level of endemicity. We claim that proper use of the control c can result in the basic reproduction number remaining below unity and, therefore, making BRDFE stable.

In order to make effective use of achievable insecticide control, and simultaneously to explain more easily to the competent authorities its effectiveness, we assume that c is constant.

We want to find c such that $\mathcal{R}_0 < 1$.

4 Dengue in Cape Verde

The simulations were carried out using the following values: $N_h = 480000$, $B = 1$, $\beta_{mh} = 0.375$, $\beta_{hm} = 0.375$, $\mu_h = 1/(71 * 365)$, $\eta_h = 1/3$, $\mu_m = 1/11$, $\mu_b = 6$, $\mu_A = 1/4$, $\eta_A = 0.08$, $\eta_m = 1/11$, $\nu_h = 1/4$, , $m = 6$, $k = 3$, $K = k * N_h$. The initial conditions for the problem were: $S_{h0} = m * N_h$, $E_{h0} = 216$, $I_{h0} = 434$, $R_{h0} = 0$, $A_{m0} = k * N_h$, $S_{m0} = m * N_h$, $E_{m0} = 0$, $I_{m0} = 0$. The final time was $t_f = 84$ days. The values related to humans describes the reality of an infected period in Cape Verde [1]. However, since it was the first outbreak that happened in the archipelago it was not possible to collect any data for the mosquito. Thus, for the *aedes Aegypti* we have selected information from Brazil where dengue is already a reality long known [8, 11].

Proposition 3. *Let us consider the parameters listed above and consider c as a constant. Then* $\mathcal{R}_0 < 1$ *if and only if* $c > 0.0837$.

For our computations let us consider $c = 0.084$. The results indicate that use of the control c is crucial to prevent that an outbreak could transform an epidemiological episode to an endemic disease. The computational experiences were carried out using Scilab [2].

Figures 1 and 2 show the curves related to human population, with and without control, respectively. The number of infected persons, even with small control, is much less than without any spray campaign.

The Figures 3 and 4 show the difference between a region with control and without control.

The number of infected mosquitoes is close to zero in a situation where control is present. Note that we do not intend to eradicate the mosquitoes but instead the number of infected mosquitoes.

5 Conclusions

It is very difficult to control or eliminate the *Aedes aegypti* mosquito because it makes adaptations to the environment and becomes resistant to natural phenomena (e.g. droughts) or human interventions (e.g. control measures).

During outbreaks emergency vector control measures can also include broad application of insecticides. It has been shown here that with a steady spray campaign it is

 ISBN 13: 978-84-613-5510-5

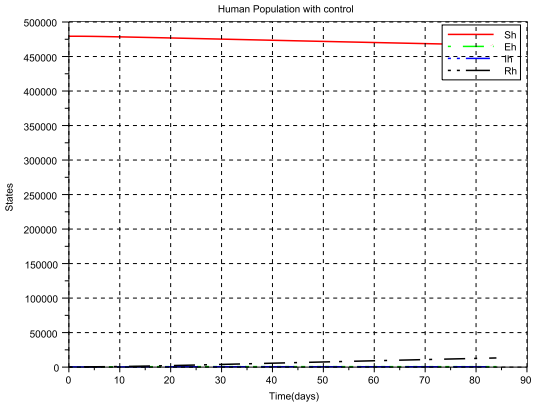


Figure 1: Human compartments using control

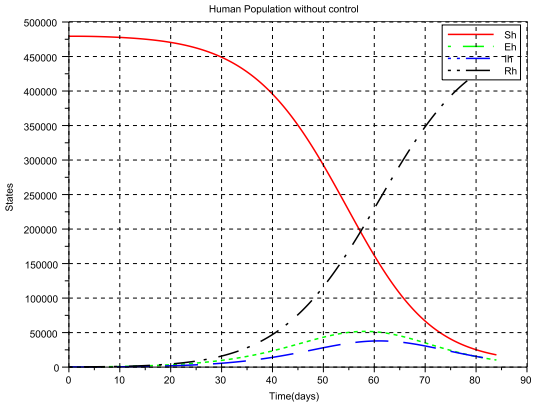


Figure 2: Human compartments with no control

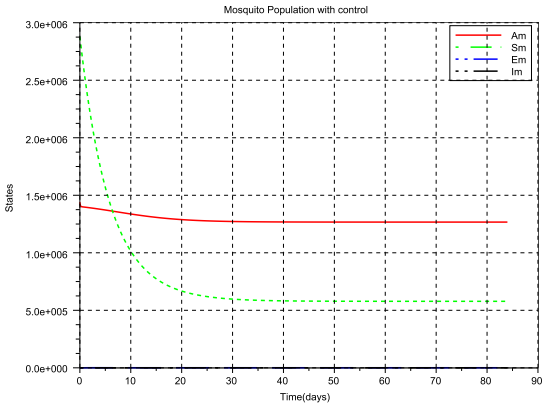


Figure 3: Mosquito compartments using control

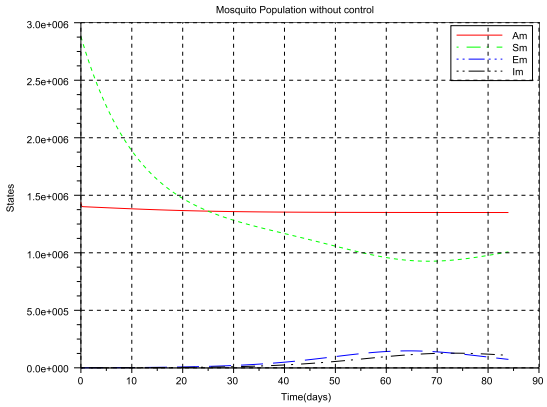


Figure 4: Mosquito compartments with no control

 ISBN 13: 978-84-613-5510-5

possible to reduce the number of infected humans and mosquitoes. Active monitoring and surveillance of the natural mosquito population should accompany control efforts to determine programme effectiveness.

Acknowledgements

Work partially supported by Portuguese Foundation for Science and Technology (FCT) through the PhD Grant SFRH/BD/33384/2008 (Rodrigues) and the R&D units Algoritmi (Monteiro) and CIDMA (Torres).

References

[1] *http://www.cdc.gov/dengue/*, April 2010.

[2] S. L. Campbell, J.-P. Chancelier, and R. Nikoukhah. *Modeling and simulation in Scilab/Scicos*. Springer, New York, 2006.

[3] M. Derouich, A. Boutayeb, and E. Twizell. A model of dengue fever. *BioMedical Engineering OnLine*, 2(1):4, 2003.

[4] Y. Dumont, F. Chiroleu, and C. Domerg. On a temporal model for the Chikungunya disease: modeling, theory and numerics. *Math. Biosci.*, 213(1):80–91, 2008.

[5] H. W. Hethcote. The mathematics of infectious diseases. *SIAM Rev.*, 42(4):599–653, 2000.

[6] M. Otero, N. Schweigmann, and H. G. Solari. A stochastic spatial dynamical model for Aedes aegypti. *Bull. Math. Biol.*, 70(5):1297–1325, 2008.

[7] H. S. Rodrigues, M. T. T. Monteiro, and D. F. M. Torres. Optimization of Dengue Epidemics: A Test Case with Different Discretization Schemes. In *Numerical Analysis and Applied Mathematics*, volume 1168 of *AIP Conference Proceedings*, pages 1385–1388. Amer. Inst. Physics, 2009.

[8] R. C. Thomé, H. M. Yang, and L. Esteva. Optimal control of Aedes aegypti mosquitoes by the sterile insect technique and insecticide. *Math. Biosci.*, 223(1):12–23, 2010.

[9] P. van den Driessche and J. Watmough. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Math. Biosci.*, 180:29–48, 2002.

[10] WHO. http://www.who.int/topics/dengue/en/, Apr 2010.

[11] H. Yang, M. Macoris, K. C. Galvani, M. T. M. Andrighett, and D. M. V. Wanderley. Assessing the effects of temperature on dengue transmission. *Epidemiol. Infect., Cambridge University Press*, 2009.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Contraction maps on spaces of partial functions endowed by the Baire quasi-metric and expoDC algorithms

S. Romaguera[1], P. Tirado[1] and O. Valero[2]

[1] *Instituto Universitario de Matemática Pura y Aplicada, Universidad Politécnica de Valencia, 46020 Valencia, Spain*

[2] *Departamento de Ciencias Matemáticas e Informática, Universidad de las Islas Baleares, 07122 Palma de Mallorca, Spain*

emails: sromague@mat.upv.es, pedtipe@mat.upv.es, o.valero@uib.es

Abstract

We deduce the existence of solution for the recurrence inequation associated to an expoDC algorithm by means of techniques of Denotational Semantics. This is done by means of a quasi-metric version of the Banach contraction principle on a space of partial functions endowed with a suitable adaptation of the Baire quasi-metric.

Key words: The Banach contraction principle, fixed point, Baire quasi-metric, recurrence, expoDC algorithm.

MSC 2000: 54E50, 54H25, 68Q25, 68Q55.

1 Introduction and preliminaries

We start with some pertinent concepts and facts on quasi-metric spaces.

The set of positive integer numbers will be denoted by $\mathbb{N}$ and the set of nonnegative integer numbers by ω.

Following the modern terminology, a quasi-metric on a set X is a function $d : X \times X \to [0, \infty)$ such that for all $x, y, z \in X$:(i) $d(x, y) = d(y, x) = 0 \Longleftrightarrow x = y$; (ii) $d(x, z) \leq d(x, y) + d(y, z)$.

If d is a quasi-metric on X, then the function $d^s : X \times X \to [0, \infty)$ given by $d^s(x, y) = \max\{d(x, y), d(y, x)\}$, is a metric on X. If d^s is a complete metric on X, we say that the quasi-metric d is bicomplete. In this case, (X, d) is said to be a bicomplete quasi-metric space.

 ISBN 13: 978-84-613-5510-5

By a contraction map on a quasi-metric space *(X, d)* we mean a self-map f of X such that $d(fx, fy) \leq kd(x, y)$ for all $x, y \in X$, where k is a constant with $0 < k < 1$. The number k is called a contraction constant for *f*.

It is clear that if f is a contraction map on a quasi-metric space (X, d) with contraction constant k, then f is a contraction map on the metric space (X, d^s) with contraction constant k.

Therefore, the classical Banach contraction principle can be generalized to the quasi-metric setting as follows (see for instance [3, Lemma 2.4])

Theorem 1. *Let f be a contraction map on a bicomplete quasi-metric space (X,d). Then, for each $x \in X$, the sequence of iterations $(f^n x)_{n\in\omega}$ is convergent in (X, d^s) to a point $x_0 \in X$ which is the unique fixed point of f.*

2 The results

The complexity quasi-metric space (introduced by M. Schellekens in [7]) provides an efficient tool to show, in a systematized way, the existence (and uniqueness) of solution for the recurrence equations or inequations typically associated to several distinguished kinds of algorithms [7, 4, 5, 2]. In particular, this approach was considered in [5] to the case of expoDC algorithm, which is carefully discussed in [1, Section 7.7], where the following recurrence inequation for this algorithm is obtained:

$$(1) \qquad T(m,n) \leq \begin{cases} 0, & \text{if } n = 1, \\ T(m, n/2) + M(mn/2, mn/2), & \text{if } n \text{ is even}, \\ T(m, n-1) + M(m, (n-1)m), & \text{otherwise}, \end{cases}$$

for all $(m, n) \in \mathbb{N} \times \mathbb{N}$.

According to [1, Section 7.7], $M(m, n)$ denotes the time needed to multiply two integers of sizes m and n, and $T(m, n)$ denotes the time spent multiplying when computing a^n, where m is the size of a.

Let $M(1, 1) = c > 0$. Then, it is constructed in the "complexity" quasi-metric space $(\mathcal{C}_{0,c}, d_{0,c})$, where

$$\mathcal{C}_{0,c} = \{f : \mathbb{N} \times \mathbb{N} \to [0, \infty) : f(m, 1) = 0, \text{ and } f(m, n) \geq c \text{ for } n > 1\},$$

and $d_{0,c}$ is the bicomplete quasi-metric on $\mathcal{C}_{0,c}$ given by

$$d_{0,c}(f, g) = \sum_{m=1}^{\infty} 2^{-m} \left[\sum_{n=2}^{\infty} 2^{-n} \max\{(\frac{1}{g(m,n)} - \frac{1}{f(m,n)}), 0\} \right]$$

The recurrence (1) induces, in a natural way, the functional Φ defined on $\mathcal{C}_{0,c}$ by

$$(2) \qquad \Phi f(m,n) = \begin{cases} 0, & \text{if } n = 1, \\ f(m, n/2) + M(mn/2, mn/2), & \text{if } n \text{ is even}, \\ f(m, n-1) + M(m, (n-1)m), & \text{otherwise}. \end{cases}$$

 ISBN 13: 978-84-613-5510-5

Then, it is proved in [5] that Φ is a contraction map on $(\mathcal{C}_{0,c}, d_{0,c})$ with contraction constant $3/4$, and thus Φ has a unique solution f_0 which is obviously a solution for T.

In Computer Science, programmers define procedures using recursion usually. In these cases, one must consider whether the mathematical specification for the procedure provides a semantically meaningless recursive definition. To discern it, the original problem is reduced to show that a fixed point equation has a solution, where such a solution is identified with the meaning of the denotational especification of the procedure. In many cases the aforementioned solution to the fixed equation is obtained as the limit of a sequence of successive approximations in such a way that each element of the sequence captures more (partial) information about the denotational meaning than the other computed before. In the described denotational analysis for high-level programming procedures the so-called partial functions have proven to be very useful. In fact, in many situations the mathematical specification of the procedure mathes up with a total function, defined recursively, which is at the same time a solution of the fixed point equation and the limit of a sequence of partial functions, also defined recursively, that capture partial information of the recursive specification meaning but they have the advantage that can be computed in a finite number of steps form their recursive specification contrary to the case of the total function (the meaning of the denotational specification). Motivated by the usefulness of partial functions in Denotational Semantics we here deduce the existence of solutions for the recurrence inequation (1) by means of techniques of Denotational Semantics based in proving that the functional Φ above is a contraction map, with contraction constant $1/2$, on a certain space of partial functions endowed by an appropriate bicomplete quasi-metric. Thus, our approach provides an improvement of the contraction constant with respect to one obtained from the complexity space $(\mathcal{C}_{0,c}, d_{0,c})$ and in addition the fact of working with partial functions yields a more visual application of our version of the Banach contraction principle because in practice one takes a partial function and its successive iterations to apply then Theorem 1.

In the sequel, given a nonempty alphabet Σ, we shall denote by Σ^∞ the domain of (finite and infinite) words over Σ, and by $\ell(x)$ we denote the length of the word x. The common prefix of x and y is denoted by $x \sqcap y$, and if x is a prefix of y we write $x \sqsubseteq y$.

The so-called Baire quasi-metric (see, for instance, [6]) is the quasi-metric $d_{\sqsubseteq}$ on Σ^∞ given by $d_{\sqsubseteq}(x, y) = 0$ if $x \sqsubseteq y$, and $d_{\sqsubseteq}(x, y) = 2^{-\ell(x \sqcap x)}$ otherwise.

Note that $(d_{\sqsubseteq})^s$ is the Baire metric on Σ^∞.

Now, put $\mathbb{N}_{\rightarrow} = \{\{1, ..., n\} : n \in \mathbb{N}\} \cup \mathbb{N}$, and $\mathcal{P} = \{f : \mathbb{N} \times B \to [0, \infty),\ B \in \mathbb{N}_{\rightarrow}\}$.

For each $f \in \mathcal{P}$ and each $m \in \mathbb{N}$, we define $f(m) : B \to [0, \infty)$ as $f(m)(n) = f(m, n)$ for all $n \in B$.

If B is finite then $f(m)$ is a partial function.

Note also that $f(m)$ can be considered as an element of Σ^∞ when $\Sigma = [0, \infty)$.

Moreover $\ell(f(l)) = \ell(f(m))$ for all $l, m \in \mathbb{N}$, and $\ell(f(m)) = \infty$ if and only if $B = \mathbb{N}$.

Next we define a function $d_{\mathcal{P}} : \mathcal{P} \times \mathcal{P} \to [0, \infty)$ by

$$d_{\mathcal{P}}(f, g) = \sup_{m \in \mathbb{N}} d_{\sqsubseteq}(f(m), g(m)),$$

 ISBN 13: 978-84-613-5510-5

It is straightforward to check that $d_{\mathcal{P}}$ is a quasi-metric on $\mathcal{P}$.

Furthermore, with the help of the well-known fact that $(d_{\sqsubseteq})^s$ is a complete metric, we can prove the following.

Theorem 2. *$d_{\mathcal{P}}$ is a bicomplete quasi-metric on $\mathcal{P}$.*

Proof. Let $(f_k)_k$ be a Cauchy sequence in $(\mathcal{P}, (d_{\mathcal{P}})^s)$. Then, for each $\varepsilon > 0$ there exists $k_\varepsilon \in \mathbb{N}$ such that $d_{\mathcal{P}}(f_j, f_k) < \varepsilon$ for all $j, k \geq k_\varepsilon$. Consequently, for each $m \in \mathbb{N}$, $(f_k(m))_k$ is a Cauchy sequence in the complete metric space $([0,\infty)^\infty, (d_{\sqsubseteq})^s)$ (recall that $(d_{\sqsubseteq})^s$ is the Baire metric on Σ^∞, where $\Sigma = [0,\infty)$), and thus there exists $f(m) \in \Sigma^\infty$ such that

$$\lim_k (d_{\sqsubseteq})^s(f(m), f_k(m)) = 0.$$

Observe that for each $l, m \in \mathbb{N}$, $\ell(f(l)) = \ell(f(m))$, because $\ell(f_k(l)) = \ell(f_k(m))$ for all $k, l, m \in \mathbb{N}$.

Let $B = \mathbb{N}$ if $\ell(f(m)) = \infty$, and $B = \{1, ..., p\}$ if $\ell(f(m)) = p$ for all $m \in \mathbb{N}$.

Then, we define $F : \mathbb{N} \times B \to [0,\infty)$ by $F(m,n) = f(m)(n)$ for all $m \in \mathbb{N}$ and $n \in B$. Clearly $F \in \mathcal{P}$ and $F(m) = f(m)$ for all $m \in \mathbb{N}$.

We shall show that

$$\lim_k (d_{\mathcal{P}})^s(F, f_k) = 0.$$

Indeed, choose $\varepsilon > 0$ and $m \in \mathbb{N}$. For each $k \geq k_\varepsilon$ there exists $j \geq k$ such that

$$(d_{\sqsubseteq})^s(f(m), f_j(m)) < \varepsilon,$$

so

$$\begin{aligned} (d_{\sqsubseteq})^s(f(m), f_k(m)) &\leq (d_{\sqsubseteq})^s(f(m), f_j(m)) + (d_{\sqsubseteq})^s(f_j(m), f_k(m)) \\ &< \varepsilon + (d_{\mathcal{P}})^s(f_j, f_k) < 2\varepsilon. \end{aligned}$$

Consequently, for each $k \geq k_\varepsilon$ we obtain

$$d_{\mathcal{P}}(F, f_k) = \sup_{m\in\mathbb{N}} d_{\sqsubseteq}(F(m), f_k(m)) = \sup_{m\in\mathbb{N}} d_{\sqsubseteq}(f(m), f_k(m)) \leq 2\varepsilon,$$

and, similarly,

$$d_{\mathcal{P}}(f_k, F) \leq 2\varepsilon.$$

We conclude that $d_{\mathcal{P}}$ is a bicomplete quasi-metric on $\mathcal{P}$.

The extension to $\mathcal{P}$ of the functional Φ defined above will be also denoted by Φ. Then we can show the following fact whose proof will be given in a full version of this paper:

For each $f, g \in \mathcal{P}$,

$$d_{\mathcal{P}}(\Phi f, \Phi g) \leq \frac{1}{2} d_{\mathcal{P}}(f, g),$$

and hence Φ has a unique fixed point f_0 which is solution of (1) (actually we have the equality in (1) for f_0).

 ISBN 13: 978-84-613-5510-5

Acknowledgements

The authors thank the support of the Spanish Ministry of Science and Innovation, grant MTM2009-12872-C02-01

References

[1] G. Brassard, P. Bratley, *Fundamentals of Algorithms*, Prentice Hall, 1996.

[2] L.M. García-Raffi, S. Romaguera and M. Schellekens, *Applications of the complexity space to the General Porbabilistic Divide and Conquer Algorithms*, J. Math. Anal. Appl. **348** (2008) 346–355.

[3] S.G. Matthews, *Partial metric topology*, in: Procedings 8th Summer Conference on General Topology and Applications, Ann. New York Acad. Sci. **728** (1994) 183–197.

[4] S. Romaguera and M. Schellekens, *Quasi-metric properties of complexity spaces*, Topology Appl. **98** (1999) 311–322.

[5] S. Romaguera, M. Schellekens, P. Tirado and O. Valero, *Contraction maps on complexity spaces and expoDC algorithms*, in: Proc. International Conference of Computational Methods in Sciences and Engineering ICCMSE 2007. AIP Conference Proceedings. **963** (2007) 1343–1346.

[6] J. Rodríguez-López, S. Romaguera, O. Valero, *Denotational semantics for programming languages, balanced quasi-metrics and fixed points*, International Journal of Computer Mathematics **85** (2008) 623–630.

[7] M. Schellekens, *The Smyth completion: a common foundation for denonational semantics and complexity analysis*, Electronic Notes Theoret. Comput. Sci. **1** (1995) 535–556.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Dynamic number of threads based on application performance and computational resources at run-time for interval branch and bound algorithms

J.F. Sanjuan-Estrada[1], L.G. Casado[1] and I. García[2]

[1] *Department of Computer Architecture and Electronics, University of Almería*

[2] *Department of Computer Architecture, University of Málaga*

emails: jsanjuan@ual.es, leo@ual.es, igarcia@ual.es

Abstract

This work investigates how to adapt the number of threads of a parallel Interval Branch and Bound algorithm based on its current performance. Basically, a thread can create a new thread that will process part of the ancestor workload. In this way, load balancing is inherent to creation of threads. The number of threads should depends on the level of parallelism, workload of the application, and on the computational resources of the architecture at a given time. The applications we are interested on use branch-and-bound algorithm which is highly irregular and therefore difficult to predict. In this way, the proposed methods can be used for more predictable algorithms as well. This research complement and does not substitute other devices that improve the exploitation of the system, like dynamic scheduling policy as work-stealing. Several approaches are presented. They differ in the used metrics and in the need or not to modify the Operating System (O.S.). The scenario for this research is just one multi-threaded application running with the O.S. in a multi-core architecture. Experimental results have been obtained for an Interval Global Optimization algorithm using POSIX-Threads running in multi-core systems with four and sixteen cores, and Linux kernel 2.6. Results show that a good number of running threads can be determined at run time, avoiding to statically establish the maximum number of threads of an application, as input parameter. The frequency with which thread creation decisions are made is very important because high frequency methods obtains better results but are time consuming. One of the presented methods does not need to establish the frequency of the decisions, obtaining the desired results.

Key words: Irregularity, Multi-threaded, Shared memory parallel processors, performance analysis, branch-and-bound, global optimization

MSC 2000: 65G20,68M20,68N25,68W10,90C26

 ISBN 13: 978-84-613-5510-5

1 Introduction

Branch and bound algorithms do an irregular and unpredictable search. These algorithms are among the most challenging to obtain a good speed up in parallel computers. A parallel version of interval branch and bound algorithm named LOCAL-PAMIGO, which has shown a good speed up in multicore systems, was presented in [1]. Local-PAMIGO uses dynamic thread creation at run time without explicit dynamic load balancing.

Some nowadays application programming interfaces (API), as OpenMP [9], TBB [11] or Cilk [4], can obtain a good performance for parallel loops and nested parallelism. For instance, OpenMP lets to use a dynamic number of threads with variable chunk size for loops. Recently, the optimal number of threads for each parallel loop in the parallel application is determined at run time based on dynamic feedbacks [7].

Previous APIs offer task parallelism, and schedule individual tasks to achieve good parallel performance for many applications. Currently, task scheduling is not exposed to the users. To reduce the run time system overhead and achieve good performance, the excessive creation of tasks has to be avoided. Cut-off for task parallelism works for many applications but does not perform well for parallel branch and bound codes [3, 8].

Local-PAMIGO approach adapts the number of threads at run time and performs well in multicore systems. Local-PAMIGO uses POSIX Thread NPTL library, instead of previous APIs, to have more control of the application execution. In Local-PAMIGO the user establishes statically the maximum number of running threads, as input parameter. In this work, we study how to adapt the number of threads to application performance and computational resources at run time, without user interaction. Recent studies show a possible direction: O.S. must change its interactions with the program run-time and parallel run-time systems must be developed that can automatically adapt programs to the architecture and usage environment [10].

Here, we present four methods that differ in the used metrics and in the modifications needed to be done in the kernel of the Operating System. Section 2 describes Local-PAMIGO schema we will compare with. Section 3 shows different strategies to decide the creation of a new thread. Section 4 shows results of the presented strategies and some conclusions are given in Section 5.

2 Parallel B&B model

B&B algorithms are well known exhaustive search methods where the initial problem is partitioned in subproblems until a solution is found. The method lets to avoid the search in some branches of the search tree. The idea of using the power of parallel computers in B&B algorithms is not new. Many researchers have devoted effort to obtain efficient parallel implementations of this general framework. Some references in the field can be found in [1]. Here we use a multithreading interval B&B algorithm running in multicore system (Local-PAMIGO), as testbed for our experimentation [1].

 ISBN 13: 978-84-613-5510-5

Following the classification presented in [5], Local-PAMIGO has a parallelism that can be classified as AMP (Asynchronous Multiple Pool).

The programming model used in Local-PAMIGO is based on threads processing their own data structures. Threads cooperate by updating information stored in shared variables. The user establish the number of process units (PUs) p to solve the problem. To avoid idle PU, a thread can generate a new thread assigning to it part of its own pending work. In this way, the application tries to maintain the p PUs busy and load balancing is inherent to the model. This model of programming multihtreaded algorithms can be extended to other type of algorithms. The use of skeletons can simplify programmer work [2]. More detailed characteristics of Local-PAMIGO programming model are the following:

- Each thread handles its own local data structures L (working storage) and Q (solution storage).
- NTh is a shared variable showing the number of running threads. The algorithm tries to keep $NTh = p$.
- VPU (Virtual PUs) is a shared vector with elements $VPU[i],\ i = 1, \ldots, p$, storing the identity of the thread assigned to slot i ($VPU[i].idthread$) and the solution storage ($VPU[i].Q$). Each thread i gets its local working storage L_i when the thread is created.
- Local-PAMIGO starts with one thread associated to $VPU[1]$, $L_1 = \{\text{Initial Problem}\}$, $VPU[1].Q = \{\}$ and $NTh = 1$.
- A thread assigned to $VPU[i]$ with L_i=$\{\}$ sets $NTh = NTh - 1$, its slot as free ($VPU[i].idthread$=-1) and terminates its execution. Notice that $VPU[i].Q$ may be nonempty.
- The condition $NTh < p$ is checked by all threads at every iteration of the algorithm. When a thread, for instance $VPU[i].idthread$, notices that $NTh < p$ and it has enough pending work in L_i, it searches a free VPU slot (for instance $VPU[j]$) and moves some pending work from L_i to L_j. Then, it generates a new thread and assigns it to $VPU[j]$ and finally sets $NTh = NTh + 1$.
- Some improvement could be achieved if the most loaded thread creates a new thread instead of the first thread detecting $NTh < p$.
- Local-PAMIGO terminates when $NTh = 0$. The solution set $\{VPU[i].Q,\ i = 1, \ldots, p\}$ contains the solution of the problem.

In this way, as soon as a virtual process unit is idle and there is enough work, a new thread is created. The time an idle PU is waiting for a new thread, is given by the time: (i) to check that $NTh < p$, (ii) to divide the work structure, (iii) to create a new thread and (iv) to migrate the new thread to the idle PU. The migration of threads is done by O.S.

Notice that Local-PAMIGO adapts the number of running threads based on the pending work in the application and on the number of idle virtual process units. Most of the time $NTh = p$ for large size problems, but $NTh < p$ can also happens due to the irregularity of B&B algorithm or small size problems. Actually, p determines the maximum number of threads that can be created. It is usually established statically

 ISBN 13: 978-84-613-5510-5

to the number of PUs in the system as input parameter. The optimum value of p, in terms of execution time of the application, will depends on the architecture, the level of parallelism of the algorithm, in the size of the problem, the ability of the programmer, etc. Table 1 shows the maximum number of running threads (MNRT) for SHCB problem with $\epsilon = 10^{-8}$ and Kowalik problem with $\epsilon = 10^{-3}$ (see [1]). Different values of p are used as input parameter. Notice that MNRT is smaller than p for low cost SHCB problem and large p values. Bold face numbers represents the values of MNRT that obtain the best execution times.

p		1	2	4	8	16	32	64	128
SHCB	MNRT	1	2	**4**	**8**	10	10	10	**9**
4 Cores	Time	0.10	0.05	0.03	0.03	0.05	0.04	0.04	0.03
Kowalik	MNRT	1	2	**4**	**8**	16	32	64	128
4 Cores	Time	683.73	364.64	170.98	170.92	173.16	196.2	210.79	215.22
SHCB	MNRT	1	2	**4**	**8**	**11**	**14**	**12**	**13**
16 Cores	Time	0.08	0.05	0.02	0.02	0.02	0.02	0.02	0.02
Kowalik	MNRT	1	2	4	8	**16**	32	64	128
16 Cores	Time	509.11	267.18	128	64.35	31.89	37.89	60.46	68.85

Table 1: Maximum number of running threads (MNRT) and execution time in seconds of Local-PAMIGO using different values of p in two, four and sixteen cores, systems.

Local-PAMIGO generates a new thread if there is an idle VPU. If p is too small, it can happen that the available parallelism of the algorithm and architecture is not exploited. On the other hand, if p is too large, even greater than the number of PUs in the system, we can found overhead due to parallelism and overhead in the scheduler of the O.S. to manage that number of threads.

Therefore, to guess a correct p value before algorithm execution is difficult and it is usually set to the number of PU. Next sections present different running time decisions to establish dynamically the number of running threads.

One can think that p can be equal to the number of idle PUs in the system at a given time. Well known Amdah's law shows that the speed-up of the parallel application is determined by the fraction of code that can be parallelized (more optimistic view can be found in [6]). For instance, if 50% of the running time can be parallelized and the other 50% is a critical section that can be executed just by one thread, then the maximum speedup is two. In such case, if the number of idle processor is greater than two and p is established to that number, the generation of new threads will not improve the execution time of the application. Therefore, other points of view have to be analyzed. Another important factor is the frequency with which decisions are made. It will be taken into account in the proposed decision methods presented below.

3 Decision methods

Every thread of the application request the possibility of generate a new thread in every iteration of the algorithm. The counters needed to evaluate a decision are reset a given

 ISBN 13: 978-84-613-5510-5

times. Some general definitions are needed.

Definition 1 *We define λ as the period of time, starting in previous reset time where decision calculations will not be done.*

Therefore, all requests in λ time will obtain a negative answer.

Definition 2 *We define τ as the time from the previous reset when calculations can be done, i.e, $\tau \geq \lambda$.*

Based on considerations presented in Section 2, four approaches to decide the creation of a new thread are described. Two of them are executed directly by the application and the other two are done by the kernel. Taken into account all previous considerations, a thread will generate a new one if:

- It has enough pending work.
- It is in τ time.
- The decision method return an positive answer.

To obtain a valuable decision, all methods presented here do the decision checking only if all threads do some work from previous checking. Following, we describe these methods.

3.1 ACW: Application decides based on completed work

This approach will increase the number of threads if the application performance, in terms of completed work per time and thread, does not decrease. More precisely we can define

$$P(n) = \frac{\sum_{i=1}^{n} NIter_i}{\tau \cdot n}, \tag{1}$$

as the performance of the application with n threads, where $NIter_i$ is the completed work per thread i in the last τ time.

Definition 3 *Decision based on completed work. A new thread is created when*

$$\frac{P(n)}{P(n-1)} > Threshold \tag{2}$$

It is important to measure $P(n)$ when all threads do some work (iterations) to obtain valuable information.

Notice that a $Threshold = 1$ means that the performance is maintained when a thread is created. $Threshold > 1$ when the performance is increased and $Threshold < 1$ when the generation of a new thread decreases the application performance. Each thread checks the thread creation decision after λ time by accessing to shared information with completed work in non-blocking schema. A positive decision reset all threads counters for next decision. The value of n decreases when a thread finishes its given work as was shown in Section 2. Therefore, a vector of size n with last $P(i),\ i = 1, \ldots, n$ values have to be stored as well.

 ISBN 13: 978-84-613-5510-5

3.2 AST-SF: Application decides based on sleeping threads by accessing to statistic files of the system

In high performance computing threads do mostly computational work. In shared memory multiprocessors with a SPMD (Single Program Multiple Data) programming model, as Local-PAMIGO algorithm, threads can be in sleeping state mainly due to simultaneous access to critical sections. The spent time in critical sections will determine the number of threads that does not increases the execution time of the application, given an infinite number of computational resources [12]. Therefore, it is important to know if a thread of the application has been in sleeping state. Nowadays operating systems, as Linux, store information of the current state of the kernel and running applications. In Linux Operating System, among other information, we can find the run-time and the state of a given thread in file *stat* of directory $/proc/[pid]/tasks/[pid_{thread}]/$. The *stat* file shows the current values for a given thread but does not store tracking information of previous values. Again, the time with which threads are not allow to access to all necessary system files with information about the application (λ value, Definition 1) is important. Smaller λ values reduce the chance to loose changes of state of the threads, at the price of increase the application and system computational time. Every time the *stat* file is read, the O.S. has to update all the information associated with the file.

Definition 4 *Decision based on sleeping thread. A new thread is created if all threads of the application have not been in sleeping state.*

Similarly to Definition 3, in Definition 4 all threads have to perform some work before checking.

To check the condition in Definition 4, current running time stored in *stat* file have to be compared with previous reading of that value. If there is a positive difference for each thread of the application and none thread sleeping state has been detected, a new thread can be created. Counters are reset after each decision checking. Depending on λ value, some sleeping states can be missed.

3.3 KST-SC: Kernel decides based on sleeping threads on demand by system calls

KST-SC uses Definition 4 decision model, but only the needed information is now processed by the O.S. A thread of the application uses a system call to request if a new thread creation is possible or not. In this way, two advantages are obtained. First, instead of thread sleeping state we can obtain now the thread sleeping time, because the system call runs at kernel level and can access to that information. Therefore, we can know if a thread was sleeping in the last τ time. Second, the amount of information evaluated from the kernel in each system call is only the needed one, i.e. running and sleeping time of threads. If all threads have a positive running time and zero sleeping time in last τ time, a positive answer is given. System calls done in less than λ time will generate automatically a negative answer (see Definition 1). Counters are reset by the Kernel after each decision checking, being this positive or negative. One drawback of this method and the following one is that the kernel has to be modified.

 ISBN 13: 978-84-613-5510-5

3.4 KITST-SC: Kernel idle threads decide based on sleeping threads on demand by system calls

In previous KST-SC method, all the necessary information has to be evaluated when the kernel receives a system call and decision has to be processed. All system calls return the possibility of thread creation or not.

In KITST-SC method, a system call performs an access to a kernel variable ($Create$) which determines the possibility of generate a new thread, instead of perform the checking of Definition 4. O.S. runs the so called *idle thread* in each processor when a processor gets idle. In this method, we modify the *idle thread* to perform Definition 4 checking and to update the previous kernel variable $Create$. A system call that found a *true* value in $Create$ set $Create$ to *false* and reset all counters associated to threads of the application. Therefore, *Idle thread* checks Definition 4 only if $Create = false$. If performed check is negative, *idle thread* reset all counters associated to threads of the application. Notice that time where checks are performed does not depends now on λ but on the existence or not of an idle processor. In this way, the following advantages are obtained:

- Computation to check the decision is done by an idle processor, i.e, when there exists computational resources. So, there are not time restrictions as λ.
- The existence of an idle processor motivates the thread creation because there exist available resources. On the other hand, when the system overload is high the thread creation usually does not increases the application performance and KITST-SC avoids checking computational costs and threads creation.
- The number of system calls done by the threads is usually large. In KITST-SC, system calls do not check Definition 4 and have low computational cost.

4 Results

Decision models presented in Section 3 have been evaluated using a modified version of Local-PAMIGO algorithm, which was described in Section 2. Thread with more workload generates a new one. In this way the number of created threads is smaller and they run more time. Only results for two problems are presented: SHCB with $\epsilon = 10^{-8}$ and Kowalik with $\epsilon = 10^{-3}$. These problems are small and medium size, respectively. The use of this small set of problems lets to avoid the presentation of large set of data. From our experience, other type of problems conclude in analogous results. Two computer architectures are used to run the algorithms:

QCore: One Quad-Core Intel Q6600, 2.40GHz and 4GB RAM.
Frida: Four Quad-Core AMD Opteron 8356, 2.30 GHz and 64GB RAM.

Both run Linux with 2.6 kernel. Numerical results of Local-PAMIGO, with different values of maximum number of threads (p), was shown in Table 1. These results are

 ISBN 13: 978-84-613-5510-5

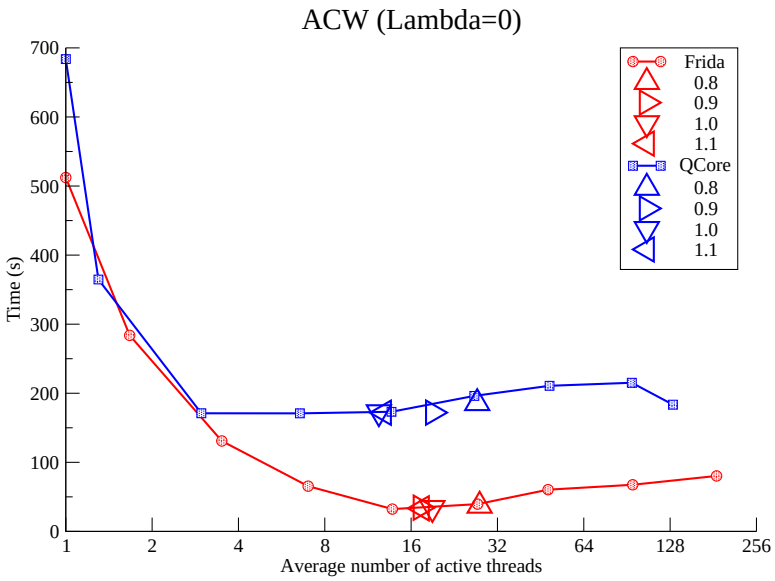


Figure 1: ACW with thresholds 0.8, 0.9, 1.0 and 1.1 for Kowalik function. $\lambda = 0$.

labelled in Figures 1 to 3 as QCore and Frida. These figures show the average number of running threads, which is smaller than p in Table 1.

Figure 1 also shows the execution time of the algorithm using ACW decision method, with $\lambda = 0$ and different $Threshold$ values for Kowalik function. The execution times and the average number of running threads obtained with $Threshold = 1$ are near to the best ones obtained in Table 1 for Qcore and Frida. $Threshold$ values around 1.0 are good. Higher values will generate few threads and lower values will generate a lot.

Figure 2 shows same experimentation of Figure 1 but fixing $Threshold = 1$ and using different values of λ. As expected, large λ values will generate few number of threads because the number of decisions is less. On the other hand, $\lambda = 0$ produces the maximum number of decisions and possible threads, and execution times are near the best ones for both architectures. From now, ACW experimentation will be done with $Threshold = 1$. Methods using λ are run with $\lambda = 0$.

Figure 3 compares all proposed methods for Kowalik function using both architectures. Following we describe all methods in ascending performance order:

AST-SF. AST-SF is the worst method in terms of average running threads and execution time. The main drawback of this method is the need to read a number of files equal to the number of running threads. In each read, the O.S. update the files with more information than needed. For $\lambda = 0$ the number of checks is the largest, but even in this case the number of threads is large. This means that this method misses some threads sleeping states.

ACW. In ACW method, the average number of active threads is slightly greater than the best one obtained in Table 1 with the corresponding increasing running time.

 ISBN 13: 978-84-613-5510-5

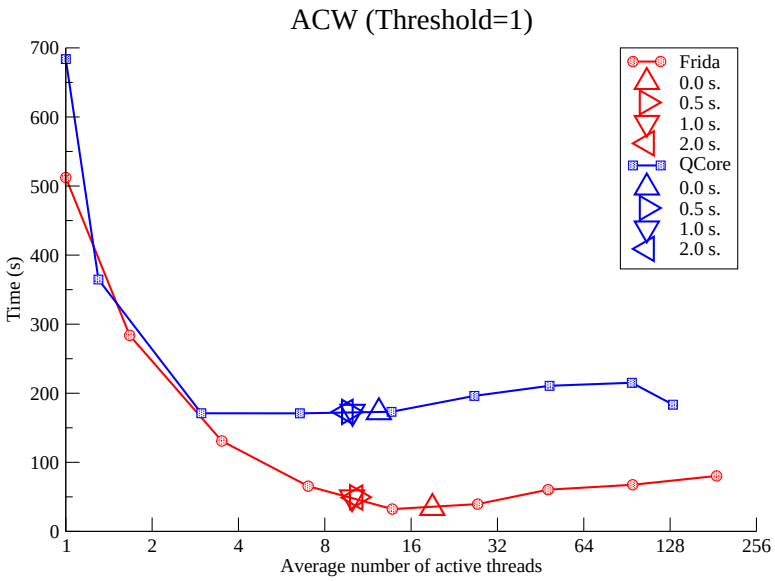


Figure 2: ACW with Threshold 1.0 for Kowalik function. λ values: 0.0, 0.5, 1.0 and 2.0 seconds.

The main advantage of ACW is that it is independent of the operating system and the architecture.

KST-SC. Comparing ACW with AST-SF we can think that completed work decision (see Definition 3) is better than sleeping thread one (see Definition 4). KST-SC obtain similar results to best ones in Table 1 using sleeping thread decision. This is achieved because the kernel only computes the necessary data in each decision. Additionally the information about sleeping threads is precise, which does not happened in AST-SF.

KITST-SC. This is the best of the four presented methods. The execution time is similar to best results obtained in Table 1, using less number of average running threads. This is because system calls of KITST-SC consume less time and decisions are taken by idle processors, independently of λ.

Figure 4 shows the changes in the number of running threads for each method in Frida and Kowalik function. Right hand side graph is a zoom of the initial stages of left hand side graph. It can be seen how Local-PAMIGO with $p = 16$ quickly increases the number of threads to 16. As soon a thread finishes other thread is created. Results for Local-PAMIGO (Frida) are those obtained with the best p value. Local-PAMIGO obtains the best execution time, because decision are based only in the number of running threads and pending work (see Section 2). Therefore, thread creation does not take into account the application performance on the system. Proposed methods have additional overhead in their decision checks and their execution time also depends on the taken decisions. We can observe this in right hand graph of Figure 4 where the different startup times of the methods are shown. These results verify the performance

 ISBN 13: 978-84-613-5510-5

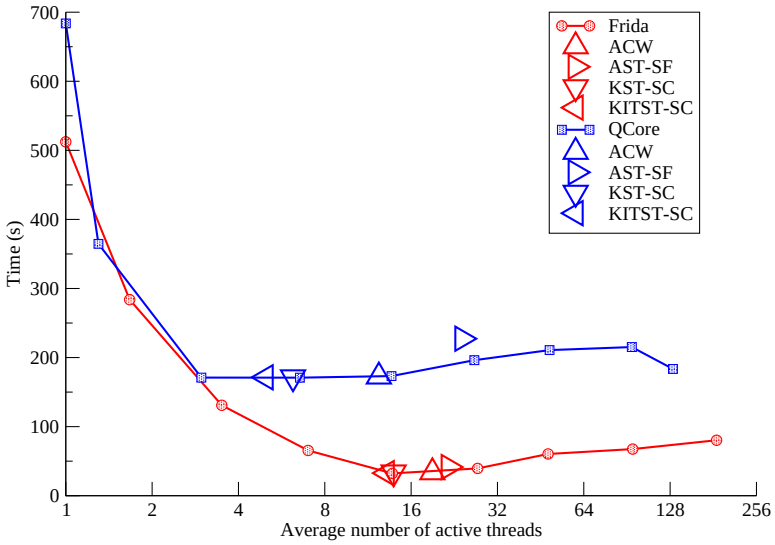


Figure 3: Comparative of proposed methods for Kowalik function.

order presented above.

The same information of Figure 4 is shown in Figure 5 for low cost SHCB problem. Left hand side graph shows application level methods and right hand side graph shows kernel level methods. File access time in AST-SF is now more representative. ACW decisions obtains the best results but near to execution times of methods at kernel level.

5 Conclusions and future work

Two methods to decide the number of active threads at run time are presented. One is based on completed work and the other in the existence of a sleeping thread in the application. Four models are evaluated. Two of them evaluate the decisions at application level and the others two do only a sleeping thread decision at kernel level. Decisions at kernel level outperform application level ones because they have better information of the system. On the other hand they need O.S. modifications. Application level decision based on completed work is easier to develop and outperform application level decision based on sleeping thread due to the cost to get the needed information from the system information files.

These methods have been used on an interval branch and bound algorithm. This type of algorithms are difficult to parallelize due to their irregularity. The proposed methods get a linear speedup and adapts the parallelism level to the achieved performance and the system resources at run time

As future work we investigate the use of combined decision methods for several application running at the same time in the system.

 ISBN 13: 978-84-613-5510-5

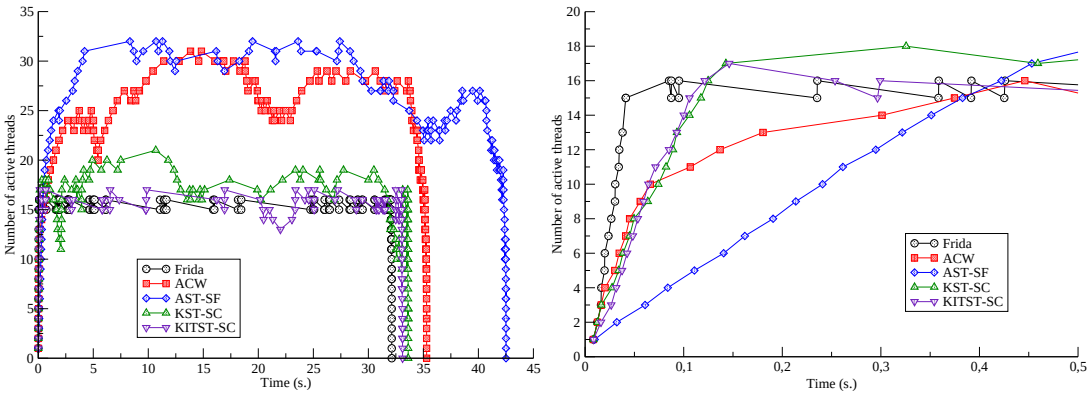


Figure 4: Number of active threads in time for Kowalik function in Frida.

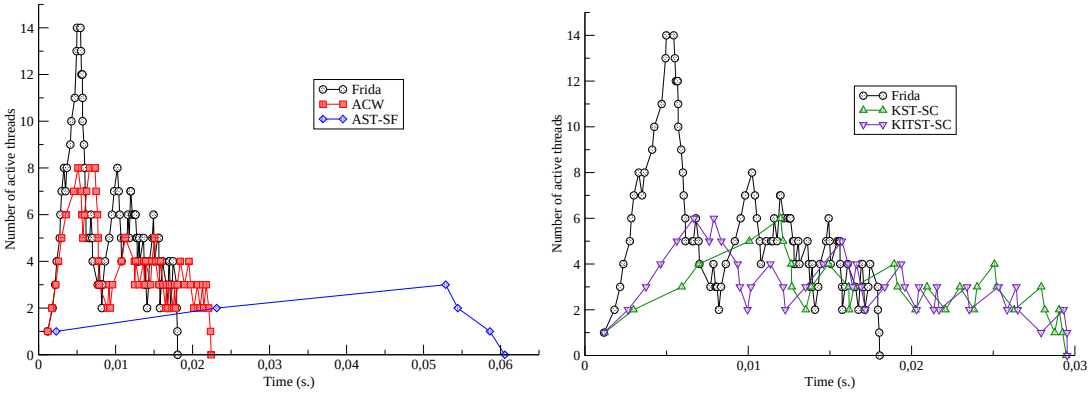


Figure 5: Number active threads in time for SHCB function in Frida.

Acknowledgements

This work has been partially funded by grants from the Spanish Ministry of Science and Innovation (TIN2008-01117) and Junta de Andalucía (P08-TIC-3518).

References

[1] L.G. Casado, J.A. Martínez, I García, and E.M.T. Hendrix. Branch-and-bound interval global optimization on shared memory multiprocessors. *Optimization Methods and Software*, 23(3):689–701, 2008.

[2] Murray Cole. Bringing skeletons out of the closet: a pragmatic manifesto for skeletal parallel programming. *Parallel Comput.*, 30(3):389–406, 2004.

[3] Alejandro Duran, Julita Corbalán, and Eduard Ayguadé. An adaptive cut-off for task parallelism. In *SC '08: Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, pages 1–11, Piscataway, NJ, USA, 2008. IEEE Press.

 ISBN 13: 978-84-613-5510-5

[4] Matteo Frigo, Charles E. Leiserson, and Keith H. Randall. The implementation of the Cilk-5 multithreaded language. In *PLDI '98: Proceedings of the ACM SIGPLAN 1998 conference on Programming language design and implementation*, pages 212–223, New York, NY, USA, 1998. ACM.

[5] Bernard Gendron and Teodor Gabriel Crainic. Parallel branch-and-bound algorithms: Survey and synthesis. *Operations Research*, 42(6):1042–1066, 1994.

[6] John L. Gustafson. Reevaluating amdahl's law. *Commun. ACM*, 31(5):532–533, 1988.

[7] Jaejin Lee, Jung-Ho Park, Honggyu Kim, Changhee Jung, Daeseob Lim, and SangYong Han. Adaptive execution techniques of parallel programs for multiprocessors. *J. Parallel Distrib. Comput.*, 70(5):467–480, 2010.

[8] Stephen L. Olivier and Jan F. Prins. Evaluating OpenMP 3.0 run time systems on unbalanced task graphs. In *IWOMP '09: Proceedings of the 5th International Workshop on OpenMP*, pages 63–78, Berlin, Heidelberg, 2009. Springer-Verlag.

[9] OpenMP Architecture Review Board. *OpenMP Application Program Interface, version 3.0.* 2008.

[10] David A. Penry. Multicore diversity: a software developer's nightmare. *SIGOPS Oper. Syst. Rev.*, 43(2):100–101, 2009.

[11] James Reinders. *Intel Threading Building Blocks*. O'Reilly, 2007.

[12] M. Aater Suleman, Moinuddin K. Qureshi, and Yale N. Patt. Feedback-driven threading: power-efficient and high-performance execution of multi-threaded workloads on cmps. In *ASPLOS XIII: Proceedings of the 13th international conference on Architectural support for programming languages and operating systems*, pages 277–286, New York, NY, USA, 2008. ACM.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Certain uncertainties in population biology revisited

Nico Stollenwerk[1], Maíra Aguiar[1], Sebastien Ballesteros[1] and Bob W. Kooi[2]

[1] *Centro de Matemática e Aplicações Fundamentais, Universidade de Lisboa, Portugal*

[2] *Faculty of Earth and Life Sciences, Department of Theoretical Biology, Vrije Universiteit Amsterdam, The Netherlands*

emails: nico@ptmat.fc.ul.pt, maira@ptmat.fc.ul.pt, sebastien@ptmat.fc.ul.pt, bob.kooi@falw.vu.nl

Abstract

In population biology, namely epidemiology and ecology, models and its parameters are much more uncertain than we are used to in physics or chemistry, for example. We give some situations from where these uncertainties orignate, and investigate case studies, mainly in epidemiology, but easily transferable to ecology and also populations of neurons, hence the wast field of brain sciences.

Key words: parameter estimation, confidence intervals, epidemic models, deterministic chaos, coexisting attractors and noise, influenza, measles, dengue fever

1 Introduction

Based on recent research experience, we demonstrate various sources of uncertainty in modelling and parameter estimation in population biological systems. Simplest models of epidemiological processes already show complex behaviour due to their inherent nonlinear structure. Crititical fluctuations are observed in the simples possible epidemiological system, the susceptible-infected-susceptible system (SIS). A simpe susceptible-infected-recovered (SIR or SIRS) system with seasanal forcing, as common in epidemiological and ecological systems, can exhibit deterministic chaos, even in simplest mean field approximation of real world stochastic systems. Multi-strain epidemiological models of SIR-type even without seasonal forcing already can give rise to deterministically chaotic attractors in wide parameter regions.

This catalogue should give a strong signal that complex behaviour should be expected in such population biological systems. But the story goes further: Not including critical fluctuations nor chaos, in parameter estimation correlations between seemingly

 ISBN 13: 978-84-613-5510-5

simple obvious parameters can give rise to large uncertainties of parameters, as we will see in the next section. Also the debate of deterministic chaos versus noisy fluctuations becomes more intriguing in situations in which only the noisy transitions of structures observed in deterministic models are a relevant descritpion of the system's behaviour, namely the hopping between co-existing attractors, including deterministically chaotic ones. A case study from measles epidemiology will illustrate the case. The two lessons to be learned from the previous argumentations might not be enough to understand slightly more complicated population biological systems of e.g. multi-strain models (in ecology, or multi-species systems in ecology). This will be demonstrated in the case study of dengue fever epidemiology. In all cases we are dealing with still very simple and basic models, no external varabilities like population growth, or age structure or other more involved complications are considered. Such models already show very complex features, either in our knowledge available from real world systems or due to intrinsic dynamical behaviour. The present considerations should facilitate the understanding of such complex features in still very simple descriptions of real world population dyanamic systems.

2 Influenza models and parameter estimation

One of the basic epidemic process is the susceptible, infected, recovered (SIR) epidemic, in which susceptible individuals S become infected on contact with already infected I with infection rate β and recover with rate γ into the R class. Eventually, the recovered and immune R can become susceptible again with rate α. For outbreaks in seasonal influenza this model can serve a first description The SIR epidemic is given by the reaction scheme

$$\begin{aligned} S + I &\xrightarrow{\beta} I + I \\ I &\xrightarrow{\gamma} R \\ R &\xrightarrow{\alpha} S \end{aligned} \tag{1}$$

giving the following master equation (stochastic Markov process in continuous time) for fixed population size N, hence $N = S + I + R$.

The master equation for the SIR system with $R = N - S - I$, hence we only need to consider the probability of S and I, and R follows from this, is given by

$$\begin{aligned} \frac{d}{dt}p(S,I,t) &= \frac{\beta}{N}(S+1)(I-1))\; p(S+1,I-1,t) + \gamma(I+1)\; p(S,I+1,t) \\ &\quad +\alpha(R+1)\; p(S-1,I,t) \\ &\quad -\left(\frac{\beta}{N}SI + \gamma I + \alpha R\right)\; p(S,I,t) \end{aligned} \tag{2}$$

with $(R+1) = N-(S-1)-I$. In order to solve the master equation we can use generating functions or charateristic functions, obtaining an eventually easier solvable partial differential equation (PDE) [1, 2] which explicitly can be solved in some simplifications.

 ISBN 13: 978-84-613-5510-5

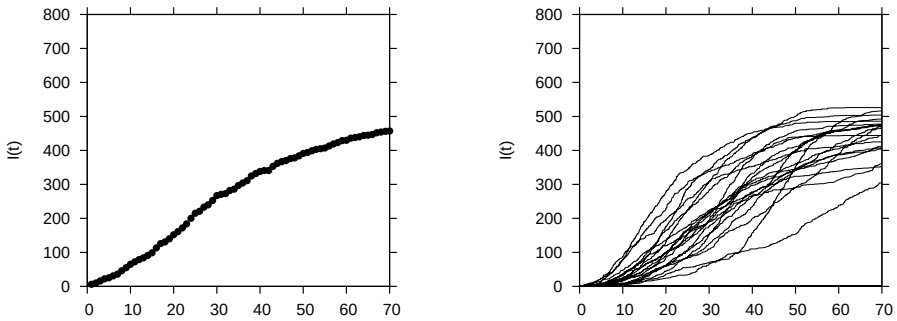


Figure 1: a) Cumulative data for influenza for the 2007 season reported in the InfluenzaNet project for the Netherlands. b) Simulations of a simple SIR system in a parameter region close to the data. The comparison between a) and b) gives the η-ball estimated likelihood of the parameters varied in the simulations in a neighbourhood of the data.

In order to apply the full SIR model to parameter estimation in data obtained from an internet based surveillance system, called InfluenzaNet [3], see Fig. 1 a), we better compare with stochastic simulations of the above given master equation (3), see Fig. 1 b). Changing parameters we can obtain an estimation for the likelihood of the model parameters involved, the η-ball method [4]. To these model parameters to be estimated, also the initial conditions, e.g. S_0, have to be considered.

The η-ball method has been tested successfully in situations where also the analytic likelihood is know [4], hence reflects reliably the likelihood for the respective parameters. In the present application we investigate the joint likelihood of the infection rate β and the recovery rate γ, see Fig. 2 a), which shows strong correlations along the diagonal of the β-γ plane. These might still be captured to some extend by linear stochastic

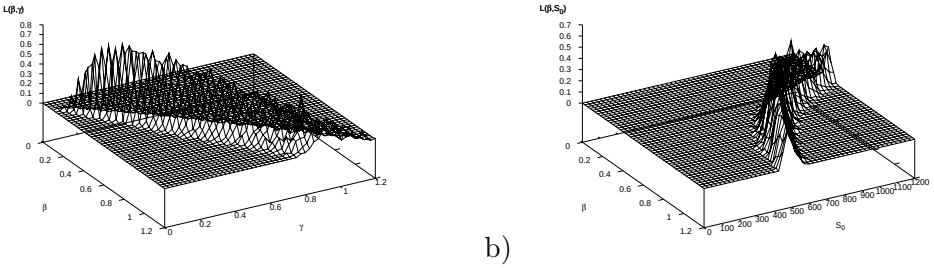


Figure 2: a) Estimated likelihood for parameters β and γ of a simple SIR model, using the η-ball method. b) Estimated likelihood for parameters β and S_0. In both cases the maximum of the joint likelihood for the two parameters involved each are ill defined, i.e. wide spread over the possible parameter ranges.

 ISBN 13: 978-84-613-5510-5

methods, but surely not the correlations in the joint likelihood of the infection rate β and the initial susceptibles S_0, a quantity which is important in influenza modelling reflecting the novelty of the present seasonal strain to the host population. The joint likelihood for β and S_0 is shown in Fig. 2 b). From this Fig. 2 b) it becomes clear that the data do not reflect well if the underlying dynamics has low infection rate and high numbers of initially susceptibles or vice versa, a common situation in parameter estimation. In seasonal influenza one has to consider seasonal forcing of the contact rate and in biologically reasonable parameter regions quickly observes bifurcations and wide parameter ranges of deterministically chaotic attractors, making parameter estimation additionally difficult. Also the replacement of influenza strains can be captured by considering reinfection as an additional process [5] giving new qualitative features to be explored further in data analysis. Here lessons from another disease modelling can help to understand the complexity to be expected, childhood diseases, and here mainly measles, see next section.

3 Coexistence of attractors and noise in measles

In childhood disease modelling classical data sets from pre-vaccination times have been extendedly investigated, especially the famous New York measles monthly reporting time series between 1928 and 1963. From this a simple two dimensional map, hereafter called measles map, can be derived from a radial basis function approximation to the data maxima return map in logarithmic scale, capturing the essential dynamics of the epidemiological system [6]. The measles map can have the following explicit form (but other analytic forms with similar shape also give qualitatively the same result)

$$\begin{aligned} x_{n+1} &= a x_n e^{-h \cdot x_n} \cdot (y_n + c)^{-g} + b \\ y_{n+1} &= x_n \end{aligned}$$

with parameters obtained from the radial basis function fit $a = 410$, $b = 135$, $c = 600$, $g = 0.327$, $h = 0.000684$. Here the parameter a controls the strength of the nonlinearity. An analysis of the bifurcation diagram shows for a wide range of parameter a the coexistence of a chaotic attractor and a stable period three attractor. Around the empirically obtained parameter value of $a = 410$, however, the chaotic attractor becomes unstable and the period three attractor is the only remaining stable attractor. For the state space plot of $a = 400$ see Fig. 3.

Also shown in Fig. 3 is another period three solution, which however is unstable. The unstable manifold of this unstable period three leads to the chaotic attractor in one direction or to the stable period three in the other direction. The stable manifold of this unstable period three gives the basin boundary between these two co-existing attractors. Computationally, this stable manifold can easily obtained by iterating the inverse map of the measles map

$$\begin{aligned} x_{n+1} &= y_n \\ y_{n+1} &= \left(\frac{x_n - b}{a \cdot y_n} \right)^{-\frac{1}{g}} e^{-\frac{h}{g} \cdot y_n} - c \end{aligned}$$

 ISBN 13: 978-84-613-5510-5

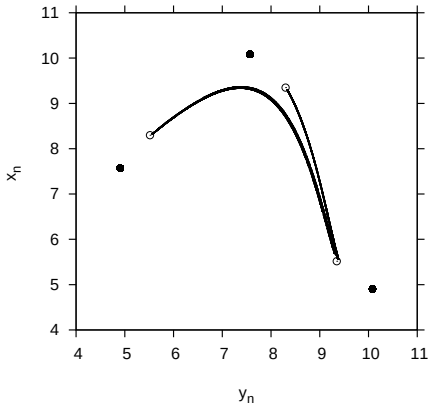


Figure 3: The state space plot of the measles maxima return map shows for $a = 400$ the co-existence of a chaotic attractor with a stable period three (full dots). the key to understanding the dynamics is the unstable period three (empty circles), which at this parameter set just touch the chaotic attractor but also mark the basin of attraction boundary between the chaotic attractor and the stable period three.

with initial conditions close to one point of the unstable period three. For $a = 410$, the empirical value, however, the basin boundary has cut into the chaotic attractor, making it unstable, a chaotic saddle or semi-attractor, see Fig. 4 a).

Now dynamic noise can drive the system from around the stable period three back to the basin of attraction of the semi-attractor, staying on the skeleton of this semi-attractor for a while before captured back around the stable period three, see Fig. 4 b). This attractor hopping is a subtil interplay between deterministic chaotic dynamics and dynamic noise. This scenario was also later found in the original Dietz ODE-model for measles.

Hence, depending on the strength of the dynamic noise the signal of a system

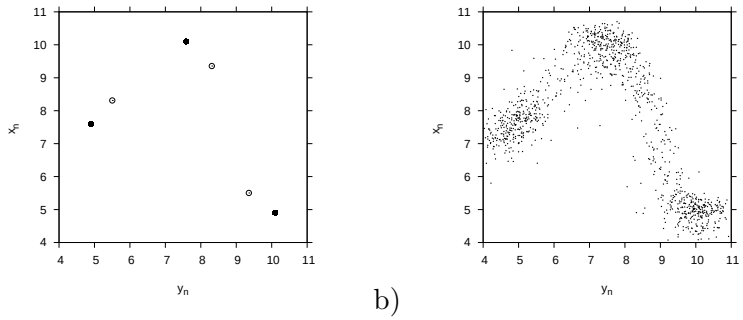
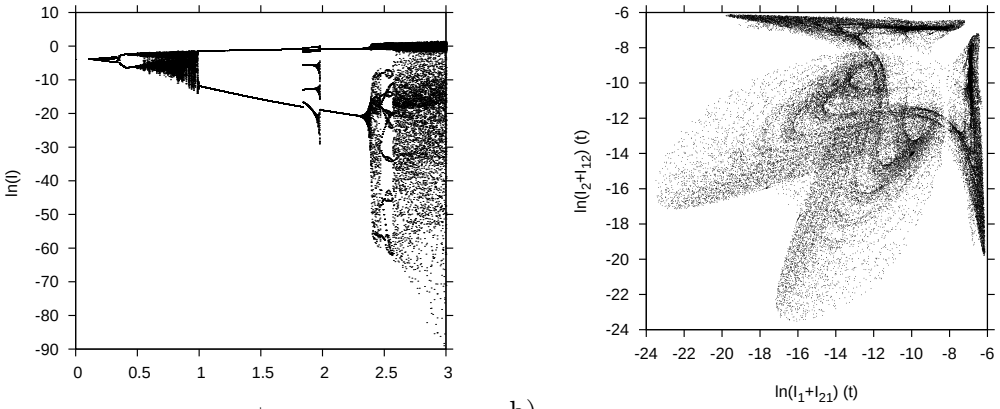


Figure 4: a) State space plot for $a = 410$. The stable period three is now the only attractor of the system, since the basin boundary of attraction has cut into the previously visible chaotic attractor. b) But noisy trajectories still reflect the skeleton of the previously existing chaotic attractor in addition to the noisy period three signal.

 ISBN 13: 978-84-613-5510-5

b



a) b)

Figure 5: a) Bifurcation diagram for the multi-strain dengue model with antibody dependence parameter ϕ in the range between zero and three. Including temporary cross immunity, for values smaller than $\phi = 1$ we already find a rich dynamic behaviour including chaotic attractors, wheras for $\phi > 1$ initially a stable period two orbit is the only attractor. b) Poincaré section of the chaotic attractor for $\phi = 0.95$.

before and after an attractor crisis, as here observed around $a = 400$ to $a = 410$ can be more similar than the deterministic bifurcation diagram would suggest. Parameter estimation would have difficulties to clearly observe at which side of the attractor crisis the system is. This theme will be investigated further in the next section for a recently studied multi-strain model for dengue fever epidemiology.

4 Antibody dependent enhancement versus hospitalization in dengue

Essential features of dengue fever epidemiology with its peculiar antibody dependent enhancement (ADE) and difference in contribution to force of infection of the secondary infection, labeled ϕ, as opposed to the first are captured by a simple extension of the SIR model to a two strain model. Including temporary cross-immunity, this model shows in wide parameter ranges bifurcations and deterministically chaotic attractors [7, 8, 9]. For the explicit model description see the references above.

The bifurcation diagram Fig. 5 a) shows deterministic chaos not only for ϕ much larger than one, as previous models without temporary cross-immunity suggested, but also for ϕ smaller than one. A Poincaré section of the deterministically chaotic attractor with positive Lyapunov exponent is shown in Fig. 5 b). This is a two dimensional graphic projection of the nine dimensional attractor. An attractor crisis happens just before $\phi = 1$, the parameter point which would show no difference of the contribution to the force of infection between primary and secondary infection. After $\phi = 1$ for some parameter range only a periodic solution is observed.

 ISBN 13: 978-84-613-5510-5

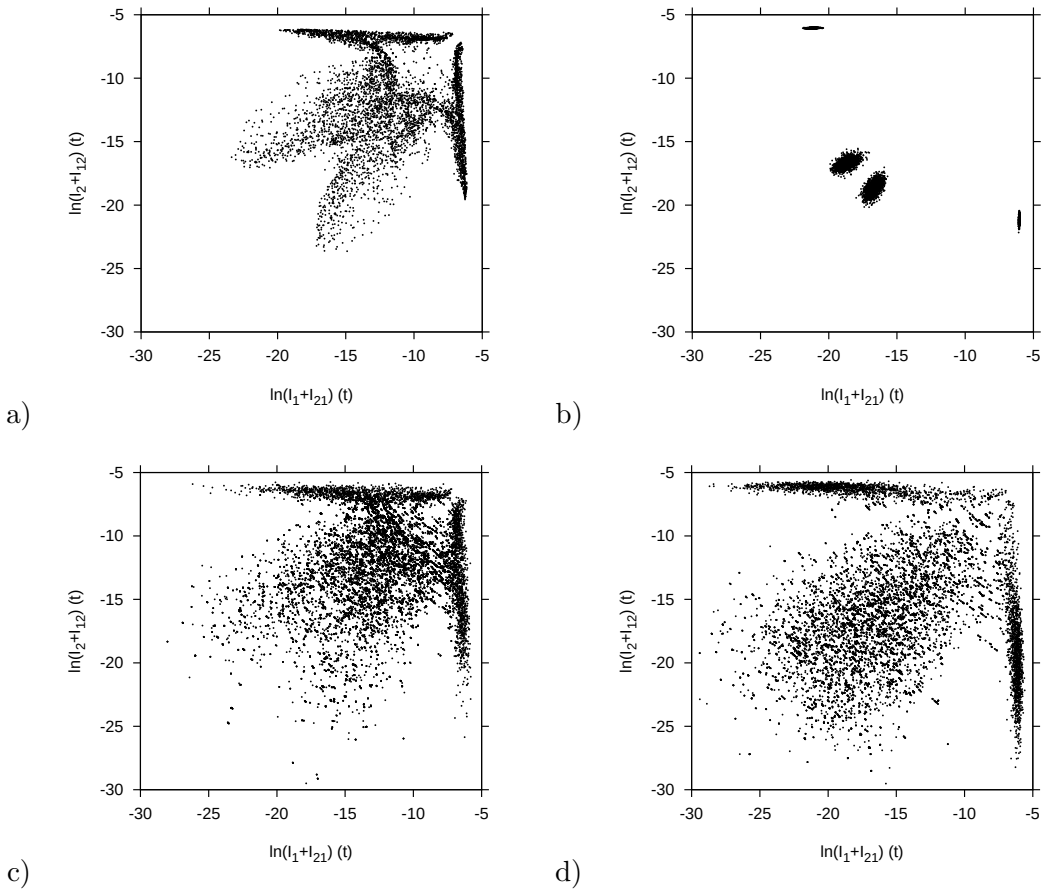


Figure 6: Noisy attractors are presented, characterized by their Poincaré sections for $\phi = 0.95$ in a) and c) and for $\phi = 1.05$ in b) and d). In the first row a) and b) we use very small dynamic noise, whereas in c) and d) slightly larger dynamic noise is applied. With this noise level the dynamic structures appear very similar for $\phi < 1$ and for $\phi > 1$, whereas for smaller noise clearly the chaotic attractor respectively the very different periodic attractors are observed.

However, including dynamic noise (in the simplest state dependent form, see [10]) in the ODE model shows that for small but not too small noise the attractors in the region $\phi < 1$ and $\phi > 1$ are qualitatively not very different, see Fig. 6 c) and d), whereas for very small noise the original noiseless attractors, for $\phi < 1$ the chaotic attractor and for $\phi > 1$ the periodic one, shine through the noise cloud.

Hence, unless data analysis would clearly show a ϕ-value far away from one, for the qualitative dynamical behaviour the chaotic attractor skeleton appearing for $\phi < 1$ would hold into the region of $\phi > 1$. First data inspection of time series especially from Thailand indicate the fingerprint of a chaotic behaviour rather than periodic behaviour under small noise. Further formal data analysis is needed in future research.

The debate on the biological implications for a long time suggested that antibody dependent enhancement would increase the contribution to force of infection of sec-

 ISBN 13: 978-84-613-5510-5

ondary dengue fever cases due to higher viral load. Without temporary cross-immunity an about three times higher viral load transmitted would be needed to explain the fluctuations seen in empirical data. However, hospitalization of dengue haemorhagic fever cases in secondary infection and other complications would rather suggest that such secondary infected do not contribute too much to the force of infection, eventually even less than secondary cases. Anyhow, due to the just described effects it might be quite difficult to obtain from epidemiological data a clear answer unless the effects are dramatic, i.e. far away from the neutral case of same contribution to force of infection of primary and secondary cases.

5 Conclusions and future research

In conclusion, we have seen that already quite simple epidemiological processes due to their non-linear structure present a rich dynamic behaviour, including subtil interplay between deterministic chaos and dynamic noise, adding to the difficulties of non-linear parameter dependence, not to speak of additional features in spatially extended stochastic systems as for example spatially restriced networks under superdiffusion as appropriate for epidemiological spreading [11]. Due to the structural similarity, e.g. the SIS epidemics has its correpondence in the Pearl–Verhulst model in ecology [10], similar effects will be expected in other population biological systems.

Acknowledgements

We would like to thank Friedhelm Drepper, Jülich, Lewi Stone, Tel Aviv, Sander van Noort and Gabriela Gomes, Oeiras, Frank Hilker, Bath, and João Boto and Jaime Combadão, Lisbon, for fruitfull discussions on the presented topics, and Luis Sanchez, Lisbon, and Ezio Venturino, Turin, for scientific support. This work has been supported by the European Union under FP7 in the EPIWORK project and by FCT, Portugal.

References

[1] S. van Noort and N. Stollenwerk, *From dynamical processes to likelihood functions: an epidemiological application to influenza*, Proceedings of 8th Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2008, ISBN 978-84-612-1982-7 (2008).

[2] S. van Noort, N. Stollenwerk and L. Stone, "Analytic likelihood function for data analysis in the starting phase of an influenza outbreak", *Proceedings of 9th Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2009*, ISBN 978-84-612-9727-6, edited by Jesus Vigo Aguiar *et al.*, Salamanca, 2009, pp. 1072–1080.

[3] InfluenzaNet, http://www.influenzanet.com/.

[4] N. Stollenwerk and K. M. Briggs, *Master equation solution of a plant disease model*, Physics Letters **A 274** (2000) 84–91.

[5] Stollenwerk, N., van Noort, S., Martins, J., Aguiar, M., Hilker, F., Pinto, A., and Gomes, G., *A spatially stochastic epidemic model with partial immunization shows in mean field approximation the reinfection threshold*, accepted for publication in Journal of Biological Dynamics (2010).

[6] Drepper, F.R., Engbert, R., and Stollenwerk, N. *Nonlinear time series analysis of empirical population dynamics*, Ecological Modelling **75/76** (1994) 171–181.

[7] M. Aguiar and N. Stollenwerk, *A new chaotic attractor in a basic multistrain epidemiological model with temporary cross-immunity*, arXiv:0704.3174v1 [nlin.CD] (2007) (accessible electronically at http://arxive.org).

[8] M. Aguiar, B.W. Kooi and N. Stollenwerk, *Epidemiology of dengue fever: A model with temporary cross-immunity and possible secondary infection shows bifurcations and chaotic behaviour in wide parameter regions*, Math. Model. Nat. Phenom. **3** (2008) 48–70.

[9] M. Aguiar, N. Stollenwerk and B.W. Kooi *Torus bifurcations, isolas and chaotic attractors in a simple dengue model with ADE and temporary cross immunity,* in Proceedings of 8th Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2008, ISBN 978-84-612-1982-7 (2008).

[10] N. Stollenwerk, F. Drepper, and H. Siegel, *Testing nonlinear stochastic models on phytoplankton biomass time series*, Ecological Modelling **144** (2001) 261–277.

[11] J.P. Boto and N. Stollenwerk, *Fractional calculus and Levy flights: modelling spatial epidemic spreading*, Proceedings of 9th Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2009, ISBN 978-84-612-9727-6, edited by Jesus Vigo Aguiar *et al.*, Salamanca, 2009, pp. 177–188.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Computational Modelling of Thermoforming Processes for Thin Polymeric Sheets, with Goal Oriented Error Estimates for Finite Element Solutions with Hyperelastic Models

D Szegda[1], J Song[1], S Shaw[2], M K Warby[2] and J R Whiteman[2]

[1] *Mechanical Engineering,, Brunel University, Uxbridge UB8 3PH, UK*

[2] *BICOM, Mathematical Sciences,, Brunel University, Uxbridge UB8 3PH, UK*

emails: Damian.Szegda@brunel.ac.uk, Jim.Song@brunel.ac.uk,
Simon.Shaw@brunel.ac.uk, Mike.Warby@brunel.ac.uk,
John.Whiteman@brunel.ac.uk

Abstract

The rapid large inflation of thin hyperelastic polymer sheets modelled as membranes is first considered. For these engineering quantities of interest (QoI) are approximated using finite element methods. Estimates for both discretization and modelling errors with finite element methods are derived using goal oriented techniques. The task of computational modelling of thermoforming processes, whereby thin polymer sheets are inflated into moulds, is then described and finite element solutions are given to an axi-symmetric thermoforming problem again using hyperelastic constitutive models. Thermoforming processes have in the past generally been applied to sheets of polymers based on oil, and products such as food packaging structures have been produced. These products are not biodegradable and contribute significantly to the solid waste produced worldwide. A thermoforming process based on a polymer derived thermoplastic starch is then modelled, using both the above hyperelastic constitutive model and an elasto-plastic constitutive model. The differences between these two computational schemes is given and it is demonstrated that the elasto-plastic model is the more accurate model to use in this application.

1 Introduction

Many packaging containers are produced by thermoforming processes in which thin polymer sheets based on oil are inflated under pressure into moulds. In such processes the

 ISBN 13: 978-84-613-5510-5

sheet is heated, clamped at its boundary, inflated under pressure into or onto a mould, cooled, and then separated from the mould. Generally most of the deformation in the process occurs in the inflation phase, which is constrained by contact with the mould. If computational modelling is to predict accurately what happens in a thermoforming process, it is paramount that this inflation phase be described well. As a result this phase has received the most attention from polymer processors and computational modellers; see e.g. DeLorenzi and co-workers in [3] and papers describing our own work (see [2, 1]). In all this work the challenge has always been to identify the characteristics of the finite deformation specific to the thermoforming conditions being modelled; such as temperature, the rate at which the pressure is applied, the contact conditions, in addition to the properties of the polymer when deformed under these conditions. Different situations lead to different models of thermoforming and as yet no one computational model can be used in all cases. This motivates continuing interest in this aspect of the modelling. The key differences between models often relate to the most appropriate constitutive model, e.g. hyperelastic, elasto-plastic, or viscoelastic, or some combination of all three, and whether the model should be dynamic or quasistatic.

In order to model thermoforming processes we here first consider the rapid free inflation of thin sheets of conventional oil based polymers where the material is temperature dependent. In this the sheet is modelled as a membrane and a hyperelastic constitutive equation is employed. In this context there is still some uncertainty in the modelling as to whether the rapid inflation requires that we solve a dynamic problem (i.e. we solve the full equations of motion) or whether it is adequate to solve a sequence of quasi-static problems. Thus if we use the simpler quasi-static model (as a coarse model) then we are likely to have modelling error as well as discretization error in attempting to compute a QoI using finite elements. Estimates of the discretization error and the modelling error can be obtained by using appropriate duality arguments, which involves solving related dual problems, and this is described in [5] for various QoIs; for example wall thickness.

In practical situations an accurate computational simulation usually requires that we cope well with several different modelling issues and we outline here what we have found that this requires for the thermoforming of polymer sheets made from thermoplastic starch which are not oil based. Such materials, which are relatively new, have the advantage that they are compostable and do not need to end up in land fill waste sites as do their oil based counterparts. Such materials are however not as well understood as more traditional oil based polymers and their properties depend additionally on moisture content. This last dependence has demanded that programmes of experimental work and of associated computational modelling be undertaken; see [4].

 ISBN 13: 978-84-613-5510-5

2 Models and results for thermoplastic starch

A computational model of the inflation phase of thermoforming requires a number of standard terms from continuum mechanics, e.g. the deformation gradient $\mathbf{F}$, stress tensors such as the Cauchy stress $\boldsymbol{\sigma}$ and the nominal stress $\boldsymbol{\Pi} = (\det \mathbf{F})\mathbf{F}^{-1}\boldsymbol{\sigma}$, equations of motion and constitutive equations. When we need to describe flow (as in plasticity) we also have the rate of deformation tensor $\mathbf{D}$ and the Jaumann stress rate $\overset{\triangledown}{\boldsymbol{\sigma}}$; these are tensors which satisfy the principal of material objectivity. A complete description of the model also needs the geometry, assumptions about contact and any appropriate simplifying assumptions such as that of quasi-static motion and of a membrane model for the thin structure. Depending on what is done, there can thus be significant differences between two different descriptions of thermoforming. For example, if the geometry is axi-symmetric, we have a membrane model and an isotropic incompressible hyperelastic material, then we have a one-space dimension problem and we can use a standard total Lagrangian description with the displacement of the mid-surface from the reference configuration as the primary unknown. On the other hand, if the model of the deformation of the sheet needs to be fully three-dimensional and an elasto-plastic model is needed, then this leads to a computational model which is more demanding. The description of the model is also quite different involving an incremental constitutive model and an Updated Lagrangian description of the equations of motion which, in each increment, involves the current deformed configuration. In this compact paper, we give only key features of these two different models (hyperelastic and elasto-plastic) in the case of a commercial thermoplastic starch material.

For an axi-symmetric hyperelastic case with an incompressible and isotropic material we can use cylindrical polars and only need to determine the mid-surface deformation. In this case the stress–stretch relations can be neatly written in the form

$$\sigma_1 = \lambda_1 \frac{\partial W}{\partial \lambda_1} \quad \text{and} \quad \sigma_2 = \lambda_2 \frac{\partial W}{\partial \lambda_2},$$

where λ_1 and λ_2 are principal stretches, and σ_1 and σ_2 are principal stresses and W is a strain energy function. Typically an Ogden form for the strain energy function W with a small number of terms is used with the parameters fitted to results of material tests. A quasi-static model using this is described in [4]. Unfortunately this computationally efficient model does not lead to accurate predictions of the wall thickness.

To get a more accurate prediction we need to use an elasto-plastic model. This is more complicated in that at each point we need to determine if it is in the elastic or plastic region and, in the latter case, we need to attribute how much of the deformation is plastic. Fortunately, commercial finite element packages such as LS-DYNA are available to deal with these issues. In this work this package was employed using explicit time integration to solve the equations of motion to advance the solution from time t to time $t + \Delta t$.

 ISBN 13: 978-84-613-5510-5

The contact was dealt with using general surface-to-surface contact elements with a high friction coefficient as we nearly have total sticking. Part of the computational expense of this approach is that the time step selected by LS-DYNA can be quite small. In the implementation a bilinear elasto-plastic model with isotropic hardening was chosen for the material of the sheet with the parameters obtained by curve fitting to experimentally obtained stress–strain curves. The algorithm itself involves getting the stress at time $t + \Delta t$ from $\boldsymbol{\sigma}(t + \Delta t) = \boldsymbol{\sigma}(t) + \dot{\boldsymbol{\sigma}}\Delta t$, $\dot{\boldsymbol{\sigma}}$ is obtained from $\overset{\triangledown}{\boldsymbol{\sigma}}$ and this is of the form

$$\overset{\triangledown}{\boldsymbol{\sigma}} = 2\mu\mathbf{D}^e + \lambda \mathrm{tr}(\mathbf{D}^e)\mathbf{I},$$

where λ and μ are Lamé constants. Here $\mathbf{D}^e$ is the rate of deformation tensor associated with the deformation gradient $\mathbf{F}^e$ where $\mathbf{F}^e$ is the elastic part of $\mathbf{F}$ in a decomposition of the form $\mathbf{F} = \mathbf{F}^e\mathbf{F}^p$. In Figure 2.1 the thickness prediction with the hyperelastic model (dotted line) and elasto-plastic model (dashed line) are compared with experimental measurements (solid line). The particle paths are also compared in Figure 2.2. The results clearly show that an elasto-plastic model is the more appropriate constitutive model to use in this context for these new materials.

REFERENCES:

[1] Jiang, W. G., Warby, M. K., Whiteman, J. R., Abbott, S., Shorter, W., Warwick, P., Wright, T., Munro, A., and Munro, B. (2003). Finite element modelling of high air pressure forming processes for polymer sheets. *Computational Mechanics*, (31):153–172.

[2] Karamanou, M., Shaw, S., Warby., M. K., and Whiteman, J. R. (2005). Model, algorithms and error estimation for computational viscoelasticity. *Comput. Methods Appl. Mech. Engrg.*, 194:245–265.

[3] Nied, H., Taylor, C. A., and deLorenzi, H. G. (1990). Three dimensional finite element simulation of thermoforming. *Polymer Engineering and Science*, 30, No. 20:1314—1322.

[4] Szegda, D., Song, J., Warby, M., and Whiteman, J. (2007). Computational modelling of a thermoforming process for thermoplastic starch. *American Institute of Physics, Proceedings*, (908):35–47.

[5] Shaw. S, Warby M.K., and Whiteman J.R. (2010). Discretization error and modelling error in the context of rapid inflation of hyperelastic membranes. IMA J Numer. Anal. 30, No1:302-333.

 ISBN 13: 978-84-613-5510-5

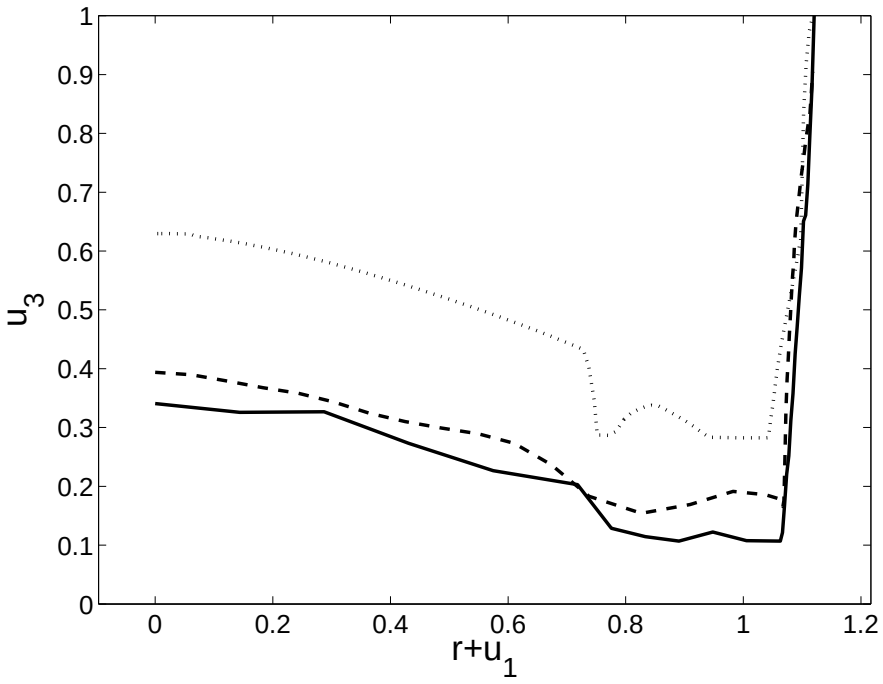


Fig. 2.1: Thickness predictions. The solid line is the experimental measurements, the dashed line is the prediction with the elasto-plastic model and the dotted line is the prediction with the hyperelastic model.

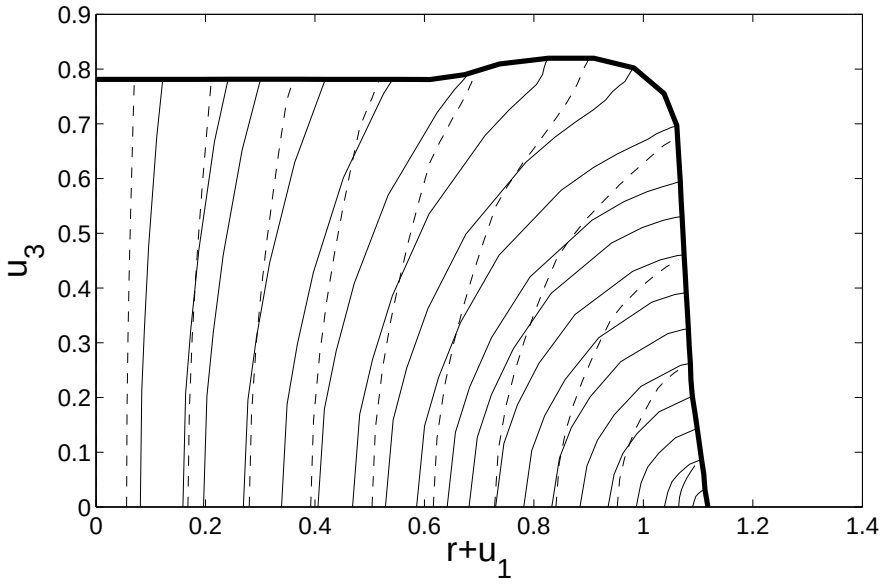


Fig. 2.2: Particle paths. Solid lines are elasto-plastic, dashed lines are hyperelastic.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Management of public parks via mathematical tools.

Lucia Tamburino[1] **and Ezio Venturino**[1]

[1] *Dipartimento di Matematica "Giuseppe Peano", Università di Torino,*
via Carlo Alberto 10, 10123 Torino

emails: lucia.tamburino@alice.it, ezio.venturino@unito.it

Abstract

In this paper we consider a mathematical model for the interaction of wild and domestic herbivores with a natural or man-maintained landscape. We focus on our attention on parks in Piedmont (NW Italy), which are characterized by the presence of meadows, woods and herbivores populations, such as deer and sheep. Even if herbivores consume mainly grass, they can occasionally debark trees, this occuring especially in the winter season, when grass availability is reduced. This may cause the death of trees. Our model evaluates some possible policies for dealing with this problem.

The results of this study show that it is wise to keep in check the number of herbivores in environments in which young trees are present. Grazing of domestic herbivores should be prohibited in newly planted tree areas.

Key words: mathematical models, herbivores, predator-prey systems, tree debarking
MSC 2000: AMS codes (92D25, 92D40, 92D50)

1 Introduction

In this paper we consider a mathematical model for the interaction of wild and domestic herbivores with a natural or man-maintained landscape. The paper is organized as follows. In the next Section we describe the situations that prompted this research. In Section 3 we present the mathematical model for the description of the populations' interactions. Its thorough mathematical analysis is carried out in the subsequently Section. One park in the wilderness is considered in Section 5, with simulations carried out using parameters relevant to this environment. Section 6 contains instead the simulations run for a urban park. Some concluding remarks are finally presented.

 ISBN 13: 978-84-613-5510-5

2 The regional parks

Natural parks in Piedmont (NW Italy) are characterized by the presence of meadows, woods and herbivores populations, such as deer and hares. In addition, these natural areas are often contiguous to agricultural ones, with breeding. In some cases cattle graze on meadows and woods of the nearby park areas, even if that is not allowed: this has sometimes occurred for instance in the "Gran Bosco di Salbertrand" park. In many parks therefore the largest herbivores, not having natural predators, exert a large pressure on the environment. Only recently, after an absence of many years, wolves have reappeared in some areas, [5].

The system represented by a herbivore population and its resource can be regarded as a predator-prey system, in which however two resource populations can be identified, namely grass and trees. In fact, even if herbivores consume mainly grass, they can occasionally debark trees. This occurs especially in the winter season, when grass availability is reduced, [13]. This fact has been observed also in the "Gran Bosco di Salbertrand" park, where the debarking by deer caused several damages. Herbivores peel off only a small piece of the bark from a tree, which is rich in cellulose and lignin and subsequently not easily assimilated. Thus the presence of this second resource for herbivores is not expected to significantly increase their population. On the other hand, peeling off even a small piece of bark may lead to the death of the whole tree, thus removing from the system a much larger amount of tree biomass. In view of the generally slow tree reproduction rate, trees cannot be replaced in a short time when dead. Therefore herbivore behavior can cause large damages to the environment. From this the need of studying deeply the system composed by herbivores, grass and trees arises, in order to predict the potential effects of these damages and eventually to prevent them. The purpose of this paper is to address this question.

In the region there are several areas that are kept at a natural state. In some cases the park is composed of several such woods and natural areas. There are a few which extend around rivers or lakes, some around towns, one of them being the Meisino park nearby Turin, many others incorporate terrains that lie in the high mountains. Among the latter, we mention the following ones: Parco naturale Orsiera Rocciavrè, Parco naturale di Salbertrand, Parco naturale Alta Valle Pesio e Tanaro, Parco naturale delle Alpi Marittime, Parco naturale Alta Valsesia. In these wild and domestic herbivores live, grazing the available pastures. However, when the grass becomes scarce, herbivores tend to debark the trees, to feed themselves. In so doing, though, they exhibit different attitudes. While some some of them, like for instance deer [10], peel off vertical stripes, without totally interrupting the communication between the roots and the leaves, sheep instead peel the tree all around the trunk [4], causing its death after a few weeks; Figure 1 shows the result of such an action.

3 The general model

To represent the situation of interacting herbivores with natural resources, we develop a three-population system: with a top predator population, i.e. the herbivores H, and

 ISBN 13: 978-84-613-5510-5

Figure 1: Damaged trees by debarking. Left: note the action of the sheep, peeling off all around the trunk; Right: the action of deer, peeling off only vertical stripes.

two prey populations, grass G and trees T. This resource splitting follows from the following two basic arguments. At first, we must consider that these resources are ecologically different, their main relevant difference lying in the fact that grass is an r-strategist population with a low carrying capacity per surface unit combined with a fast growth-rate, while trees are k-strategist, they grow in a slow way but can reach a high size. In addition, herbivores show a different level of preference for the two types of resources. Grass is their preferred prey, so that they switch their attention to trees only occasionally, with a higher frequency when grass availability decreases.

The functions $H(t)$, $G(t)$, $T(t)$ represent the biomass of the corresponding populations in our model at a given time t.

The model we propose is given by:

$$\begin{aligned} \dot{H} &= -\mu H + e\frac{HG}{c+H+\alpha G} + f\frac{HT}{g+H+\beta T+\alpha G} \\ \dot{G} &= r_1 G\left(1-\frac{G}{K_1}\right) - \frac{HG}{c+H+\alpha G} \\ \dot{T} &= r_2 T\left(1-\frac{T}{K_2}\right) - \frac{HT}{g+H+\beta T+\alpha G} \end{aligned} \tag{1}$$

The meaning of some parameters is as follows: μ represents the metabolic rate of herbivores, e and f are assimilation coefficients, r_1 and r_2 are the growth rates respectively

 ISBN 13: 978-84-613-5510-5

of grass and trees and K_1 and K_2 represent instead their respective carrying capacities.

Note that the function modeling grass consumption is chosen to follow the Michaelis-Menten model [9], namely

$$\frac{GH}{c+\alpha G+H}$$

where α and c denote positive constants. This, because as the populations become large, we have

$$\lim_{H\to\infty}\frac{GH}{c+\alpha G+H}=G \qquad \text{and} \qquad \lim_{G\to\infty}\frac{GH}{c+\alpha G+H}=\frac{H}{\alpha}$$

meaning that when H grows, the consumption can be at most the whole amount of grass in the system and when grass availability is unlimited, the consumption of grass by herbivores has an upper bound proportional to the number of herbivores: $\alpha^{-1}H$. These are reasonable results, making this function biologically suitable. From the second property, we deduce that α^{-1} represents the herbivores per-capita consumption rate in presence of an unlimited availability of grass. Instead, the parameter c, called the Michaelis constant, or sometimes the half saturation constant, affects only the speed at which consumption approaches its asymptotes.

The function representing the consumption of trees is taken in a similar form, but borrowing ideas of feeding switching, proposed in the classical paper [15] and several other more recent ones, [7, 8, 14, 11, 3], we modify it to model the fact that tree consumption by herbivores grows when the availability of grass decreases, as observed above. This amounts to introducing G in the denominator as well. Specifically, the function is given by:

$$\frac{TH}{g+\alpha G+\beta T+H}$$

Grass and tree grow in a condition of intra-specific competition. We have however excluded the inter-specific competition follows from the consideration that even in so-called "natural" parks there are actually only small completely protected areas. The majority of forests are exploited for timber and fuelwood. Meadows are instead used for grazing so that woods cannot freely expand and replace grass. Moreover, several Piedmontese natural parks cover mountain areas, where most of meadows are located above the tree limit line, so that they are not in competition with woods.

4 Mathematical analysis of the model

4.1 Boundedness of the system

Under certain parameter conditions we now show that the system is bounded. Let us introduce the total system biomass $P \equiv H+G+T$. Then a $\bar{t}>0$ and $B>0$ exist such that for all $t>\bar{t}$ we have $P(t)<B$. For an arbitrary $\eta>0$ we have the following

 ISBN 13: 978-84-613-5510-5

estimate

$$\begin{aligned}\dot{P}+\eta P &\leq (\eta-\mu)H+(e-1)\frac{HG}{c+H+\alpha G}+(f-1)\frac{HT}{g+H+\beta T+\alpha G}\\ &+(r_1+\eta)G-r_1\frac{G^2}{K_1}+(r_2+\eta)T-r_2\frac{T^2}{K_2}\end{aligned}$$

Using $e, f < 1$ and replacing the parabolas in G and in H with their maxima, we find

$$\begin{aligned}\dot{P}+\eta P &\leq (\eta-\mu)H+\left[\frac{(r_1+\eta)^2}{4r_1}K_1+\frac{(r_2+\eta)^2}{4r_2}K_2\right]\\ &\leq \frac{(r_1+\eta)^2}{4r_1}K_1+\frac{(r_2+\eta)^2}{4r_2}K_2\equiv M^*\end{aligned}$$

having taken $\eta < \mu$. We thus find

$$\dot{P}+\eta P \leq M^*$$

from which using the theory of differential inequalities the boundedness of P and ultimately of each population in the model is obtained. All system solutions are therefore confined in a compact region of $\mathbb{R}^3_+$.

4.2 Equilibria and their stability

The boundary equilibria in the $H-G-T$ phase space are the origin E_0, the points $E_1 \equiv (0, K_1, 0)$ and $E_2 \equiv (0, 0, K_2)$ Furthermore the points $E_3 \equiv (0, K_1, K_2)$ and $E_4 \equiv (H, G, 0)$. There could be also the coexistence equilibrium, $E_5 \equiv (H, G, T)$.

To investigate the stability of these points, we construct the Jacobian $J = (J_{kj})$ of the system (1),

$$\begin{aligned}J_{11}&=-\mu+eG\frac{c+aG}{(c+H+aG)^2}+fT\frac{g+aG+bT}{(g+H+aG+bT)^2},\\ J_{12}&=\frac{eH(c+H)}{(c+H+aG)^2}-\frac{faHT}{(g+H+aG+bT)^2},\quad J_{13}=\frac{fH(g+H+aG)}{(g+H+aG+bT)^2},\\ J_{21}&=-\frac{G(c+aG)}{(c+H+aG)^2},\\ J_{22}&=r_1-\frac{2r_1G}{K_1}-\frac{H(c+H)}{(c+H+aG)^2},\\ J_{23}&=0,\quad J_{31}=-T\frac{g+aG+bT}{(g+H+aG+bT)^2},\\ J_{33}&=r_2-\frac{2r_2T}{K_2}-\frac{H(g+H+aG)}{(g+H+aG+bT)^2},\quad J_{32}=\frac{aHT}{(g+H+aG+bT)^2}.\end{aligned}$$

It follows then that the origin is unstable since it has the eigenvalues $-\mu < 0$, $r_1 > 0$ and $r_2 > 0$. Also E_1 and E_2 are both unstable, their eigenvalues respectively being

$$-\mu+\frac{eK_1}{c+\alpha K_1},\quad -r_1$$

 ISBN 13: 978-84-613-5510-5

and

$$r_2, \quad -\mu + \frac{fK_2}{g+\beta K_2}, \quad r_1, \quad -r_2.$$

The equilibrium E_3 has eigenvalues

$$-r_1, \quad -r_2, \quad -\mu + e\frac{K_1}{c+\alpha K_1} + f\frac{K_2}{g+\alpha K_1+\beta K_2}.$$

It is therefore conditionally stable, namely if

$$e\frac{K_1}{c+\alpha K_1} + f\frac{K_2}{g+\alpha K_1+\beta K_2} < \mu \tag{2}$$

To determine the points E_4 from (1), we need to solve the following nonlinear system

$$\begin{aligned} c + H + \alpha G &= \frac{e}{\mu}G, \\ r_1 K_1(c + H + \alpha G) &= HK_1 + r_1 G(c + H + \alpha G). \end{aligned}$$

Solving from the first one for H we have,

$$H_1 = \frac{e}{\mu}G_1 - c - \alpha G_1,$$

and by replacing the value of H_1 into the second equation, we obtain the following parabola in G whose roots give the sought value G_1,

$$a_2 G^2 + a_1 G + a_0 = 0 \tag{3}$$

where $a_0 = K_1 c$ and

$$\begin{aligned} a_1 &= \frac{K_1 r_1 e}{\mu} - \frac{K_1 e}{\mu} + K_1\alpha \\ a_2 &= -\frac{r_1 e}{\mu}. \end{aligned}$$

Clearly, $a_0 > 0$ and $a_2 < 0$, so that there are two real solutions, one positive and the other one negative. But this is not sufficient to ensure the existence of a biologically feasible equilibrium, since H_1 must also be positive. The feasibility condition in this case then becomes

$$G_1 > \frac{c}{e/\mu - \alpha}. \tag{4}$$

The numerical integration of the model (1) show that this equilibrium can in fact be reached, for suitable parameter values. Moveover, for the interior equilibrium, other simulations reported in the next Sections, indicate that it exists and is stable in adequate parameter ranges.

 ISBN 13: 978-84-613-5510-5

5 Case study: the "Gran Bosco di Salbertrand" park

The natural park "Gran Bosco di Salbertrand" covers an area encompassing parts of the municipality of Chiomonte, Exilles, Oulx, Pragelato, Salbertrand, Sauze d'Oulx and Usseaux (all in the Turin district). Its altitude ranges from 1000-1200 meters to about 2600 meters, with the maximum at 2692 meters (Gran Pelà). In the park a deer population (*Cervus elaphus*) is living, which has caused severe damages to its woods, especially during the eighties of the last century. The most affected trees were firs, whose natural renovation has been compromised. Then, these damages drastically decreased with the reduction of the deer population. In fact, according to the census periodically carried out by rangers, the deer population slightly increased until 1995, when a peak of 240 individuals was reached. Afterwards, deers suddenly dropped to 111 individuals and in the following years they remained around 100 individuals, with a maximum of 150 and a minimum of 78. According to the park rangers, this decrease is likely due to predation, because the presence of a stable pack of wolves in the park is well established since at least 1997.

To apply the model to this park, as first step we choose the units: for time we use the day and for the biomass the average weight of a deer, in order to facilitate the reading of the plots. According to [5], the mean weight of a male is 200 kilos, of a female 90, of a young one 40 and of a fawn born in the current year 20 kilos. Using on these data and the census reports, we calculate the weighted mean of the mean weights of deers, subdivided on the basis of gender and age class. The result represents our biomass unit: 115 kilos.

In the second step we assign values to the parameters. In [6] tables are reported providing estimations of the phytomass and of the annual NPP (Net Primary Production) of several natural environments. These allow to estimate the growth rate of grass and trees at the various interested latitudes and altitudes. These informations enable us to set $r_1 = 0.003$ and $r_2 = 0.0002$. But if we restrict the rate only to the growth period, which can be estimated to be about 120 days, we then obtain $r_1 = 0.01$ and $r_2 = 0.0006$. From [1] the extension of the park (3774 ha) is obtained, together with other relevant data such as the percentage of the different kinds of soil, like rocks, anthropic areas and, what is important for our work, meadows (1256.6 ha) and woods (2277.2 ha). Using these informations and the tables of [6], we could estimate also the values of the carrying capacities in kilos or tons. Once converted in the chosen unit, we have $K_1 = 220,000$ and $K_2 = 4,950,000$. However a part of the meadows cannot be included in the deer-grass-tree system, because they are used for cattle grazing and hence they cannot be used by deers. According to [1], there are only 139.6 ha of not-used meadows besides 299 ha of rock meadows, having a reduced NPP. That leads to a different value for K_1, namely $K_1 = 36500$.

In [12] information about the diet of ruminants can be found, and an empirical method — in fact widely used — to estimate the food daily consumption of a ruminant is suggested. According to this rule, the food ingested can be approximated as a percentage of the weight of the ruminant itself. The percentages however depend on the particular species. Even if precise data are not given for deer, useful indications can

 ISBN 13: 978-84-613-5510-5

be obtained from data referring to similar species. This allows us to assign a value to α. The latter represents the reciprocal of the grass amount consumed by a single herbivore when the grass is unlimited (see Section 3). We set $\mu = 0.03$, meaning that a herbivore with no food available dies in about 30 days. This is consistent with the starvation time of similar mammals, namely mammals not accustomed to lethargic periods and with a weight comparable with the one of deers. Recalling that $e < 1$ and assuming that an available adequate amount of grass can satisfy the metabolic needs of herbivores, i.e. $e > \alpha\mu$, we then set $e = 0.62$ as reference value, varying it within the previous bounds.

An estimation of the tree consumption due to herbivores is more difficult. When a herbivore switches its attention to a tree rather than to grass, even if it takes just a small piece of bark, it may cause the death of the whole tree. The weight of a young tree is comparable with the mean weight of a deer. Therefore, recalling that β represents the reciprocal of the per-capita tree consumption by herbivores (similarly as for α), this would imply a value for β close to 1. However, in case of deer, the tree death does not occur with a high probability, because, as remarked in Section 2, deer in general peel off vertical stripes of bark without totally interrupting the communication between leaves and roots. On these considerations we assign a smaller value to β^{-1}, i.e. a larger one to β. We set as a reference value $\beta = 8$. Of a whole tree, a deer can assimilate at most the piece of bark it really takes, but as already remarked even that is only partially assimilated, in view of the high lignin and cellulose content of barks. Therefore, the assimilation coefficient from trees f must be very small: we set it to be $f = 0.001$.

To sum up, for the numerical simulations, we used the following reference set of parameter values:

$$\begin{aligned} K_1 = 36500 \quad r_1 = 0.003, \quad K_2 = 4,950,000 \quad r_2 = 0.0005, \quad \alpha = 20, \\ \mu = 0.03, \quad e = 0.62, \quad \beta = 8, \quad f = 0.001. \end{aligned} \tag{5}$$

We have then varied each parameter within a certain reasonable, biologically feasible range.

5.1 Simulations

Our simulations show that the system can reach a stable coexistence equilibrium. Figures 2 and 3 show the results on the system obtained with the same data. The main difference among the two plots consists in the fact that the second one (Figure 3) accounts for the seasons. More specifically, in this simulation we alternate periods of vegetative growth, lasting about 120 days per year, i.e. about four months, with periods of vegetative pause, the remaining 245 days of the year, in which $r_1 = r_2 = 0$. This does not modify the behavior of the system, but leads the populations G, T to stabilize on lower values. In particular the values at which H settles are consistent with the real values of the deer population living in the park, in absence of wolves.

Comparing the two curves in Figure 3, it is possible to assess the impact of the deer population on trees. In fact the latter grow more slowly than in absence of the deer and furthermore they do not reach the carrying capacity. The damages are very limited, but they depend on the number of deers. With slight differences in the parameter

 ISBN 13: 978-84-613-5510-5

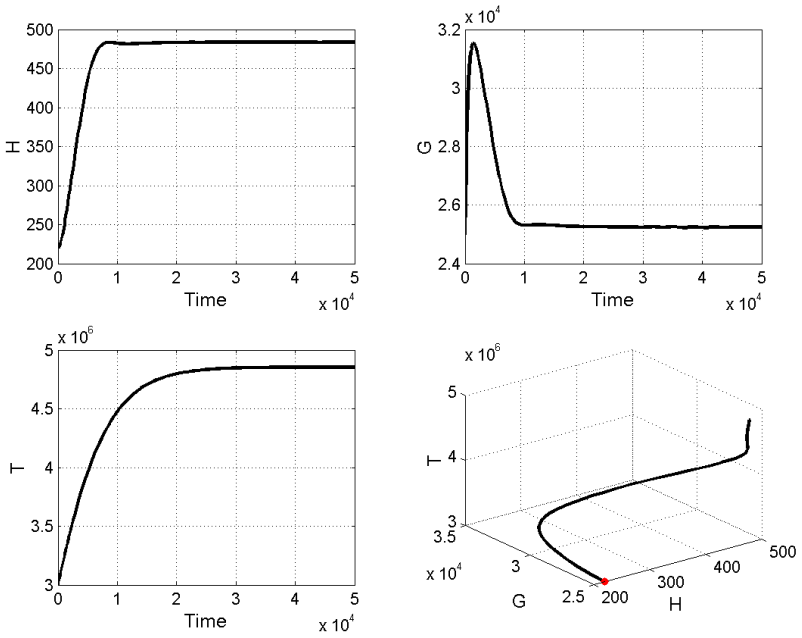


Figure 2: Evolution of the system with parameters set like in (5)

values, simulations show that the deer population could grow to higher levels: in this case the entity of damages would be much larger. Even limited damages can however have relevant consequences if they are not uniformly distributed, but concentrated in some particular area. This can be seen in Figure 4, showing the case in which only the fir woods interact with the rest of the system. According to [1], fir woods cover an area of 612.3 ha. This allows us to estimate the tree carrying capacity for the plot of Figure 4, to the value $K_1 = 106000$. This scenario is not far from the real case, because deers debark especially the youngest and smallest trees with a thin bark. Hence damages interest at most the growing woods with a big percentage of young trees as well as copse woods. In addition, there are some species of trees that are never debarked by deers, like *Larix decidua*, *Pinus cembra* and *Fagus sylavatica*. More generally, even if the overall damage is small with respect to the total tree biomass, since it is concentrated on the youngest trees, it can compromise the natural renewal processes of the woods. This suggests the necessity of avoiding an excessive growth of the deer population and of adopting some measures to protect trees, including for instance conversion of copse into high forests.

6 Case study: the Meisino park

The Meisino is a urban park, on the immediate outskirts of the town of Turin. It provides a study case for our model because here the Turin municipality has partially

 ISBN 13: 978-84-613-5510-5

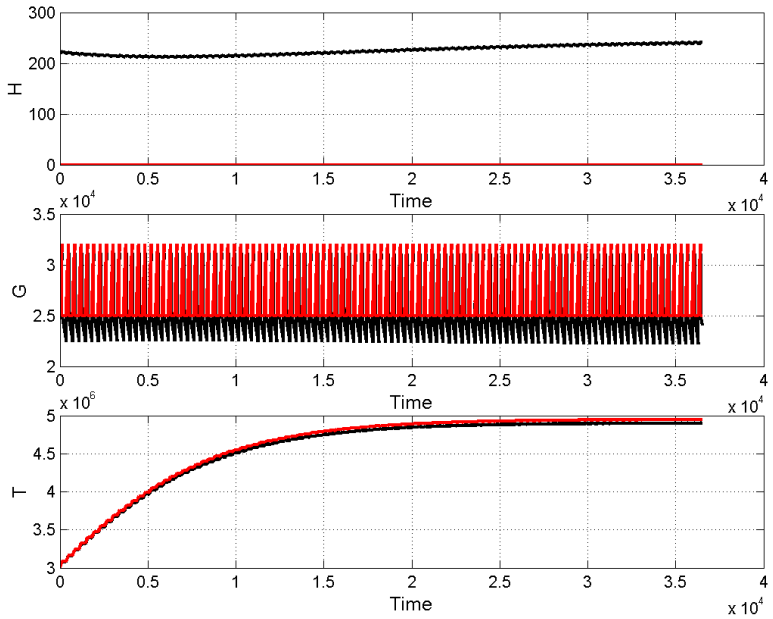


Figure 3: Evolution of the system over a century with alternating seasons. Parameters are set like in (5), with the exception of r_1 and r_2, which have set respectively to the values 0.01 and 0.0006 during the summer and both to 0 during the winter. The upper curve shows the evolution of the same system without herbivores: damages to trees are very limited with this set of parameter values.

replaced the traditional tools for lawn maintenance by sheep grazing. Since 2007 in fact, 500 "Biellese" meat-sheep have been introduced in the park for a 2 months period every year.

In order to assign meaningful values to the parameters, as for the case of the "Gran Bosco di Salbertrand" park, we based our considerations on [6] and [12]. In addition, we used information about the Meisino park contained in the project, on top of the information about the average weight of the Biellese sheep (75 kg), available in [2]. For the parameter β, i.e. the inverse of the per-capita consumption of trees by sheep, as already outlined, we considered that sheep have a peculiar way of debarking: unlike deer, they peel trees all around the trunk [4], causing its death after a few weeks almost certainly, or at least with a probability close to 1. This implies a larger consumption of trees by sheep than by deer. Hence in this case, we assign a larger value to β^{-1} and therefore a smaller one to β: we set $\beta = 3$. To sum up, for the numerical simulations, we used the following reference set of parameter values:

$$\begin{aligned} K_1 = 12500, \quad r_1 = 0.004, \quad K_2 = 25000, \quad r_2 = 0.0005, \\ \mu = 0.03, \quad \alpha = 20, \quad e = 0.62, \quad \beta = 3, \quad f = 0.001. \end{aligned} \tag{6}$$

 ISBN 13: 978-84-613-5510-5

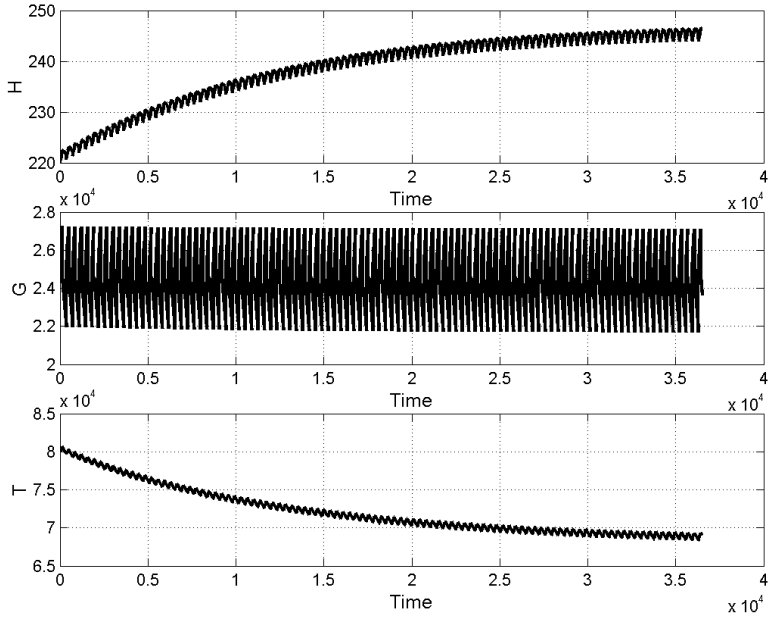


Figure 4: Evolution of the system over a century with alternating seasons and $K_2 = 106,000$, i.e. considering only the fir woods. In this case damages are much more relevant.

As for the previous case, we have then varied each parameter within a certain reasonable, biologically feasible range. Of course, these ranges are more restrictive for some parameters and larger for the others, for which the estimations are more uncertain, like for instance K_2.

In the case of the Meisino park, it has been possible to obtain a validation of our model, since there exist precise data about the trees damaged by sheep: there are more than 100 trees damaged every year, most of them in a non-reversible way. Simulations show that our model is able to reproduce the real case: with 500 sheep over a 60 days period the grass decreases, the sheep grow; further, the trees decrease by amounts consistent with the reported damages.

6.1 Simulations

With the set of values (6), simulations show that the system reaches an equilibrium. But the latter is very sensitive to some parameters, in particular to e, i.e. the grass assimilation coefficient by sheep. We explored also different parameter combinations. Interesting results emerge by setting all parameters as given in (6), but for f, i.e. the tree assimilation coefficient by sheep, which is set to a slightly higher value, namely $f = 0.05$.

Figures 5 and 6 show the same Hopf bifurcation diagram, obtained varying e, but

 ISBN 13: 978-84-613-5510-5

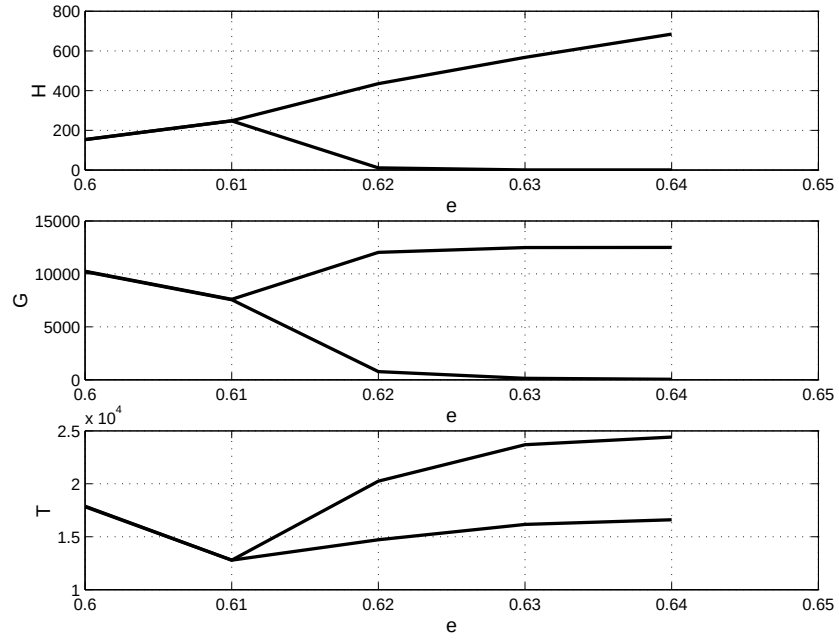


Figure 5: Hopf bifurcation diagram obtained varying e, i.e. the grass assimilation coefficient by sheep, with $f = 0.05$ and the other parameters set as in (6)

in the latter we set $T = 0$. It is worth noting that in the latter the threshold above which the system starts to oscillate shifts to the right. That implies the existence of a range of values for e, for which the system reaches an equilibrium only without trees. Figures 7 and 8 show two plots obtained with a value of e in such a range: $e = 0.62$. Both plots are carried out with the same parameter values. In the former there are no trees, and herbivores and grass reach a stable equilibrium level. In the latter trees are present; here all populations after reaching a higher peak, start to oscillate. In the low peaks of oscillations, herbivores reach values very close to zero and hence run the risk of extinction. To sum up, for such values of e, the presence of the second resource results in a disadvantage for H. Moreover, there is only a very small range of values for e (around 0.61) in which the system can attain an equilibrium in both cases; below this threshold there exists an equilibrium with sheep only if trees are simultaneously present: without trees the system does not oscillate but sheep cannot survive, Figure 6.

6.2 Discussion

In order to evaluate the management choice of the Turin municipality of introducing sheep in the park, we carried out simulations in which sheep where introduced and then removed from the system for two months every year, like in the real case. We have accounted for 150 days per year, i.e. 5 months, i.e. the average vegetative growth period in the Turin area. Recalling that the goal of the Turin municipality policy is that

 ISBN 13: 978-84-613-5510-5

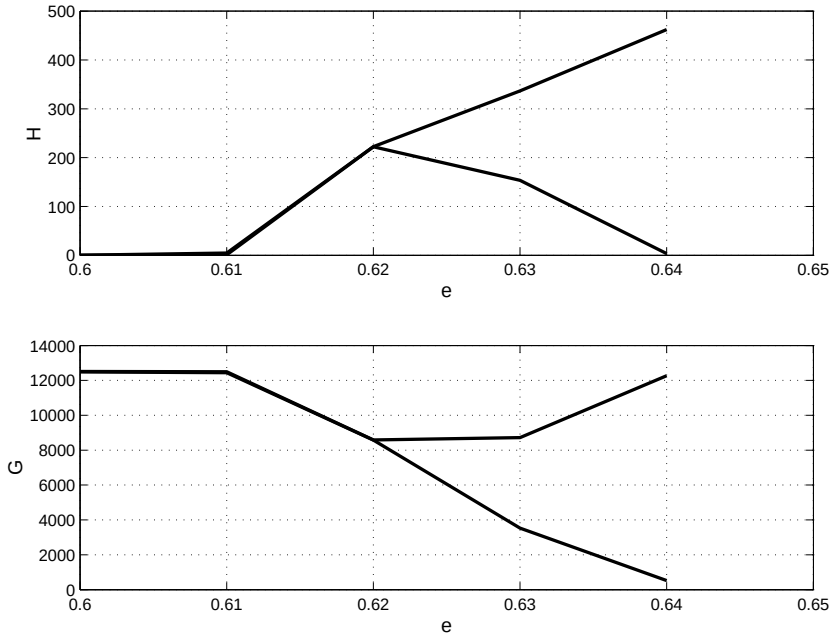


Figure 6: Hopf bifurcation diagram obtained varying e, i.e. the grass assimilation coefficient by sheep, with the same parameter values as in Figure 5: $f = 0.05$ and the other parameters set as in (6). Here we set $T = 0$.

the grass should be controlled via grazing, we could understand better the situation by setting a quantitative value for this containment. So let us assume that by this intervention we want to keep the grass at say 80% of the natural carrying capacity of the Meisino park. Simulations show that we can obtain this goal with a number of sheep between 400 and 450. However in this case the damage suffered by trees is quite large: over a timespan of 25 years the tree biomass is 35% lower than in absence of sheep (see Figure 9). Therefore the tree canopy expected from the original project for the park will never be reached, causing a net loss in the money invested to plant the trees. An additional cost is incurred by the loss of the environmental services provided by trees, which are lost when they die.

In spite of this, the municipality is planning to reintroduce sheep. To prevent bark grazing, this year (2010), fences have been put around trunks, but their setting is certainly insufficient. For instance, in the same area we can now find trees with and without fences. But if a certain area is going to be grazed by sheep, then all trees in it need to be protected. Otherwise, if an area is not used, no tree in it needs protection. In addition, there are cases in which fences have been placed even around trees that are already damaged and dead (see Figure 10), and around trees which have been logged only a few days later, (see Figure 11). Due to this inappropriate use, the fences represent an additional increase in the management costs.

 ISBN 13: 978-84-613-5510-5

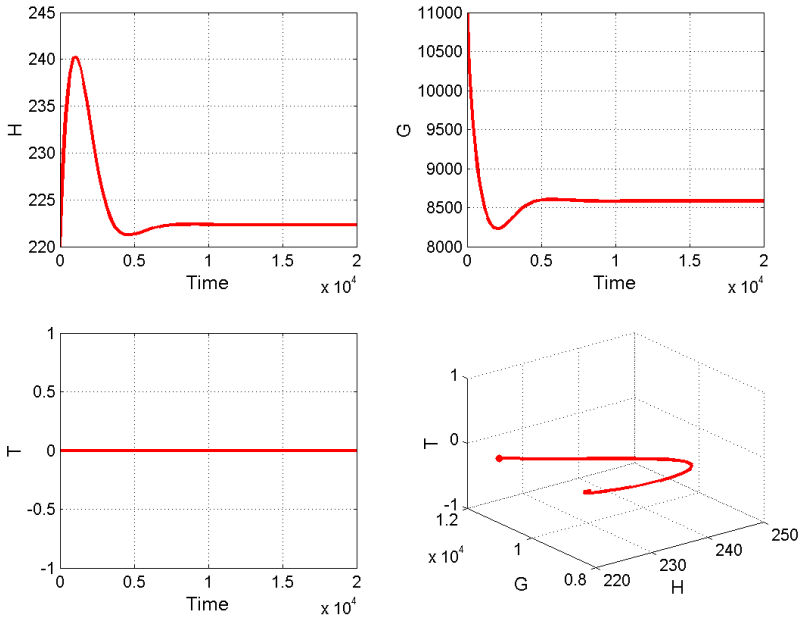


Figure 7: This plot has been obtained with $e = 0.62$, $f = 0.05$ and other parameter like in (6): in absence of trees, the system settles to a stable equilibrium.

7 Conclusions

In this paper we have proposed a mathematical model for the assessment of the interactions between wild and domestic herbivores with the natural or man-maintained system of meadows and woods. Damages to the trees caused by debarking, occurring all the time, but in particular when grass is scarcer, are accounted for. The results of this study show that it is wise to keep in check the number of herbivores in environments in which young trees are present, to prevent their peeling off of the barks of the latter, causing damages to the natural renewal process of woods. In the natural parks this can be achieved by reducing their numbers, either by culling the herbivores or by the action of their natural predators, ensured for instance by the presence of wolves. The copse woods are more prone to this debarking risk than high forests. This occurs in view of the reduced tree dimensions and thus also of their bark thickness. Whenever possible, it would then be preferable to convert the copse woods into high forests. The latter is a wood type more apt to natural parks and therefore ecologically richer and more resilient. The domestic herbivores instead should be prevented to enter in areas in which young trees without an enough thick bark are present. In particular then, grazing should be prohibited in newly planted tree areas.

Acknowledgments. The authors thank Fabio Schiari for the photograph of deer debarking, Figure 1 right, and Elisa Ramassa, a ranger of the "Gran Bosco di Salbertrand" park, for the useful informations provided.

 ISBN 13: 978-84-613-5510-5

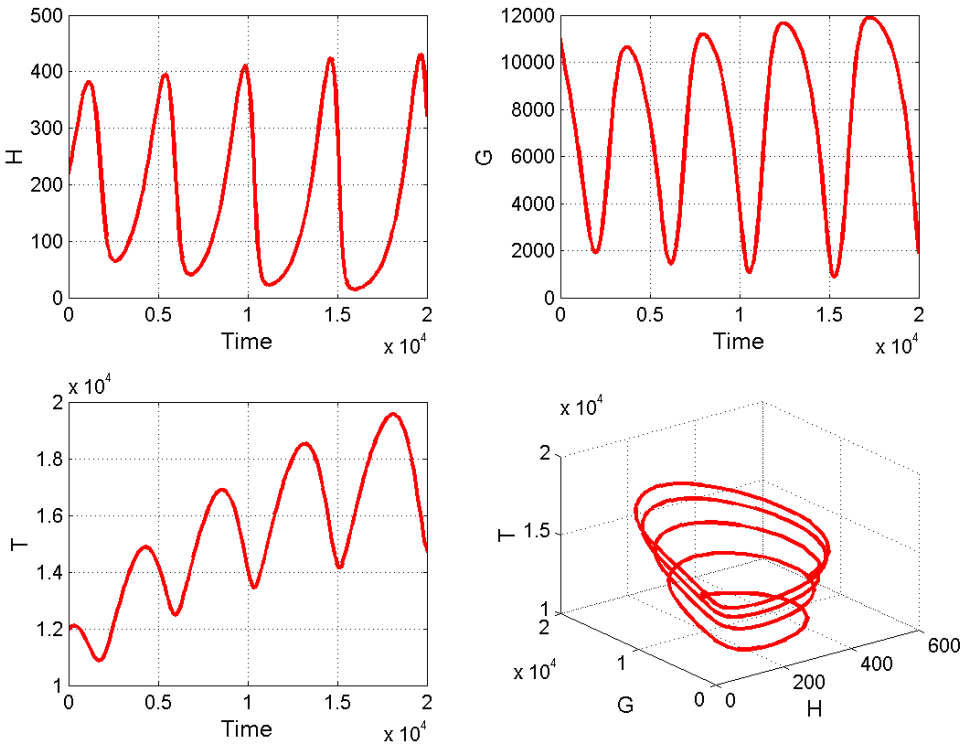


Figure 8: This plot has been obtained with the same values of the plot in figure 7: with H and both resources T and G, the system does not reach a stable equilibrium, rather it starts to oscillate, but at every minima in the oscillations, H risks extinction.

References

[1] AA. VV. *Revisione e integrazione del Piano naturalistico e del piano di assestamento relativamente alle aree di ampliamento (Revision and integration of the naturalistic plan and of the settling plan of the enlargement areas)*, Regione Piemonte, Settore Pianificazione Aree Protette, Torino 2002.

[2] AA. VV., *Le razze ovine autoctone del Piemonte (Autochthonous sheep breeds of Piedmont)*, in www.associazionerare.it, 2009.

[3] V. Ajraldi and E. Venturino, *Stabilizing Effect of Prey Competition for Predators Exhibiting Switching Feeding Behavior*, WSEAS Transactions on Biology & Biomedicine, **5**, (2008) 65–74.

[4] G. W. Anderson, M. Hawke and R. W. Moore, *Pine needle consumption and bark stripping by sheep grazing annual pastures in a young stands of widely spaced Pinus radiata and P. pinaster*, Agroforestry Systems, **3**, (1985) 37-45.

 ISBN 13: 978-84-613-5510-5

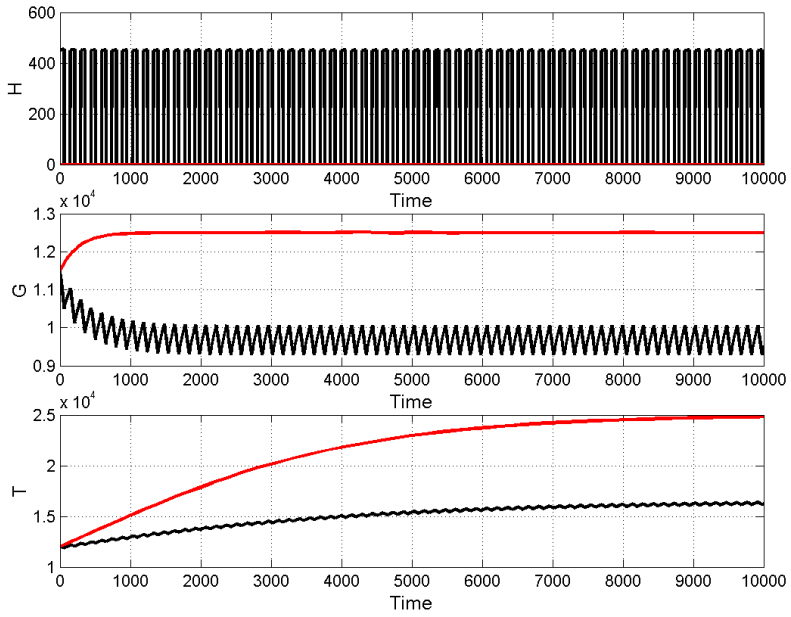


Figure 9: The figure represents the result of the simulation introducing 450 sheep, to reach the goal of keeping the grass at 80% of its carrying capacity. The damage suffered from the trees amounts in this case to 35% over the timespan of 25 years.

 ISBN 13: 978-84-613-5510-5

Figure 10: Already debarked, but still fenced, tree.

[5] M. Borgia, *Il ritorno del lupo nelle valli torinesi (The return of the wolf in the Turin valleys)*, Luna Nuova, Avigliana (TO), 2003.

[6] T. Fujimori, *Ecological and silvicultural strategies for sustainable forest management*, Elsevier, 2001.

[7] Q. J. A. Khan, E. Balakrishnan and G. C. Wake, *Analysis of a predator-prey system with predator switching*, Bull. Math. Biol., **66**, 109-123, 2004.

[8] Q. J. A. Khan, E. V. Krishnan and M. A. Al-Lawatia, *A stage structure model for the growth of a population involving switching and cooperation*, ZAMM,

 ISBN 13: 978-84-613-5510-5

Figure 11: A logged young tree.

82, 125-135, 2002.

[9] J. D. Murray, *Mathematical Biology*, Springer Verlag, Berlin, 1989.

[10] A. J. de Nahlik, *Wild deer*, Faber and Faber Ltd., London, 1959.

[11] S. Palomino Bean, A. C. S. Vilcarromero, J. F. R. Fernandes, O. Bonato, *Co-existência de Espécies em Sistemas Presa-predador com Switching*, /rm TEMA Tend. Mat. Apl. Comput., 7 (2006) 317-326.

[12] G. Pulina and R. Bencini, *Dairy Sheep Nutrition*, CABI Publishing, Cambridge, MA, 2004.

[13] S. H. Sharrow, D. H. Carlson, W. H. Emmingham and D. P. Lavender, *Direct impacts of sheep upon Douglas-fir in two agrosilvopastoral systems*, Agroforestry Systems, (1992), **19**, 223-232.

[14] A. C. S. Vilcarromero, S. Palomino Bean, J. F. R. Fernandes, O. Bonato, *Switching Behaviour and Stability in Predator-Preys Systems*, Proceedings Of Congreso Latino Americano de Biomatematica, **X**, (2001) Alab-V Elaem.

[15] M. Tansky, *Switching effects in prey-predator system*, J. Theor. Biol., **70**, (1978) 263-271.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Solving a Nonlinear Forward-Backward Differential Equation from Nerve Conduction Theory

M.F. Teodoro[1,2], P.M. Lima[2], N.J. Ford[3] and P.M. Lumb[3]

[1] *Departamento de Matemática, Escola Superior de Tecnologia de Setúbal, Instituto Politécnico de Setúbal, Estefanilha, 2910-761 Setúbal, Portugal*

[2] *CEMAT, Instituto Superior Técnico, Universidade Técnica de Lisboa, 1049-001 Lisboa, Portugal*

[3] *Department of Mathematics, University of Chester, CH1 4BJ, Chester, UK*

emails: mteodoro@est.ips.pt, plima@math.ist.utl.pt, njford@chester.ca.uk, p.lumb@chester.ca.uk

Abstract

Mixed type (forward-backward) functional differential equations (MTFDEs) are, in general, ill-posed. This issue of MTFDEs poses problems for both classical and numerical analysts. It is done a brief review of our preliminary work with autonomous and non-autonomous linear MTFDEs using collocation, least squares and finite element methods. In particular, this paper is concerned with the approximate solution of a nonlinear mixed type functional differential equation (MTFDE) with deviating arguments arising from nerve conduction theory. The considered equation describes conduction in a myelinated nerve axon in which the myelin totaly insulates the membrane. As a consequence, the potential change jumps from node to node. As described in [1], this process is modeled by a first order nonlinear functional-differential equation with deviated arguments. A solution of this equation is searched. Following the approach introduced previously in [5], [6], [2], using collocation and least squares methods and in [4], [7], using a finite element method, some new computational methods for the solution of this problem are proposed and analysed. Numerical results are obtained and compared with the ones presented in [1].

Key words: Mixed-type functional differential equation, method of steps, collocation, Newton method, nonlinear

MSC 2000: 34K06; 34K10; 34K28; 65Q05

 ISBN 13: 978-84-613-5510-5

1 Introduction

In this paper we consider equations of the type

$$\delta v'(t) = F(v(t)) + \beta(t)v(t-\tau) + \gamma(t)v(t+\tau), \tag{1}$$

where v is the unknown function, δ is a known constant, $\beta(t)$, $\gamma(t)$ and $F(v(t))$ are known functions. These methods were extended to the non-autonomous case (when $\delta = 1$ and α, β and γ are smooth functions of t). In particular, we are concerned about a nonlinear mixed type functional differential equation (MTFDE) with deviating arguments arising from nerve conduction theory. The considered equation describes conduction in a myelinated nerve axon in which the myelin totaly insulates the membrane. As an immediate consequence, the potential jumps from node to node (figure 1). The modelling equation corresponds to a boundary value problem (BVP) of first order. We search for a solution of a boundary value problem defined in $\mathbb{R}$, which takes given values at $\pm\infty$.

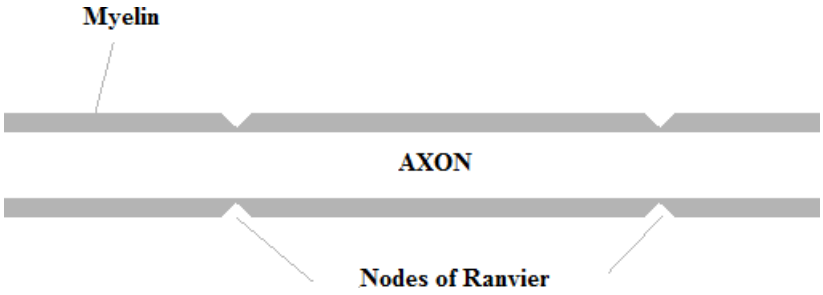


Figure 1: Nerve axon

In this work is considered the following nonlinear MTFDE

$$RCv'(t) = F(v(t)) + v(t-\tau) + v(t+\tau), \tag{2}$$

where $-\infty < t < +\infty$, $v(-\infty) = 0$ and $v(+\infty) = 1$. The equation (2) is also considered in [1] where we find a detailed derivation of the model. The unknown $v(t)$ represents the transmembrane potential at a node in a myelinated axon, in the nerve conduction model. F reflects the current-voltage model. R and C are respectively axomatic nodal resistivity and nodal capacity. This model is described with detail in [1]. It is considered a mathematical model formulated from an equivalent electric circuit model which assumes the so called pure saltatory conduction (PSC). It means that the myelin has higher resistance and lower capacitance when compared with the membrane; if the membrane is depolarized at a node, myelin tends to jump the next node and excite the membrane there. In this model the myelin insulates the membrane.

 ISBN 13: 978-84-613-5510-5

2 Numerical Method

As stated before, the problem arose from nerve conduction theory and is modeled by equation (2). It is not an easy problem to solve numerically: it is advanced-retarded and is a BVP defined on $\mathbb{R}$ with τ unknown. It is assumed pure saltatory conduction. In addition, the circuit model supposes that the nodes are uniformly spaced and electrically identical, the axon is infinite in extent and in the axon cross-sectional variations in potential are negligible. Several models can be obtained using different current-voltage expressions. Using FitzHugh-Nagumo dynamics for the nodal membrane, without a recovery term, we assume that a supra-threshold stimulus begins a propagated axon potential and consequently travels down the axon from node to node. With adequate variable substitutions, one can get the following non-dimensional model:

$$v'(t) = f(v(t)) + v(t-\tau) + v(t+\tau) - 2v(t), \tag{3}$$

where $-\infty < t < +\infty$, $v(-\infty) = 0$ and $v(+\infty) = 1$. The function f is given by $f(v) = bv(v-a)(1-v)$ with a the threshold potential in the non dimensional problem $(0 < a < 1)$, τ the non dimensional time delay and b related with the strength of the ionic current density $(b > 0)$. The solution at any node should be monotone increasing. This arises from the current-voltage relation $f(v)$, once a node is turned on, it cannot return to rest potential $v = 0$.

2.1 Scheme

Instead of model (3) we propose

$$v'(t) = f(v(t)) + v(t-\tau) + v(t+\tau) - 2v(t), \tag{4}$$

where $-L < t < +L$, with the boundary conditions at $[-L-\tau, -L]$ and $[L, L+\tau]$ for some positive integer L. L is considered large enough. We want to solve (4) using an adapted method of steps, similar to the work found in [5, 6, 2, 7, 4], which we call an enhanced method of steps. The algorithm is described below and it is based on the idea of predictor-corrector methods.

Enhanced Method of Steps

1. **Compute τ**

   (a) In order to compute the boundary conditions $[-L-\tau, -L]$ and $[L, L+\tau]$, we must find the characteristic roots for the linearized equation, at L and L, as done in [1];

   (b) Knowing the solution v and its k first derivatives at $-L-\tau$ and $-L$ (k it is the number of steps), we can determine by recurrence formulae, using an initial guess for τ; the value of $v(0)$ (limit from left);

 ISBN 13: 978-84-613-5510-5

(c) Knowing the solution v and its k first derivatives at L and $L+\tau$, we can determine by recurrence formulae, using an initial guess for τ, the value of $v(0)$ (limit from right);

(d) Then, in order to get v continuous at $t=0$, we adjust the parameter τ, till the limits from left and from right will coincide at $t=0$.

2. **For the computed value of τ, construct the first iterate of the solution at** $[-L, L]$

(a) Use values of the initial guess for the solution on the interval $[-L, -L+\tau]$.

(b) Construct the auxiliary solution at sucessive intervals with amplitude τ using the ordinary method of steps;

(c) Correct the solution at sucessive intervals with amplitude τ by solving equation (4) in each interval and using the auxiliary solution at the neighbour intervals;

(d) Continue until we reach the interval $[L, L+\tau]$. We get a vector $W = (w(L), w(L+h), w(L+2h), ..., w(L+\tau))$, where w represents the approximate solution.

3. **Compute the Jacobian matrix**

(a) For each grid point t_i, at the interval replace the value $w(t_i)$ of the initial guess by $w(t_i)+\epsilon$ and repeat step 2 with the new initial approximation. This will give the vector

$$W_i = (w_i(L), w_i(L+h), w_i(L+2h), ..., w_i(L+\tau)), \quad i = 1, \dots N;$$

(b) For each vector W_i, compute the $i-th$ row of the jacobian matrix.

4. **Apply the Newton method**

(a) Obtain the components of the right-hand side of the jacobian and solve the linear system;

(b) Update the initial guess of the solution.

5. **Iterate this process until the norm of the difference between iterates is less than a certain tolerance value**

 ISBN 13: 978-84-613-5510-5

So, in summary

- Based on the approximation of the solution and on the method of steps, compute the value of τ, for which the solution of the problem exists;
- The nonlinear equation (4) with the respective boundary conditions is reduced by the Newton method to a sequence of linear BVP;
- The numerical solution of each linear BVP is obtained by the collocation method.

A question arised was how to choose the initial approximation to guarantee convergence?

The solution of test problem can then be used as initial approximation for the numerical solution of the target problem. The idea, proposed by Chi et al in [1], is to use a continuation method.

The continuation method consists on the approach of the target problem using a test problem, by formulae (5)

$$f_a(v) = \alpha f_{test}(v) + (1-\alpha) f_{target}(v), \quad 0 \leq \alpha \leq 1. \tag{5}$$

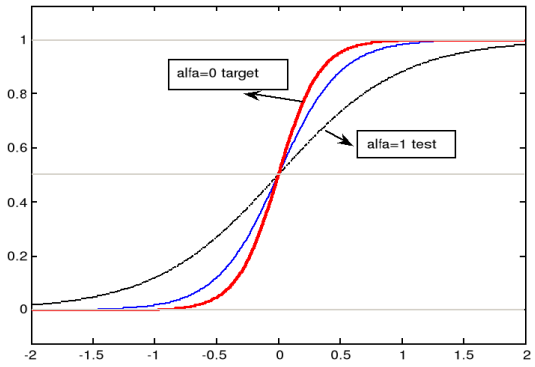


Figure 2: Continuation method. $a = 0.05$, $b = 15$, $N = 81$.

In figure 2 is computed the approximate solution for target problem considering $a = 0.05$, $b = 15$ and $N = 81$.

3 Final Remarks

To analyse the convergence of the numerical scheme we take into account some test problems with known solutions. Numerical results are compared with the results obtained by other methods presented in [1]. A question in study is how the solution of equation (4) will be affected by changing the parameters of the problem and the deviating argument τ.

 ISBN 13: 978-84-613-5510-5

4 Acknowledgements

M. F. Teodoro would like to acknowledge the finantial support from Fundação para a Ciência e Tecnologia, FCT, through grant SFRH/BD/37528/2007.

References

[1] H. Chi, J. Bell and B. Hassard, *Numerical solution of a nonlinear advance-delay-differential equation from nerve conduction theory*, J. Math. Biol., 24, 583-601 (1986).

[2] P.M. Lima, M.F. Teodoro, N.J. Ford and P.M. Lumb, *Analytical and Numerical Investigation of Mixed Type Functional Differential Equations*, Journal of Computational and Applied Mathematics (2010), avaialable electronically doi:10.1016/j.cam.2010.01.039

[3] P.M. Lima, M.F. Teodoro, N.J. Ford and P.M. Lumb, *Analytical and Numerical Investigation of Mixed Type Functional Differential Equations (Extended version)*(to appear as a tech. report at the University of Chester).

[4] P.M. Lima, M.F. Teodoro, N.J. Ford and P.M. Lumb, *Finite Element Solution of a Linear Mixed-Type Functional Differential Equation*, submitted to Numerical Algorithms, Springer.

[5] M.F. Teodoro, P.M. Lima, N.J. Ford, P. M. Lumb, *New approach to the numerical solution of forward-backward equations*, Front. Math. China,V.4, N.1, 155-168 (2009).

[6] M.F. Teodoro, N. Ford, P.M. Lima, P. Lumb, *Numerical modelling of a functional differential equation with deviating arguments using a collocation method*, AIP Proc., Inter. Conference on Numerical Analysis and Applied Mathematics, Kos 2008, vol 1048, pp. 553-557 (2008).

[7] M. F. Teodoro, P. M. Lima, N.J. Ford and P.M. Lumb, *Numerical Approximation of Forward-Backward Differential Equations by a Finite Element Method*, Proceedings of CMMSE 2009, 9th International Computational and Mathematical Methods in Science and Engineering, Gijon, Spain, V.3, 1010-1019, ISBN:978-84-612-9727-6 (2009)

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A model for the human papilloma virus infection

Giulia Toniolo[1], Silvia Martorano Raimundo[2] and Ezio Venturino[1]

[1] *Dipartimento di Matematica "Giuseppe Peano", Università di Torino, via Carlo Alberto 10, 10123 Torino*

[2] *Faculdade de Medicina, Universidade de São Paulo, Rua Teodoro Sampaio, 115 São Paulo, SP, Brazil CEP: 05405-000*

[2] *Fondazione ISI, Villa Gualino, viale Settimio Severo 65, 10133 Torino - Italia*

emails: toniolo.icg@libero.it, silviamr@dim.fm.usp.br, ezio.venturino@unito.it

Abstract

We present a dynamical system modeling a disease that has a weak and a strong form, for which a vaccine is available. The latter however does not confer permanent immunity, but allows disease relapses. The mathematical model is investigated, providing a framework for the understanding of the epidemilogy of the human papilloma virus infection.

Key words: epidemics, vaccination, basic reproduction number
MSC 2000: AMS codes (92C60, 92D25, 92D30)

1 Introduction

The human papilloma virus (HPV) is a virus belonging to the group of papillomaviruses couniting more than 100 types of viruses. Infections due to HPV are largely diffused, causing skin diseases among other disorders. The virus spreads by contact. If the infection is caused by papilloma of type 6, 11 or other milder ones, there are therapies. The infection caused by types 16, 18 or other high risk ones leads instead to tumors. All tumors of the cervix are caused by HPV. In the course of their life, an estimated 70% of women becomes infected by HPV, in most cases the infection being of short duration, although the latency period for the cervical cancer may last decades. The infection is usually asymptomatic, although detected by routine screening examinations, like the PAP test.

External treatments are available, like interferon and imiquimod, or podofillotoxine and podofilline. There are also available vaccines, like Gardasil, against genotypes 16-18 of HPV, which are responsible of about 70% of tumors, as well as genotypes 6 e

 ISBN 13: 978-84-613-5510-5

11, and Cervarix, active against genotypes 16 e 18. These are injected three times; for the vaccine to be fully effective, it is important that all the injections be performed. At first the vaccine was thought to provide immunity for life, more recent studies have confirmed an effetive immunity lasting only about five years, [1].

Simulation models for this situation have been considered in the literature, [3], [4]. Here, we would like to introduce a mathematical model to describe the situation, having in mind particularly that the vaccine can lose its effects. Using the power of the dynamical systems tools, we investigate its long term behavior.

2 The mathematical model

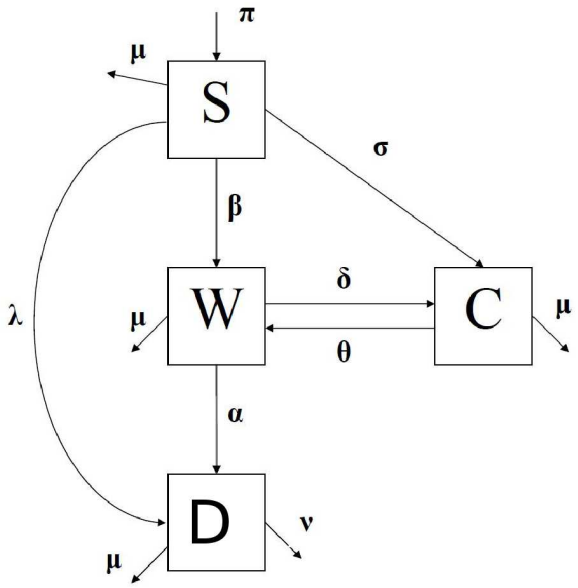


Figure 1: The compartments in the model with their possible transitions and transition rates.

To model the situation, we partition the total population into four classes: the susceptibles S, the weakly diseased W, the ones that have the disease in the strong form D, and those that have been exposed to the virus, either by vaccination or by recovery from the weak form of the disease, C. We assume that only individuals in classes W and D can spread the virus by contact. The form of the disease incidence assumed here is the mass action one, or Holling-type I. The disease is unrecoverable, i.e. it constitutes what is classically called an SI epidemic. The relationships between the various classes are depicted in Figure 1 outlining the possible transitions among them. Note in particular the contacts with weakly (strongly) infected lead to new weak (strong) cases of the disease, and the weak form of the disease may degenerate into a strong form. Also, as mentioned earlier, the "contaminated" individuals, those that

 ISBN 13: 978-84-613-5510-5

have come in contact with the virus, either recovering from the weak form of the disease or by the vaccine inoculation, may get a disease relapse. With these assumptions, the model reads then as follows.

$$\begin{aligned} \frac{dS}{dt} &= \pi - \beta SW - \lambda SD - (\mu + \sigma)\, S \\ \frac{dC}{dt} &= \delta W + \sigma S - (\vartheta + \mu)\, C \\ \frac{dW}{dt} &= \beta SW + \vartheta C - (\delta + \mu + \alpha)\, W \\ \frac{dD}{dt} &= \alpha W + \lambda SD - (\mu + \nu)\, D \end{aligned} \tag{1}$$

The first equation describes the dynamics of the susceptibles. The are recruited at a constant rate π, are subject like all other classes to natural mortality μ, may leave the class via vaccination at rate σ or by contacts with diseased individuals, at rates β and λ respectively for the weak form and the strong form of the infection.

The evolution of the contaminated individuals appears in the second equation. They can enter this class by vaccination, from the susceptible set, or by overcoming the weak form of the disease, at rate δ. From this class, as mentioned, only relapses into the weak form of the infection are allowed, at rate ϑ, apart from natural mortality.

The third equation models the evolution of weakly infected individuals. New cases can arise only from a susceptible after a contact with a weak infected at rate β. The class can be left by natural mortality, recovery at rate δ giving a new "exposed" or contaminated individual, or by regression to the strong form of the disease at rate α.

The strongly infected individuals, modeled by the last equation, enter only if the weak form of the disease degenerates, or if a susceptible gets the strong form of the disease by contact with a strongly infected one at rate λ. In addition to natural mortality μ, also disease-related mortality is here allowed, at rate ν.

The total population is $N = S + W + D + C$. It is constant if $\pi - \mu N - \nu D = 0$. In case in which the strongly affected ones are missing, this condition simplifies to $\pi - \mu N = 0$, giving a stable equilibrium $N^\dagger = \frac{\pi}{\mu}$. The dynamics of the total population when $D \neq 0$ is therefore bounded above by the one of the model with $D = 0$, so that it will have an equilibrium $N^* \leq N^\dagger$.

For the later stability analysis, we need the generic Jacobian of (1):

$$\mathbf{J} = \begin{pmatrix} -\beta W - \lambda D - \mu - \sigma & 0 & -\beta S & -\lambda S \\ \sigma & -\vartheta - \mu & \delta & 0 \\ \beta W & \vartheta & \beta S - (\delta + \mu + \alpha) & 0 \\ \lambda D & 0 & \alpha & \lambda S - (\mu + \nu) \end{pmatrix}$$

3 Boundedness

From the total population $N = S + C + W + D$, calculating the derivative we find

$$\frac{dN}{dt} = \frac{dS}{dt} + \frac{dC}{dt} + \frac{dW}{dt} + \frac{dD}{dt} = \pi - \mu S - \mu W - (\mu + \nu)D - \mu C = \pi - \mu N - \nu D.$$

 ISBN 13: 978-84-613-5510-5

For an arbitrary $0 < \varphi < \mu$ we find

$$\begin{aligned}\frac{dN}{dt} + \varphi N &= \pi - \mu(S+C+W+D) - \nu D + \varphi(S+C+W+D) = \\ &= \pi + S(\varphi-\mu) + C(\varphi-\mu) + W(\varphi-\mu) + D(\varphi-\mu-\nu) = \\ &= \pi + (\varphi-\mu)(S+C+W) + D(\varphi-\mu-\nu),\end{aligned}$$

so that $\frac{dN}{dt} + \varphi N \leq \pi$ and the differential inequality has a solution bounded above, implying that all subpopulations are bounded as well.

4 The system with no vaccination

Let us take the model without vaccination rate, $\sigma = 0$ and let us assume that the system is at the equilibrium $N^* \equiv \mu^{-1}[\pi - \nu D]$. Using the definition of total population, setting

$$a^* = \frac{\beta\frac{\pi}{\mu} + \vartheta\frac{\delta}{\vartheta+\mu} - (\delta+\mu+\alpha)}{\beta\left(1+\frac{\nu}{\mu}\right)}, \qquad b^* = \frac{\frac{\delta}{\vartheta+\mu}+1}{1+\frac{\nu}{\mu}},$$

we find the endemic equilibrium $\widehat{E}$ in absence of vaccine administration as

$$\begin{aligned}\widehat{W} &= \frac{a^*\left[\frac{\lambda}{\beta}\left(\frac{\beta\pi}{\mu} + \frac{\vartheta\delta}{\vartheta+\mu} - (\delta+\mu+\alpha)\right) - \left(\lambda\frac{\pi}{\mu} - \mu - \nu\right)\right]}{\alpha - b^*\left(\lambda\frac{\pi}{\mu} - \mu - \nu\right) + a^*\lambda\left(\frac{\delta}{\vartheta+\mu}+1\right)} \equiv \frac{h}{k}, \\ \widehat{D} &= a^* - b^*\frac{h}{k}, \qquad \widehat{C} = \frac{\delta}{\vartheta+\mu}\frac{h}{k}, \qquad \widehat{S} = N - \widehat{W} - \widehat{C} - \widehat{D}. \end{aligned} \tag{2}$$

In addition we find also the equilibria $\widehat{E}_0 \equiv \left(\frac{\pi}{\mu}, 0, 0, 0\right)$ and

$$E_1 = (S_1, 0, 0, D_1), \quad S_1 = \frac{\lambda+\mu}{\lambda}, \quad D_1 = \frac{\lambda\pi - \mu^2 - \mu\nu}{\lambda(\mu+\nu)}.$$

In this case the characteristic equation factors into the product of two quadratics,

$$\Lambda^2 + \Lambda(\lambda D_1 + \mu) + \lambda^2 D_1 S_1 = 0, \quad \Lambda^2 + \Lambda[\vartheta + 2\mu + \lambda + \delta + \alpha - \beta S_1] + [\delta + \mu + \alpha - \beta S_1](\vartheta + \mu + \lambda) - \vartheta\delta = 0.$$

The former is easily seen to have always negative roots, while for the latter this happens if only if

$$\vartheta + 2\mu + \alpha + \lambda + \delta > \frac{\beta}{\lambda}(\mu+\nu) > \frac{\vartheta(\mu+\alpha) + (\mu+\lambda)(\delta+\mu+\alpha)}{\vartheta+\mu+\lambda}.$$

4.1 Determination of $\mathcal{R}_0$

At $\widehat{E}_0$, we find two explicit eigenvalues, $\lambda\pi\mu^{-1} - \mu - \nu$ and $-\mu$, while the other ones are the roots of the quadratic

$$\Lambda^2 + \Lambda\left(\vartheta + 2\mu + \delta + \alpha - \beta\frac{\pi}{\mu}\right) - \vartheta\beta\frac{\pi}{\mu} + \vartheta\mu + \vartheta\alpha - \beta\pi + \mu\delta + \mu^2 + \mu\alpha = 0.$$

 ISBN 13: 978-84-613-5510-5

This equilibrium is therefore stable if the Routh Hurwitz criterion for the quadratic is satisfied, leading to

$$\beta < \beta_1 = \frac{\mu}{\pi}\left(\vartheta + 2\mu + \delta + \alpha\right), \tag{3}$$

$$\beta < \beta_2 = \frac{\mu}{\pi}\left(\frac{\vartheta\mu + \vartheta\alpha + \mu\delta + \mu^2 + \mu\alpha}{\mu + \vartheta}\right). \tag{4}$$

and if and only if the first eigenvalue is negative,

$$\lambda < \lambda_1 = \frac{\mu}{\pi}\left(\mu + \nu\right) \tag{5}$$

i.e., from the latter we can define a first basic reproduction number,

$$\mathcal{R}_0^{\lambda} = \frac{\lambda}{\lambda_1} < 1.$$

The basic reproduction number $\mathcal{R}_0$ represents the number of all secondary infections arising from a single infected individual in an entirely susceptible population, [5]. Furthermore, it is easy to verify that $\beta_2 < \beta_1$ and the equations (3) and (4) are satisfied if the condition (3) holds, i.e., if

$$\mathcal{R}_0^{\beta} = \frac{\beta}{\beta_2} < 1.$$

Therefore, if $\mathcal{R}_0^{\lambda} < 1$ and $\mathcal{R}_0^{\beta} < 1$ then the eigenvalues have negative real parts. If instead either one of $\mathcal{R}_0^{\lambda}$ or $\mathcal{R}_0^{\beta}$ exceeds one, then the disease becomes endemic in the population.

From (3) and (4) note that purely imaginary eigenvalues are impossible since the condition for them would lead to the inequality $\mu(\theta+\mu)+\theta(\theta+\mu+\alpha) < 0$, and therefore Hopf bifurcations do not arise around this equilibrium.

5 The interior equilibrium

For the case with vaccination, i.e. $\sigma \neq 0$, seeking the coexistence equilibrium in which all populations survive at nonzero level, a process of elimination of the nonlinear terms leads to the following solution

$$W^* = \frac{\vartheta\sigma S^*}{A - (\vartheta+\mu)\beta S^*}, \quad C^* = \frac{\delta}{\vartheta+\mu}W^* - \frac{\sigma S^*}{\vartheta+\mu}, \quad D^* = \frac{\alpha W^*}{((\mu+\nu) - \lambda S^*)}$$

where $A = \vartheta\mu + \vartheta\alpha + \mu\delta + \mu^2 + \mu\alpha$ and S^* now is a root of the following cubic equation

$$\widetilde{a}S^3 + \widetilde{b}S^2 + \widetilde{c}S + \widetilde{d} = 0, \tag{6}$$

in which we set

$$\widetilde{a} = \lambda\beta\mu\left(\vartheta+\mu\right)\left(\vartheta+\mu+\sigma\right), \qquad \widetilde{d} = -A\pi\left(\vartheta+\mu\right)\left(\mu+\nu\right),$$

$$\widetilde{b} = -\pi\beta\lambda\left(\vartheta+\mu\right)^2 - \mu\left(\vartheta+\mu+\sigma\right)\left(A\lambda + \vartheta\mu\beta + \vartheta\nu\beta + \mu^2\beta + \mu\nu\beta\right) - \vartheta\sigma\lambda\mu\left(\delta+\vartheta+\mu\right),$$

$$\widetilde{c} = \pi A\lambda\left(\vartheta+\mu\right) + \pi\left(\vartheta+\mu\right)^2\left(\mu+\nu\right)\beta + A\mu\left(\mu+\nu\right)\left(\mu+\nu+\sigma\right) +$$

$$+\vartheta\sigma\mu\left(\mu+\nu\right)\left(\delta+\vartheta+\mu\right) + \left(\vartheta+\mu\right)\left(\mu+\nu\right)\alpha\vartheta\sigma.$$

 ISBN 13: 978-84-613-5510-5

An alternative solution of the system leads to the following algebraic system

$$\Gamma_1 : aWC + bW^2 + cC + dW = 0, \quad \Gamma_2 : fC + gC^2 + hWC + kW^2 + pW = q \tag{7}$$

with

$$S = \frac{1}{\sigma}\left[(\vartheta+\mu)C - \delta W\right], \quad D = \frac{1}{\sigma(\mu+\nu)}\left[\sigma\pi - \left(\sigma\mu + \mu\vartheta + \mu^2\right)C + (\mu\delta - \mu\sigma)W\right]$$
$$a = \beta(\vartheta+\mu), \quad b = -\beta\delta, \quad c = \sigma\vartheta, \quad d = -\sigma(\mu+\delta+\alpha),$$
$$f = \sigma\left(\lambda\pi\vartheta + \lambda\pi\mu + \mu^2\sigma + \mu^2\vartheta + \mu^3 + \sigma\nu\mu + \nu\mu\vartheta + \nu\mu^2\right), \quad q = \sigma^2(\mu+\nu)\pi,$$
$$g = -\lambda(\vartheta+\mu)\left(\sigma\mu + \mu\vartheta + \mu^2\right), \quad h = \lambda\left(2\vartheta\mu\delta - \vartheta\mu\sigma + 2\mu^2\delta - \mu^2\sigma + \sigma\delta\mu\right),$$
$$k = -\lambda\delta\mu(\delta - \sigma), \quad p = -\sigma\left(\lambda\delta\pi + \mu^2\delta - \mu^2\sigma + \nu\mu\delta - \nu\mu\sigma - \sigma\alpha\mu - \sigma\alpha\nu\right).$$

Since all parameters are positive, the sign of some coefficients in (7) can be determined,

$$a > 0, \quad b < 0, \quad c > 0, \quad d < 0, \quad f > 0, \quad g < 0, \quad q > 0, \tag{8}$$

while for k we find

$$k > 0 \quad \text{for} \quad \sigma > \delta, \qquad k < 0 \quad \text{for} \quad \sigma < \delta. \tag{9}$$

The signs of h and p are unknown. But we will see that $p < 0$ is allowed, as the case $p > 0$ does not lead to new findings with respect to what already found above.

System (7) represents the intersection of two conic sections. Let the general conic be

$$Ax^2 + 2Hxy + By^2 + 2Gx + 2Fy + C = 0. \tag{10}$$

Its invariants are then

$$\mathbf{D} = \begin{vmatrix} A & H & G \\ H & B & F \\ G & F & C \end{vmatrix}, \qquad \mathbf{C} = AB - H^2, \qquad \text{and} \qquad \mathbf{I} = A + B. \tag{11}$$

To better understand this problem, we observe that one of the invariants of Γ_1 is

$$\mathbf{C} = -\frac{a^2}{4} < 0,$$

showing that it is a hyperbola, with

$$\mathbf{D} = \frac{acd}{8} + \frac{acd}{8} - \frac{bc^2}{4} = \frac{c}{4}(ad - bc) \tag{12}$$

so that it is not degenerate if $\mathbf{D} \neq 0$ i.e. if $ad \neq bc$. Similarly, for Γ_2 we find

$$\mathbf{C} = kg - \frac{h^2}{4}, \tag{13}$$

so that in view of the signs of the coefficients, (8) and (9) we have: if $\sigma > \delta$ then $\mathbf{C} < 0$ giving a hyperbola; if $\sigma < \delta$ then if $\mathbf{C} > 0$ it gives an ellipse, otherwise a hyperbola.

 ISBN 13: 978-84-613-5510-5

To investigate their intersections, note that Γ_1 can be written as a function $W = W(C)$, which crosses the C axis at $W = 0$ and $W = -\frac{d}{b}$. For later purposes, we also calculate the derivative of the conic at these points,

$$\left.\frac{\partial \Gamma_1}{\partial C}\right|_{(0,0)} = -\frac{c}{d} > 0, \quad \left.\frac{\partial \Gamma_1}{\partial C}\right|_{\left(0,-\frac{d}{b}\right)} = W'\left(-\frac{d}{b}\right) = \frac{c}{d} - \frac{a}{b}. \tag{14}$$

For Γ_2 instead we find two intersections with $C = 0$, namely

$$W_1 = \frac{-p + \sqrt{p^2 + 4kq}}{2k}, \qquad W_2 = \frac{-p - \sqrt{p^2 + 4kq}}{2k}.$$

These roots have opposite signs in the following two cases: $p < 0$, $q > 0$ and $k > 0$ with $\sigma > \delta$; $p > 0$, $q > 0$ and $k > 0$ with $\sigma > \delta$. We also find

$$\left.\frac{\partial \Gamma_2}{\partial C}\right|_{(0,W_1)} = -\frac{hW_1 + f}{2kW_1 + p}, \quad \left.\frac{\partial \Gamma_2}{\partial C}\right|_{(0,W_2)} = -\frac{hW_2 + f}{2kW_2 + p}.$$

The intersection with the vertical axis does not produce points other that the origin for Γ_1, while for Γ_2 we find

$$C_1 = \frac{-f + \sqrt{f^2 + 4gq}}{2g}, \quad C_2 = \frac{-f - \sqrt{f^2 + 4gq}}{2g},$$

with the same signs in view of $g < 0$, $f > 0$, $q > 0$. Moreover,

$$\left.\frac{\partial \Gamma_2}{\partial W}\right|_{(C_1,0)} = \frac{-p - hC_1}{2gC_1 + f}, \qquad \left.\frac{\partial \Gamma_2}{\partial W}\right|_{(C_2,0)} = \frac{-p - hC_2}{2gC_2 + f}.$$

The oblique asymptotes can also be determined. For Γ_1 we find

$$W = -\frac{d}{2b} \quad W = -\frac{a}{b}C - \frac{d}{2b}.$$

from which the second derivative (14) is seen to be positive. Figure 2 shows a possible graph of the conic Γ_1.

For Γ_2 the oblique asymptotes are

$$W_\pm = \frac{-h \pm \sqrt{h^2 - 4kg}}{2k}C - \frac{p}{2k}.$$

There arise the following two cases.

If $h < 0$, $k > 0$ for $\sigma > \delta$, $g < 0$ and $p < 0$ the asymptote W_+ has positive slope, while W_- has negative slope, since $\left|\sqrt{h^2 - 4kg}\right| > h$. Their intercepts with the W axis have a positive height. If instead $p > 0$, the asymptotes cross the W axis at the negative height $-\frac{p}{2k}$.

If $h > 0$, $k > 0$ for $\sigma > \delta$, $g < 0$ and $p < 0$ the asymptote W_+ has positive slope, while $W_=$ has a negative one and $-\frac{p}{2k} > 0$, while for $p > 0$ we have instead $-\frac{p}{2k} < 0$.

 ISBN 13: 978-84-613-5510-5

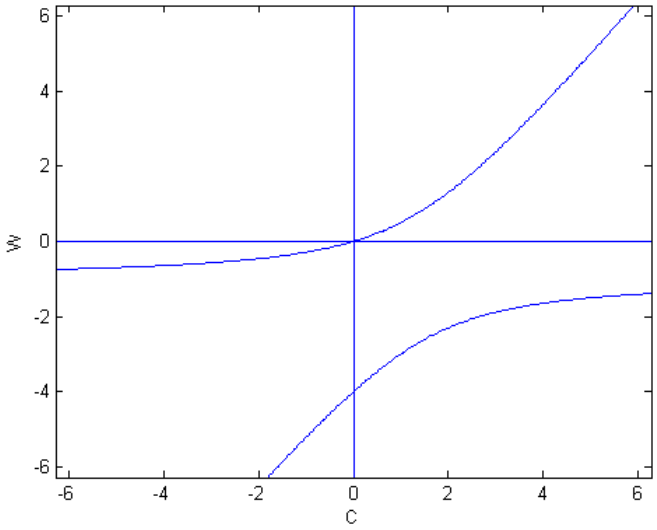


Figure 2: Graph of the conic $\Gamma_1 : aWC + bW^2 + cC + dW = 0$.

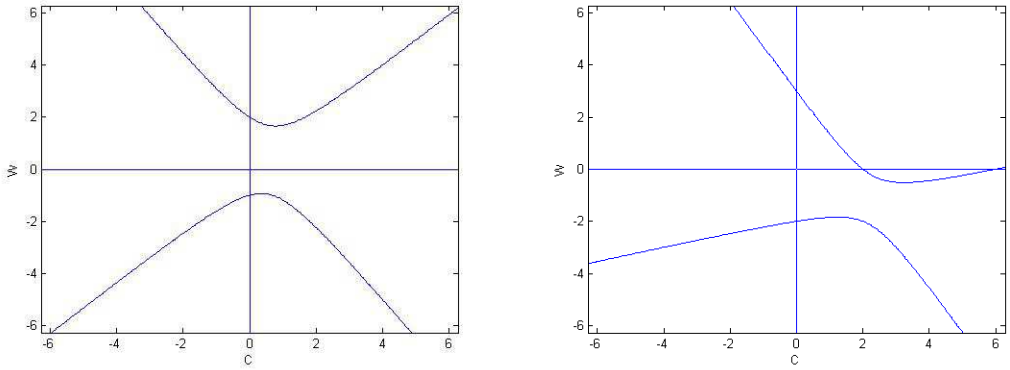


Figure 3: Γ_2: the two situations with $\Delta < 0$ (left) and $\Delta > 0$ (right).

Γ_2 intercepts $C = 0$ and $W = 0$ respectively at

$$W_{3,4} = \frac{-p \pm \sqrt{p^2 + 4kq}}{2k}, \quad C_{3,4} = \frac{-f \pm \sqrt{\Delta}}{2g}, \quad \Delta = f^2 + 4gq.$$

The sign of the discriminant Δ depends on $g < 0$ and $q > 0$. For $\Delta > 0$ the lower branch of Γ_2 intersects the W axis, otherwise it lies entirely in the lower half plane $C < 0$, see Figure 3. The possible intersections of the two conics are depicted in Figure 4. Combining all these considerations, sufficient conditions for an intersection in the first quadrant can be derived.

Since $-\frac{p}{2k} > 0$, a sufficient condition is provided by $p < 0$ and $-\frac{a}{b} > \frac{-h+\sqrt{h^2-4kg}}{2k}$.

 ISBN 13: 978-84-613-5510-5

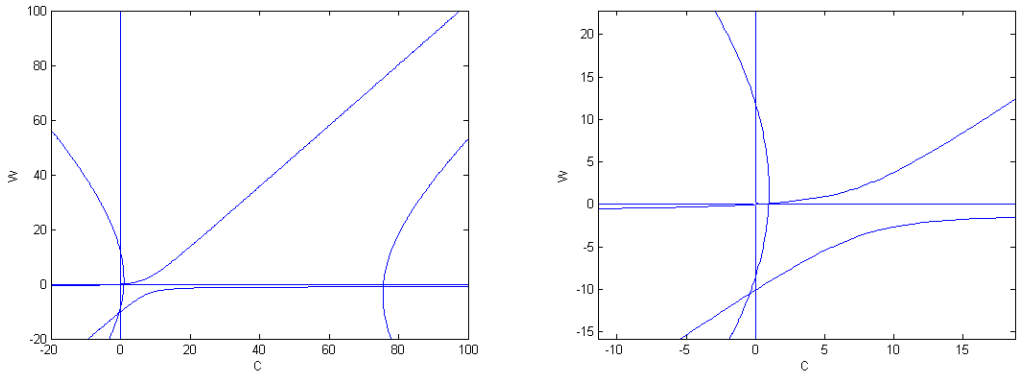


Figure 4: Conics intersections (left). On the right a blow up of a neighborhood of the origin.

Also, for $p > 0$ and $-\frac{d}{2b} < -\frac{p}{2k}$ it follows $-\frac{a}{b} > \frac{-h+\sqrt{h^2-4kg}}{2k}$. For $p > 0$ and $-\frac{d}{2b} > -\frac{p}{2k}$ we instead have $-\frac{a}{b} < \frac{-h+\sqrt{h^2-4kg}}{2k}$. The last two inequalities represent necessary but not sufficient conditions for an intersection to occur in the first quadrant.

6 Simulations

We have carried out a few numerical simulations to substantiate our theoretical findings. We show some results in the Figures 5-6, where counterclockwise from the upper left corner, we have plotted the populations S, C, W and D. The simulations have been run over very long time spans, in order to pinpoint the stable nature of the equilibria.

The interior equilibrium, Fig. 5, is obtained for the following set of parameter values

$$\pi = 1000, \quad \alpha = 0.8, \vartheta = 0.02, \quad \beta = 0.01, \quad \delta = 0.01,$$
$$\lambda = 0.02, \quad \mu = 0.0118, \quad \nu = 0.07, \quad \sigma = 0.06.$$

The interior equilibrium for the particular case $\sigma = 0$, Fig. 6, is obtained for the same parameters as above but the last one.

The equilibrium E_1, for $\sigma = 0$, Fig. 7, is instead obtained for the following choice of the parameters

$$\pi = 1000, \quad \alpha = 0.01, \vartheta = 0.02, \quad \beta = 0.01, \quad \delta = 0.01,$$
$$\lambda = 0.02, \quad \mu = 0.0118, \quad \nu = 0.2, \quad \sigma = 0.0.$$

7 Conclusions

In the model proposed we have discovered that a disease-free equilibrium exists, as well as an endemic one. Conditions for the existence of the latter have been identified, basic

 ISBN 13: 978-84-613-5510-5

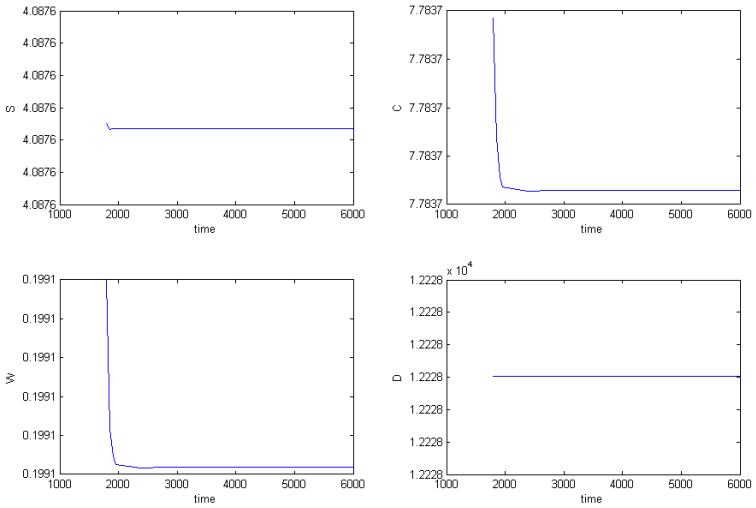


Figure 5: Internal equilibrium of system (1).

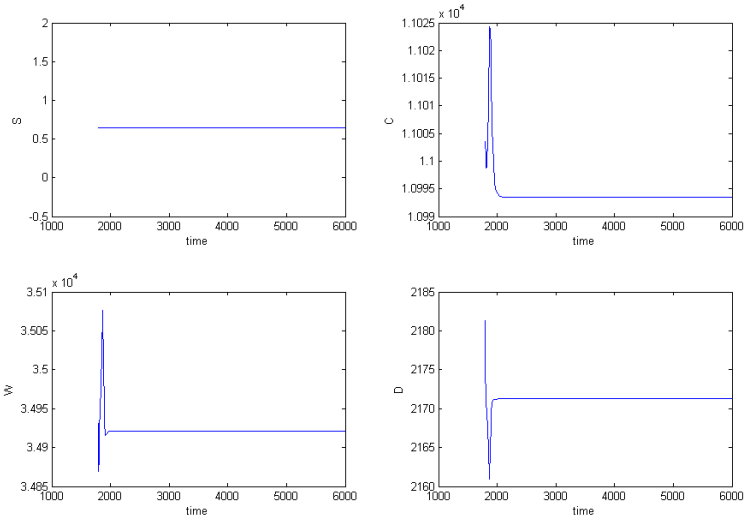


Figure 6: Internal equilibrium of system (1) for the particular case $\sigma = 0$.

reproduction numbers for the disease to become endemic have been identified.

References

[1] M. Adams, B. Jasani, A. Fiander *Human papilloma virus (HPV) prophylactic vaccination: challenges for public health and implications for screening*, Vaccine

 ISBN 13: 978-84-613-5510-5

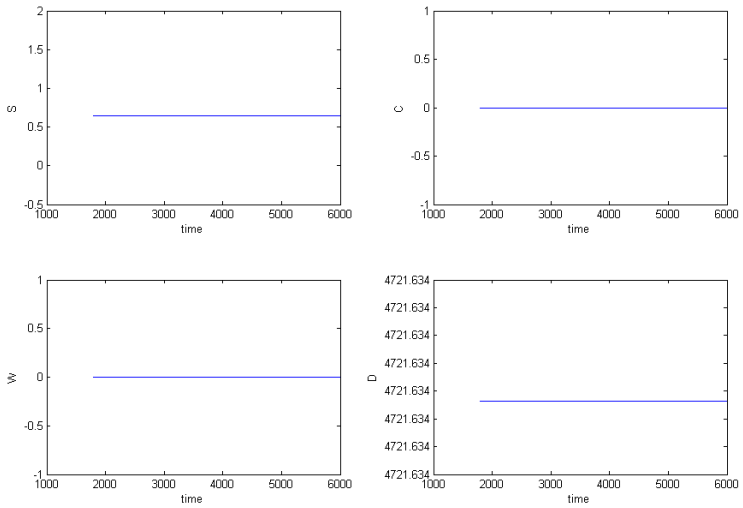


Figure 7: The equilibrium E_1 of system (1) for $\sigma = 0$.

25 (2007) 3007–30013.

[2] R. Cavoretto, S. Remogna, E. Venturino, *A model of a mildly-severely staged disease*, Proceedings Mathmod 09 Full Papers CD Volume, Argesim Report **35** (2009) Vienna, Austria, 1786–1798.

[3] E. H. Elbasha, E. J. Dasbach, R. P. Insinga, *Model for assessing human papillomavirus vaccination strategies*, Emerging infectious diseases **13** (1) (2007) 28–41.

[4] G. P. Garnett, J. J. Kim, K. French, S. J. Goldie, *Chapter 21: modelling the impact of HPV vaccine on cervical cancer and screening programmes*, Vaccine **24S3** (2006) S3/178–S3/186.

[5] H. Inaba, H. Nishiura *The basic reproduction number of an infectious disease in a stable population: the impact of population growth rate on the eradication threshold*, Math. Model. Nat. Phenom. **3** (7) (2008) 194–228.

[6] H. Malchow, S. Petrovskii, E. Venturino, *Spatiotemporal patterns in Ecology and Epidemiology*, CRC, Boca Raton, 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Automatic Data Layout at Multiple Levels for CUDA

Yuri Torres[1], Arturo González-Escribano[1] and Diego R. Llanos[1]

[1] *Departamento de Informática, University of Valladolid*

emails: yuri.torres@alumnos.uva.es, arturo@infor.uva.es, diego@infor.uva.es

Abstract

Trasgo is a source-to-source compiler system that translates simple high-level specifications of parallel algorithms to lower-level native programs, with data partition and communication details generated automatically. Hitmap is the run-time library used by the back-ends of Trasgo for hierarchical tiling and mapping of arrays, currently built on top of the MPI message-passing interface. Hitmap includes a plug-in system for automatic data-layouts. In this paper we extend Hitmap with a new type of data-layout techniques suitable for the CUDA parallel programming model. The combination with the previous type of data-layout techniques allow to generate data distributions, at multiple levels of parallelism, for GPU clusters. The new Hitmap version hides to the programmer the details about the machine structure and thread management, allowing to easily generate programs with multiple levels of parallelism in heterogeneous systems. This work opens the road to develop a new back-end for the Trasgo compiler system to automatically generate CUDA programs.

Key words: Data layout, CUDA, GPUs, heterogeneous systems

1 Introduction

1.1 The Hitmap run-time library

Trasgo [GL09] is a parallel programming system based on high-level and nested-parallel specifications. It provides a C-like front-end language with nested-parallel coordination extensions. The front-end language allows to easily represent abstract specifications of parallel algorithms, with no detail about threads management or inter-process communications. It uses a common scheme to express hierarchical combinations of data- and task-parallelism. The high-level coordination language provided by Trasgo is translated internally to an XML intermediate representation, to allow easier data-flow analysis and code rewriting. Different back-ends may translate the result to native code using different parallel tools or models. Currently, Trasgo have a complete back-end that efficiently exploits the MPI message-passing interface.

 ISBN 13: 978-84-613-5510-5

The Trasgo back-ends are supported by a runtime library for hierarchical tiling and mapping of arrays, named Hitmap. This library also includes a plug-in system of modules for automatic creation of virtual topologies, and data-partition and layout. The modules are invocated in the code, but applied at run-time with architecture and topology information supplied by the underlying system. Moreover, the resulting layout objects contain all the information needed to map data to the local processor, and to find neighbors on the virtual topology which have data affinities. Thus, the programmer never reasons in terms of system resources, and does not need to know the implementation details of the partition, scheduling, or communication.

Virtual topology functions identify the hosts or cores that are available for the parallel code execution, and use the available topology information to generate a mapping function. Layout functions use the virtual topology information and the index domain of a data structure to generate tiles of the proper grain size for the virtual processors. Hitmap includes several virtual topology functions (such a parallelepiped multidimensional topologies with different restrictions), and several data-layout functions (such as blocks, cyclic, exponential distributions, or dynamic workload balancing of weighted tasks). These techniques work for any special circumstances. For example, they do not need the data elements to be a multiple of the number of virtual processors, and they automatically may assign groups of processors to single data elements if necessary.

2 CUDA programming model

CUDA [NBGS08, NVI10] was introduced by NVIDIA to exploit the parallel compute engine in NVIDIA GPUs. Although, CUDA design approach is appropriate for efficient GPU programming, it is conceived as a general purpose parallel computing model. However, there are important differences with other popular parallel computing models, such as message-passing, OpenMP, or PGAS. CUDA works with a shared-memory architecture model. In other shared-memory programming models, such as OpenMP, each task is expected to be launched in an independent CPU core. The memory hierarchy is hidden, and the programmer typically takes into account the number of cores to produce coarse-grain computations to process data in big memory chunks. On the other hand, in CUDA each task will be launched in a SM (streaming multiprocessor) composed by a fixed number (eight) of cores. Inside the SM, the computation model is SIMD (Single Instruction, Multiple Data). Each SM has its own small shared memory, and synchronization system. All SMs in the same device (GPU card) share a bigger global memory. Several devices may work in parallel, receiving, processing and returning data pieces to the main host memory. CUDA places on the programmer the burden of managing the memory hierarchy, and taking decisions about the how to organize the fine grain synchronized tasks which are grouped and pipelined through the streaming multiprocessors. Practical experience shows that this approach is often tedious and error-prone, needing abstractions to hide details and help the programmer [HA09]. Moreover, working with several GPU devices in parallel adds another level of complexity.

 ISBN 13: 978-84-613-5510-5

3 Data-Layout combinations at multiple levels

In previous versions of Hitmap, the run-time system was oriented to create coarse-grain data partitions. It was designed for efficient SPMD implementations in programming models based on interprocess communication, such as message-passing. In this work we extent the functionalities of Hitmap, to support combinations of multiple levels of coarse-grain, and fine-grain layouts.

Consider a system with several GPU cards. Although, the synchronization and communication system across them is limited, we may describe the topology of GPU devices. In CUDA, it is possible to automatically obtain information about the current GPU devices at run-time. Thus, the current Hitmap topology functions and coarse-grain data-partition modules are perfectly suitable to automatically distribute computations, across several GPU devices, with coarse-grain techniques.

However, inside the GPU device we have a different level of parallelism. The computation pieces should be distributed with a different, fine-grain, approach. In CUDA, the number of SPs and SMs in a device is not as important as the number of threads supported by a single SM. Threads are grouped in packs which are executed in the same SM. Groups are pipelined through the processing elements. Thus, threads are grouped and executed in a two-level nested-parallel model. The threads on each group share a local memory, and all groups share the device global memory. Communication or synchronization across groups is possible with atomic operations on the global memory. Thus, the identification of the group and thread are relevant for the computation, not the identification of the processing element.

We define a μlayout (or *ulayout*) as a function to make a domain partition in terms of the number of data elements to be processed together, instead of the number of processing elements (number of SPs and SMs). The output of a μlayout function is a structure of groups of domain elements. Each group will have an appropriate number of domain elements to be processed as a block, and CUDA will be responsible of the assignment of each block to the corresponding SMs. Restrictions to the group size, or shape, may be also imposed by the application, by other μlayout results, by the SM local memory limits, or by the programming model itself (in CUDA the maximum number of threads in a block is limited to 512). We define the first μlayout functions for multidimensional blockings, or cyclic assignment of elements. The Hitmap plug-in system for classical layouts has been replicated for μlayouts. Programmers may add new μlayout functions as modules.

The output of a μlayout may be used by the Hitmap coarse-grain layout functions to distribute the groups across several processing elements or devices. A parallel computation may be deployed on a GPU system using: (1) the new Hitmap μlayout functions to adapt the computation grain and the distribution of the data to the internals of the GPU device, and (2) the topology and layout functions to distribute sets of medium-grain computations across several devices. Thus, a good data-locality may be achieved at the lower level, and a good load balance at the higher level.

 ISBN 13: 978-84-613-5510-5

4 Conclusions

Hitmap is a run-time library for hierarchical tiling, and automatic mapping of tiled arrays. It is designed to support code generation by the back-ends of Trasgo, a source-to-source compiler system.

This work introduces a new type of data-layout techniques into Hitmap. Previous techniques in Hitmap focused on the automatic creation of coarse-grain data distributions in terms of the underlaying machine topology information. The new type focuses on the creation of groups of fine-grain computations to map into a GPU device. The combination of both types allow to develop multiple levels of automatic data-layout techniques for heterogeneous systems in CUDA. This work opens the possibility to develop a Trasgo back-end to generate efficient CUDA programs.

Acknowledgements

This research is partly supported by the Ministerio de Educación y Ciencia, Spain (TIN2007-62302), Ministerio de Industria, Spain (FIT-350101-2007-27, FIT-350101-2006-46, TSI-020302-2008-89, CENIT MARTA, CENIT OASIS), Junta de Castilla y León, Spain (VA094A08), and also by the Dutch government STW/PROGRESS project DES.6397. Part of this work was carried out under the HPC-EUROPA project (RII3-CT-2003-506079), with the support of the European Community - Research Infrastructure Action under the FP6 "Structuring the European Research Area" Programme. The authors wish to thank the members of the Trasgo Group for their support, and Dr. Valentín Cardeñoso-Payo, and Prof. Arjan van Gemund, for many helpful discussions during the early stages of this research.

References

[GL09] Arturo González-Escribano and Diego Llanos. Trasgo: a nested-parallel programming system. *The Journal of Supercomputing*, 2009.

[HA09] Tianyi David Han and Tarek S. Abdelrahman. hiCUDA: a high-level directive-based language for gpu programming. In *GPGPU-2: Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units*, pages 52–61, New York, NY, USA, 2009. ACM.

[NBGS08] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with CUDA. *ACM Queue*, 6(2):40–53, 2008.

[NVI10] NVIDIA. NVIDIA CUDA ProgrammingGuide 3.0. 2010.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Mining Rules for Disclosing Sensitive Information

Luigi Troiano[1], Luis J. Rodríguez-Muniz[2], José Ranilla[3] and Irene Díaz[3]

[1] *Department of Engineering, University of Sannio*

[2] *Department of Statistics and O.R., University of Oviedo*

[3] *Department of Computer Science, University of Oviedo*

emails: troiano@unisannio.it, luisj@uniovi.es, ranilla@uniovi.es, sirene@uniovi.es

Abstract

In this paper we address the problem of controlling the disclosure of sensible information by inferring them by the other attributes made public. This threat to privacy is commonly known as prediction or attribute disclosure. Our approach is based on identifying those rules able to link sensitive information to the other attributes being released. In particular, the method presented in this paper is based on mining fuzzy rules. The fuzzy approach is compared to (crisp) decision trees in order to highlight pros and cons of it.

Key words: disclosure control, fuzzy rules, privacy

1 Introduction

In order to provide a richer set of data to analyze, statistical agencies and offices release information regarding individuals, companies and other organizations. If availability of microdata makes possible to investigate trends and relationships more accurately, on the other side it poses relevant concerns regarding the risk of revealing sensitive information about the respondents. Indeed, publishing aggregate or individual data carries always the risk that individuals or organizations could be identified and confidential information about them could be released. Therefore, on one side there is a need of providing information in order to perform statistical analysis, whereas on the other it is necessary that some relevant information is not revealed.

Statistical Disclosure Control (SDC) aims at releasing statistical records while protecting confidentiality of information at the same time. Among the different threats, there is the possibility that some sensitive information can be obtained by other known data regarding some entity. In this case, the risk is that hidden information is inferred

 ISBN 13: 978-84-613-5510-5

using public information as premise. Discovering a link between hidden and public information is possible can help SDC to prevent such a risk.

In this work we explore the rules mined from data as means to identify possible paths that if own by an intruder would be able to disclose sensitive information. In addition, we want to check if the fuzzyfication of the public part of a database can help to protect the confidential information. The reminder of this extended abstract is organized as follows: First some preliminaries regarding information disclosure are presented; Secondly, Data Mining is presented as means to prevent disclosure of sensitive information; finally experimental set-up is designed.

2 Information Disclosure

Information disclosure has place when an entity (i.e. a person or an organization) is able to learn something regarding another entity by released microdata sets. For example, illness regarding patients could be released via medical databases, or competitors' financial figures by business databases.

Microdata attributes of interest for statistical disclosure control can refer to respondent identity (key attributes), or to relevant information (sensitive attributes). In order to preserve the respondent's privacy, the direct linkage between key and sensitive attributes is hidden by SDC. This process is known as data anonymization. However, an intruder can still attack data anonymization by reconstructing the original link with respect to some records.

In particular, there are two types of disclosure associated to microdata [13]: (i) identity disclosure when the entity is (re-)associated to some sensitive data in an anonymized database; (2) prediction disclosure when some sensitive data is inferred by the other attributes for some known entity. The first is also known as *re-identification*, the second as *attribute disclosure*.

Different metrics for measuring the level of privacy guaranteed by SDC have been proposed over the time. Among them, k-anonymity [11], l-diversity [4], p-sensitive [12] and t-closeness [3] each of these metrics is able to drive data anonymization with respect to same aspect, but all of them share the common idea that having more records within a group associable to an entity enforce privacy protection.

However privacy should be related to the extent some information can be considered sensitive. For instance, disclosing that incomes are within a given range, can be considered as much as sensitive than more precise information. This case is known in literature as similarity attack.

Therefore diversification, obtaining by altering the initial information, does not necessarily lead to a stronger privacy protection. Even masking or removing a sensitive attribute could be not enough to avoid attribute disclosure.

The aim of this paper is to show evidence that, even if there is no correlation between data, it is still possible to find a link, although approximated, between public and sensitive variables. The simpler this link is, the most likely it can be discovered or known by intruder, representing thus a threat to no-disclosure of sensitive information.

 ISBN 13: 978-84-613-5510-5

3 Data Mining

Domingo [1] establishes the connections between data mining and statistical disclosure. The problem in attribute disclosure is basically finding an inferential path from released attributes to sensitive information. Such a path can be due to background knowledge. Mining rules, able to reconstruct the hidden linkage from given patterns of the other attributes, can put into evidence that if some knowledge is discovered by intruders, this can be used to break privacy protections. In addition, similarity is inherently a fuzzy concept.

Data mining approaches can be seen as a way of knowledge discovery which is essential for solving problems. Data mining techniques build a model to predict or classify a problem like an expert. In this sense, data mining techniques may infer relationships allowing us to study the problem of Information Disclosure.

There exists a huge number of machine learning approaches to cope with the problem of classifying an example. Some of them, as Neural Networks or Support Vector Machines (SVM) are very effective and efficient but the model they build is little informative for the Statistical Disclosure problem (for example, SVM provides the weights of the support vectors) [2].

On the other hand, Literature reports a considerable number of ID3-based systems [5] and several fuzzy versions of decision trees [6].

4 Our Contribution

In this paper we use a data mining strategy based on building a decision tree to try to infer a path from released attributes to sensitive information. In addition we will study the pros and cons of crisp and fuzzy decision trees to protect sensitive information.

We think that the latter approach has two main advantages: first, the induction rules provide us with some information about the most sensible attributes, and, second, as the input information is fuzzified the disclosure risk is supposed to be decreased. This is due in many cases of practical interest it is not important that precise values of sensible attributes are disclosed but rather the ranges they belong to, discovering treats of disclosing sensitive information can be regarded as a classification problem, where an intruder could be able to associate non-sensitive data to a class of sensitive information.

To check these hypothesis, first of all it is necessary to identify the variables to protect. Once selected, we want to measure how strong is this protection if we attack the data with a data mining rule. Since the data mining system predicts only one class at once, we must construct one classifier for each variable to protect.

To provide the fuzzy rules, it is used a system based on $C4.5$ [7], the so-called ARNI, and its fuzzy extension, the so-called *FArni*. But despite of using *Information gain* as in $C4.5$, both systems ARNI and *FArni* use a measure called *Imputity Level* (*IL*) for determining the quality of the rules induced from examples [10]. *IL* [8] explicitly takes into account not only the probability of success p, but also the difficulty of attaining that amount of examples of class C. Later, once the fuzzy decision tree is induced,

 ISBN 13: 978-84-613-5510-5

FArni returns compact fuzzy rule sets after applying a pruning process inherited from ARNI and Fan [9]. *FArni* is presented in detail in [10].

The experiments will show which system mines simpler rules. The threat of disclosing attributes is related to the possibility of being aware or building such links as background knowledge. Simplicity of rules derives from the number of predicates involved in the antecedents and by interpretability of them. The system which results in less structured and easier to understand rules will be able to find a higher number of threats from released data to sensitive information.

Acknowledgements

Authors acknowledge financial support by Grant MTM2008-01519 from Ministry of Science and Innovation and Grant TIN2007-61273 from Ministry of Education and Science, Government of Spain

References

[1] Josep Domingo-Ferrer and Vicenç Torra. On the connections between statistical disclosure control for microdata and some artificial intelligence tools. *Inf. Sci. Inf. Comput. Sci.*, 151:153–170, 2003.

[2] T. Joachims. Making large-scale support vector machine learning practical. In A. Smola B. Scholkopf, C. Burges, editor, *Advances in Kernel Methods: Support Vector Machines*. MIT Press, Cambridge, MA, 1998.

[3] Ninghui Li and Tiancheng Li. t-closeness: Privacy beyond k-anonymity and ?-diversity. In *In Proceedings of IEEE International Conference on Data Engineering*, 2007.

[4] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *22nd IEEE International Conference on Data Engineering*, 2006.

[5] Nikhil R. Pal and Sukumar Chakraborty. Fuzzy rule extraction from id3-type decision trees for real data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 31(5):745–754, 2001.

[6] D. T. Pham, S. Bigot, and S. S. Dimov. Rules-f: a fuzzy inductive learning algorithm. In *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, pages 1433–1444, 2006.

[7] Ross Quinlan. *c4.5: Programs of machine learning*. 1993.

[8] J. Ranilla, O. Luaces, and A. Bahamonde. A heuristic for learning decision trees and pruning them into classification rules. *AICom (Artificial Intelligence Communication)*, 16(2):in press, 2003.

 ISBN 13: 978-84-613-5510-5

[9] Jose Ranilla and Bahamonde Antonio. Fan: Finding accurate inductions. *International Journal of Human Computer Studies*, 56(4):445–474, 2002.

[10] Jose Ranilla and Luis J. Rodriguez-Muniz. A heuristic approach to learning rules from fuzzy databases. *IEEE Intelligent Systems*, 22(2):62–68, 2007.

[11] Pierangela Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13:1010–1027, 2001.

[12] Xiaoxun Sun, Hua Wang, Jiuyong Li, and David Ross. Achieving p-sensitive k-anonymity via anatomy. In *Proceedings of the 2009 IEEE International Conference on e-Business Engineering*, pages 199–205, Washington, DC, USA, 2009. IEEE Computer Society.

[13] Leon Willenborg and Ton de Waal. *Statistical Disclosure Control in Practice*. Springer Verlag, 1996.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Nabla Analytic Functions on Time Scales

Ünal Ufuktepe [1] **and Sinan Kapcak**[1]

[1] *Department of Mathematics, Izmir University of Economics, Izmir, TURKEY*

emails: unal.ufuktepe@ieu.edutr, skapcak@hotmail.com

Abstract

The concept of analyticity for complex functions on time scale complex plane was introduced by Bohner and Guseinov in 2005. They developed completely delta differentiability, delta analytic functions on products of two time scales, and Cauchy-Riemann equations for delta case.

In this the paper we study on continuous, discrete and semi-discrete analytic functions and developed completely nabla differentiability, nabla analytic functions on products of two time scales, and Cauchy-Riemann equations for nabla case.

1 Introduction

Gusein Sh. Guseinov and Martin Bohner gave the differential calculus, integral calculus and developed line integration along time scale curves [2, 3, 7] . They also developed the concept of analytic functions on Time Scales in [3].
The study of analytic functions on $\mathbb{Z}^2$ has a history of more than sixty years. The pioneer in that field is Rufus Isaacs [10], who introduced two difference equations, both of which are discrete counterparts of the Cauchy-Riemann equation in one complex variable.

The discrete analogues of analytic functions have been called by several names. Duffin calls them discrete analytic functions [5], Ferrand calls them preholomorphic functions [6], and Isaacs calls them monodiffric functions [10]. We use the definition given by Isaacs.

2 Discrete Analytic Functions

Definition 2.1 *A complex-valued function f defined on a subset A of $\mathbb{Z} + i\mathbb{Z}$ is said to be holomorphic in the sense of Isaacs or monodiffric (discrete analytic) of the first*

 ISBN 13: 978-84-613-5510-5

kind if the equation

$$\frac{f(z+1)-f(z)}{1}=\frac{f(z+i)-f(z)}{i} \tag{1}$$

holds for all $z \in A$ such that also $z+1$ and $z+i$ belong to A.

It should be noted that this definition is not the one used by Duffin or Ferrand in their works on discrete analytic functions.

In [10] Isaacs defined also monodiffric functions of the second kind, in which the condition

$$\frac{f(z_0+1+i)-f(z_0)}{1+i}=\frac{f(z_0+i)-f(z_0+1)}{i-1} \tag{2}$$

is required instead of (1).

Here, we propose a new kind of monodiffric function which we call backward monodiffric function. To avoid confusion we call the monodiffric function of first kind as forward monodiffric function.

Definition 2.2 *A complex-valued function f defined on a subset A of $\mathbb{Z}+i\mathbb{Z}$ is said to be backward monodiffric if the equation*

$$\frac{f(z)-f(z-1)}{1}=\frac{f(z)-f(z-i)}{i} \tag{3}$$

holds for all $z \in A$ such that also $z-1$ and $z-i$ belong to A.

There are many articals which concern discrete analogues for analytic functions by Duffin and Isaacs. In each of these, either a discrete analogue to the Cauchy-Riemann equations or in the case of Ferrand, a discrete version of Morera's theorem is used to define discrete analytic functions (as (1) and (2)).

3 Semi-Discrete Analytic Functions

Semi-discrete analytic functions are single-valued functions of one continuous and one discrete variable defined on a semi-lattice, a uniformly spaced sequence of lines parallel to the real axis.
The appropriate semi-discrete analogues of analytic functions are defined in [12] from the classic Cauchy-Riemann equations on replacing the y-derivative by either a non-symmetric difference

$$\frac{\partial f(z)}{\partial x}=[f(z+ih)-f(z)]/ih, \quad z=x+ikh, \tag{4}$$

or a symmetric difference

$$\frac{\partial f(z)}{\partial x}=[f(z+ih/2)-f(z-ih/2)]/ih, \quad z=x+ikh/2. \tag{5}$$

 ISBN 13: 978-84-613-5510-5

Definition 3.1 *Semi-discrete functions which satisfy (4) or (5) are called, respectively, semi-discrete analytic functions of the first, second kind.*

We study the semi-discrete analytic functions of the first kind since we propose the backward version of semi-discrete analytic functions, to avoid confusion we call the first kind, forward semi-discrete analytic functions.

Definition 3.2 *The function defined on* $\mathbb{R} + ih\mathbb{Z}$ *that satisfies the condition*

$$\frac{\partial f(z)}{\partial x} = [f(z) - f(z - ih)]/ih, \quad z = x + ikh, \tag{6}$$

is called backward semi-discrete analytic function.

Let $f(z) = u(x, y) + iv(x, y)$ be semi-discrete analytic function on $D \subset \mathbb{R} + ih\mathbb{Z}$, where u and v are real valued semi-discrete functions, equating real and imaginary parts of (4) and (5) respectively yields the semi-discrete Cauchy-Riemann equations. For Type I

$$\frac{\partial u(x,y)}{\partial x} = \frac{1}{h}[v(x, y + h) - v(x, y)],$$
$$\frac{\partial v(x,y)}{\partial x} = \frac{1}{h}[u(x, y) - u(x, y + h)],$$

and for Type II

$$\frac{\partial u(x,y)}{\partial x} = \frac{1}{h}[v(x, y + \frac{h}{2}) - v(x, y - \frac{1}{2})],$$
$$\frac{\partial v(x,y)}{\partial x} = \frac{1}{h}[u(x, y - \frac{h}{2}) - u(x, y + \frac{h}{2})].$$

Helmbold [8] considers functions on a semi-lattice which satisfies the following semi-discrete analogue of Laplace's equation:

$$\frac{d^2u(x,k)}{dx^2} + [u(x, k + 1) - 2u(x, k) + u(x, k - 1)] = 0. \tag{7}$$

He calls this function semi-discrete harmonic.

Let us introduce following operators on D which are defined by Kurowski in [12].

(a) $\Delta_1 f(z) = f(z + ih) - f(z)$,

(b) $\Delta_2 f(z) = f(z + ih/2) - f(z - ih/2)$,

(c) $\Delta_j^{n+1} = \Delta_j[\Delta_j^n f(z)], \quad n \geq 1$,

(d) $\nabla_j f(z) = \frac{\partial^2 f(z)}{\partial x^2} + \Delta_j^2 f(z)$,

(e) $2S_j f(z) = \frac{\partial f(z)}{\partial x} - \frac{i}{h}\Delta_j f(z)$,

(f) $2\bar{S}_j f(z) = \frac{\partial f(z)}{\partial x} + \frac{i}{h}\Delta_j f(z)$,

 ISBN 13: 978-84-613-5510-5

(g) $2\bar{S}_B f(z) = \frac{\partial f(z)}{\partial x} + \frac{i}{h}\Delta_1 f(z - ih)$.

Since

$$4S_j[\bar{S}_j(f)] = 4\bar{S}_j[S_j(f)] = \nabla_j(f),$$

if $f(z)$ is semi-discrete analytic on D, then $\nabla_j(f) = 0$ for all $z \in D^0$ and consequently

$$\nabla_j(u) = \nabla_j(v) = 0.$$

Semi-discrete functions g such that $\nabla_j(g) = 0$ are called semi-discrete harmonic functions of the first or second kind. The semi-discrete functions of the second kind are the semi-discrete harmonic functions considered by Helmbold [8], see equation (7), who called such functions 1/2-harmonic.

4 Functions of Two Real Time Scale Variables

Let T_1 and T_2 be time scales. Let us set $T_1 \times T_2 = \{(x, y) : x \in T_1,\ \ y \in T_2\}$. The set $T_1 \times T_2$ is a complete metric space with the metric (distance) d defined by

$$d((x, y), (x', y')) = \sqrt{(x - x')^2 + (y - y')^2} \qquad \text{for} \qquad (x, y), (x', y') \in T_1 \times T_2.$$

For a given $\delta > 0$, the δ-neighborhood $U_\delta(x_0, y_0)$ of a given point $(x_0, y_0) \in T_1 \times T_2$ is the set of all points $(x, y) \in T_1 \times T_2$ such that $d((x_0, y_0), (x, y)) < \delta$.

Let σ_1 and σ_2 be the forward jump operators for T_1 and T_2, respectively. Further, let ρ_1 and ρ_2 be the backward jump operators for T_1 and T_2, respectively. Let $u : T_1 \times T_2 \to \mathbb{R}$ be a function. The first order delta derivatives of u at a point $(x_0, y_0) \in \mathbb{T}_1^\kappa \times \mathbb{T}_2^\kappa$ are defined to be

$$\frac{\partial u(x_0, y_0)}{\Delta_1 x} = \lim_{x \to x_0, x \neq \sigma_1(x_0)} \frac{u(\sigma_1(x_0), y_0) - u(x, y_0)}{\sigma_1(x_0) - x}$$

and

$$\frac{\partial u(x_0, y_0)}{\Delta_2 y} = \lim_{y \to y_0, y \neq \sigma_2(y_0)} \frac{u(x_0, \sigma_2(y_0)) - u(x_0, y)}{\sigma_2(y_0) - y}.$$

Similarly, we define nabla derivatives of u at a point $(x_0, y_0) \in \mathbb{T}_{1\kappa} \times \mathbb{T}_{2\kappa}$ as

$$\frac{\partial u(x_0, y_0)}{\nabla_1 x} = \lim_{x \to x_0, x \neq \rho_1(x_0)} \frac{u(x, y_0) - u(\rho_1(x_0), y_0)}{x - \rho_1(x_0)}$$

and

$$\frac{\partial u(x_0, y_0)}{\nabla_2 y} = \lim_{y \to y_0, y \neq \rho_2(y_0)} \frac{u(x_0, y) - u(x_0, \rho_2(y_0))}{y - \rho_2(y_0)}.$$

5 Completely Delta Differentiable Functions

Definition 5.1 *A function $u : \mathbb{T} \to \mathbb{R}$ is called completely delta differentiable at a point $x_0 \in \mathbb{T}^\kappa$ if there exists a number A such that*

$$u(x_0) - u(x) = A(x_0 - x) + \alpha(x_0 - x) \quad \textit{for all} \quad x \in U_\delta(x_0) \tag{8}$$

 ISBN 13: 978-84-613-5510-5

and

$$u(\sigma(x_0)) - u(x) = A[\sigma(x_0) - x] + \beta[\sigma(x_0) - x] \quad for\, all \quad x \in U_\delta(x_0) \tag{9}$$

where $\alpha = \alpha(x_0, x)$ *and* $\beta = \beta(x_0, x)$ *are equal zero at* $x = x_0$ *and*

$$\lim_{x \to x_0} \alpha(x_0, x) = 0 \quad and \quad \lim_{x \to x_0} \beta(x_0, x) = 0.$$

Now we can give the definition for two variable case:

Definition 5.2 *We say that a function* $u : T_1 \times T_2 \to \mathbb{R}$ *is* ***completely delta differentiable*** *at a point* $(x_0, y_0) \in T_1^\kappa \times T_2^\kappa$ *if there exist numbers* A_1 *and* A_2 *independent of* $(x, y) \in T_1 \times T_2$ *(but, in general, dependent on* (x_0, y_0)*) such that*

$$u(x_0, y_0) - u(x, y) = A_1(x_0 - x) + A_2(y_0 - y) + \alpha_1(x_0 - x) + \alpha_2(y_0 - y), \tag{10}$$

$$u(\sigma_1(x_0), y_0) - u(x, y) = A_1[\sigma_1(x_0) - x] + A_2(y_0 - y) + \beta_{11}[\sigma_1(x_0) - x] + \beta_{12}(y_0 - y), \tag{11}$$

$$u(x_0, \sigma_2(y_0)) - u(x, y) = A_1(x_0 - x) + A_2[\sigma_2(y_0) - y] + \beta_{21}[x_0 - x] + \beta_{22}[\sigma_2(y_0) - y] \tag{12}$$

for all $(x, y) \in U_\delta(x_0, y_0)$*, where* $\delta > 0$ *is sufficiently small,* $\alpha_j = \alpha_j(x_0, y_0; x, y)$ *and* $\beta_{jk} = \beta_{jk}(x_0, y_0; x, y)$ *are defined on* $U_\delta(x_0, y_0)$ *such that they are equal to zero at* $(x, y) = (x_0, y_0)$ *and*

$$\lim_{(x,y) \to (x_0,y_0)} \alpha_j(x_0, y_0; x, y) = \lim_{(x,y) \to (x_0,y_0)} \beta_{jk}(x_0, y_0; x, y) = 0 \quad for \quad j, k \in \{1, 2\}$$

Note that in case $T_1 = T_2 = \mathbb{Z}$, the neighborhood $U_\delta(x_0, y_0)$ contains the single point (x_0, y_0) for $\delta < 1$. Therefore, in the case, the condition (10) disappears, while the conditions (11) and (12) hold with $\beta_{jk} = 0$ and with

$$A_1 = u(x_0 + 1, y_0) - u(x_0, y_0) = \frac{\partial u(x_0, y_0)}{\Delta_1 x} \tag{13}$$

and

$$A_2 = u(x_0, y_0 + 1) - u(x_0, y_0) = \frac{\partial u(x_0, y_0)}{\Delta_2 y}. \tag{14}$$

Lemma 5.3 *Let the function* $u : T_1 \times T_2 \to \mathbb{R}$ *be completely delta differentiable at the point* $(x_0, y_0) \in T_1^\kappa \times T_2^\kappa$*, then it is continuous at that point and has at* (x_0, y_0) *the first order partial delta derivatives equal to* A_1 *and* A_2*, namely*

$$\frac{\partial u(x_0, y_0)}{\Delta_1 x} = A_1 \quad and \quad \frac{\partial u(x_0, y_0)}{\Delta_2 y} = A_2.$$

Proof. The continuity of u follows, in fact, from any one of (10), (11) and (12). Indeed (10) obviously yields the continuity of u at (x_0, y_0). Let now (11) holds. In the case $\sigma_1(x_0) = x_0$, (11) immediately gives the continuity of u at (x_0, y_0). If $\sigma_1(x_0) > x_0$, except of $u(x, y)$, each term in (11) has a limit as $(x, y) \to (x_0, y_0)$. Therefore $u(x, y)$ also has a limit as $(x, y) \to (x_0, y_0)$, and we have

$$u(\sigma_1(x_0), y_0) - \lim_{(x,y) \to (x_0,y_0)} u(x, y) = A_1[\sigma_1(x_0) - x_0].$$

 ISBN 13: 978-84-613-5510-5

Further, letting $(x, y) = (x_0, y_0)$ in (11), we get

$$u(\sigma_1(x_0), y_0) - u(x_0, y_0) = A_1[\sigma_1(x_0) - x_0].$$

Comparing the last two relations gives

$$\lim_{(x,y)\to(x_0,y_0)} u(x, y) = u(x_0, y_0)$$

which shows the continuity of u at (x_0, y_0). Next, setting $y = y_0$ in (11) and dividing both sides by $\sigma_1(x_0) - x$ and passing to limit as $x \to x_0$ we get $\frac{\partial u(x_0,y_0)}{\Delta_1 x} = A_1$. By similar approach $\frac{\partial u(x_0,y_0)}{\Delta_2 y} = A_2$ can be obtained.

6 Completely Nabla Differentiable Functions

Before the definition of completely nabla differentiability on products of two time scales, we first give the definition for one-variable case.

Definition 6.1 *A function $u : \mathbb{T} \to \mathbb{R}$ is called completely nabla differentiable at a point $x_0 \in \mathbb{T}_\kappa$ if there exists a number B such that*

$$u(x) - u(x_0) = B(x - x_0) + \alpha(x - x_0) \quad for\, all \quad x \in U_\delta(x_0) \tag{15}$$

and

$$u(x) - u(\rho(x_0)) = B[x - \rho(x_0)] + \beta[x - \rho(x_0)] \quad for\, all \quad x \in U_\delta(x_0) \tag{16}$$

where $\alpha = \alpha(x_0, x)$ and $\beta = \beta(x_0, x)$ are equal zero at $x = x_0$ and

$$\lim_{x\to x_0} \alpha(x_0, x) = 0 \quad and \quad \lim_{x\to x_0} \beta(x_0, x) = 0.$$

Now we can give the definition for two variable case:

Definition 6.2 *We say that a function $u : T_1 \times T_2 \to \mathbb{R}$ is **completely nabla differentiable** at a point $(x_0, y_0) \in \mathbb{T}_{1\kappa} \times \mathbb{T}_{2\kappa}$ if there exist numbers B_1 and B_2 independent of $(x, y) \in T_1 \times T_2$ (but, in general, dependent on (x_0, y_0)) such that*

$$u(x, y) - u(x_0, y_0) = B_1(x - x_0) + B_2(y - y_0) + \alpha_1(x - x_0) + \alpha_2(y - y_0), \tag{17}$$

$$u(x, y) - u(\rho_1(x_0), y_0) = B_1[x - \rho_1(x_0)] + B_2(y - y_0) + \beta_{11}[x - \rho_1(x_0)] + \beta_{12}(y - y_0), \tag{18}$$

$$u(x, y) - u(x_0, \rho_2(y_0)) = B_1(x - x_0) + B_2[y - \rho_2(y_0)] + \beta_{21}[x - x_0] + \beta_{22}[y - \rho_2(y_0)] \tag{19}$$

for all $(x, y) \in U_\delta(x_0, y_0)$, where $\delta > 0$ is sufficiently small, $\alpha_j = \alpha_j(x_0, y_0; x, y)$ and $\beta_{jk} = \beta_{jk}(x_0, y_0; x, y)$ are defined on $U_\delta(x_0, y_0)$ such that they are equal to zero at $(x, y) = (x_0, y_0)$ and

$$\lim_{(x,y)\to(x_0,y_0)} \alpha_j(x_0, y_0; x, y) = \lim_{(x,y)\to(x_0,y_0)} \beta_{jk}(x_0, y_0; x, y) = 0 \quad for \quad j, k \in \{1, 2\}$$

 ISBN 13: 978-84-613-5510-5

With $T_1 = T_2 = \mathbb{Z}$, similarly we have

$$B_1 = u(x_0, y_0) - u(x_0 - 1, y_0) = \frac{\partial u(x_0, y_0)}{\nabla_1 x}$$

and

$$B_2 = u(x_0, y_0) - u(x_0, y_0 - 1) = \frac{\partial u(x_0, y_0)}{\nabla_2 y}.$$

This results and (13) and (14) show that each function $u : \mathbb{Z} \times \mathbb{Z} \to \mathbb{R}$ is completely delta and nabla differentiable at every point.

Lemma 6.3 *Let the function $u : T_1 \times T_2 \to \mathbb{R}$ be completely nabla differentiable at the point $(x_0, y_0) \in \mathbb{T}_{1\kappa} \times \mathbb{T}_{2\kappa}$, then it is continuous at that point and has at (x_0, y_0) the first order partial nabla derivatives equal to B_1 and B_2, namely*

$$\frac{\partial u(x_0, y_0)}{\nabla_1 x} = B_1 \quad \text{and} \quad \frac{\partial u(x_0, y_0)}{\nabla_2 y} = B_2.$$

Proof. The proof is similar with the Lemma 5.3.

Theorem 6.4 *If the function $u : T_1 \times T_2 \to \mathbb{R}$ is continuous and have the first order partial nabla derivatives $\frac{\partial u(x,y)}{\nabla_1 x}$, $\frac{\partial u(x,y)}{\nabla_2 y}$ in some δ-neighborhood $U_\delta(x_0, y_0)$ of the point $(x_0, y_0) \in \mathbb{T}_{1\kappa} \times \mathbb{T}_{2\kappa}$ and if these derivatives are continuous at (x_0, y_0), then u is completely ∇-differentiable at (x_0, y_0).*

To prove this theorem we first give the mean value theorem for one-variable case.

Theorem 6.5 *Let a and b be two arbitrary points in $\mathbb{T}$ and let us set $\alpha = min\{a, b\}$ and $\beta = max\{a, b\}$. Let, further, f be a continuous function on $[\alpha, \beta]$ that has a nabla derivative at each point of $(\alpha, \beta]$. Then there exist $\xi, \xi' \in (\alpha, \beta]$ such that*

$$f^{\nabla}(\xi)(a - b) \leq f(a) - f(b) \leq f^{\nabla}(\xi')(a - b).$$

Proof. (for Theorem 6.4) For better clearness of the proof we first consider the single variable case. So, let $u : \mathbb{T} \to \mathbb{R}$ be a function that has a nabla derivative $u^{\nabla}(x)$ in some δ-neighbourhood $U_\delta(x_0)$ of the point $x_0 \in \mathbb{T}_\kappa$ (note that, in contrast to the multivariable case, in the single variable case existence of the derivative at a point implies continuity of the function at that point). The relation (16) with $B = u^{\nabla}(x_0)$ follows immediately from the definition of the nabla derivative

$$u(x) - u(\rho(x_0)) = u^{\nabla}(x_0)[x - \rho(x_0)] + \beta[x - \rho(x_0)], \tag{20}$$

where $\beta = \beta(x_0, x)$ and $\beta \to 0$ as $x \to x_0$. In order to prove (15), we consider all possible cases separately;

(i) If the point x_0 is isolated in $\mathbb{T}$, then (15) is satisfied independent of B and α, since in this case $U_\delta(x_0)$ consists of the single point x_0 for sufficiently small $\delta > 0$.

 ISBN 13: 978-84-613-5510-5

(ii) Let x_0 be left-dense. Regardless whether x_0 is right-scattered or right-dense, we have in this case $\rho(x_0) = x_0$ and (20) coincides with (15).

(iii) Finally, let x_0 be right-dense and left-scattered. Then for sufficiently small $\delta > 0$, any point $x \in U_\delta(x_0) - \{x_0\}$ must satisfy $x > x_0$. Applying Theorem 6.5, we obtain

$$u^\nabla(\xi)(x - x_0) \leq u(x) - u(x_0) \leq u^\nabla(\xi')(x - x_0),$$

where $\xi, \xi' \in (x, x_0]$. Since $\xi \to x_0$ and $\xi' \to x_0$ as $x \to x_0$, by the continuity of the nabla derivative, we get

$$u^\nabla(x_0) = \lim_{x \to x_0} \frac{u(x) - u(x_0)}{x - x_0}.$$

Therefore

$$\frac{u(x) - u(x_0)}{x - x_0} = u^\nabla(x_0) + \alpha,$$

where $\alpha = \alpha(x_0, x)$and $\alpha \to 0$ as $x \to x_0$. Consequently, in the considered case we obtain (15) with $B = u^\nabla(x_0)$ as well.

Now we consider the two-variable case as it is stated in the theorem. To prove (17), we take the difference

$$u(x, y) - u(x_0, y_0) = [u(x, y) - u(x, y_0)] + [u(x, y_0) - u(x_0, y_0)]. \tag{21}$$

By the one-variable case considered above,for $(x, y_0) \in U_\delta(x_0, y_0)$ we have

$$u(x, y_0) - u(x_0, y_0) = \frac{\partial u(x_0, y_0)}{\nabla_1 x}(x - x_0) + \alpha_1(x - x_0) \tag{22}$$

where $\alpha_1 = \alpha_1(x_0, y_0; x)$ and $\alpha_1 \to 0$ as $x \to x_0$. Further, applying the one-variable mean value result, Theorem 6.5, for fixed x and variable y, we have

$$\frac{\partial u(x, \xi)}{\nabla_2 y}(y - y_0) \leq u(x, y) - u(x, y_0) \leq \frac{\partial u(x, \xi')}{\nabla_2 y}(y - y_0), \tag{23}$$

where $\xi, \xi' \in (\alpha, \beta]$ and $\alpha = min\{y_0, y\}$, $\beta = max\{y_0, y\}$. Since $\xi \to y_0$ and $\xi' \to y_0$ as $y \to y_0$, by the continuity of the partial derivatives at (x_0, y_0) we have

$$\lim_{(x,y)\to(x_0,y_0)} \frac{\partial u(x, \xi')}{\nabla_2 y} = \lim_{(x,y)\to(x_0,y_0)} \frac{\partial u(x, \xi)}{\nabla_2 y} = \frac{\partial u(x_0, y_0)}{\nabla_2 y}.$$

Therefore from (23) we obtain

$$u(x, y) - u(x, y_0) = \frac{\partial u(x_0, y_0)}{\nabla_2 y}(y - y_0) + \alpha_2(y - y_0), \tag{24}$$

where $\alpha_2 = \alpha_2(x_0, y_0; x, y)$ and $\alpha_2 \to 0$ as $(x, y) \to (x_0, y_0)$. Substituting (22) and (24) in (21), we get a relation of the form (17) with $B_1 = \frac{\partial u(x_0,y_0)}{\nabla_1 x}$ and $B_2 = \frac{\partial u(x_0,y_0)}{\nabla_2 y}$. To prove (18) we take the difference

$$u(x, y) - u(\rho_1(x_0), y_0) = [u(x, y) - u(x, y_0)] + [u(x, y_0) - u(\rho_1(x_0), y_0)]. \tag{25}$$

 ISBN 13: 978-84-613-5510-5

By the definition of the partial nabla derivative we have

$$u(x,y_0)-u(\rho_1(x_0),y_0)=\frac{\partial u(x_0,y_0)}{\nabla_1 x}[x-\rho_1(x_0)]+\beta_{11}[x-\rho_1(x_0)], \tag{26}$$

where $\beta_{11}=\beta_{11}(x_0,y_0;x)$ and $\beta_{11}\to 0$ as $x\to x_0$. Now substituting (26) and (24) into (25), we obtain a relation of the form (18) with $B_1=\frac{\partial u(x_0,y_0)}{\nabla_1 x}$ and $B_2=\frac{\partial u(x_0,y_0)}{\nabla_2 y}$. The equality (19) can be proved similarly by considering the difference

$$u(x,y)-u(x_0,\rho_2(y_0))=[u(x,y)-u(x_0,y)]+[u(x_0,y)-u(x_0,\rho_2(y_0))].$$

7 Cauchy-Riemann Equations on Time Scale Complex Plane

For given time scales T_1 and T_2, let us set

$$T_1+iT_2=\{z=x+iy : x\in T_1, y\in T_2\}, \tag{27}$$

where $i=\sqrt{-1}$ is the imaginary unit. The set T_1+iT_2 is called the **time scale complex plane** and is a complete metric space with the metric d defined by

$$d(z,z')=|z-z'|=\sqrt{(x-x')^2+(y-y')^2} \tag{28}$$

where $z=x+iy, z'=x'+iy'\in T_1+iT_2$.
Any function $f:T_1+iT_2\to\mathbb{C}$ can be represented in the form

$$f(z)=u(x,y)+iv(x,y) \quad \text{for} \quad z=x+iy\in T_1+iT_2,$$

where $u:T_1\times T_2\to\mathbb{R}$ is the real part of f and $v:T_1\times T_2\to\mathbb{R}$ is the imaginary part of f.
Let σ_1 and σ_2 be the forward jump operators for T_1 and T_2, respectively. For $z=x+iy\in T_1+iT_2$, let us set

$$z^{\sigma_1}=\sigma_1(x)+iy \quad \text{and} \quad z^{\sigma_2}=x+i\sigma_2(y).$$

Let ρ_1 and ρ_2 be the backward jump operators for T_1 and T_2, respectively. For $z=x+iy\in T_1+iT_2$, we set

$$z^{\rho_1}=\rho_1(x)+iy \quad \text{and} \quad z^{\rho_2}=x+i\rho_2(y).$$

8 Delta Analytic Functions

Definition 8.1 *A complex-valued function $f:T_1+iT_2\to\mathbb{C}$ is **delta differentiable** (or **delta analytic**) at a point $z_0=x_0+iy_0\in T_1^\kappa+iT_2^\kappa$ if there exists a complex number A (depending in general on z_0) such that*

$$f(z_0)-f(z)=A(z_0-z)+\alpha(z_0-z) \tag{29}$$

 ISBN 13: 978-84-613-5510-5

$$f(z_0^{\sigma_1}) - f(z) = A(z_0^{\sigma_1} - z) + \beta(z_0^{\sigma_1} - z) \tag{30}$$

$$f(z_0^{\sigma_2}) - f(z) = A(z_0^{\sigma_2} - z) + \gamma(z_0^{\sigma_2} - z) \tag{31}$$

for all $z \in U_\delta(z_0)$, where $U_\delta(z_0)$ is a δ-neighborhood of z_0 in $T_1 + iT_2$, $\alpha = \alpha(z_0, z)$, $\beta = \beta(z_0, z)$ and $\gamma = \gamma(z_0, z)$ are defined for $z \in U_\delta(z_0)$, they are equal to zero at $z = z_0$, and

$$\lim_{z \to z_0} \alpha(z_0, z) = \lim_{z \to z_0} \beta(z_0, z) = \lim_{z \to z_0} \gamma(z_0, z) = 0.$$

*Then the number A is called the **delta derivative** (or Δ-**derivative**) of f at z_0 and is denoted by $f^\Delta(z_0)$.*

Theorem 8.2 *Let the function $f : T_1 + iT_2 \to \mathbb{C}$ have the form*

$$f(z) = u(x, y) + iv(x, y) \quad for \quad z = x + iy \in T_1 + iT_2.$$

Then a necessary and sufficient condition for f to be Δ-differentiable (as a function of the complex variable z) at the point $z_0 = x_0 + iy_0 \in T_1^\kappa + iT_2^\kappa$ is that the functions u and v be completely Δ-differentiable (as a function of the two real variables $x \in T_1$ and $y \in T_2$) at the point (x_0, y_0) and satisfied the Cauchy-Riemann equations

$$\frac{\partial u}{\Delta_1 x} = \frac{\partial v}{\Delta_2 y} \quad and \quad \frac{\partial u}{\Delta_2 y} = -\frac{\partial v}{\Delta_1 x} \tag{32}$$

at (x_0, y_0). If these equations are satisfied, then $f^\Delta(z_0)$ can be represented in any of the forms

$$f^\Delta(z_0) = \frac{\partial u}{\Delta_1 x} + i\frac{\partial v}{\Delta_1 x} = \frac{\partial v}{\Delta_2 y} - i\frac{\partial u}{\Delta_2 y} = \frac{\partial u}{\Delta_1 x} - i\frac{\partial u}{\Delta_2 y} = \frac{\partial v}{\Delta_2 y} + i\frac{\partial v}{\Delta_1 x}, \tag{33}$$

where the partial derivatives are evaluated at (x_0, y_0).

Proof. First we show necessity. Assume that f is Δ-differentiable at $z_0 = x_0 + iy_0$ with $f^\Delta(z_0) = A$. Then (29)-(31) are satisfied. Letting

$$f = u + iv, \quad A = A_1 + iA_2, \quad \alpha = \alpha_1 + i\alpha_2, \quad \beta = \beta_1 + i\beta_2, \quad \gamma = \gamma_1 + i\gamma_2,$$

we get from (29)-(31), equating the real and imaginary parts of both sides in each of these equations,

$$\begin{cases} u(x_0, y_0) - u(x, y) = A_1(x_0 - x) - A_2(y_0 - y) + \alpha_1(x_0 - x) - \alpha_2(y_0 - y) \\ u(\sigma_1(x_0), y_0) - u(x, y) = A_1[\sigma_1(x_0) - x] - A_2(y_0 - y) + \beta_1[\sigma_1(x_0) - x] - \beta_2(y_0 - y) \\ u(x_0, \sigma_2(y_0)) - u(x, y) = A_1(x_0 - x) - A_2[\sigma_2(y_0) - y] + \gamma_1(x_0 - x) - \gamma_2[\sigma_2(y_0) - y] \end{cases}$$

and

$$\begin{cases} v(x_0, y_0) - v(x, y) = A_2(x_0 - x) - A_1(y_0 - y) + \alpha_2(x_0 - x) - \alpha_1(y_0 - y) \\ v(\sigma_1(x_0), y_0) - v(x, y) = A_2[\sigma_1(x_0) - x] - A_1(y_0 - y) + \beta_2[\sigma_1(x_0) - x] - \beta_1(y_0 - y) \\ v(x_0, \sigma_2(y_0)) - v(x, y) = A_2(x_0 - x) - A_1[\sigma_2(y_0) - y] + \gamma_2(x_0 - x) - \gamma_1[\sigma_2(y_0) - y] \end{cases}$$

 ISBN 13: 978-84-613-5510-5

Hence, taking into account that $\alpha_j \to 0$, $\beta_j \to 0$, and $\gamma_j \to 0$ as $(x,y) \to (x_0,y_0)$, we get that the functions u and v are completely Δ-differentiable (as functions of the two real variables $x \in T_1$ and $y \in T_2$) and that

$$A_1 = \frac{\partial u(x_0,y_0)}{\Delta_1 x}, \quad -A_2 = \frac{\partial u(x_0,y_0)}{\Delta_2 y}, \quad A_2 = \frac{\partial v(x_0,y_0)}{\Delta_1 x}, \quad A_1 = \frac{\partial v(x_0,y_0)}{\Delta_2 y}.$$

Therefore the Cauchy-Riemann equations (32) hold and we have the formulas (33).

Now we show sufficiency. Assume that the functions u and v, where $f = u + iv$, are completely Δ-differentiable at the point (x_0,y_0) and that the Cauchy-Riemann equations (32) hold. Then we have

$$\begin{cases} u(x_0,y_0) - u(x,y) = A_1'(x_0-x) - A_2'(y_0-y) + \alpha_1'(x_0-x) - \alpha_2'(y_0-y) \\ u(\sigma_1(x_0),y_0) - u(x,y) = A_1'[\sigma_1(x_0)-x] - A_2'(y_0-y) + \beta_{11}'[\sigma_1(x_0)-x] - \beta_{12}'(y_0-y) \\ u(x_0,\sigma_2(y_0)) - u(x,y) = A_1'(x_0-x) - A_2'[\sigma_2(y_0)-y] + \beta_{21}'(x_0-x) - \beta_{22}'[\sigma_2(y_0)-y] \end{cases}$$

and

$$\begin{cases} v(x_0,y_0) - v(x,y) = A_1''(x_0-x) - A_2''(y_0-y) + \alpha_1''(x_0-x) - \alpha_2''(y_0-y) \\ v(\sigma_1(x_0),y_0) - v(x,y) = A_1''[\sigma_1(x_0)-x] - A_2''(y_0-y) + \beta_{11}''[\sigma_1(x_0)-x] - \beta_{12}''(y_0-y) \\ v(x_0,\sigma_2(y_0)) - v(x,y) = A_1''(x_0-x) - A_2''[\sigma_2(y_0)-y] + \beta_{21}''(x_0-x) - \beta_{22}''[\sigma_2(y_0)-y] \end{cases}$$

where α_j', β_{ij}' and α_j'', β_{ij}'' tend to zero as $(x,y) \to (x_0,y_0)$ and

$$A_1' = \frac{\partial u(x_0,y_0)}{\Delta_1 x} = \frac{\partial v(x_0,y_0)}{\Delta_2 y} = A_2'' =: A_1$$

and

$$-A_2' = -\frac{\partial u(x_0,y_0)}{\Delta_2 y} = \frac{\partial v(x_0,y_0)}{\Delta_1 x} = A_1'' =: A_2.$$

Therefore

$$\begin{aligned} f(z_0) - f(z) &= (A_1 + iA_2)(z_0 - z) + \alpha(z_0 - z), \\ f(z_0^{\sigma_1}) - f(z) &= (A_1 + iA_2)(z_0^{\sigma_1} - z) + \beta(z_0^{\sigma_1} - z), \\ f(z_0^{\sigma_2}) - f(z) &= (A_1 + iA_2)(z_0^{\sigma_2} - z) + \gamma(z_0^{\sigma_2} - z), \end{aligned}$$

where

$$\begin{aligned} \alpha &= (\alpha_1' + i\alpha_1'')\frac{x_0 - x}{z_0 - z} + (\alpha_2' + i\alpha_2'')\frac{y_0 - y}{z_0 - z}, \\ \beta &= (\beta_{11}' + i\beta_{11}'')\frac{\sigma_1(x_0) - x}{z_0^{\sigma_1} - z} + (\beta_{12}' + i\beta_{12}'')\frac{y_0 - y}{z_0^{\sigma_1} - z}, \\ \gamma &= (\beta_{21}' + i\beta_{21}'')\frac{x_0 - x}{z_0^{\sigma_2} - z} + (\beta_{22}' + i\beta_{22}'')\frac{\sigma_2(y_0) - y}{z_0^{\sigma_2} - z}. \end{aligned}$$

Since

$$\begin{aligned} |\alpha| &\le |\alpha_1' + i\alpha_1''| \left|\frac{x_0 - x}{z_0 - z}\right| + |\alpha_2' + i\alpha_2''| \left|\frac{y_0 - y}{z_0 - z}\right| \\ &\le |\alpha_1' + i\alpha_1''| + |\alpha_2' + i\alpha_2''| \le |\alpha_1'| + |\alpha_1''| + |\alpha_2'| + |\alpha_2''|, \end{aligned}$$

we have $\alpha \to 0$ as $z \to z_0$. Similarly, $\beta \to 0$ and $\gamma \to 0$ as $z \to z_0$. Consequently, f is Δ-differentiable at z_0 and $f^{\Delta}(z_0) = A_1 + iA_2$.

 ISBN 13: 978-84-613-5510-5

Remark 8.3 *(See [2]) If the functions $u, v : T_1 \times T_2 \to \mathbb{R}$ are continuous and have the first order partial delta derivatives $\frac{\partial u(x,y)}{\Delta_1 x}$, $\frac{\partial u(x,y)}{\Delta_2 y}$, $\frac{\partial v(x,y)}{\Delta_1 x}$, $\frac{\partial v(x,y)}{\Delta_2 y}$ in some δ-neighborhood $U_\delta(x_0, y_0)$ of the point $(x_0, y_0) \in T_1^\kappa \times T_2^\kappa$ and if these derivatives are continuous at (x_0, y_0), then u and v are completely Δ-differentiable at (x_0, y_0).*
Therefore in this case, if in addition the Cauchy-Riemann equations (32) are satisfied, then $f(z) = u(x, y) + iv(x, y)$ is Δ-differentiable at $z_0 = x_0 + iy_0$.

Example 8.4 *(i) The function $f(z) =$constant on $T_1 + iT_2$ is Δ-analytic everywhere and $f^\Delta(z) = 0$.*
(ii) The function $f(z) = z$ on $T_1 + iT_2$ is Δ-analytic everywhere and $f^\Delta(z) = 1$.
(iii) Consider the function

$$f(z) = z^2 = (x + iy)^2 = x^2 - y^2 + i2xy \quad on \quad T_1 + iT_2.$$

Hence $u(x, y) = x^2 - y^2, v(x, y) = 2xy$, and

$$\frac{\partial u(x,y)}{\Delta_1 x} = x + \sigma_1(x), \quad \frac{\partial u(x,y)}{\Delta_2 y} = -y - \sigma_2(y), \quad \frac{\partial v(x,y)}{\Delta_1 x} = 2y, \quad \frac{\partial v(x,y)}{\Delta_2 y} = 2x.$$

Therefore the Cauchy-Riemann equations become

$$x + \sigma_1(x) = 2x \quad and \quad -y - \sigma_2(y) = -2y,$$

which hold simultaneously if and only if $\sigma_1(x) = x$ and $\sigma_2(y) = y$ simultaneously. It follows that the function $f(z) = z^2$ is not Δ-analytic at each point of $\mathbb{Z} + i\mathbb{Z}$. So, the product of two Δ-analytic functions need not be Δ-analytic.
(iv) The function $f(z) = x^2 - y^2 + i(2xy + x + y)$ is Δ-analytic everywhere on $\mathbb{Z} + i\mathbb{Z}$. Since, Cauchy-Riemann equations are satisfied:

$$\frac{\partial u(x,y)}{\Delta_1 x} = x + \sigma_1(x) = 2x + 1, \quad \frac{\partial u(x,y)}{\Delta_2 y} = -y - \sigma_2(y) = -2y - 1,$$

$$\frac{\partial v(x,y)}{\Delta_1 x} = 2y + 1, \quad \frac{\partial v(x,y)}{\Delta_2 y} = 2x + 1.$$

This function is not analytic anywhere on $\mathbb{R} + i\mathbb{R} = \mathbb{C}$.
(v) The function $f(z) = x^2 - y^2 + i3xy$ is Δ-analytic everywhere on $\mathbb{T} + i\mathbb{T}$ where $\mathbb{T} = \{2^n : n \in \mathbb{Z}\} \cup \{0\}$.
Since, Cauchy-Riemann equations are satisfied:

$$\frac{\partial u(x,y)}{\Delta_1 x} = x + \sigma_1(x) = 3x = \frac{\partial v(x,y)}{\Delta_2 y},$$

$$\frac{\partial u(x,y)}{\Delta_2 y} = -y - \sigma_2(y) = -3y = -\frac{\partial v(x,y)}{\Delta_1 x}.$$

Remark 8.5 *(i) If $T_1 = T_2 = \mathbb{R}$, then $T_1 + iT_2 = \mathbb{R} + i\mathbb{R} = \mathbb{C}$ is the usual complex plane and the three condition (29)-(31) of Definition 8.1 coincide and reduce to the classical definition of analyticity (differentiability) of functions of a complex variable.*

 ISBN 13: 978-84-613-5510-5

(ii) Let $T_1 = T_2 = \mathbb{Z}$. Then $T_1 + iT_2 = \mathbb{Z} + i\mathbb{Z} = \mathbb{Z}[i]$ is the set of Gaussian integers. The neighborhood $U_\delta(z_0)$ of z_0 contains the single point z_0 for $\delta < 1$. Therefore, in this case, the condition (29) disappears, while the conditions (30) and (31) reduce to the single condition

$$\frac{f(z_0+1)-f(z_0)}{1} = \frac{f(z_0+i)-f(z_0)}{i} \tag{34}$$

with $f^{\Delta}(z_0)$ equal to the left (and hence also to the right) hand side of (34). The condition (34) coincides with the condition of forward monodiffric functions in (1).
(iii) If $T_1 = \mathbb{R}$ and $T_2 = h\mathbb{Z} = \{hk : k \in \mathbb{Z}\}$ where $h > 0$, then (29) and (30) coincide and in any of them, dividing both sides by $(z_0 - z)$ where $z = x + ikh$ and $z_0 = x_0 + ik_0h$, and taking limit as $z \to z_0$ (which just means $x \to x_0$), with $k = k_0$ we have

$$\lim_{x \to x_0} \frac{f(x_0 + ik_0h) - f(x + ik_0h)}{x_0 - x} = A.$$

Similarly by (31) we get

$$\frac{f(z_0 + ih) - f(z)}{ih} = A.$$

Equating this two results gives the condition of forward semi-discrete analytic functions:

$$\frac{\partial f(z)}{\partial x} = [f(z + ih) - f(z)]/ih, \quad z = x + ikh.$$

Remark 8.6 *We can combine Cauchy-Riemann equations for delta analytic functions in one complex equation as follows:*

$$\frac{\partial f}{\Delta_1 x} = \frac{1}{i}\frac{\partial f}{\Delta_2 y} \tag{35}$$

(i) *If $T_1 = T_2 = \mathbb{R}$, then we have*

$$\begin{aligned} \frac{\partial f}{\partial x} &= \frac{1}{i}\frac{\partial f}{\partial y} \\ \frac{\partial u}{\partial x} + i\frac{\partial v}{\partial x} &= \frac{1}{i}\left(\frac{\partial u}{\partial y} + i\frac{\partial v}{\partial y}\right) \end{aligned}$$

Equating real and imaginary parts, we get the usual Cauchy-Riemann equations:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} \quad \text{and} \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}.$$

(ii) *If we take $T_1 = T_2 = \mathbb{Z}$ then the equation (35) becomes*

$$\Delta_x f = \frac{1}{i}\Delta_y f$$

which is exactly the condition for forward monodiffric functions

$$\frac{f(z+1)-f(z)}{1} = \frac{f(z+i)-f(z)}{i}.$$

(iii) *With $T_1 = \mathbb{R}$ and $T_2 = h\mathbb{Z}$, the equation (35) becomes*

$$\frac{\partial f}{\partial x} = \frac{1}{i}\frac{f(z+ih)-f(z)}{h}.$$

which is the equation for forward semi-discrete analytic functions.

 ISBN 13: 978-84-613-5510-5

References

[1] Atici, F.M. and Guseinov, G.Sh., 2002. "On Green's functions and positive solutions for boundary value problems on time scales", *Journal of Computational and Applied Mathematics*, Vol. 141 pp. 7599.

[2] Bohner, M. and Guseinov, G.Sh., 2004. "Partial Differentiation on time scales", *Dynam. Systems Appl.* Vol. 13, pp. 351-379.

[3] Bohner, M. and Guseinov, G.Sh., 2005. "An Introduction to Complex Functions on Product of Two Time Scales", *J. Difference Equ. Appl.*

[4] Bohner, M. and Peterson, A., 2001. *Dynamic Equations on Time Scales: An Introduction with Applications*, (Birkhäuser, Boston), Chapter 1.

[5] Duffin, R.J., 1956. "Basic properties of discrete analytic functions", *Duke Math. J.*, Vol. 23, pp. 335-363.

[6] Ferrand, J., 1944. "Fonctions préharmoniques et fonctions preholomorphes", *Bull. Sci. Math.* Vol. 68, second series, pp. 152-180.

[7] Guseinov, G.Sh., 2003. "Integration on time scales", *J. Math. Anal. Appl.* Vol. 285, pp. 107-127.

[8] Helmbold, R.L., "Semi-discrete potential theory", *Carnegie Institute of Technology, Technical Report* No. 34, DA-34-061-ORD-490, Office of Ordnance Research, U.S. Army.

[9] Hilger, S., 1997. "Differential and difference calculus — unified.", *Nonlinear Analysis, Theory, Methods and Applications*, Vol. 30, pp. 2683-2694.

[10] Isaacs, R.P., 1941. "A finite difference function theory", *Univ. Nac. Tucuman Rev.*, Vol. 2, pp. 177-201.

[11] Kiselman, C.O., 2005. "Functions on discrete sets holomorphic in the sense of Isaacs, or monodiffric functions of the first kind", *Science in China, Ser. A Mathematics*, Vol. 48 Supp. 1-11.

[12] Kurowski, G.J., 1963. "Semi-Discrete Analytic Functions", *Transaction of the American Society*, Vol. 106, No. 1, pp. 1-18.

[13] Kurowski, G.J., 1966. "Further results in the theory of monodiffric functions", *Pasific Journal of Mathematics*, Vol. 18, No. 1.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Application of the generalized finite difference method to seismic wave propagation in 2-D.

Francisco Ureña[1], J.J. Benito[2], Eduardo Salete[2] and Luis Gavete[3]

[1] *Departamento de Matemática Aplicada, Universidad de Castilla-La Mancha*

[2] *Departamento de Construcción y Fabricación, Universidad Nacional de Educación a Distancia*

[3] *Departamento de Matemática Aplicada a los Recursos Naturales, Universidad Politécnica de Madrid*

emails: francisco.urena@uclm.es, jbenito@ind.uned.es, esalete@ind.uned.es, lu.gavete@upm.es

Abstract

This paper shows the application of generalized finite difference method (GFDM) to the problem of seismic wave propagation. We investigated stability and star dispersion in 2-D.
We obtained indepedent stability conditions and star dispersion for the P and S waves. Also, we are obtained P and S-wave group velocity..

Key words: meshless methods, generalized finite difference method, moving least squares, seismic waves, instructions
MSC 2000: 65M06, 65M12, 74S20, 80M20

1 Introduction

The Generalized finite difference method (GFDM) is evolved from classical finite difference method (FDM). GFDM can be applied over general or irregular clouds of points. The basic idea is to use moving least squares (MLS) approximation to obtain explicit difference formulae which can be included in partial differential equation to establish, together with an explicit method, a recursive relationship. The authors have made many contributions to the development of this method [1], [2], [3], [4] and [5].
In this paper, this meshless method is applied to seismic wave propagation . Stability conditions and grid dispersion relations in 2-D we derived.

 ISBN 13: 978-84-613-5510-5

2 Explicit Generalized Differences Schemes for the seismic waves propagation problem for a perfectly elastic, homogeneous and isotropic medium

2.1 Equation of motion

The equation of motion and Hooke's law for a perfectly elastic, homogeneous, isotropic medium in 2-D are

$$\begin{cases} \dfrac{\partial^2 U(x,y,t)}{\partial t^2} = \alpha^2 \dfrac{\partial^2 U(x,y,t)}{\partial x^2} + \beta^2 \dfrac{\partial^2 U(x,y,t)}{\partial y^2} + (\alpha^2 - \beta^2) \dfrac{\partial^2 V(x,y,t)}{\partial x \partial y} \\ \dfrac{\partial^2 V(x,y,t)}{\partial t^2} = \beta^2 \dfrac{\partial^2 V(x,y,t)}{\partial x^2} + \alpha^2 \dfrac{\partial^2 V(x,y,t)}{\partial y^2} + (\alpha^2 - \beta^2) \dfrac{\partial^2 U(x,y,t)}{\partial x \partial y} \end{cases} \tag{1}$$

with the initial conditions

$$U(x,y,0) = f_1(x,y); V(x,y,0) = f_2(x,y)$$
$$\frac{\partial U(x,y,0)}{\partial t} = f_3(x,y); \frac{\partial V(x,y,0)}{\partial t} = f_4(x,y) \tag{2}$$

and the boundary condition

$$\begin{cases} a_1 U(x_0,y_0,t) + b_1 \frac{\partial U(x_0,y_0,t)}{\partial n} = g_1(t) \\ a_2 V(x_0,y_0,t) + b_2 \frac{\partial V(x_0,y_0,t)}{\partial n} = g_2(t) \end{cases} \quad en \quad \Gamma \tag{3}$$

where $f_1(x,y)$, $f_2(x,y)$, $f_3(x,y)$, $f_4(x,y)$, $g_1(t)$ y $g_2(t)$ are showed functions,

$$\alpha = \sqrt{\frac{\lambda + 2\mu}{\rho}}, \quad \beta = \sqrt{\frac{\mu}{\rho}}$$

ρ is the density, λ and μ are Lamé elastic coefficients and Γ is the boundary of Ω.

2.2 Explicit Generalized Differences Schemes

The aim is to obtain explicit linear expressions for the approximation of partial derivatives in the points of the domain. First of all, an irregular grid or cloud of points is generated in the domain $\Omega \cup \Gamma$. On defining the central node with a set of nodes surrounding that node, the star then refers to a group of established nodes in relation to a central node. Every node in the domain has an associated star assigned to it.
Following ([5]), the explicit difference formulae are obtained,

$$\begin{cases} \frac{\partial^2 U(x_0,y_0,n\triangle t)}{\partial t^2} = \frac{u_0^{n+1} - 2u_0^n + u_0^{n-1}}{(\triangle t)^2} \\ \frac{\partial^2 V(x_0,y_0,n\triangle t)}{\partial t^2} = \frac{v_0^{n+1} - 2v_0^n + v_0^{n-1}}{(\triangle t)^2} \end{cases} \tag{4}$$

 ISBN 13: 978-84-613-5510-5

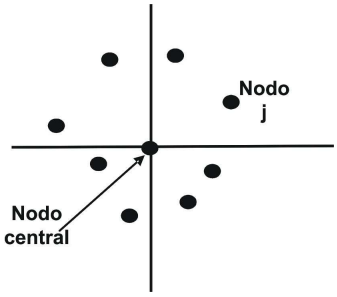


Figure 1: Irregular star (9 nodes)

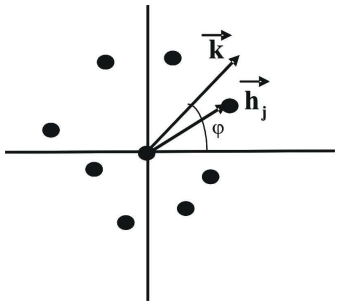


Figure 2: the wavenumber $\vec{k}$

$$\begin{aligned}
&\frac{\partial^2 U(x_0,y_0,n\triangle t)}{\partial x^2} = -m_0 u_0^n + \sum_{j=1}^{N} m_j u_j^n; \quad \frac{\partial^2 V(x_0,y_0,n\triangle t)}{\partial x^2} = -m_0 v_0^n + \sum_{j=1}^{N} m_j v_j^n \\
&\frac{\partial^2 U(x_0,y_0,n\triangle t)}{\partial y^2} = -\eta_0 u_0^n + \sum_{j=1}^{N} \eta_j u_j^n; \quad \frac{\partial^2 V(x_0,y_0,n\triangle t)}{\partial y^2} = -\eta_0 v_0^n + \sum_{j=1}^{N} \eta_j v_j^n \\
&\frac{\partial^2 U(x_0,y_0,n\triangle t)}{\partial x \partial y} = -\zeta_0 u_0^n + \sum_{j=1}^{N} \zeta_j u_j^n; \quad \frac{\partial^2 V(x_0,y_0,n\triangle t)}{\partial x \partial y} = -\zeta_0 v_0^n + \sum_{j=1}^{N} \zeta_j v_j^n
\end{aligned} \tag{5}$$

where $u_j (j = 0, \cdots, N)$ are the second order approximations of the variables in the star nodes. Throughout subindex 0 refers to the central node. The replacement in equation 1 of the explicit expressions obtained for the partial derivatives leads to

$$\begin{cases}
u_0^{n+1} = 2u_0^n - u_0^{n-1} + (\triangle t)^2[\alpha^2(-m_0 u_0^n + \sum_1^N m_j u_j^n) + \beta^2(-\eta_0 u_0^n + \sum_1^N \eta_j u_j^n) \\
\qquad +(\alpha^2 - \beta^2)(-\zeta_0 v_0^n + \sum_1^N \zeta_j v_j^n)] \\
v_0^{n+1} = 2v_0^n - v_0^{n-1} + (\triangle t)^2[\beta^2(-m_0 v_0^n + \sum_1^N m_j v_j^n) + \alpha^2(-\eta_0 v_0^n + \sum_1^N \eta_j v_j^n) \\
\qquad +(\alpha^2 - \beta^2)(-\zeta_0 u_0^n + \sum_1^N \zeta_j u_j^n)]
\end{cases} \tag{6}$$

3 Stability Criterion

For the stability analysis the first idea is to make a harmonic decomposition of the approximated solution at grid points and at a given time level (n). Then we can write the finite difference approximation in the nodes of the star at time n, as

$$u_0^n = A\xi^n e^{i\boldsymbol{k}^T \boldsymbol{x_0}}; \quad u_j^n = A\xi^n e^{i\boldsymbol{k}\boldsymbol{u}^T \boldsymbol{x_j}}; \quad v_0^n = B\xi^n e^{i\boldsymbol{k}^T \boldsymbol{x_0}}; \quad v_j^n = B\xi^n e^{i\boldsymbol{k}^T \boldsymbol{x_j}} \tag{7}$$

 ISBN 13: 978-84-613-5510-5

where: ξ is the amplification factor,

$$\boldsymbol{x_j} = \boldsymbol{x_0} + \boldsymbol{h_j}; \qquad \xi = e^{-iw\triangle t}$$

$\boldsymbol{k}$ (fig. 2) is the column vector of the wave numbers

$$\boldsymbol{k} = \left\{ \begin{array}{c} k_x \\ k_y \end{array} \right\} = k \left\{ \begin{array}{c} \cos\varphi \\ \sin\varphi \end{array} \right\}$$

Then we can write the stability condition as: $\|\xi\| \leq 1$.
Including 7 into 6, cancelation of $\xi^n e^{i\boldsymbol{\nu}^T \boldsymbol{x_0}}$, leads to

$$\begin{aligned} A\xi = 2A - \frac{A}{\xi} + (\triangle t)^2[\alpha^2(-Am_0 + A\sum_1^N m_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + \beta^2(-A\eta_0 + A\sum_1^N \eta_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + \\ (\alpha^2 - \beta^2)(-B\zeta_0 + B\sum_1^N \zeta_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}})] \\ B\xi = 2B - \frac{B}{\xi} + (\triangle t)^2[\beta^2(-Bm_0 + B\sum_1^N m_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + \alpha^2(-B\eta_0 + B\sum_1^N \eta_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + \\ (\alpha^2 - \beta^2)(-A\zeta_0 + A\sum_1^N \zeta_j e^{i\boldsymbol{k}^T\boldsymbol{h_j}})] \end{aligned} \quad (8)$$

where

$$m_0 = \sum_1^N m_j; \quad \eta_0 = \sum_1^N \eta_j; \quad \zeta_0 = \sum_1^N \zeta_j \quad (9)$$

Including 9 into 8, the system of equations is obtained

$$\begin{aligned} A[\xi - 2 + \frac{1}{\xi} + (\triangle t)^2\alpha^2 \sum_1^N m_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + (\triangle t)^2\beta^2 \sum_1^N \eta_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}})] \\ + B(\triangle t)^2(\alpha^2 - \beta^2)\sum_1^N \zeta_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) = 0 \\ A(\triangle t)^2(\alpha^2 - \beta^2)\sum_1^N \zeta_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) + B[\xi - 2 + \frac{1}{\xi} + (\triangle t)^2\beta^2 \sum_1^N m_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}}) \\ + (\triangle t)^2\alpha^2 \sum_1^N \eta_j(1 - e^{i\boldsymbol{k}^T\boldsymbol{h_j}})] = 0 \end{aligned} \quad (10)$$

 ISBN 13: 978-84-613-5510-5

If B is obtained from the second equation and is included into the first equation, then

$$[2\cos w\triangle t-2+(\triangle t)^2(\alpha^2\sum_1^N m_j(1-e^{i\boldsymbol{k}^T\boldsymbol{h_j}})+\beta^2\sum_1^N\eta_j(1-e^{i\boldsymbol{k}^T\boldsymbol{h_j}}))]\times$$
$$[2\cos w\triangle t-2+(\triangle t)^2(\beta^2\sum_1^N m_j(1-e^{i\boldsymbol{k}^T\boldsymbol{h_j}})+\alpha^2\sum_1^N\eta_j(1-e^{i\boldsymbol{k}^T\boldsymbol{h_j}}))]=$$
$$=(\triangle t)^4(\alpha^2-\beta^2)^2[\sum_1^N\zeta_j(1-e^{i\boldsymbol{k}^T\boldsymbol{h_j}})]^2 \quad (11)$$

Operating, the following conditions are obtained:
Real part

$$(1-\cos w\triangle t)^2-2(1-\cos w\triangle t)\frac{(\triangle t)^2}{4}(\alpha^2+\beta^2)\sum_1^N(m_j+\eta_j)(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})+$$
$$\frac{(\triangle t)^4}{4}[(\alpha^2\sum_1^N m_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})+\beta^2\sum_1^N\eta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j}))(\beta^2\sum_1^N m_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})$$
$$+\alpha^2\sum_1^N\eta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j}))-(\alpha^2\sum_1^N m_j\sin\boldsymbol{k}^T\boldsymbol{h_j}+\beta^2\sum_1^N\eta_j\sin\boldsymbol{k}^T\boldsymbol{h_j})(\beta^2\sum_1^N m_j\sin\boldsymbol{k}^T\boldsymbol{h_j}$$
$$+\alpha^2\sum_1^N\eta_j\sin\boldsymbol{k}^T\boldsymbol{h_j})-(\alpha^2-\beta^2)^2[(\sum_1^N\zeta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j}))^2-(\sum_1^N\zeta_j\sin\boldsymbol{k}^T\boldsymbol{h_j})^2]]=0 \quad (12)$$

Imaginary part

$$2(1-\cos w\triangle t)(\alpha^2+\beta^2)\sum_1^N(m_j+\eta_j)\sin\boldsymbol{k}^T\boldsymbol{h_j}-(\triangle t)^2[(\alpha^2\sum_1^N m_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j}$$
$$+\beta^2\sum_1^N\eta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})(\beta^2\sum_1^N m_j\sin\boldsymbol{k}^T\boldsymbol{h_j}+\alpha^2\sum_1^N\eta_j\sin\boldsymbol{k}^T\boldsymbol{h_j})+$$
$$(\alpha^2\sum_1^N m_j\sin\boldsymbol{k}^T\boldsymbol{h_j}+\beta^2\sum_1^N\eta_j\sin\boldsymbol{k}^T\boldsymbol{h_j})(\beta^2\sum_1^N m_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})+$$
$$\alpha^2\sum_1^N\eta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j}))+2(\alpha^2-\beta^2)^2(\sum_1^N\zeta_j(1-\cos\boldsymbol{k}^T\boldsymbol{h_j})\sum_1^N\zeta_j\sin\boldsymbol{k}^T\boldsymbol{h_j}]=0 \quad (13)$$

Operating with the equations 12 and 13, canceling with conservative criteria, the condition for stability of star is obtained.

$$\triangle t<\sqrt{\frac{4}{(\alpha^2+\beta^2)[(|m_0|+|\eta_0|)+\sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}} \quad (14)$$

 ISBN 13: 978-84-613-5510-5

4 Star dispersion

4.1 Star-dispersion relations for the P and S waves

The equation 12 leads to

$$\omega = \frac{1}{\triangle t}\arccos\Phi \tag{15}$$

where

$$\Phi = 1 - \frac{(\triangle t)^2}{4}((\alpha^2+\beta^2)(a_1+a_3) + ((\alpha^2+\beta^2)^2(a_1+a_3)^2 + 4[(\alpha^2-\beta^2)^2(a_5^2-a_6^2) + (\alpha^2a_2+\beta^2a_4)(\beta^2a_2+\alpha^2a_4) - (\alpha^2a_1+\beta^2a_3)(\beta^2a_1+\alpha^2a_3)])^{\frac{1}{2}}) \tag{16}$$

with

$$a_1 = \sum_1^N m_j(1-\cos \boldsymbol{k}^T\boldsymbol{h_j}) \Rightarrow \frac{\partial a_1}{\partial k} = a_{1,k} = \sum_1^N m_j d\sin kd$$

$$a_2 = \sum_1^N m_j \sin \boldsymbol{k}^T\boldsymbol{h_j} \Rightarrow \frac{\partial a_2}{\partial k} = a_{2,k} = \sum_1^N m_j d\cos kd$$

$$a_3 = \sum_1^N \eta_j(1-\cos \boldsymbol{k}^T\boldsymbol{h_j}) \Rightarrow \frac{\partial a_3}{\partial k} = a_{3,k} = \sum_1^N \eta_j d\sin kd$$

$$a_4 = \sum_1^N \eta_j \sin \boldsymbol{k}^T\boldsymbol{h_j} \Rightarrow \frac{\partial a_4}{\partial k} = a_{4,k} = \sum_1^N \eta_j d\cos kd$$

$$a_5 = \sum_1^N \zeta_j(1-\cos \boldsymbol{k}^T\boldsymbol{h_j}) \Rightarrow \frac{\partial a_5}{\partial k} = a_{5,k} = \sum_1^N \zeta_j d\sin kd$$

$$a_6 = \sum_1^N \zeta_j \sin \boldsymbol{k}^T\boldsymbol{h_j} \Rightarrow \frac{\partial a_6}{\partial k} = a_{6,k} = \sum_1^N \zeta_j d\cos kd \tag{17}$$

and

$$\boldsymbol{k}^T\boldsymbol{h_j} = k(h_{jx}\cos\varphi + h_{jy}\sin\varphi) = kd$$

Is known

$$\omega = 2\pi\frac{c^{grid}}{\lambda^{grid}} \tag{18}$$

where c^{grid} and λ^{grid} are the phase velocity (α^{grid} or β^{grid}) and the wavelength (λ_P^{grid} or λ_S^{grid}) in the star respectively.
Defining the relations:

$$s = \frac{2}{\lambda_S^{grid}\sqrt{(r^2+1)[(|m_0|+|\eta_0|) + \sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}} \tag{19}$$

 ISBN 13: 978-84-613-5510-5

$$s_P = \frac{2}{\lambda_P^{grid}\sqrt{(r^2+1)[(|m_0|+|\eta_0|)+\sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}} \tag{20}$$

$$p = \frac{\beta\triangle t\sqrt{(r^2+1)[(|m_0|+|\eta_0|)+\sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}}{2} \tag{21}$$

$$r = \frac{\alpha}{\beta} \tag{22}$$

$$s_P = \frac{s}{r} \tag{23}$$

Including the relations 15, 20, 21 and 23 into equation 18, the star-dispersion relations are obtained:

$$\frac{\alpha^{grid}}{\alpha} = \frac{\arccos\Phi}{2\pi sp} \tag{24}$$

$$\frac{\beta^{grid}}{\beta} = \frac{\arccos\Phi}{2\pi sp} \tag{25}$$

4.2 Star-dispersion for group velocity

By definition the group velocity is the derivative of w, 15, in respect k, thus

$$\alpha_{group}^{grid} = \frac{\partial w}{\partial k} = \frac{\triangle t}{4}\frac{\beta^2\Upsilon}{\sqrt{1-\Phi^2}} \tag{26}$$

where

$$\begin{aligned}\Upsilon = (r^2+1)(a_{1,k}+a_{3,k}) + \frac{1}{2}[2(r^2+1)^2(a_1+a_3)(a_{1,k}+a_{3,k})+\\ 4[2(r^2-1)^2(a_5a_{5,k}-a_6a_{6,k})+(r^2a_{2,k}+a_{4,k})(a_2+r^2a_4)+\\ (r^2a_2+a_4)(a_{2,k}+r^2a_{4,k})-(r^2a_{1,k}+a_{3,k})(a_1+r^2a_3)-\\ (r^2a_1+a_3)(a_{1,k}+r^2a_{3,k})]]\times[(r^2+1)^2(a_1+a_3)^2+\\ 4[(r^2-1)^2(a_5^2-a_6^2)+(r^2a_2+a_4)(a_2+r^2a_4)-(r^2a_1+a_3)(a_1+r^2a_3)]]^{-\frac{1}{2}}\end{aligned} \tag{27}$$

defining

$$\begin{aligned}F = (r^2+1)(a_1+a_3)+[(r^2+1)^2(a_1+a_3)^2+\\ 4[(r^2-1)^2(a_5^2-a_6^2)+(r^2a_2+a_4)(a_2+r^2a_4)-(r^2a_1+a_3)(a_1+r^2a_3)]]\frac{1}{2}]\end{aligned} \tag{28}$$

and including the expressions 21 and 28 into 26, the star-dispersion for waves P and S are

$$\frac{\alpha_{group}^{grid}}{\alpha} = \frac{1}{2\sqrt{2}r}\frac{\Upsilon}{\sqrt{F-(\frac{pF}{\sqrt{(r^2+1)[(|m_0|+|\eta_0|)+\sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}\sqrt{2}})^2}} \tag{29}$$

 ISBN 13: 978-84-613-5510-5

$$\frac{\beta_{group}^{grid}}{\beta} = \frac{1}{2\sqrt{2}} \frac{\Upsilon}{\sqrt{F - (\frac{pF}{\sqrt{(r^2+1)[(|m_0|+|\eta_0|) + \sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}\sqrt{2}})^2}} \tag{30}$$

4.3 Irregularity of the star and dispersion

The coefficients, m_0, η_0, ζ_0, included in the stability criterion and star-dispersion are functions of:

- The number of nodes in the star
- The coordinates of each star node refered to the central node of the star
- The weighting function ([1])

If the number of nodes by star is fixed, in this case 9 ($N = 8$), and the weighting function

$$w(h_{jx}, h_{jy}) = \frac{1}{(\sqrt{h_{jx}^2 + k_{jy}^2})^3} \tag{31}$$

the expression

$$\frac{1}{\sqrt{(r^2+1)[(|m_0|+|\eta_0|) + \sqrt{(m_0+\eta_0)^2+\zeta_0^2}]}} \tag{32}$$

is function of the coordinates of each node of star refered to its central node.
The coefficients, m_0, η_0, ζ_0, are functions of $\frac{1}{h_{jx}^2 + h_{jy}^2}$.
Denoting τ_l the average of the distances between of the nodes of the star l and its central node. Denoting τ the average of the τ_l values in the stars of the mesh, then

$$\boldsymbol{h_j} = \tau \left\{ \begin{array}{c} \overline{h_{jx}} \\ \overline{h_{jy}} \end{array} \right\} \tag{33}$$

$$m_0 = \overline{m_0}\frac{1}{\tau^2}; \quad \eta_0 = \overline{\eta_0}\frac{1}{\tau^2}; \quad \zeta_0 = \overline{\zeta_0}\frac{1}{\tau^2} \tag{34}$$

The stability criterion can be rewritten

$$\triangle t < \frac{2\tau}{\beta\sqrt{(r^2+1)}\sqrt{(|\overline{m_0}|+|\overline{\eta_0}|) + \sqrt{(\overline{m_0}+\overline{\eta_0})^2+\overline{\zeta_0}^2}}} \tag{35}$$

For the regular mesh case of 8 nodes, the expression 35 is

$$\triangle t < \frac{\tau}{\beta\sqrt{r^2+1}} \frac{2(\sqrt{2}-1)\sqrt{3}}{\sqrt{5}} \tag{36}$$

Multiplying the member second of the expression 36 by the factor

$$\frac{\sqrt{5}(\sqrt{2}+1)}{\sqrt{3(|\overline{m_0}|+|\overline{\eta_0}| + \sqrt{(\overline{m_0}+\overline{\eta_0})^2+\overline{\zeta_0}^2})}} \tag{37}$$

 ISBN 13: 978-84-613-5510-5

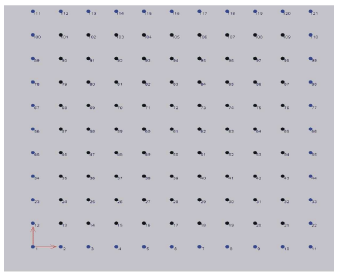


figure 3

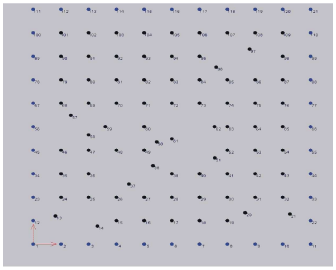


figure 4

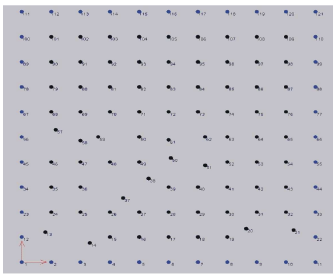


figure 5

the expression 35 is obtained.
The inverse of the value defined by 37, which for a regular star (scheme of 8 nodes plus the central node) is the unity, is an irregularity indicator of the star. The irregularity indicator of the mesh is the minimum value of the irregularity indicator of each star of the mesh. If the number of nodes in the star is increased, the irregularity indicator of the mesh may decrease, thus, the $\triangle t$ value.

5 Numerical Results

The solution of equation 1, with $\Omega = [0,1] \times [0,1] \subset \mathbf{R}^2$, Dirichlet boundary conditions and initial conditions

$$U(x,y,0) = \sin x \sin y; V(x,y,0) = \cos x \cos y; \quad \frac{\partial U(x,y,0)}{\partial t} = 0; \frac{\partial V(x,y,0)}{\partial t} = 0 \quad (38)$$

using the regular meshes (see figure 3 with 121 nodes) and irregular meshes (see figures 4 and 5) with 121 nodes. The analytical solution is

$$U(x,y,t) = \cos(\sqrt{2}\beta t) \sin x \sin y; \quad V(x,y,t) = \cos(\sqrt{2}\beta t) \cos x \cos y \quad (39)$$

The weighting function is function 31 and the criterion for the selection of star nodes is the quadrant criterion [1]. The global error is evaluated for each time increment, in the last time step considered, using the following formula

$$Global \quad error = \frac{\sqrt{\frac{\sum_{j=1}^{NT}(sol(j)-exac(j))^2}{NT}}}{|exac_{max}|} \times 100 \quad (40)$$

where $sol(j)$ is the GFDM solution at the node j $exac(j)$ is the exact value of the solution at the node j, $exac_{max}$ is the maximum value of the exact solution in the cloud of nodes considered and NT is the total number of nodes of the domain.
Tables 1 and 2 show the global errors, with $\triangle t = 0.01$, for several values of α and β, in regular meshes.
Table 3 shows the values of the global error for several values of $\triangle t$, using the irregular mesh with 121 nodes (see figure 4), with $IIM = 0.6524$. Table 4 shows the values of the global error for several values of $\triangle t$, using the irregular mesh with 121 nodes (see figure 5), with $IIM = 0.8944$.

 ISBN 13: 978-84-613-5510-5

Table 1: Influence of the number of nodes in the global error with $\alpha = 1; \beta = 0.6$

N of Nodes	Global Error U	Global Error$V$
121	0.002816	0.003851
289	0.001166	0.001618
441	0.000652	0.000896
676	0.000328	0.000443

Table 2: Influence of the number of nodes in the global error with $\alpha = 1; \beta = 0.5$

N of Nodes	Global Error U	Global Error$V$
121	0.001569	0.001754
289	0.000604	0.000679
441	0.000386	0.000431
676	0.000245	0.000275

6 Conclusions

This paper shows a scheme in generalized finite differences, for seismic wave propagation in 2-D. The von Neumann stability criterion has been expressed as a function of the coefficients of the star equation and velocity ratio.
The investigated star dispersion has been related with the irregularity of the star using the irregularity indicator of the mesh. The use of irregular meshes, adjusted to the geometry of the problem, may create high dispersion in certain stars which is related to high values of the irregularity index of the mesh (IIM). In this case the mesh is redefined by an adaptive process until a mesh whit suitable dispersion and irregularity index values is obtained.

Acknowledgements

The authors acknowledge the support from Ministerio de Ciencia e Innovación of Spain, project CGL2008 − 01757/CLI.

References

[1] *Influence several factors in the generalized finite difference method.*, J. J. Benito; F. Urena; L. Gavete, Applied Mathematical Modeling (2001), vol 25, pp. 1039–1053.

 ISBN 13: 978-84-613-5510-5

Table 3: Influence of the $\triangle t$ in the global error with $\alpha = 1; \beta = 0.5$

$\triangle t$	Global Error U	Global Error V
0.0316	0.015060	0.007900
0.0223	0.010170	0.006447
0.01	0.003566	0.003099
0.007	0.002245	0.001945

Table 4: Influence of the $\triangle t$ in the global error with $\alpha = 1; \beta = 0.5$

$\triangle t$	Global Error U	Global Error V
0.0316	0.004900	0.010400
0.0223	0.003900	0.007300
0.01	0.002020	0.002460
0.007	0.001520	0.001530

[2] *An h-adaptive method in the generalized finite difference.*, J. J. Benito; F. Urena; L. Gavete; R. Alvarez, Comput. Methods Appl. Mech. Eng.(2003), vol 192, pp. 735–759.

[3] *Solving parabolic and hyperbolic equations by Generalized Finite Difference Method.*, J. J. Benito; F. Urena; L. Gavete; B. Alonso, Journal of Computational and Applied Mathematics (2007), vol 209 Issue 2, pp. 208–233.

[4] *Application of the Generalized Finite Difference Method to improve the approximated solution of pdes.*, J. J. Benito; F. Urena; L. Gavete; B. Alonso, Computer Modelling in Engineering & Sciences.(2009), vol 38, pp. 39–58.

[5] *Leading-Edge Applied Mathematical Modelling Research (chapter 7)*, J. J. Benito; F. Urena; L. Gavete, Nova Science Publishers, New York, 2008.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Solving third and fourth order partial differential equations using GFDM. Application to solve problems of plates.

Francisco Ureña[1], Eduardo Salete[2], J.J. Benito[2] and Luis Gavete[3]

[1] *Departamento de Matemática Aplicada, Universidad de Castilla-La Mancha*

[2] *Departamento de Construcción y Fabricación, Universidad Nacional de Educación a Distancia*

[3] *Departamento de Matemática Aplicada a los Recursos Naturales, Universidad Politécnica de Madrid*

emails: francisco.urena@uclm.es, esalete@ind.uned.es, jbenito@ind.uned.es, lu.gavete@upm.es

Abstract

This paper describes the generalized finite difference method to solve second-order partial differential equation systems and fourth-order partial differential equations. This method is applied to solve problem of thin and thick elastic plates.

Key words: meshless methods, generalized finite difference method, moving least squares, plates, instructions
MSC 2000: 65M06, 65M12, 74S20, 80M20

1 Introduction

The Generalized finite difference method (GFDM) is evolved from classical finite difference method (FDM). GFDM can be applied over general or irregular clouds of points. The basic idea is to use moving least squares (MLS) approximation to obtain explicit difference formulae which can be included in the partial differential equations. Benito, Ureña and Gavete have made interesting contributions to the development of this method ([1, 3, 4, 6, 7]).

 ISBN 13: 978-84-613-5510-5

2 The generalized finite difference method

Let us to consider a problem governed by

$$\alpha_1\frac{\partial U}{\partial x}+\alpha_2\frac{\partial U}{\partial y}+\alpha_3\frac{\partial^2 U}{\partial x^2}+\alpha_4\frac{\partial^2 U}{\partial y^2}+\alpha_5\frac{\partial^2 U}{\partial x\partial y}+\alpha_6\frac{\partial^3 U}{\partial x^3}+\alpha_7\frac{\partial^3 U}{\partial x^2\partial y}+$$
$$\alpha_8\frac{\partial^3 U}{\partial x\partial y^2}+\alpha_9\frac{\partial^3 U}{\partial y^3}+\alpha_{10}\frac{\partial^4 U}{\partial x^4}+\alpha_{11}\frac{\partial^4 U}{\partial x^3\partial y}+\alpha_{12}\frac{\partial^4 U}{\partial x^2\partial y^2}+$$
$$\alpha_{13}\frac{\partial^4 U}{\partial x\partial y^3}+\alpha_{14}\frac{\partial^4 U}{\partial y^4}=f(x,y)\quad in\quad \Omega \tag{1}$$

with boundary condition

$$\beta\frac{\partial U}{\partial n}+\gamma U=g(x,y)\quad in\quad \Gamma \tag{2}$$

where $\Omega\subset R^2$ with boundary Γ; $\alpha_i, i=1,\cdots,14,\beta$ and γ are constant coefficients; and f,g are two known smoothed functions.
On defining the composition central node with a set of $N\geq 14$ points surrounding it (henceforth referred as nodes), the star then refers to the group of established nodes in relation to a central node. Each node in the domain have an associated star assigned. If u_0 is an approximation of fourth-order for the value of the function at the central node (U_0) of the star, with coordinates (x_0,y_0) and u_i is an approximation of fourth-order for the value of the function at the rest of nodes, of coordinates (x_i,y_i) with $i=1,\cdots,N$, on including the explicit expressions for the values of the partial derivatives in 1 the star equation is obtained as

$$-m_0u_0+\sum_{i=1}^{N}m_iu_i=f(x_0,y_0) \tag{3}$$

with

$$m_0=\sum_{i=1}^{N}m_i \tag{4}$$

If this process is carried out for each node of the domain a linear equations system is obtained, where the unknowns are the values u_i. On solving this system, the approximated values of the function in the nodes of the domain are obtained and the partial derivatives may easily be calculated from the aforementioned.

3 Application of GFDM to Plates

3.1 Thin Elastic Plates

The partial differential equation, frequently called Lagrange's equation, which relates the rectangular coordinates, the load, the deflections, and the physical and elastic

 ISBN 13: 978-84-613-5510-5

constants of a laterally loaded plate, is well known. Its application to the solution of problems of bending of plates is justified if the following conditions are met: **a**) the plate is composed of material which may be assumed to be homogeneous, isotropic, and elastic, **b**) the plate is of a uniform thichness which is small as compared with its lateral dimensions, **c**) the deflections of the loaded plate are small as compared with its thickness. The additional differential expressions relating the deflections to the boundary conditions, moments, and shears are perhaps equally well known.

$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = -\frac{q(x,y)}{D}; \quad D = \frac{Et^3}{12(1-\mu^2)} \tag{5}$$

where: $w(x,y)$ is deflection in each point of the plate ; $q(x,y)$ is intensity of pressure in each point, normal to the plane of the plate; μ is Poisson's ratio for the material of the plate; E is Young's modulus for the material of the plate; t is the thichness of plate.
On including the explicit expressions for the values of the partial derivatives in 5 and in the boundary conditions for each nodes of the domain a linear equations system is obtained, where the unknowns are the values w_i. On solving this system, the approximated values of the function in the nodes of the domain are obtained.

3.2 Thick Elastic Plates

The partial differential equations are:

$$\begin{cases} \dfrac{t^3}{12}\boldsymbol{H^T C_f H \theta} + t\boldsymbol{C_c}(\boldsymbol{\nabla} w - \boldsymbol{\theta}) = 0 \\ -\boldsymbol{\nabla^T}(t\boldsymbol{C_c})\boldsymbol{\theta} + \boldsymbol{\nabla^T}(t\boldsymbol{C_c})\boldsymbol{\nabla} w = -q \end{cases} \tag{6}$$

where

$$\boldsymbol{H} = \begin{pmatrix} \dfrac{\partial}{\partial x} & 0 \\ 0 & \dfrac{\partial}{\partial y} \\ \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{pmatrix}; \quad \boldsymbol{C_f} = \frac{E}{1-\nu^2}\begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \dfrac{1-\nu}{2} \end{pmatrix}$$

$$\boldsymbol{C_c} = \frac{\alpha E}{2(1+\nu)}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad \boldsymbol{\theta} = \begin{pmatrix} \theta_x \\ \theta_y \end{pmatrix}$$

On including the explicit expressions for the values of the partial derivatives in 6 and in the boundary conditions for each nodes of the domain a linear equations system is obtained, where the unknowns are the values w_i, θ_x, θ_y. On solving this system, the approximated values in the nodes of the domain are obtained.

4 Numerical Results

4.1 Academics Example

This section provides some of the numerical results when solving partial differential equations in a square domain of unit side, with Dirichlet boundary conditions, using

 ISBN 13: 978-84-613-5510-5

the weighting function

$$\Omega(h_j, k_j) = \frac{1}{(\sqrt{h_j^2 + k_j^2})^3} \tag{7}$$

The global exact error can be calculated as

$$Global \quad exact \quad error = \frac{\sqrt{\frac{\sum_{i=1}^{N} e_i^2}{N}}}{exac_{max}} \tag{8}$$

where N is the number of nodes in the domain, $exac_{max}$ is the maximum exact value of function in the domain, e_i is the exact error in the node i.

4.1.1 Example 1

Application of the GFDM to solve the partial differential equation

$$\triangle^2(U) = 0 \tag{9}$$

The cloud of points employed was irregular and is indicated as cloud 1 in fig. 1. The analytical solution is

$$U(x, y) = x^4 + y^4 - 6x^2y^2 \tag{10}$$

The global error is: 0.00001471%

4.1.2 Example 2

Application of the GFDM to solve the partial differential equation

$$-\frac{\partial^3 U}{\partial x^3} + \frac{\partial^3 U}{\partial y^3} + \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \tag{11}$$

The cloud of points employed was irregular of 121 nodes and is indicated as cloud 1 in fig. 1. The analytical solution is

$$U(x, y) = x^3 + y^3 - 3x^2y - 3xy^2 \tag{12}$$

The global error is: 0.0001769%

4.1.3 Example 3

Application of the GFDM to solve the systems

$$\begin{cases} \dfrac{\partial^2 U}{\partial x^2} + \dfrac{\partial^2 V}{\partial x \partial y} = 0 \\ \dfrac{\partial^2 v}{\partial y^2} + \dfrac{\partial^2 u}{\partial x \partial y} = 0 \end{cases} \tag{13}$$

The cloud of points employed was irregular and is indicated as cloud 1 in fig. 1. The analytical solution is

$$U(x, y) = e^x \sin y; \quad V(x, y) = e^x \cos y \tag{14}$$

The global errors are: $errorU = 0.0000425\%$; $errorV = 0.0000464\%$

 ISBN 13: 978-84-613-5510-5

4.2 Plates

4.2.1 Thin Elastic Plates

Tables 1 and 2 show the results of the maximum displacement at the node located at $(0.5; 0.5)$, using regular meshes of 49, 81, 289 and 441 nodes, of a 1×1 thin plate, (t=0.05), with the borders completely fixed (movements and rotations restrained), with uniform load and with punctual load at $(0.5; 0.5)$. The error is evaluated using the following formula

$$error = \frac{|displacement - exact.max.displacement|}{exact.max.displacement} \times 100 \tag{15}$$

Table 1: Fixed plate with uniform load.

Exact. max. displacement= 0.001260[**2**]		
nodes	displacement	% error
49	0.001719	36.42
81	0.001526	21.11
289	0.001332	5.71
441	0.001282	1.75

Table 2: Fixed plate with punctual load.

Exact. max. displacement= 0.005600[**2**]		
nodes	displacement	% error
49	0.005436	2.93
81	0.005488	2.00
289	0.005568	0.57
441	0.005600	0.00

Tables 3 and 4 show the results of the maximum displacement at the node located at $(0.5; 0.5)$, using regular meshes of 49, 81, 289 and 441 nodes, of a 1×1 thin plate, (t=0.05), simply supported (movements restrained at the borders), with uniform load and with punctual load at $(0.5; 0.5)$.

 ISBN 13: 978-84-613-5510-5

Table 3: Simply supported plate with uniform load.

Exact. max. displacement= 0.004062[**2**]		
nodes	displacement	% error
49	0.004420	8.81
81	0.004234	4.23
289	0.004112	1.23
441	0.004094	0.78

Table 4: Simply supported plate with punctual load.

Exact. max. displacement= 0.01160[**2**]		
nodes	displacement	% error
49	0.01095	5.60
81	0.011136	4.00
289	0.011456	1.24
441	0.0115	0.86

4.2.2 Thick Elastic Plates

Table 5 shows the results of the maximum displacement at the node located at $(0.5; 0.5)$ of a thick plate with its borders completely fixed and uniform load, using the GDFM with a regular mesh of 961 nodes. The results are provided for different values of the thickness of the plate.

Table 5: Fixed plate with uniform load.

Maximum displacement	
thickness	displacement
0.05	0.001177
0.1	0.00144
0.2	0.00216
0.3	0.003236
0.4	0.004724

4.2.3 Comparison of results with other methods

The following figure 1 shows the displacement of the node located at $(0.5; 0.5)$ for the fixed 1×1 plate, as the thickness is increased. The results obtained from the GFDM

 ISBN 13: 978-84-613-5510-5

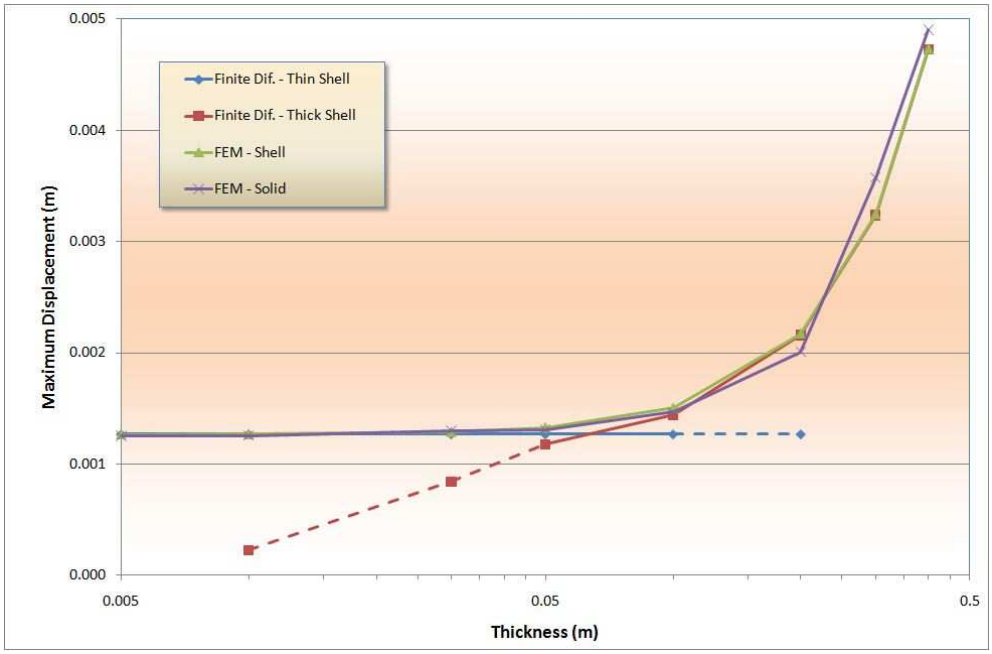


Figure 1: Comparison with other methods

have been compared with the ones obtained using a finite elements commercial software. In order to better understand the differences, two models have been created.
The first finite elements model uses 2500 shell elements with six degrees of freedom per node. The element used is a 4-node element suitable for analyzing thin to moderately-thick shell structures.
The second finite elements model uses 25000 brick 8-node elements with three degrees of freedom per node.
The following figures 2 and 3 show the deformation of a fixed irregular shaped plate with a punctual load on the node located at $(0.5; 0.5)$ using the GFDM and the finite element method.

5 Conclusions

The GFDM has been used to obtain the solution of up to fourth order differential equations.
A series of academic examples have been tested to compare the GFDM results with the analytical results. It has been observed that with a reduced number of nodes the error is low.
The method has been applied to solve thin and thick plates.
A 1.0×1.0 square plate with a punctual load and uniform loads and with fixed or simply supported borders has been analyzed, varying the number of nodes. The obtained

 ISBN 13: 978-84-613-5510-5

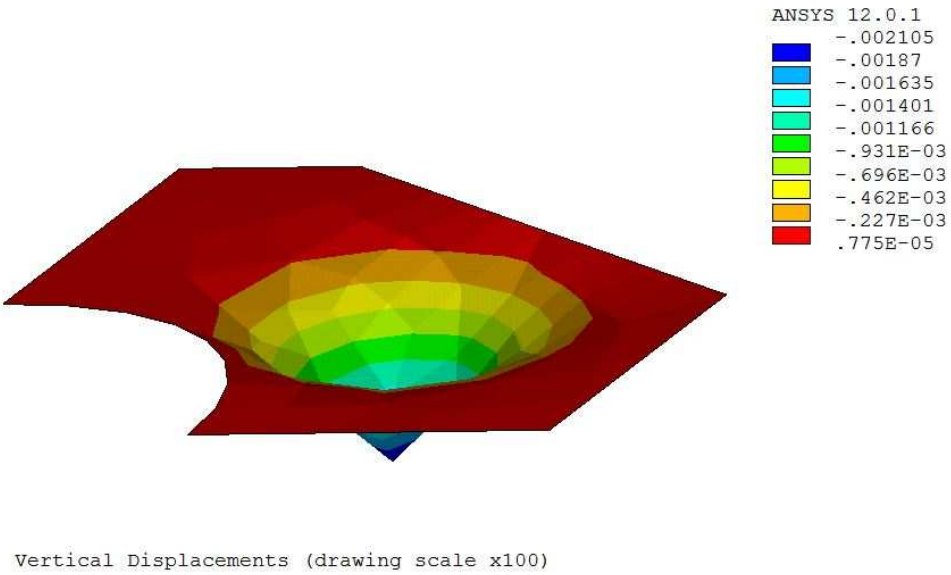


Figure 2: Deformed plate using ANSYS

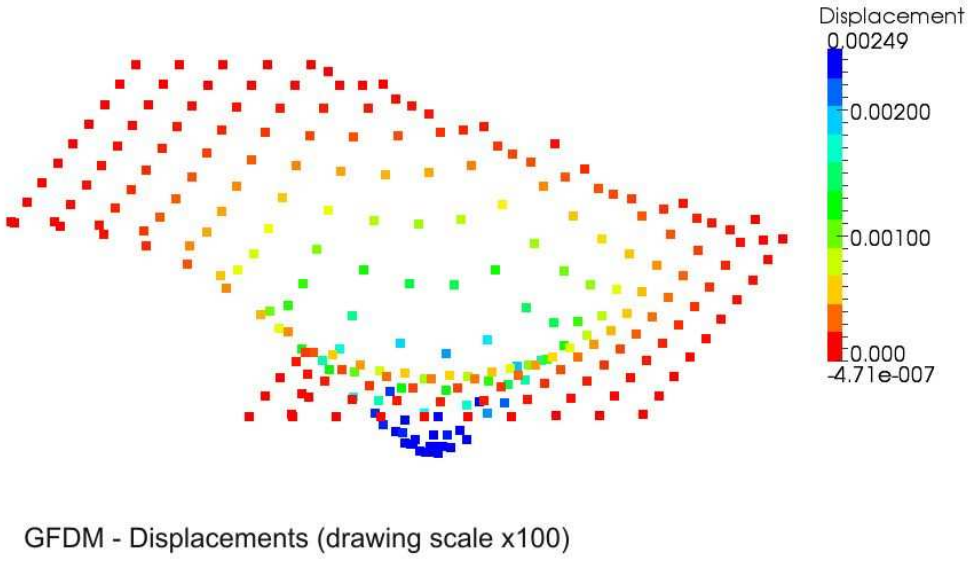


Figure 3: Deformed plate using GFDM

 ISBN 13: 978-84-613-5510-5

solution has been compared with the analytical solution. Even though the numerical solution approaches the theoretical solution as the number of nodes increases, with a low number of nodes an accurate result is provided.
An analysis has been carried out varying the thickness of the plate and comparing the results with a finite element commercial software. For a range between 0.05 and 0.10 of the thickness/length ratio of the plate, the results are similar (7% and 13% difference respectively). For a ratio greater than 0.10 the thick plate formulation should be used, while for a ration under 0.05 the thin plate formulation provides far more accurate results. This confirms the validity of the applied procedure.
Finally, an irregular plate with a punctual load has been tested, comparing the results obtained with the GFDM with the ones obtained with the commercial finite elements software. Both the maximum displacement and the deformed shape match between the two methods.

Acknowledgements

The authors acknowledge the support from Ministerio de Ciencia e Innovación of Spain, project CGL2008 – 01757/CLI.

References

[1] J. J. Benito, F. Urena and L. Gavete, *Leading-Edge Applied Mathematical Modelling Research (chapter 7)*, Nova Science Publishers, New York, 2008.

[2] O. C. Zienkiewicz, R. L. Taylor, *El Método de los Elementos Finitos, Vol 2*, CIMNE, Barcelona, 1994.

[3] J.J. Benito, F. Urena, L. Gavete, *Influence of several factors in the generalized finite difference method*, Applied Mathematical Modelling **25** (2001) 1039–1053.

[4] J.J. Benito, F. Urena, L. Gavete, R. Alvarez, *An h-adaptive method in the generalized finite differences*, Computer methods in Applied Mechanics and Engineering **192** (2003) 735–759.

[5] J.J. Benito, F. Urena, L. Gavete, *Solving parabolic and hyperbolic equations by Generalized Finite Difference Method*, Journal of Computational and Applied Mathematics **209**, Issue 2, (2007) 208–233.

[6] J.J. Benito, F. Urena, L. Gavete, B. Alonso, *A posteriori error estimator and indicator in Generalized Finite Differences. Application to improve the approximated solution of elliptic pdes*, International Journal of Computer Mathematics **85** (2008) 359–370.

 ISBN 13: 978-84-613-5510-5

[7] J.J. Benito, F. Urena, L. Gavete, B. Alonso, *Application of the Generalized Finite Difference Method to improve the approximated solution of pdes*, Computer Modelling in Engineering & Sciences **38** (2009) 39–58.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

A GPU-based implementation of the Classification using Markov Random Fields algorithm in the ITK package*

Pedro Valero[1], José Luis Sánchez[2], Diego Cazorla[2] and Enrique Arias[2]

[1] *Albacete Research Institute of Informatics, University of Castilla-La Mancha*
Avda. España s/n, 02071-Albacete (Spain)

[2] *Computing Systems Dept., University of Castilla-La Mancha*
Avda. España s/n, 02071-Albacete (Spain)

emails: pedro.valerolara@uclm.es, jose.sgarcia@uclm.es,
diego.cazorla@uclm.es, enrique.arias@uclm.es

Abstract

The analysis of medical image, in particular Magnetic Resonance Imaging (MRI), is a very useful tool to help the neurologists on the diagnosis. One of the stages on the analysis of MRI is given by a classification based on the Markov Random Fields (MRF) method. It is possible to find in the literature several packages to carry out this analysis, and of course the classification tasks. One of them is the Insight Segmentation and Registration Toolkit (ITK). However, the analysis of MRI is an expensive computational task. In order to reduce the execution time spent on the analysis of MRI, parallelism techniques can be used. Currently, Graphics Processing Units (GPUs) are becoming a good choice to reduce the execution time of several applications at a low cost. In this paper, the authors present a GPU-based classification using MRF from the sequential implementation that appears in the ITK package. The experimental results show a spectacular execution time reduction being the GPU-based implementation up to 118 times faster than the sequential implementation included in the ITK. The solution provided in this work will be integrated on future versions of ITK package.

Key words: Magnetic Resonance Imaging, Markov Random Fields, Insight Toolkit, Graphics Processing Units

*This work has been supported by the Spanish MICT under grant TSI-020110-2009-362.

 ISBN 13: 978-84-613-5510-5

1 Introduction

The main goal of the Spanish project *Alztools* (TSI-020110-2009-362) is to develop a High Performance Computing tool for the analysis of Magnetic Resonance Imaging (MRI) from the brain in order to prevent Alzheimer diseases.

One of the most useful algorithms used on medical image analysis is the so called Markov Random Fields (MRF). In the literature, MRF algorithm has been applied in different areas as, for instance, speech recognition [1], analysis of satellite images [2], etc.

Focusing on the analysis of MRI according to the objectives of *Alztools* project, there is a set of widely used tools by the neurologists: SPM [3], Freesurfer [4], FSL [5], 3DSlicer [6] and ITK [7], among others. In *Alztools* project, ITK library has been chosen due to the fact that is opensource under GNU licence, it is mantained and updated with more efficient algorithms, it can be integrated with others libraries, as for example 3DSlicer, etc.

Evidently, in ITK the MRF classification algorithm, which is the subject of this paper, is present.

However, as far as the authors know, there is not any implementation of the MRF algorithm of ITK based on a GPU platform. There are some initiatives to introduce GPU computation on ITK as CUDAITK [8] or CITK [9]. But, till this moment CUDAITK initiative has carried out only GPU implementations of basic filters in ITK. On the other hand, CITK iniciative provides some guides or best practices to link efficiently CUDA in ITK.

The paper is structured as follows: Section 2 briefly introduces the classification process in the image analysis. Section 3 describes the sequential implementation of the classification using Markov Random Fields included in the ITK package. Section 4 shows the great computational capacity of the current GPUs. Using the sequential code as the starting point, in Section 5 the GPU-based implementation of the MRF-based classification algorithm is introduced. Section 6 shows the experimental results and the analysis of performance. Finally, Section 7 outlines the conclusions and future work.

2 Image classification

Given a image, multivariate data (feature vectors) are observed at respective pixels. Feature vector $(v_1, v_2, \ldots, v_n)$ describe the image in terms of several, n, attributes. Image classification is a problem of classifying pixels into several homogeneous regions by learning the feature vectors and the adjacency relationships of the pixels in the image. The classification of a pixel into one category is an important and fundamental problem in image pattern analysis [10].

Image classification analyzes the properties of various image features and organizes data into categories [11]. Such categories are defined by decision rules that are known or can be trained. In the second case, classification algorithms typically employ two phases of processing: training and testing. In the training phase, characteristic properties of image features are isolated and a unique description of each classification category

 ISBN 13: 978-84-613-5510-5

(training class), is created. In the testing phase, those categories are used to classify image features.

In image classification, normal distributions are frequently used for analyzing multivariate data in a feature space, and Markov Random Fields (MRFs) are used for modeling the distribution of categories in the image [12]. It is usually assumed that the category labels follow the MRF. The estimation of parameters specifying the MRF is not an easy task because the probability distribution cannot be expressed in a closed form. Hence, the pseudo-likelihood is frequently used for this purpose. The key issue is the estimation of pixel labels of test data. Computer-intensive methods [13, 14] can be used for the estimation, but the implementation is often difficult because of computational complexity.

Markov Random Fields (MRFs) are ubiquitous in low-level computer vision, finding applications in image restoration, stereo matching, texture analysis, segmentation and elsewhere. MRFs are probabilistic models that use the correlation between pixels in a neighborhood to decide the object region or category.

In order to consider the neighborhood information, a simple approach consists in estimating the joint posterior probability for the whole image's labeling configuration. The size of the images can become a problem even not considering high resolution. For example, for a 620 x 540 image, there will be 334800 pixels need to be considered. After determining features, each pixel will have a corresponding feature, which generally will be larger than 1. If color is considered, the dimension is 3; if texture is managed and 8 Gabor filters are used, the dimension of each feature will be 8. The huge amount of information to manage will make the joint of the whole image hard to compute.

An alternative approach is to consider this problem in the Bayesian framework. From the Bayesian statistics' point of view, the contextual information is actually some prior knowledge, while the data will contribute to the computation of the likelihood. This indicates that the prior modeling can be designed to capture the contextual information, while the likelihood can still be computed locally. If an efficient approach can be found to compute the joint prior information, the computation issue can be solved at the same time that the spatial information is considered. This is the main reason for applying MRF models.

Different alternatives have been used to model MRF. In this paper, we focus on that included in the ITK package, contributing at image classification process.

3 MRF-based classification in the ITK package

The Classification using Markov Random Field (CMRF) algorithm implemented in ITK is used, mainly, to make a first-stage filter of a medical imaging.

As it was introduced previously, Markov Random Fields are probabilistic models that use the correlation between pixels in a neighborhood, given by the windows size, to decide the object region. This model is implemented in ITK at *itk::Statistics::MRFImageFilter* which uses the maximum a posteriori (MAP) [20] estimates for modeling the MRF. The object traverses the data set and uses the model generated by the Mahalanobis dis-

 ISBN 13: 978-84-613-5510-5

tance classifier [21] to get the distance between each pixel in the data set to a set of known classes, updates the distances by evaluating the influence of its neighboring pixels (based on a MRF model) and finally, classifies each pixel to the class which has the minimum distance to that pixel (taking the neighborhood influence under consideration). The energy function minimization is done using the iterated conditional modes (ICM) algorithm [14].

For instance, if the algorithm is run using three classes then there are three medias mu_0, mu_1 and mu_2. If a value of the MRI is closed to the value of the media mu_i, then the label i is assigned to this value of the MRI.

The main use of the *itk::Statistics::MRFImageFilter* method is for refining an initial classification by introducing the spatial coherence of the labels. The user should provide two images as input. The first image is the one to be classified while the second image is an image of labels representing an initial classification.

The algorithm 1 sums up the sequential implementation of the CMRF algorithm implemented in ITK.

Algorithm 1 Pseudocode of CMRF implemented in ITK

Function $I\_Class = \texttt{MRFImageFilter}(I\_MRI, I\_Label, Classes, ConVar, Window)$
Inputs:
I_MRI : *MRI image*
I_Label : *Labeled image*
$Classes$: *Classes considered to classify*
$ConVar$: *Number of pixels changed at the previous iteration*
$Window$: *Size of the window*
Output:
I_Class : *Classified image*

1: **while** (**do**$(iter\_actual < MAX\_ITER)$ and $(Error > MAX\_ERROR)$)
2: **while** (**do**No_Final_Image)
3: Compute the influence of each $Class$ on the central pixel of the $Window$ over I_MRI.
4: Compute the influence of each $Class$ of all the pixels of the $Window$ around the central pixel of the I_MRI.
5: Compute the Mahalanobis Distance of the central pixel of the $Window$ of the I_MRI
6: Compute the new label of the image, according to I_Label and the Mahalanobis Distance previously computed.
7: **end while**
8: Compute Error
9: **end while**

 ISBN 13: 978-84-613-5510-5

4 Graphics Processing Units

Current Graphics Processing Units (GPUs) consist of a high number (e.g. 128-240) of fragment processors with high memory bandwidth. They can offer 10x higher main memory bandwidth and use data parallelism to achieve up to 10x more floating point throughput than the CPUs [15].

GPUs are traditionally used for interactive applications, and are designed to achieve high rasterization performance. However, their characteristics have led to the opportunity to other more general applications to be accelerated in GPU-based platforms. This trend is now called General Purpose Computing on GPU (GPGPU) [16], or what is the same, the usage of GPUs for applications for which they were not originally designed. These general applications must have parallel characteristics and an intense computational load to obtain a good performance.

Perhaps the biggest problem lies in the programmability of this kind of devices. In previous years, there were attempts to use graphics-oriented languages [17, 18] in order to accelerate specific parts of code using GPUs. More recently, the GPU manufacturers, like NVIDIA or ATI, have proposed new languages or even extensions for the most common used high level programming languages. As example, NVIDIA proposes CUDA [19], which is a software platform for massively parallel high-performance computing on the company powerful GPUs.

In CUDA, the calculations are distributed in a mesh or grid of thread blocks, all thread blocks are the same size (number of threads). These threads run the GPU code, known as kernel. The dimensions of the mesh and thread blocks should be carefully chosen for maximum performance based on the specific problem being treated. CUDA includes C/C++ software development tools, function libraries, and a hardware abstraction mechanism that hides the GPU hardware from developers such as an Application Programming Interface (API).

Current GPUs are being used for solving image processing problems (robotics, visual inspection, video conferencing, video-on-demand, image databases, data visualization, medical imaging). In the particular case of medical images, it is increasingly the number of applications that are being parallelized for GPUs. As mentioned in Section 2, image processing based on MRF has high computational complexity. Therefore, it is reasonable that GPUs are used for running this kind of applications.

5 A GPU based implementation of the Classification using MRF method in the ITK package

In this paper, a coarse-grain implementation of the sequential CMRF algorithm is considered. Thus, the parallel implementation using a GPU consists in carrying out all the operations over the image at the same time.

According to step 5 of the sequential implementation, a set of operations are carried out in order to compute the Mahalanobis distance, and then to decide if the value of the central pixel has to change or not. These operations are considered for any window

 ISBN 13: 978-84-613-5510-5

that have to be done in a determined order.

The sequential version imposes an order in the operations. Thus, the parallel implementation needs to launch different kernels from steps 2 to 5 on the algorithm 2, launching as many threads as the number of windows in the image. The number of threads is given by the image size and the windows size.

The algorithm 2 is similar to the one presented in algorithm 1, but with a *While* iterative sentence less.

Algorithm 2 Pseudocode of GPU-CMRF algorithm

Function $I\_Class = \texttt{MRFImageFilter}(I\_MRI, I\_Label, Classes, ConVar, Window)$
Inputs:
I_MRI : *MRI image*
I_Label : *Labeled image*
$Classes$: *Classes considered to classify*
$ConVar$: *Number of pixels changed at the previous iteration*
$Window$: *Size of the window*
Output:
I_Class : *Classified image*

1: **while** (**do**$(iter\_actual < MAX\_ITER)$ and $(Error > MAX\_ERROR)$)
2: Compute the influence of each *Class* on the central pixel of the *Window* over I_MRI.
3: Compute the influence of each *Class* of all the pixels of the *Window* around the central pixel of the I_MRI.
4: Compute the Mahalanobis Distance of the central pixel of the *Window* of the I_MRI
5: Compute the new label of the image, according to I_Label and the Mahalanobis Distance previously computed.
6: Compute Error
7: **end while**

6 Experimental results

In this section a performance analysis will be carried out considering the sequential implementation of the CMRF algorithm and the GPU-based CMPRF implementation presented in this work.

The algorithm has been run in two different platforms with the following features:

Platform 1

- CPU: Intel Core 2 at 2.66GHz and 4GB of main memory.
- GPU: GTX 285 with 240 cores and a main memory of 1 GB.

Platform 2

 ISBN 13: 978-84-613-5510-5

– CPU: Intel Core 2 at 2.26GHz and 4GB of main memory.
– GPU: 9600M GT with 32 cores and a main memory of 512 MB.

In this work two platforms have been considered due to the fact that they can be used in different environments. The GPU 9600M GT is yet integrated in current laptops, being this platform widely used. On the other hand, GPU GTX285 is a more powerful and last generation product that has to be integrated, almost, in a desktop.

In this, work different case studies have been considered according to the following parameters:

- Resolution of MRI: 0.488 mm, 0.5 mm, 1 mm and 2 mm.
- Image size: 628×544 pixels if the resolution is 0.488 mm, 364×436 pixels if the resolution is 0.5 mm, 182×218 pixels if the resolution is 1 mm, and 91×109 pixels if the resolution is 2 mm.
- Window size: 3×3, 5×5, and 7×7.
- Number of classes: 3 and 4.

Tables 1 and 2 sum up the results in terms of execution time, by considering the different resolutions (0.488 mm. first row, 0.5 mm. second row, 1 mm. thrid row and 2 mm. fourth row) of MRI, for GPU 9600M GT and GTX 285, respectively.

Sequential time (sec.)						Parallel time (sec.)					
3 Classes			4 Classes			3 Classes			4 Classes		
3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7
4.13	14.89	48.40	4.53	18.49	30.46	0.89	2.71	4.82	0.64	2.78	4.77
1.13	3.39	22.14	8.35	14.49	19.56	0.25	1.13	2.25	0.45	1.32	2.24
0.33	0.69	1.50	0.44	2.00	1.76	0.15	0.41	0.67	0.16	0.43	0.63
0.13	0.33	0.47	0.18	0.22	0.30	0.11	0.16	0.27	0.12	0.17	0.29

Table 1: Execution time of the sequential and GPU parallel implementation on a GPU 9600M GT

In order to know the gain of velocity by employing a GPU platform, Figures 1-4 show the speed-up considering the different resolutions of MRI, for 3 and 4 classes and for GPU 9600M GT and GTX 285.

According to the experimental results, the execution time has been dramatically reduced by using the GPUs. These results have been obtained after several optmizations: usage of shared memory, loop unrolling, and reduction of the number of records needed for the kernel.

From this general conclusion, the following conclusions can be outlined:

- As the resolution of the MRI increases, that is, the size of the image increases in terms of number of pixels, the execution time evidently also increases. Thus, the

 ISBN 13: 978-84-613-5510-5

Sequential time (sec.)						Parallel time (sec.)					
3 Classes			4 Classes			3 Classes			4 Classes		
3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7	3×3	5×5	7×7
3.61	12.84	41.60	3.90	16.27	26.78	0.18	0.25	0.35	0.20	0.31	0.42
0.90	2.90	19.00	7.24	12.51	17.09	0.10	0.15	0.19	0.12	0.18	0.23
0.24	0.6	1.31	0.27	1.71	1.46	0.05	0.06	0.08	0.06	0.08	0.09
0.08	0.28	0.40	0.11	0.16	0.27	0.05	0.06	0.07	0.05	0.06	0.08

Table 2: Execution time of the sequential and GPU parallel implementation on a GPU GTX 285

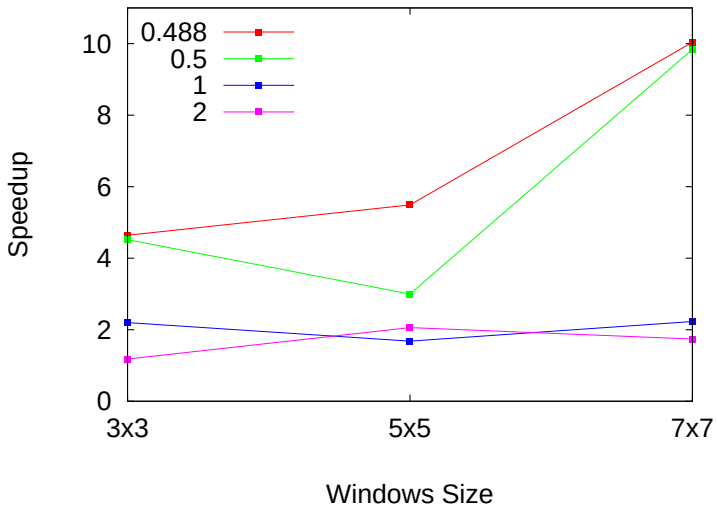


Figure 1: Speedup of the GPU parallel implementation on a GPU 9600M GT considering 3 classes

 ISBN 13: 978-84-613-5510-5

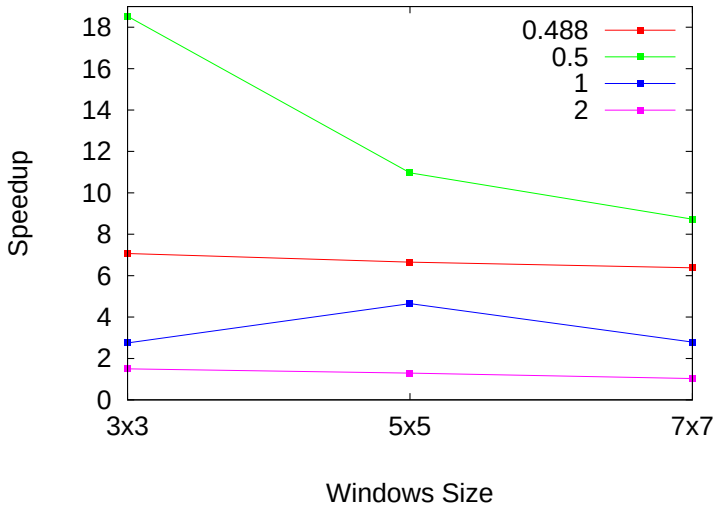


Figure 2: Speedup of the GPU parallel implementation on a GPU 9600M GT considering 4 classes

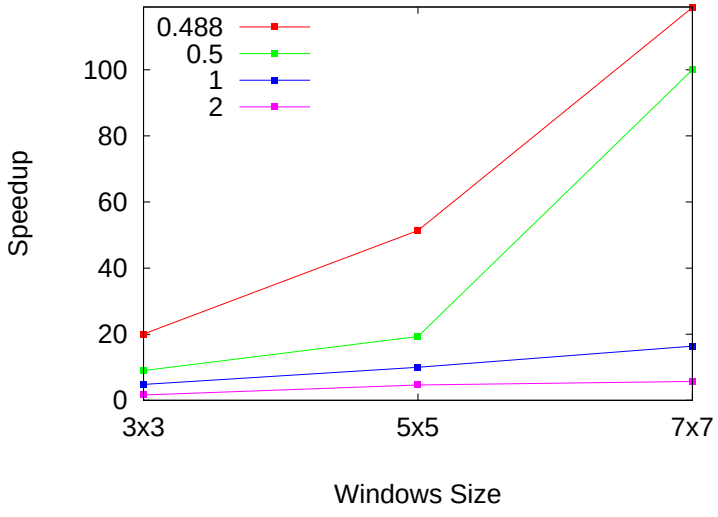


Figure 3: Speedup of the GPU parallel implementation on a GPU GTX285 considering 3 classes

 ISBN 13: 978-84-613-5510-5

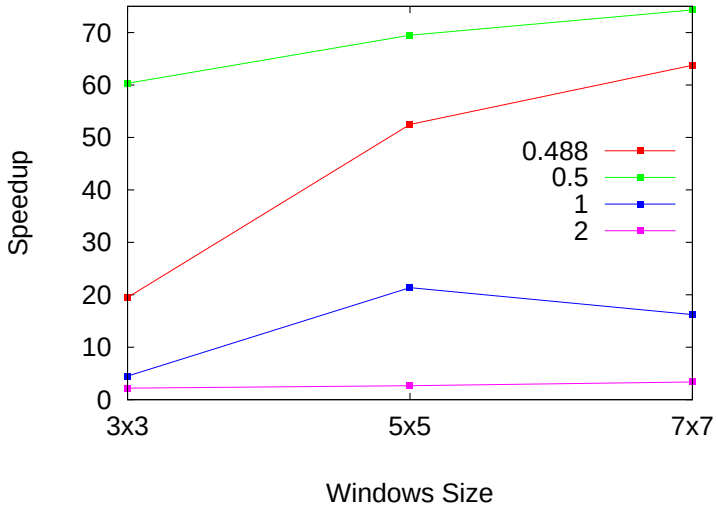


Figure 4: Speedup of the GPU parallel implementation on a GPU GTX285 considering 4 classes

use of GPUs provides better performance in terms of speedup considering a high resolution MRI. From the medical point of view, a high resolution MRI is most interesting due to the fact that it provides better accuracy.

- On the other hand, as the size of window considered on CMRF method increases, the exeution time also increases, but again the accuracy increases. Newly, the use of the GPUs is benefited with respect to the sequential implementation as the problem size increases.

- The behaviour of both platforms considered in this work is quite similar, highlighting that the GPU GTX285 is more powerful than GPU 9600M GT, and as a consequence the execution time is the lowest.

To sum up, from the medical point of view it is interesting to deal with high resolution MRI and a high size of window due to the fact that more accuracy is obtained. Under these conditions, but no constraints, the GPU-based implementation is up to 118 times faster than the sequential implementation.

7 Conclusions and future work

Image analysis is becoming an important discipline applied to different areas in science and engineering. In the framework of *Alztools* project, where this work has been developed, the authors are focused on the analysis of Magnetic Resonance Imaging (MRI). In this context several MRI analysis packages have appeared. Among them, one of the most widely used is the Insight Segmentation and Registration Toolkit (ITK).

 ISBN 13: 978-84-613-5510-5

An important task on image analysis is the classification of images. In ITK the Classification using Markov Random Fields (CMRF) is used.

In this work, a GPU-based implementation of CMRF algorithm in ITK has been developed. As far as the authors know, there are not any GPU implementation of this algorithm.

According to the experimental results, it is remarkable the spectacular execution time reduction by the use of the GPUs reaching a speedup close to 118.

At this stage of *Alztools* project, only 2D images have been used. Even so, the performance obtained encourage to continue not only considering 3D images, but also considering new algorithms useful by the neurologist as Hiden Markov Model, Nonlinear Registration, etc.

References

[1] Guillaume Gravier, Marc Sigelle and G?erard Chollet. *A Markov Random Field Model for Automatic Speech Recognitio.* Proceedings of the International Conference on Pattern Recognition (ICPR'00), ISSN:1051-4651, IEEE Computer Society, Vol. 3, pp. 3258, 2000.

[2] Orfeo Toolbox. *http://www.orfeo-toolbox.org/otb/.*

[3] Statistical Parametric Mapping. *http://www.fil.ion.ucl.ac.uk/spm/.*

[4] FreeSurfer. *http://surfer.nmr.mgh.harvard.edu/fswiki.*

[5] FSL. *http://www.fmrib.ox.ac.uk/fsl/index.html.*

[6] 3DSlicer. *http://www.slicer.org/.*

[7] ITK - Segmentation & Registration Toolkit. *http://www.itk.org/.*

[8] CUDAITK. *http://sourceforge.net/projects/cudaitk/*

[9] CITK. *http//code.google.com/p/cuda-insight-toolkit/.*

[10] K.V. Mardia. *Multi-dimensional multivariate Gaussian Markov random fields with application to image processing*, J. Multivariate Anal. 24 (1988) 265284.

[11] R.O. Duda, P.E. Hart, D.G. Stork. *Pattern classification.* A Wiley-Interscience Publication, second edition, 2000.

[12] S.Z. Li. *Modeling image analysis problems using Markov random fields*, Handbook of Statistics, vol. 20, Wiley, New York, 2000, pp. 143.

[13] S. Geman, D. Geman. *Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images*, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 721-741.

[14] J. Besag. *On the statistical analysis of dirty pictures*, J. Roy. Statist. Soc. Ser. B 48 (1986) 259302.

 ISBN 13: 978-84-613-5510-5

[15] W.-c. Feng, D. Manocha. *High-performance computing using accelerators*, Parallel Computing, Elsevier, 33 (2007), 645-647.

[16] GPGPU. *General-purpose computation using graphics hardware.* `http://www.gpgpu.org`.

[17] R.J. Rost. *OpenGL Shading Language.* Addison-Wesley, 2005.

[18] W.R. Mark, S.R. Glanville, K. Akeley, M.J. Kilgard. *Cg: a system for programming graphics hardware in a C-like language.* In SIGGRAPH 03: ACM SIGGRAPH 2003 Papers, pages 896907, New York, NY, USA, 2003. ACM Press.

[19] NVIDIA. *NVIDIA CUDA Compute Unified Device Architecture-Programming Guide, Version 2.3* 2009, `http://www.nvidia.com/object/cuda_home.html`,

[20] M. DeGroot. *Optimal Statistical Decisions*, McGraw-Hill, 1970.

[21] Geoffry J. McLachlan *Discriminant Analysis and Statistical Pattern Recognition*, Wiley Interscience, 1992.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Symplectic exponentially-fitted modified Runge-Kutta Gauss methods

Guido Vanden Berghe[1] **and Marnix Van Daele**[1]

[1] *Department of Applied Mathematics and Computer Science, Ghent University*
Belgium

emails: Guido.VandenBerghe@UGent.be, Marnix.VanDaele@UGent.be

Abstract

The construction of symmetric and symplectic exponentially-fitted Runge-Kutta methods of Gauss type for the numerical integration of Hamiltonian systems with oscillatory solutions is revisited. In this paper new such two-, three- and four-step integrators are constructed and discussed by making use of the six-step procedure of Ixaru and Vanden Berghe (*Exponential fitting*, Kluwer Academic Publishers, 2004). Numerical experiments for some oscillatory problems are presented and compared to the results obtained by previous methods for three-stage methods.

Key words: exponential fitting, symplecticness, RK-methods oscillatory Hamiltonian
MSC 2000: 65L05, 65L06

1 Introduction

The construction of Runge-Kutta (RK) methods for the numerical solution of ODEs with periodic or oscillating solutions has been considered extensively in the literature [1]-[13]. In an exponential fitting approach the available information on the solutions is used in order to derive more accurate and/or efficient algorithms than the general purpose algorithms for such type of problems. In [14] a particular six-step flow chart is proposed by which specific exponentially-fitted (RK) algorithms can be constructed. In this review paper we shall introduce this procedure in all its aspects for the construction of symplectic RK methods of Gauss type.
In recent years, the construction of numerical integration schemes for ordinary differential equations of the form

$$y'(t) = f(t, y(t)), \quad y(t_0) = y_0 \in \mathbb{R}^m \,, \tag{1}$$

that preserve qualitative properties of the solutions has been studied by many researchers. In this paper we are interested in the class of differential equations that are

 ISBN 13: 978-84-613-5510-5

derived from Hamilton's equations. In case of Hamiltonian systems $m = 2d$ and there exits a scalar Hamiltonian function $H = H(t, y)$, so that $f(t, y) = -J\nabla_y H(t, y)$, where J is the $2d$-dimensional skew symmetric matrix

$$J = \begin{pmatrix} 0_d & I_d \\ -I_d & 0_d \end{pmatrix}, \quad J^{-1} = -J,$$

and where $\nabla_y H(t, y)$ is the column vector of the derivatives of $H(t, y)$ with respect to the components of $y = (y_1, y_2, \ldots, y_{2d})^T$. The Hamiltonian system can then be written as

$$y'(t) = -J\nabla_y H(t, y(t)), \quad y(t_0) = y_0 \in \mathbb{R}^{2d} . \tag{2}$$

Volume-preservation in the phase space (or energy conservation) is an important characteristics for a Hamiltonian system given in (1). Several classical integration methods exist where artificial damping or excitation may lead to misleading results. Several researchers have developed volume-preserving integrators (VPIs) that overcome these difficulties in solving Hamiltonian systems [15]-[19]. For each fixed t_0 the flow map of (2) will be denoted by $\phi_h : \mathbb{R}^m \to \mathbb{R}^m$ so that $\phi_h(y_0) = y(t_0 + h; t_0, y_0)$. In particular, in the case of Hamiltonian systems, ϕ_h is a symplectic map for all h in its domain of definition, i.e. the Jacobian matrix of $\phi_h(y_0)$ satisfies

$$\phi_h'(y_0) J \phi_h'(y_0)^T = J .$$

A desirable property of a numerical method ψ_h for the numerical integration of a Hamiltonian system is to preserve qualitative properties of the original flow ϕ_h such as the symplecticness, in addition to provide an accurate approximation of the exact ϕ_h. It has been widely recognized by several authors [15, 20, 21] that symplectic integrators have some advantages for the preservation of qualitative properties of the flow over the standard integrators when they are applied to Hamiltonian systems. In this sense it may be appropriate to consider symplectic EFRK methods that preserve the structure of the original flow. In [13] the well-known theory of symplectic RK methods is extended to modified EFRK methods of the following form:

$$y_1 = \psi_h(y_0) = y_0 + h \sum_{i=1}^{s} b_i f(t_0 + c_i h, Y_i) , \tag{3}$$

$$Y_i = \gamma_i y_0 + h \sum_{i=1}^{s} a_{ij} f(t_0 + c_j h, Y_j), \quad i = 1, \ldots, s , \tag{4}$$

where the real parameters c_i and b_i are respectively the nodes and the weights of the method. The parameters γ_i make the method modified with repect to the classical RK method, where $\gamma_i = 1, i = 1, \ldots, s$. The s-stage modified RK-method (3)-(4) can also be represented by means of its Butcher's tableau

$$\begin{array}{c|c|ccc} c_1 & \gamma_1 & a_{11} & \ldots & a_{1s} \\ c_2 & \gamma_2 & a_{21} & \ldots & a_{2s} \\ \vdots & \ldots & \vdots & \ddots & \vdots \\ c_s & \gamma_s & a_{s1} & \ldots & a_{ss} \\ \hline & & b_1 & \ldots & b_s \end{array} \tag{5}$$

 ISBN 13: 978-84-613-5510-5

or equivalently by the quartet (c, γ, A, b). Van de Vyver has shown that a modified RK-method (3)-(4) for solving the Hamiltonian system (2) is symplectic if the following conditions are satisfied

$$m_{ij} \equiv b_i b_j - \frac{b_i}{\gamma_i} a_{ij} - \frac{b_j}{\gamma_j} a_{ji} = 0, \quad 1 \leq i, j \leq s \; . \tag{6}$$

Van de Vyver [13] was able to derive a two-stage fourth-order symplectic modified EFRK method of Gauss type. Calvo *et al.* [2]-[4] have studied two-stage as well as three-stage symplectic methods. In their applications they consider pure EFRK methods as well as modified EFRK methods. Their set of functions is the trigonometric polynomial one consisting essentially of the functions $\exp(\pm\lambda t)$ combined with $\exp(\pm 2\lambda t)$ and sometimes $\exp(\pm 3\lambda t)$ or a kind of mixed set type where $\exp(\pm\lambda t)$ is combined with $1, t$ and t^2. In all cases they constructed fourth-order (two-stage case) and sixth-order (three-stage case) methods of Gauss type with fixed or frequency dependent knot points. A theoretical study of high order symmetric and symplectic trigonometrically fitted RK methods with an even number of stages has been performed by Calvo *et al.* [5].
In addition it has been pointed out in [25] that symmetric methods show a better long time behaviour than non-symmetric ones when applied to reversible differential systems.
In [22]-[24] the full exponentially-fitted approach based on a set

$$\{1, t, \ldots, t^K, \exp(\pm\lambda t), t\exp(\pm\lambda t), \ldots, t^P \exp(\pm\lambda t)\} \; , \tag{7}$$

has been applied for the construction of respective fourth-order, sixth-order and eighth-order methods. In this paper we shall give a review of these results and discuss the construction of two-stage (fourth-order), three-stage (sixth-order) and four-stage (eighth-order) symmetric and symplectic (modified) EFRK methods which integrate exactly first-order differential systems whose solutions can be expressed as linear combinations of functions present in the set (7). Our purpose consists in deriving accurate and efficient modified EF geometric integrators based on the combination of the EF approach, followed from the sixth step flow chart [14], and symmetry and symplecticness conditions. The paper is organized as follows. In Section 2 we present the notations and definitions used in the rest of the paper as well as some properties of symplectic and symmetric methods and we introduce the six-step procedure. In Section 3 we present the classes of new two-, three- and four-stage symplectic modified EFRK integrators with frequency dependent nodes. In Section 4 we present some numerical experiments for sixth-order methods with oscillatory Hamiltonian systems and we compare them with the results obtained by other symplectic (EF)RK Gauss integrators given in [4, 15].

2 Notations and definitions, the six-step procedure

It has been remarked by Hairer *et al.* [25] that symmetric numerical methods show a better long time behaviour than nonsymmetric ones when applied to reversible differential equations, as it is the case of conservative mechanical systems. In [3] it is

 ISBN 13: 978-84-613-5510-5

observed that for modified RK methods whose coefficients are even functions of h the symmetry conditions are given by

$$c(h)+Sc(h)=e, \quad b(h)=Sb(h), \quad \gamma(h)=S\gamma(h), \quad SA(h)+A(h)S=\gamma(h)b^T(h)\ , \tag{8}$$

where

$$e=(1,...,1)^T\in\mathbb{R}^s \quad \text{and } S=(s_{ij})\in\mathbb{R}^{s\times s} \quad \text{with } s_{ij}=\begin{cases} 1, & \text{if } i+j=s+1,\\ 0, & \text{if } i+j\neq s+1.\end{cases}$$

Since for symmetric EFRK methods the coefficients are even functions of h, the symmetry conditions can be written in a more convenient form by putting [3]

$$c(h)=\frac{1}{2}e+\theta(h), \quad A(h)=\frac{1}{2}\gamma(h)b^T(h)+\Lambda(h)\ , \tag{9}$$

where

$$\theta(h)=(\theta_1,\ldots,\theta_s)^T\in\mathbb{R}^s \quad \text{and } \Lambda=(\lambda_{ij})\in\mathbb{R}^{s\times s}\ .$$

Therefore, for a symmetric EFRK method whose coefficients a_{ij} are defined by

$$a_{ij}=\frac{1}{2}\gamma_i b_j+\lambda_{ij}, \quad 1\le i,j\le s,$$

the symplecticness conditions (6) reduce to

$$\mu_{ij}\equiv\frac{b_i}{\gamma_i}\lambda_{ij}+\frac{b_j}{\gamma_j}\lambda_{ji}=0, \quad 1\le i,j\le s\ . \tag{10}$$

Taking into account the symmetry relations (8) or (9) and the symplecticness conditions (10) the Butcher tableau for the two-, three- and four-stage methods can be expressed as follows:

- Two-stage method:
  four unknowns remain, $\theta,\gamma_1,\lambda_{12}$ and b_1 and the Butcher tableau reads:

$$\begin{array}{c|c|cc} \frac{1}{2}-\theta & \gamma_1 & \dfrac{\gamma_1 b_1}{2} & \dfrac{\gamma_1 b_1}{2}+\lambda_{12} \\ \frac{1}{2}+\theta & \gamma_1 & \dfrac{\gamma_1 b_1}{2}-\lambda_{12} & \dfrac{\gamma_1 b_1}{2} \\ \hline & & b_1 & b_1 \end{array} \tag{11}$$

- Three-stage method:
  eight unknowns remain, $\theta,\gamma_1,\gamma_2,\lambda_{12}=-\alpha_2,\lambda_{13}=-\alpha_3,\lambda_{21}=-\alpha_4,b_1$ and b_2 and the Butcher tableau reads:

$$\begin{array}{c|c|ccc} \frac{1}{2}-\theta & \gamma_1 & \dfrac{\gamma_1 b_1}{2} & \dfrac{\gamma_1 b_2}{2}-\alpha_2 & \dfrac{\gamma_1 b_1}{2}-\alpha_3 \\ \frac{1}{2} & \gamma_2 & \dfrac{\gamma_2 b_1}{2}-\alpha_4 & \dfrac{\gamma_2 b_2}{2} & \dfrac{\gamma_2 b_1}{2}+\alpha_4 \\ \frac{1}{2}+\theta & \gamma_1 & \dfrac{\gamma_1 b_1}{2}+\alpha_3 & \dfrac{\gamma_1 b_2}{2}+\alpha_2 & \dfrac{\gamma_1 b_1}{2} \\ \hline & & b_1 & b_2 & b_1 \end{array} \tag{12}$$

 ISBN 13: 978-84-613-5510-5

- Four-stage method:
  eight unknowns remain when we choose $\gamma_i = 1, i = 1 \ldots 4$, i.e. $\theta_1, \theta_2, \lambda_{12} = \alpha_1, \lambda_{13} = \alpha_2, \lambda_{14} = \alpha_3, \lambda_{23} = \alpha_4, b_1$ and b_2 and the Butcher array reads:

$$\begin{array}{c|c|cccc} \frac{1}{2}-\theta_1 & 1 & \frac{b_1}{2} & \frac{b_2}{2}+\alpha_1 & \frac{b_2}{2}+\alpha_2 & \frac{b_1}{2}+\alpha_3 \\ \frac{1}{2}-\theta_2 & 1 & \frac{b_1}{2}-\beta\alpha_1 & \frac{b_2}{2} & \frac{b_2}{2}+\alpha_4 & \frac{b_1}{2}+\beta\alpha_2 \\ \frac{1}{2}+\theta_2 & 1 & \frac{b_1}{2}-\beta\alpha_2 & \frac{b_2}{2}-\alpha_4 & \frac{b_2}{2} & \frac{b_1}{2}+\beta\alpha_1 \\ \frac{1}{2}+\theta_1 & 1 & \frac{b_1}{2}-\alpha_3 & \frac{b_2}{2}-\alpha_2 & \frac{b_2}{2}-\alpha_1 & \frac{b_1}{2} \\ \hline & & b_1 & b_2 & b_2 & b_1 \end{array} \tag{13}$$

  with $\beta = \dfrac{b_1}{b_2}$.

The idea of constructing symplectic EFRK taking into account the six-step procedure [14] has been introduced for the first time in [22]–[24]. We briefly shall survey this procedure and suggest some adaptation in order to make the comparison with previous work more easy.
In step (i) we define the appropriate form of an operator related to the discussed problem. Each of the s internal stages (4) and the final stage (3) can be regarded as being a generalized linear multistep method on a nonequidistant grid; we can associate with each of them a linear functional , i.e.

$$\mathcal{L}_i[h, \mathbf{a}]y(t) = y(t + c_i h) - \gamma_i y(t) - h \sum_{j=1}^{s} a_{ij} y'(t + c_j h), \quad i = 1, 2, \ldots s \ . \tag{14}$$

and

$$\mathcal{L}[h, \mathbf{b}]y(t) = y(t + h) - y(t) - h \sum_{i=1}^{s} b_i y'(t + c_i h) \ . \tag{15}$$

We further construct the so-called moments which are for Gauss methods the expressions for $L_{i,j}(h, \mathbf{a}) = \mathcal{L}_i[h, \mathbf{a}]t^j, j = 0, \ldots, s-1$ and $L_i(h, \mathbf{b}) = \mathcal{L}[h, \mathbf{b}]t^j, j = 0, \ldots, 2s-1$ at $t = 0$, respectively.
In step (ii) the linear systems

$$L_{ij}(h, \mathbf{a}) = 0, \quad i = 1, \ldots, s, \quad j = 0, 1, \ldots, s-1 \ ,$$

$$L_i(h, \mathbf{b}) = 0, \quad i = 0, 1, \ldots, 2s-1 \ .$$

are solved to reproduce the classical Gauss RK collocation methods, showing the maximum number of functions which can be annihilated by each of the operators.
The steps (iii) and (iv) can be combined in the present context. First of all we have to define all reference sets of s and $2s$ functions which are appropriate for the internal

 ISBN 13: 978-84-613-5510-5

and final stages respectively. These sets are in general hybrid sets of the following form

$$\{1, t, t^2, \ldots, t^K \text{ or } t^{K'}\} \cup$$
$$\{\exp(\pm\lambda t), t\exp(\pm\lambda t), \ldots, t^P \exp(\pm\lambda t) \text{ or } t^{P'} \exp(\pm\lambda t)\}, \tag{16}$$

where for the internal stages $K + 2P = s - 3$ and for the final stage $K' + 2P' = 2s - 3$. The set in which there is no classical component is identified by $K = -1$ and $K' = -1$, while the set in which there is no exponential fitting component is identified by $P = -1$ or $P' = -1$. It is important to note that such reference sets should contain all successive functions in between. Lacunary sets are in principle not allowed.
Once the sets chosen the operators (14)-(15) are applied to the members of the sets, in this particular case by taking into account the symmetry and the symplecticness conditions described above. The obtained independent expressions are put to zero and in step (v) the available linear systems are solved. The numerical values for $\lambda_{ij}(h)$, $b_i(h), \gamma_i(h)$ and $\theta_i(h)$ are expressed for real values of λ (the pure exponential case) or for pure imaginary $\lambda = i\ \omega$ (oscillatory case). In order to make the comparison with previous work transparable we have opted in the rest of the paper to denote the results for real λ-values.
After the coefficients in the Butcher tableau have been filled in, the principal term of the local truncation error can be written down (step (vi)). Essentially, we know [12] that the algebraic order of the EFRK methods remains the same as the one of the classical Gauss method when this six-step procedure is followed, in other words the algebraic order is $\mathcal{O}(h^{2s})$, while the stage order is $\mathcal{O}(h^s)$. Explicit expressions for this local truncation error will not be discussed here.

3 Two-, three- and four-stage methods

Since $K + 2P = s - 3$ and $K' + 2P' = 2s - 3$ we summarize in table 1 for each s values the occurring (K, P) and (K', P') pairs.

Table 1: The (K, P) and (K', P') values for $s = 2, 3$ and 4.

s	2	3	4
(K, P)	$(1, -1), (-1, 0)$	$(2, -1), (0, 0)$	$(3, -1), (1, 0), (-1, 1)$
(K', P')	$(3, -1), (1, 0),$ $(-1, 1)$	$(5, -1), (3, 0), (1, 1),$ $(-1, 2)$	$(7, -1), (5, 0), (3, 1),$ $(1, 2), (-1, 3)$

The corresponding hybrid sets follow for each (K, P) or (K', P') from the general definition (16). The application of the operators (14) and (15) to sets related to the specific combinations $(K = s - 1, P = -1)$ and $(K' = 2s - 1, P = -1)$, i.e. the polynomial set, gives rise to the well-known order conditions for the corresponding s-stage Gauss method of order $2s$ (for more details see [22]–[24]). Solving these derived

 ISBN 13: 978-84-613-5510-5

equations results in Butcher's tableaus for Gauss methods of order 4, 6 and 8 with $\gamma_i = 1, i = 1, 2, \ldots, s$, i.e.

$$\begin{array}{c|c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & 1 & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & 1 & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & & \frac{1}{2} & \frac{1}{2} \end{array} \qquad \begin{array}{c|c|ccc} \frac{1}{2}-\frac{\sqrt{15}}{10} & 1 & \frac{5}{36} & \frac{2}{9}-\frac{\sqrt{15}}{15} & \frac{5}{36}-\frac{\sqrt{15}}{30} \\ \frac{1}{2} & 1 & \frac{5}{36}+\frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36}-\frac{\sqrt{15}}{24} \\ \frac{1}{2}+\frac{\sqrt{15}}{10} & 1 & \frac{5}{36}+\frac{\sqrt{15}}{30} & \frac{2}{9}+\frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

and the tableau (13) with

$$\theta_1 = \frac{1}{2}\sqrt{\frac{15+2\sqrt{30}}{35}} = \frac{c}{70}, \qquad \theta_2 = \frac{1}{2}\sqrt{\frac{15-2\sqrt{30}}{35}} = \frac{a}{70},$$
$$b_1 = \frac{1}{4} - \frac{\sqrt{30}}{72}, \qquad b_2 = \frac{1}{4} + \frac{\sqrt{30}}{72},$$

with $a = \sqrt{525 - 70\sqrt{30}}$ and $c = \sqrt{525 + 70\sqrt{30}}$ and

$$\alpha_1 = \frac{a}{1470} - \frac{5b}{2352} - \frac{c}{420} - \frac{d}{336}, \quad \alpha_2 = -\frac{a}{1470} + \frac{5b}{2352} - \frac{c}{420} - \frac{d}{336},$$
$$\alpha_3 = -\frac{c}{105} + \frac{d}{168}, \quad \alpha_4 = -\frac{a}{105} - \frac{b}{168},$$

with $b = \sqrt{630 - 84\sqrt{30}}$ and $d = \sqrt{630 + 84\sqrt{30}}$.
Equivalent results have been reported by Hairer *et al.* in [25].

3.1 Two-stage methods

Let us remark that considering the $(K = 1, P = -1)$ set for the internal stages gives rise to $\gamma_1 = 1$, a value which is not compatible with the additional symmetry, symplecticity and order conditions imposed. Therefore in what follows we combine the $(K = -1, P = 0)$ case with either $(K' = 1, P' = 0)$ or $(K' = -1, P' = 1)$.

Case $(K' = 1, P' = 0)$:

The operators (14) and (15) are applied to the functions present in the occurring hybrid sets, taking into account the structure of the Butcher tableau (11). Following equations arise with $z = \lambda h$:

$$2b_1 = 1, \tag{17}$$
$$2b_1 \cosh(z/2)\cosh(\theta z) = \frac{\sinh(z)}{z}, \tag{18}$$
$$\lambda_{12}\cosh(\theta z) = -\frac{\sinh(\theta z)}{z}, \tag{19}$$
$$\lambda_{12}\sinh(\theta z) - \frac{\cosh(\theta z)}{z} = -\frac{\gamma_1}{z}\cosh(z/2), \tag{20}$$

 ISBN 13: 978-84-613-5510-5

resulting in the results

$$b_1 = 1/2, \qquad \theta = \frac{\operatorname{arccosh}\left(\dfrac{2\sinh(z/2)}{z}\right)}{z}, \qquad \lambda_{12} = -\frac{\sinh(\theta z)}{z\cosh(\theta z)},$$
$$\gamma_1 = \frac{\left(\dfrac{\sinh(\theta z)^2}{z\cosh(\theta z)} + \dfrac{\cosh(\theta z)}{z}\right) z}{\cosh(z/2)} .$$

The series expansions for these coefficients for small values of z are given by

$$\begin{aligned}
\theta &= \sqrt{3}\left(\frac{1}{6} + \frac{1}{2160}z^2 - \frac{1}{403200}z^4 + \frac{1}{145152000}z^6 + \frac{533}{9656672256000}z^8 + \dots\right) , \\
\lambda_{12} &= \sqrt{3}\left(-\frac{1}{6} + \frac{1}{240}z^2 - \frac{137}{1209600}z^4 + \frac{143}{48384000}z^6 - \frac{81029}{1072963584000}z^8 + \dots\right) , \\
\gamma_1 &= 1 - \frac{1}{360}z^4 + \frac{11}{30240}z^6 - \frac{71}{1814400}z^8 + \frac{241}{59875200}z^{10} + \dots ,
\end{aligned}$$

showing that for $z \to 0$ the classical values are retrieved.

Case $(K' = -1, P' = 1)$:

In this approach equations (18)-(20) remain unchanged and they deliver expressions for b_1, γ_1 and λ_{12} in terms of θ. Only (17) is replaced by

$$b_1(\cosh(\theta z)\,(2\cosh(z/2) + z\sinh(z/2)) + 2\theta z\cosh(z/2)\sinh(\theta z)) = \cosh(z). \qquad (21)$$

By combining (18) and (21) one obtains an equation in θ and z, i.e.:

$$\theta\sinh(z)\sinh(\theta z) = \cosh(\theta z)\left(\cosh(z) - \frac{\sinh(z)}{z} - \sinh^2(z/2)\right).$$

It is not anymore possible to write down an analytical solution for θ, but iteratively a series expansion can be derived. We give here those series expansions as obtained for the four unknowns

$$\begin{aligned}
\theta &= \sqrt{3}\left(\frac{1}{6} + \frac{1}{1080}z^2 + \frac{13}{2721600}z^4 - \frac{1}{7776000}z^6 - \frac{1481}{1810626048000}z^8 + \dots\right) , \\
b_1 &= \frac{1}{2} - \frac{1}{8640}z^4 + \frac{1}{1088640}z^6 + \frac{1}{44789760}z^8 + \dots , \\
\lambda_{12} &= \sqrt{3}\left(-\frac{1}{6} + \frac{1}{270}z^2 - \frac{223}{2721600}z^4 + \frac{17}{9072000}z^6 - \frac{259513}{5431878144000}z^8 + \dots\right) , \\
\gamma_1 &= 1 - \frac{1}{480}z^4 + \frac{17}{60480}z^6 - \frac{2629}{87091200}z^8 + \frac{133603}{43110144000}z^{10} + \dots .
\end{aligned}$$

 ISBN 13: 978-84-613-5510-5

3.2 Three-stage methods

Following the ideas developed in this paper it should be obvious that we combine the $(K = 0, P = 0)$ case with the three non-polynomial cases for the final stage. However keeping the 1 in the hybrid set for $(K = 0, P = 0)$ delivers $\gamma_1 = \gamma_2 = 1$, a result which is not compatible with the only remaining symplecticity condition.

$$\frac{b_1}{\gamma_1}\alpha_2 + \frac{b_2}{\gamma_2}\alpha_4 = 0. \tag{22}$$

Therefore we choose for the internal stages the hybrid set $\{\exp(\pm\lambda t)\}$, omitting the constant 1; in other words we accept exceptionally a lacunary set, what is principally not allowed by the six-step procedure [14]. Under these conditions, and taking into account the symmetry conditions the $\alpha_i, (i = 2, 3, 4)$ parameters are the solutions in terms of θ, γ_1 and γ_2 of the following three equations [4]:

$$\begin{aligned} &1 - \gamma_2 \cosh(z/2) - 2z\alpha_4 \sinh(\theta z) = 0, \\ &\cosh(\theta z) - \gamma_1 \cosh(z/2) + z\alpha_3 \sinh(\theta z) = 0, \\ &\sinh(\theta z) - z\alpha_3 \cosh(\theta z) - z\alpha_2 = 0. \end{aligned} \tag{23}$$

with the following solution:

$$\begin{aligned} \alpha_2 &= \frac{\cosh(2\theta z) - \gamma_1 \cosh(z/2)\cosh(\theta z)}{z \sinh(\theta z)}, \\ \alpha_3 &= \frac{\gamma_1 \cosh(z/2) - \cosh(\theta z)}{z \sinh(\theta z)}, \quad \alpha_4 = \frac{1 - \gamma_2 \cosh(z/2)}{2z \sinh(\theta z)}. \end{aligned} \tag{24}$$

The solution for the other parameters depends essentially on the chosen values of K' and P'.

Case $(K' = 3, P' = 0)$:

The operators (14) and (15) are applied to the functions present in the ocurrring hybrid set, taking into account the symmetry conditions; we derive three independent equations in b_1, b_2 and θ, i.e.

$$2b_1 + b_2 = 1, \tag{25}$$

$$b_1\theta^2 = \frac{1}{24}, \tag{26}$$

$$b_2 + 2b_1 \cosh(\theta z) = \frac{2\sinh(z/2)}{z}. \tag{27}$$

Taking into account (25) and (27) b_1 and b_2 can be expressed in terms of θ:

$$b_1 = \frac{z - 2\sinh(z/2)}{2z(1 - \cosh(\theta z))}, \quad b_2 = \frac{2\sinh(z/2) - z\cosh(\theta z)}{z(1 - \cosh(\theta z))}.$$

These expressions combined with (26) results in the following equation for θ:

$$\theta^2 - \frac{z(1 - \cosh(\theta z))}{12(z - 2\sinh(z/2))} = 0\,.$$

 ISBN 13: 978-84-613-5510-5

If now the symplecticness condition (22) is imposed, the parameter γ_1 is determined by

$$\gamma_1 = \frac{\gamma_2(2\sinh(z/2) - z)\cosh(2\theta z)}{2\sinh(z/2) - \gamma_2\sinh(z) + (\gamma_2\sinh(z) - z)\cosh(\theta z)}.$$

The obtained parameters define a familiy of EFRK methods which are symmetric and symplectic for all $\gamma_2 \in \mathbb{R}$. Following [4] we choose from now on $\gamma_2 = 1$.
Now it is easy to give the series expansions for all the coefficients for small values of z:

$$\begin{aligned}
\theta &= \sqrt{15}\left(\frac{1}{10} + \frac{1}{21000}z^2 - \frac{131}{1058400000}z^4 + \frac{13487}{48898080000000}z^6\right. \\
&\quad \left. - \frac{1175117}{3203802201600000000}z^8 - \frac{505147}{9153720576000000000}z^{10} + \ldots\right), \\
\gamma_1 &= 1 - \frac{3}{70000}z^6 + \frac{13651}{1176000000}z^8 - \frac{2452531}{862400000000}z^{10} + \ldots, \\
b_1 &= \frac{5}{18} - \frac{1}{3780}z^2 + \frac{167}{190512000}z^4 - \frac{23189}{8801654400000}z^6 + \frac{7508803}{1153368792576000000}z^8 \\
&\quad - \frac{87474851}{8073581548032000000000}z^{10} + \ldots, \\
b_2 &= \frac{4}{9} + \frac{1}{1890}z^2 - \frac{167}{95256000}z^4 + \frac{23189}{4400827200000}z^6 - \frac{7508803}{576684396288000000}z^8 \\
&\quad + \frac{87474851}{4036790774016000000000}z^{10} + \ldots, \\
\alpha_2 &= \sqrt{15}\left(\frac{1}{15} - \frac{1}{6000}z^2 + \frac{11623}{3175200000}z^4 - \frac{213648613}{73347120000000}z^6\right. \\
&\quad \left. + \frac{1669816359863}{2135868134400000000}z^8 - \frac{409429160306437}{2135868134400000000000}z^{10} + \ldots\right),
\end{aligned}$$

$$\begin{aligned}
\alpha_3 &= \sqrt{15}\left(\frac{1}{30} + \frac{3}{14000}z^2 - \frac{24739}{793800000}z^4 + \frac{14753813}{2993760000000}z^6 - \right. \\
&\quad \left. \frac{7187933379103}{6407604403200000000}z^8 + \frac{48242846122937}{1779890112000000000000}z^{10} + \ldots\right), \\
\alpha_4 &= \sqrt{15}\left(-\frac{1}{24} + \frac{13}{67200}z^2 - \frac{37}{12700800}z^4 + \frac{19922401}{469421568000000}z^6\right. \\
&\quad \left. - \frac{733072729}{12204960768000000000}z^8 + \frac{1539941201}{1830744115200000000000}z^{10} + \ldots\right).
\end{aligned}$$

Case $(K' = 1, P' = 1)$:

The equations (25) and (27) remain unchanged. Equation (26) is replaced by the equation obtained by applying the operator (15) with $s = 3$ on $t\exp(\pm\lambda t)$ resulting in:

$$2b_1 z^2\theta\sinh(\theta z) = z\cosh(z/2) - 2\sinh(z/2)\ . \tag{28}$$

 ISBN 13: 978-84-613-5510-5

Taking into account (27) and (28) b_1 and b_2 can be expressed in terms of θ:

$$b_1 = \frac{z\cosh(z/2) - 2\sinh(z/2)}{2z^2\theta\sinh(\theta z)}, \tag{29}$$

$$b_2 = \frac{-\cosh(\theta z)z\cosh(z/2) + 2\cosh(\theta z)\sinh(z/2) + 2\sinh(z/2)z\theta\sinh(\theta z)}{z^2\theta\sinh(\theta z)}. \tag{30}$$

Introducing these results for b_1 and b_2 into (25) reproduces an equation for θ:

$$\frac{(1-\cosh(\theta z))\,(z\cosh(z/2) - 2\sinh(z/2)) + z\theta\sinh(\theta z)\,(2\sinh(z/2) - z)}{z^2\theta\sinh(\theta z)} = 0\,.$$

From the symplecticness condition (22) an expression for γ_1 follows:

$$\gamma_1 = \frac{\gamma_2\cosh(2\theta z)(z\cosh(z/2) - 2\sinh(z/2))}{\cosh(\theta z)(z\cosh(z/2) - 2\sinh(z/2)) - 2\sinh(z/2)z\theta\sinh(\theta z)(1-\gamma_2\cosh(z/2))}\,. \tag{31}$$

Again by chosing $\gamma_2 = 1$ series expansions for the different parameters can be obtained (see [23]).

Case $(K' = -1, P' = 2)$:

The equations (27) and (28) remain unchanged. A third equation is added which follows from the application of the operator (15) with $s = 3$ on $t^2\exp(\pm\lambda t)$, i.e.:

$$b_1\cosh(z\theta)\left(2\cosh(z/2) + \frac{1}{2}z\sinh(z/2) + 2z\theta^2\sinh(z/2)\right) - \cosh(z) \tag{32}$$

$$+2b_1\sinh(z\theta)\,(2\theta\sinh(z/2) + z\theta\cosh(z/2)) + b_2\left(\cosh(z/2) + \frac{1}{4}z\sinh(z/2)\right) = 0\,.$$

The formal expression for b_1 and b_2 remain respectively (29) and (30). Introducing these expression for b_1 and b_2 into (32) gives us an equation for θ. From the symplecticness condition (22) again the expression (31) for γ_1 follows. Again by chosing $\gamma_2 = 1$, the series expansion of the different parameters follow (see [23]).

3.3 Four-stage methods

In theory we can combine the $(K = 1, P = 0)$ or the $(K = -1, P = 1)$ (see table 1) with the four non-polynomial cases for the final stage to construct EFRK methods. This gives rise to eight different EFRK methods of the Gauss type. However in most of the cases the values of the occurring parameters can only be determined numerically by solving quite complicated nonlinear equations. Therefore we have opted to analyse the simplest method which follows from the combination of the sets $(K = 1, P = 0)$ and $(K' = 5, P' = 0)$. A detailed description of these results can be found in [24].

 ISBN 13: 978-84-613-5510-5

4 Numerical experiments

In this section we report on some numerical experiments where we test the effectiveness of the new and the previous [4] modified Runge-Kutta sixth-order methods when they are applied to the numerical solution of several differential systems. All the considered codes have the same qualitative properties for the Hamiltonian systems. In the figures we show the decimal logarithm of the maximum global error versus the number of steps required by each code in logarithmic scale. All computations were carried out in double precision and series expansions are used for the coefficients when $|z| < 0.1$. In all further displayed figures following results are shown: the method of Calvo *et al.* [4] with constant nodes (const) and with variable nodes (var), the classical Gauss results (class) [25] and the results obtained with the new methods with $P' = 0$ (P0), $P' = 1$ (P1) and $P' = 2$ (P2). Let us remark that for the fourth- and eighth-order methods quite similar results are obtained (see [22, 24]).

Problem 1: Kepler's plane problem defined by the Hamiltonian function

$$H(p,q) = \frac{1}{2}(p_1^2 + p_2^2) - (q_1^2 + q_2^2)^{-1/2} ,$$

with the initial conditions $q_1(0) = 1-e, q_2(0) = 0, p_1(0) = 0, p_2(0) = ((1+e)/(1-e))^{\frac{1}{2}}$, where $e, (0 \leq e < 1)$ represents the eccentricity of the elliptic orbit. The exact solution of this IVP is a 2π-periodic elliptic orbit in the (q_1, q_2)-plane with semimajor axis 1, corresponding the starting point to the pericenter of this orbit. In the numerical experiments presented here we have chosen the same values as in [4], i.e. $e = 0.001, \lambda = i\omega$ with $\omega = (q_1^2 + q_2^2)^{-\frac{3}{2}}$ and the integration is carried out on the interval $[0, 1000]$ with the steps $h = 1/2^m, m = 1, \ldots, 4$. The numerical behaviour of the global error in the solution is presented in figure 1. The results obtained by the three new constructed methods are falling together. One cannot distinguish the results. They are comparable to the ones obtained by Calvo and more accurate than the results of the classical Gauss method of the same order.

Problem 2: A perturbed Kepler's problem defined by the Hamiltonian function

$$H(p,q) = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{(q_1^2 + q_2^2)^{1/2}} - \frac{2\epsilon + \epsilon^2}{3(q_1^2 + q_2^2)^{3/2}} ,$$

with the initial conditions $q_1(0) = 1, q_2(0) = 0, p_1(0) = 0, p_2(0) = 1 + \epsilon$, where ϵ is a small positive parameter. The exact solution of this IVP is given by

$$q_1(t) = \cos(t + \epsilon t), \quad q_2(t) = \sin(t + \epsilon t), \quad p_i(t) = q_i'(t), i = 1, 2 .$$

As in [4] the numerical results are computed with the integration steps $h = 1/2^m, m = 1, \ldots, 4$. We take the parameter $\epsilon = 10^{-3}, \lambda = i\omega$ with $\omega = 1$ and the problem is integrated up to $t_{end} = 1000$. The global error in the solution is presented in figure 2. For our methods we have the same conclusions as for the Problem 1. On the contrary

 ISBN 13: 978-84-613-5510-5

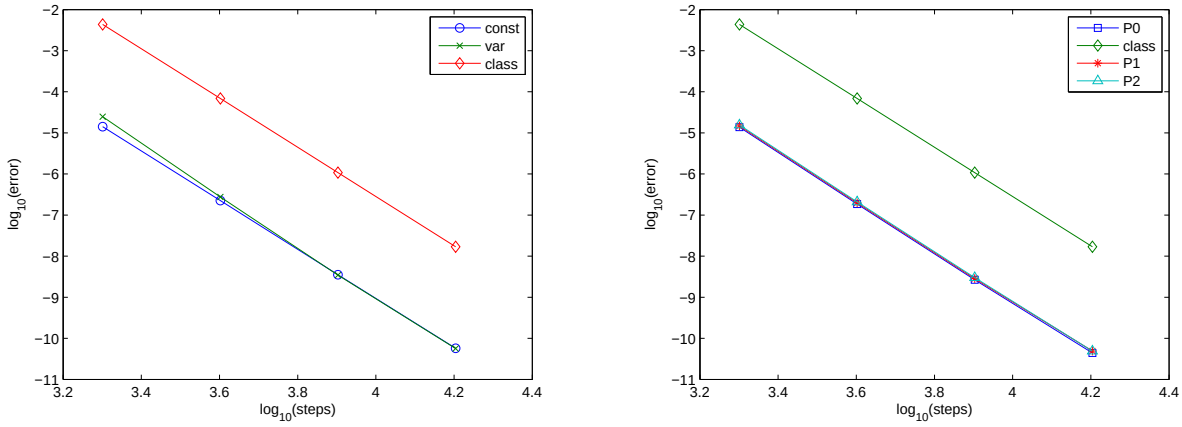


Figure 1: Maximum global error in the solution of Problem 1. In the left figure the results obtained by the methods of Calvo *et al.* [4] are displayed. In the right figure the results obtained with the methods of order six derived in this paper are shown.

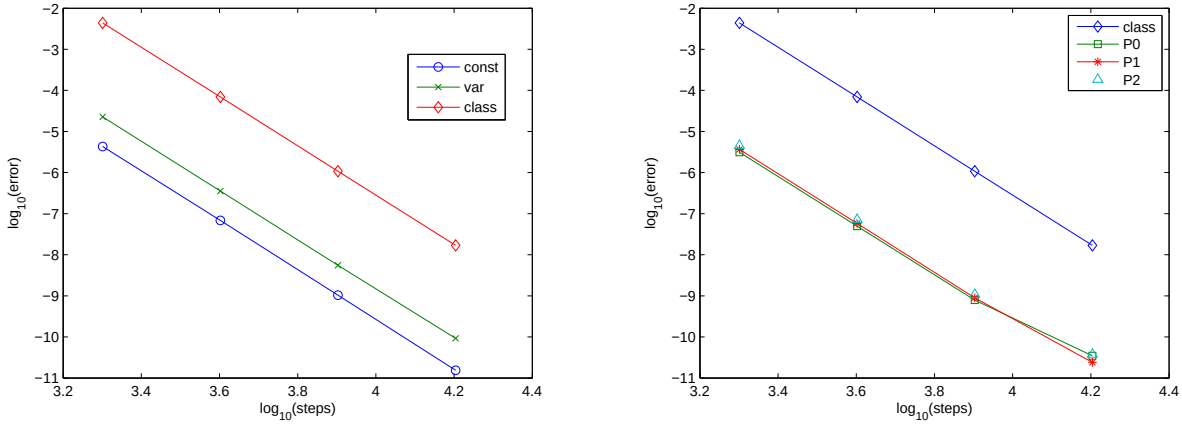


Figure 2: Maximum global error in the solution of Problem 2. In the left figure the results obtained by the methods of Calvo *et al.* [4] are displayed. In the right figure the results obtained with the methods of order six derived in this paper are shown.

 ISBN 13: 978-84-613-5510-5

for the results of Calvo *et al* the results obtained with fixed θ-values are more accurate than the ones obtained by variable θ-values.

Problem 3: Euler's equations that describe the motion of a rigid body under no forces

$$\dot{q} = f(q) = ((\alpha - \beta)q_2 q_3, (1-\alpha)q_3 q_1, (\beta - 1)q_1 q_2)^T \ ,$$

with the initial values $q(0) = (0, 1, 1)^T$, and the parameter values $\alpha = 1 + \dfrac{1}{\sqrt{1.51}}$ and $\beta = 1 - \dfrac{0.51}{\sqrt{1.51}}$. The exact solution of this IVP is given by

$$q(t) = \left(\sqrt{1.51}\,\mathrm{sn}(t, 0.51), \mathrm{cn}(t, 0.51), \mathrm{dn}(t, 0.51)\right)^T \ ,$$

it is periodic with period $T = 7.45056320933095$, and $\mathrm{sn}, \mathrm{cn}, \mathrm{dn}$ stand for the elliptic Jacobi functions. Figure 3 shows the numerical results obtained for the global error computed with the interation steps $h = 1/2^m, m = 1, \ldots, 4$, on the interval $[0, 1000]$, and $\lambda = i2\pi/T$. The results of Calvo *et al* are all of the same accuracy while in our approach the EF methods are still more accurate than the classical one. In this problem the choice of the frequency is not so obvious and therefore the differentiation between the classical and the EF methods is not so pronounced.

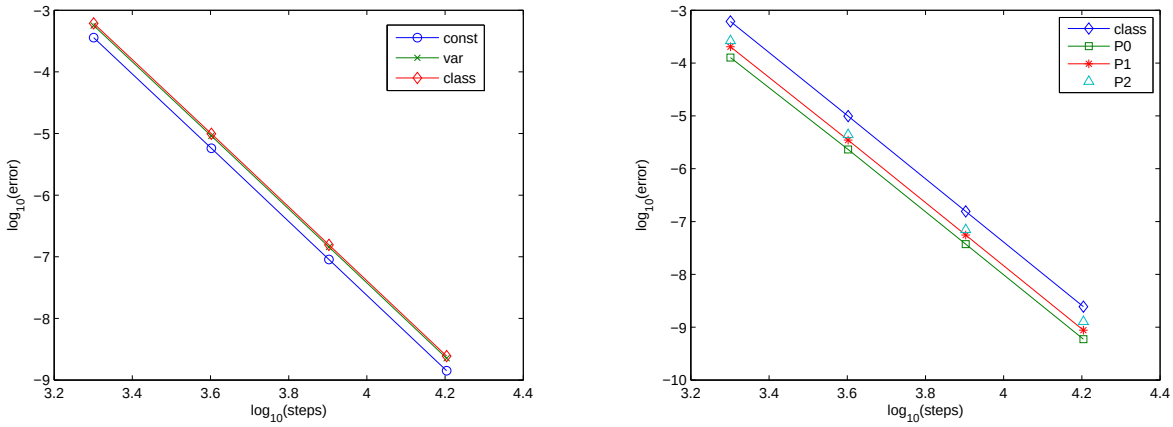


Figure 3: Maximum global error in the solution of Problem 3. In the left figure the results obtained by the methods of Calvo *et al.* [4] are displayed. In the right figure the results obtained with the methods of order six derived in this paper are shown.

References

[1] D. G. Bettis, Runge-Kutta algorithms for oscillatory problems, J. Appl. Math. Phys. (ZAMP) **30** (1979) 699–704.

 ISBN 13: 978-84-613-5510-5

[2] M. Calvo, J. M. Franco, J. I. Montijano, L. Rández, Structure preservation of exponentially fitted Runge-Kutta methods, Journ. Comp. Appl. Math. **218** (2008) 421-434.

[3] M. Calvo, J. M. Franco, J. I. Montijano, L. Rández, Sixth-order symmetric and symplectic exponentially fitted Runge-Kutta methods of the Gauss type, Comp. Phys. Commun. **178** (2008) 732–744.

[4] M. Calvo, J. M. Franco, J. I. Montijano, L. Rández, Sixth-order symmetric and symplectic exponentially fitted modified Runge-Kutta methods of the Gauss type, Journ. Comp. Appl. Math. **223** (2009) 387–398 .

[5] M. Calvo, J. M. Franco, J. I. Montijano, L. Rández, On high order symmetric and symplectic trigonometrically fitted Runge-Kutta methods with an even number of stages, BIT Numer. Math. **50** (2010) 3–21.

[6] J. M. Franco, Runge-Kutta methods adapted to the numerical integration of oscillatory problems, Appl. Numer. Math. **50** (2004) 427-443.

[7] K. Ozawa, A functional fitting Runge-Kutta method with variable coefficients, Japan J. Indust. Appl. Math. **18** (2001) 107–130.

[8] T. E. Simos, An exponentially-fitted Runge-Kutta method for the numerical integration of initial-value problems with periodic or oscillating solutions, Comp. Phys. Commun. **115** (1998) 1–8.

[9] T. E. Simos, J. Vigo-Aguiar, Exponentially-fitted symplectic integrator, Phys. Rev. E67 (2003) 1–7.

[10] G. Vanden Berghe, H. De Meyer, M. Van Daele, T. Van Hecke, Exponentially-fitted explicit Runge-Kutta methods, Comp. Phys. Commun. **123** (1999) 7–15.

[11] G. Vanden Berghe, H. De Meyer, M. Van Daele, T. Van Hecke, Exponentially-fitted Runge-Kutta methods, Journ. Comp. Appl. Math. **125** (2000)107–115.

[12] G. Vanden Berghe, M. Van Daele, H. Van de Vyver, Exponentially-fitted Runge-Kutta methods of collocation type: Fixed or variable knot points?, Journ. Comp. Appl. Math. **159** (2003) 217–239.

[13] H. Van de Vyver, A fourth order symplectic exponentially fitted integrator, Comp. Phys. Commun. **174** (2006) 255–262.

[14] L. Gr. Ixaru, G. Vanden Berghe, *Exponential Fitting*, Mathematics and its applications vol. 568, Kluwer Academic Publishers, 2004.

 ISBN 13: 978-84-613-5510-5

[15] E. Hairer, C. Lubich and G. Wanner,*Geometric Numerical Integration, Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer, Berlin 2002.

[16] E. Hairer, C. Lubich and G. Wanner, Geometric numerical integration illustrated by the Störmer /Verlet method, Acta Numerica **12** (2003) 399–450.

[17] M. Hochbruck and Ch. Lubich, A Gautschi-type method for oscillatory second-order differential equations, Numer. Math. **83** (1999) 403–426.

[18] M. Van Daele and G. Vanden Berghe, Geometric numerical integration by means of exponentially fitted methods, Applied Numerical Mathematics **57** (2007) 415–435.

[19] G. Vanden Berghe and M. Van Daele, Exponentially-fitted Störmer/Verlet methods, Journal of Numerical Analysis, Industrial and Applied Mathematics (JNAIAM) **1** (2006) 237-251.

[20] J. M. Sanz-Serna, Symplectic integrators for Hamiltonian problems: An overview, Acta Numerica **1** (1992) 243–286.

[21] J. M. Sanz-Serna, M. P. Calvo, *Numerical Hamiltonian Problems*, Chapman and Hall, London 1994.

[22] G. Vanden Berghe, M. Van Daele, Fourth-order symplectic exponentially-fitted modified Runge-Kutta methods of the Gauss type: a review, Journal of the Academia de Ciencias de Zaragoza (submitted).

[23] G. Vanden Berghe, M. Van Daele, Symplectic exponentially-fitted modified Runge-Kutta methods of the Gauss type: revisited, Annals of the European Academy of Sciences (on computational mathematics) (in press).

[24] G. Vanden Berghe, M. Van Daele, Symplectic exponentially-fitted four-stage Runge-Kutta methods of the Gauss type, Numerical Algorithms (submitted).

[25] E. Hairer, S. P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations I, Nonstiff Problems*, Springer-Verlag Berlin, Heidelberg 1993.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Evaluating the sparse matrix vector product on multi-GPUs

F. Vázquez[1], G. Ortega[1], J.J. Fernández[1] and E.M. Garzón[1]

[1] *Dpt Computer Architecture.Cra Sacramento s/n Almeria 04120 Spain, Almeria University*

emails: f.vazquez@ual.es, gloriaortega@ual.es, jjfdez@ual.es, gmartin@ual.es

Abstract

The sparse matrix vector product (SpMV) is considered as a key operation in engineering and scientific computing. Graphics Processing Units (GPUs) have recently emerged as platforms that yield outstanding acceleration factors. SpMV implementations for GPUs have already appeared on the scene. This work proposes and evaluates the parallel ELLR-T algorithm to compute SpMV on multi-GPUs architecture. A comparative analysis of ELLR-T against a variety of previous proposals on one GPU has been carried out based on a representative set of test matrices. The results show that ELLR-T reaches the best performance on one GPU device, above all if two parameters related to ELLR-T are optimized. A model to help to tune ELLR-T is described. Finally, the parallel ELLR-T is evaluated on a multi-GPU architecture based on Tesla C1060.

1 Introduction

The Matrix-Vector product (MV) is a key operation for a wide variety of scientific applications, such as image processing, simulation, control engineering and so on. For many applications based on MV, the matrix is large and sparse. Sparse matrices are involved in linear systems, eigensystems and partial differential equations from a wide spectrum of scientific and engineering disciplines [14]. For these problems the optimization of the sparse matrix vector product (SpMV) is a challenge because of the irregular computation of large sparse operations. Therefore, additional effort must be spent to accelerate the computation of SpMV. This effort is focused on the design of appropriate data formats to store the sparse matrices, since the performance of SpMV is directly related to the used format as shown [11, 12, 15].

Currently, Graphics Processing Units (GPUs) offer massive parallelism for scientific computations. The use of GPUs for general purpose applications has exceptionally

 ISBN 13: 978-84-613-5510-5

increased in the last few years thanks to the availability of Application Programming Interfaces (APIs), such as Compute Unified Device Architecture (CUDA) and OpenCL [10], that greatly facilitate the development of applications targeted at GPUs.

From the programmer's point of view, the GPU is considered as a set of SIMT (Single Instruction, Multiple Threads) multiprocessors. Each kernel (parallel code) is executed as a batch of threads organized as a grid of thread blocks (hereinafter *BS* denotes the size of every thread block). For the execution, each block is assigned to a Streaming Multiprocessor (SM) composed by eight cores called Scalar Processors (SP). The blocks in turn are divided into sets of 32 threads called warps. In order to optimize the exploitation of the NVIDIA GPU architecture the programmer has to attend to maximize: (1) the *multiprocessor occupancy*, that is the ratio between the number of active warps per multiprocessor and the maximum number of possible active warps. This goal can be achieved by choosing optimum value of *BS*, balancing the threads workload and avoiding the flow control instructions which can cause the divergence of threads, i.e. mantaining the multiprocessors on the device as busy as possible; and (2) the *bandwidth memory*, the memory management can be optimized if the access pattern of the different threads belonging to every half-warp (16 threads) verifies the coalescence and alignment conditions, then, it can be performed in parallel by all of them and the memory latency would be the same as that of a single access. Moreover, the use of texture memory improves the performance when the searched word is located within it [10].

Recently, several implementations of SpMV have been developed with CUDA and evaluated on GPUs [1, 3–5, 9]. Devising GPU-friendly matrix storage formats has been a key in these implementations. This work aims at presenting and evaluating a new approach to increase the performance of SpMV on multi-GPUs which relies on the storage format for the sparse matrix, ELLPACK-R [13]. This format is a GPU-friendly variant of one previously designed for vector architectures, ELLPACK [7]. An extensive performance evaluation of this new approach has been carried out based on a representative set of test matrices. The comparative study has drawn the conclusion that the implementations ELLR-T based on ELLPACK-R proves to outperform the most common and efficient formats for SpMV on GPUs used so far.

Next Section 2 reviews the different formats to compress sparse matrices, given that the selection of an appropriate format is the key to optimize SpMV on GPUs. Section 3 introduces the parallel ELLR-T algorithm to compute SpMV on multi-GPUs. In Section 4 the performance of ELLR-T measured on a NVIDIA Geforce GTX 285 and on a multi-GPU architecture based on Tesla C1060 with a set of representative sparse matrices is presented. The results clearly show that the ELLR-T gets the best performance for all the test matrices and its performance can be optimized on one GPU device. The speed-up got on the multi-GPU architecture before mentioned is analyzed. Finally, Section 5 summarizes the main conclusions.

 ISBN 13: 978-84-613-5510-5

2 An overview of SpMV and its challenges

Let $u = Av$ be sparse matrix vector product where A is the sparse matrix, v and u are the input and output vectors respectively, every specific algorithm to compute $u = Av$ exploiting a particular architecture is related to a specific format to store A. Next, the main formats to compress sparse matrices and their corresponding algorithms are described, focusing on the formats specifically designed for SIMD architectures such as vector architectures and GPUs:

The *coordinate storage scheme (COO)* to compress a sparse matrix is a direct transformation from the dense format. Let Nz be the total number of non-zero entries of the matrix. A typical implementation of COO uses three one-dimensional arrays of size Nz. One array, $A[\,]$ of floating-point numbers, contains the non-zero entries. The other two arrays of integer numbers, $I[\,]$ and $J[\,]$, contain the corresponding row and column indices for each non-zero entry. The performance of SpMV based on COO is penalized because it does not implicitly include the information about the ordering of the coordinates, and, additionally, for multi-threaded implementations of SpMV atomic data access must be included when the elements of the output vector are written.

Compressed Row Storage (CRS) is the most extended format to store sparse matrices on superscalar processors. Let N and Nz be the number of rows of the matrix and the total number of non-zero entries of the matrix, respectively; the data structure consists of the following arrays: (1) $A[\,]$ array of floats of dimension Nz, which stores the entries; (2) $J[\,]$ array of integers of dimension Nz, which stores their column index; and (3) $start[\,]$ array of integers of dimension $N + 1$, which stores the pointers to the beginning of every row in $A[\,]$ and $J[\,]$, both sorted by row index. The code to compute SpMV based on CRS has several drawbacks that hamper the optimization of the performance of this code on superscalar architectures. First, the access locality of vector $v[\,]$ is not maintained due to the indirect addressing. Second, the fine grained parallelism is not exploited because the number of iterations of the inner loop is small and variable [6]. Despite these drawbacks, several optimizations have made possible to improve the performance of sparse computation on current processors [8,15]. In particular, the Intel Math Kernel Library (MKL) improves the performance of sparse BLAS operations, based on CRS, by optimizing the memory management and exploiting the ILP on Intel processors.

ELLPACK or ITPACK [7] was introduced as a format to compress a sparse matrix with the purpose of solving large sparse linear systems with ITPACKV subroutines on vector computers. This format stores the sparse matrix on two arrays, one float $A[\,]$, to save the entries, and one integer $J[\,]$, to save the column index of every entry. Both arrays are of dimension $N \times Max\_nzr$ at least, where N is the number of rows and Max_nzr is the maximum number of non-zeros per row in the matrix, with the maximum being taken over all rows. Note that the size of all rows in these compressed arrays $A[\,]$ and $J[\,]$ is the same, because every row is padded with zeros. Therefore, ELLPACK can be considered as an approach to fit a sparse matrix in a regular data structure similar to a dense matrix. Consequently, this format is appropriate to compute operations with sparse matrices on vector architectures.

 ISBN 13: 978-84-613-5510-5

Focusing our interest on the GPU architecture and if every element i of vector u is computed by a thread identified by index $x = i$ and the arrays store their elements in column-major order, then the SpMV based on ELLPACK can improve the performance due to: (1) the coalesced global memory access, thanks to the column-major ordering used to store the matrix elements into the data structures. Then, the thread identified by index x accesses to the elements in the x row: $A[x+k*N]$ with $\{0 \leq k < Max\_nzr\}$ where k is the column index into the new data structures $A[\,]$ and $J[\,]$. Consequently, two threads x and $x+1$ access to consecutive memory address, thereby fulfilling the conditions of coalesced global memory access; (2) non-synchronized execution between different thread blocks. Every thread block can complete its computation without synchronization with other blocks. However, if the percentage of zeros is high in the ELLPACK data structure and there is a relevant amount of padding zeros, then the performance decreases. This penalty even remains when conditional branches are included to avoid the memory access and arithmetic operations with padding zeros.

Recently, different *proposals of kernels to compute SpMV on GPUs* have been described and analyzed [1, 3–5, 9]. They can be classified in two groups according to their relationship with CRS or ELLPACK formats.

On the one hand, the kernel called CRS(vector) evaluated in [3] is based on CRS format. This kernel computes every output vector element with the collaboration of the 32 threads of every warp. So, one warp computes the float products related to the entries of one row in a cyclic fashion, followed by a parallel reduction in shared memory in order to obtain the final result of output vector element. Similarly, another kernel to compute SpMV on GPUs based on CRS format has been recently proposed in [1]. Here the collaboration of 16 threads (half warp) computes every output vector element, and zero-padding are added to every row to complete a length multiple of 16, in order to fulfill the memory alignment requirements and improve the coalesced memory access. It has been included on the SpMV4GPU library [2], and hereinafter it will be referred to the same name.

On the other hand, the kernels related to the format called HYB (which stands for hybrid) proposed by [3] seem to yield the best performance on GPUs so far. This format combines the ELLPACK and COO formats with the goal of improving the performance of ELLPACK. Let A be a sparse matrix stored with CRS format, then a preprocessing step is required to store it with HYB format in order to compute: (1) parameter Max_nzr, (2) distribution function of rows according to their number of entries, (3) subset of a specific percentage of rows with less entries, for example 2/3 [3], and its corresponding parameter Max_nzr', and, finally, (4) two data structures to store A. Max_nzr' entries of every row are stored in ELLPACK format, and if any entries remain, they are stored with COO format. In other words, HYB stores the sparse matrix with ELLPACK avoiding the elements which overfill some rows and storing them with COO format. So, the corresponding computation of SpMV based on GPU is split in several kernels related to the different formats, hopefully with an appropriate value of Max_nzr' the main kernel related to ELLPACK can reach high performance on GPU, but the kernels related to COO format adds relevant penalties due mainly to un-coalesced memory access and the need to use atomic functions for the memory

 ISBN 13: 978-84-613-5510-5

write operations. This drawback could be relevant especially for any kind of patterns of sparse matrices where the computation of Max_nzr' does not reach optimum value.

Lately, the format called *Sliced ELLPACK* has been proposed and evaluated in [9]. In order to compress the matrix, the N rows of A are partitioned in sets of S rows and every set is stored with ELLPACK format. Moreover, the τ threads into every block collaborate in the computation related to every set of rows. It achieves high performance when a preprocess with reordering of rows is considered and the optimum values of the parameters S and τ are selected. Other format, called BELLPACK, has been proposed in [5], this proposal compresses the sparse matrix by small dense entries blocks. Then, this approach reaches better performance for those sparse matrices with their pattern including small blocks of entries. Both approaches, *Sliced ELLPACK* and BELLPACK, include complex pre-processing of the sparse matrix.

3 Computing SpMV with ELLR-T algorithm on multi-GPUs

ELLPACK-R consists of two arrays, $A[\ ]$ (float) and $J[\ ]$ (integer) of dimension $N \times Max\_nzr$; and, moreover, an additional integer array called $rl[\ ]$ of dimension N (i.e. the number of rows) is included with the purpose of storing the actual length of every row, regardless of the number of the zero elements padded.

According to the mapping of threads in the computation of every row, several implementations of SpMV based on ELLPACK-R can be developed. Thus, when T threads compute the element $u[i]$ accessing to the i-th row, the implementation is referred as ELLR-T. So, the i-th row is split in sets of T elements. Then, in order to compute the element $u[i]$, T threads compute $\lceil rl[i]/T \rceil$ iterations of the inner loop of SpMV, every thread stores its partial computation in the shared memory. Finally, to generate the value of $u[i]$, one reduction of the T values computed and stored in shared memory has to be included. The value of parameter T can be explored in order to obtain the best performance with every kind of sparse matrices. Figure 1 illustrates the code of ELLR-T algorithm. The algorithms ELLR-T to compute SpMV with GPUs take advantage of: (1) *Coalesced and aligned global memory access.* The access to read the elements of A, J and rl are coalesced and aligned thanks to the column-major ordering used to store the matrix elements and the zeros-padding to complete the length of every row as multiple of 16. Consequently, the highest possible memory bandwidth of GPU is exploited. (2) *Homogeneous computing within the warps.* The threads belonging to one warp do not diverge when executing the kernel to compute SpMV. The code does not include flow instructions that cause serialization in warps since every thread executes the same loop, but with different number of iterations. Every thread stops as soon as its loop finishes, and the remaining threads continue the execution.(3) *Reduction of useless computation and unbalance of the threads of one warp.* Let S_i be the set of T threads which are collaborating on the computation of $u[i]$, the k-loop reaches the maximum value of $k = \lceil rl[i]/T \rceil \leq \lceil Max\_nzr/T \rceil$ for specific sets, S_i, into the warp. Then, the run-time of every warp is proportional to maximum element of the sub-vector $\lceil rl[i]/T \rceil$

 ISBN 13: 978-84-613-5510-5

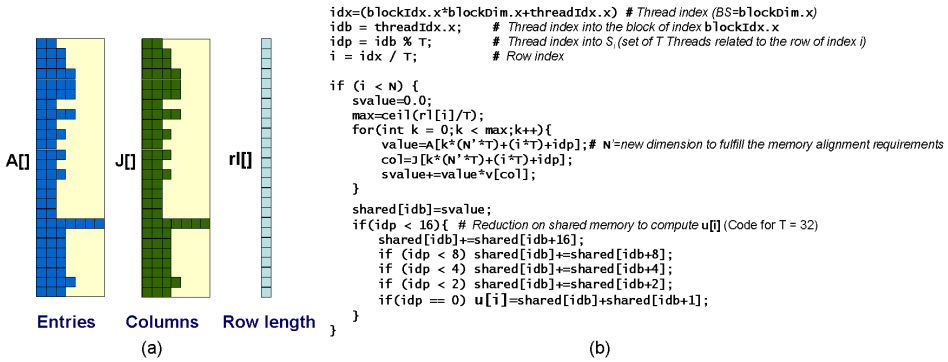


Figure 1: (a) ELLPACK-R Format and (b)ELLR-T code to compute SpMV on GPUs

related with every warp, and it is not necessary that the k-loop for all threads reaches $k = \lceil Max\_nzr/T \rceil$, then, there are not useless iterations and the control of loops of this implementation is reduced comparing with SpMV based on ELLPACK. (4) *High occupancy*. High occupancy levels are reached as it will be shown in next section, if optimal value of thread block size (BS) is used.

If several GPU devices are available, other parallelism level can be exploited to compute SpMV. Every device has its local memory space. In order to compute $u = Av$ on a multi-GPU architecture, the matrix A is distributed by row blocks between D devices. Then, the ELLR-T algorithm is computed with every sub-matrix A_d where $1 \leq d \leq D$. Therefore the output vector, u, is distributed between the D devices without any additional communication process because every local memory stores the whole vector v. Figure 2 illustrates the distributed computing of SpMV based on ELLR-T algorithm with several GPUs devices. This approach can help to accelerate the SpMV,

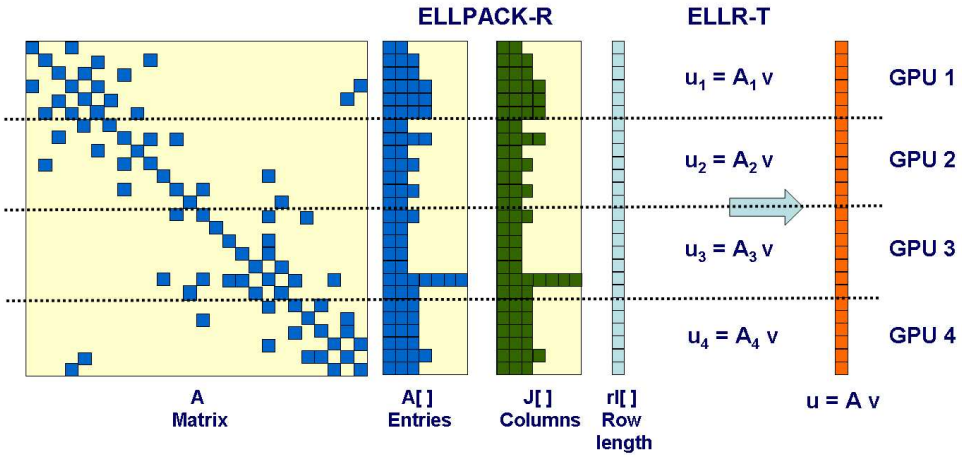


Figure 2: Distributed computing of SpMV based on ELLR-T algorithm with four GPUs devices

 ISBN 13: 978-84-613-5510-5

specially when the dimension of sparse matrix is large. However, the performance can be penalized by the load unbalance due to the irregular distribution of matrix entries between the GPU devices. Next section includes an analysis of the performance achieved by ELLR-T on multi-GPU architectures.

4 Evaluation

In order to evaluate ELLR-T on multi-GPU architectures two analysis have been carried out. First, ELLR-T has been evaluated on a GeForce GTX 285 with a set of test sparse matrices from different disciplines of science and engineering which are described in Table 1. The results comparative analysis shows that the best performance on one GPU is achieved with ELLR-T, specially, if the optimum values of two parameters are selected by means the proposed selection model. And, second, parallel ELLR-T* has been evaluated on a multi-GPU architecture based on Tesla C1060, where $\star$ denotes ELLR-T with optimum value parameters. The results show that parallel ELLR-T* on multi-GPU relevantly accelerates the SpMV.

4.1 Comparative evaluation of ELLR-T on GPU

The SpMV computations with GPU based on the following formats to store the matrix have been evaluated: CRS, CRS(vector), SpMV4GPU, ELLPACK, HYB and ELLR-T*. Table 1 summarizes the characteristic parameters related to specific patterns of the set of test matrices: number of rows (N), total number of non-zeros elements (*Entries*), average number of entries per row (Av), the difference between the maximum number of entries in a row and Av (IAv), percentage of relative standard deviation of entries by row($\frac{\sigma}{Av}$). Moreover Table 1 shows the bandwidth (BW) and speed-up (sp) reached with ELLR-T* on the GPU GeForce GTX 285. The values of these parameters are key to justify the differences between the performance achieved by SpMV with the different formats, which are primarily related to the variability or dispersion of the number of entries by row of the matrices. All matrices are real of dimensions $N \times N$.

The programming interface, CUDA, allows the programmer to specify which variables are to be stored in the texture cache within the memory hierarchy [10]. Here, the vector v has been stored binding to the texture memory for all kernels evaluated, since only the vector v is reused throughout the products with the different rows of the matrix, in the computation of $u = Av$.

The evaluation results show that the best average performance is got by ELLR-1 followed by HYB and ELLPACK, and the worst average performance is obtained by CRS, CRS(vector) and SpMV4GPU. However, the performance of ELLR-T can be highly increased if the values of two parameters are appropriately selected: the thread block size, BS, before mentioned in Section 1, and T recently mentioned. The possible values of BS are the powers of two from 16 to 512. The experimental results have shown that only for $BS = 128, 256, 512$ the kernels ELLR-T reach 100% occupancy of GPU, and for $BS = 16, 32, 64$ the occupancy is equal to or less than 50%. Then, it is predictable that the performance decreases with the smaller values of BS. On the

 ISBN 13: 978-84-613-5510-5

Table 1: Set of test matrices, characteristic parameters related to entries distribution on the rows; Effective Bandwidth of memory access (BW) and net speed-up (sp) of SpMV with ELLR-T* on the GPU GeForce GTX 285

Matrix	N	*Entries*	Av	IAv	$\frac{\sigma}{Av}$	BW	sp
qh1484	1.484	6.110	4,1	8,9	38,9	6,5	1,4
dw2048	2.048	10.114	4,9	3,1	10,2	10,8	1,9
rbs480a	480	17.087	35,6	0,4	1,4	14,0	1,2
gemat12	4.929	33.111	6,7	37,3	44,8	19,9	3,5
dw8192	8.192	41.746	5,1	2,9	12,0	34,6	5,3
mhd3200a	3.200	68.026	21,3	11,7	27,4	41,2	4,2
e20r4000	4.241	131.556	31,0	31,0	49,6	53,5	5,2
bcsstk24	3.562	159.910	45,0	12,0	25,6	69,9	13,9
mac_econ	206.500	1.273.389	6,2	37,8	71,9	38,1	7,9
qcd5_4	49.152	1.916.928	39,0	1,0	0,0	120,2	15,9
mc2depi	525.825	2.100.225	4,0	0,0	1,9	118,0	23,5
rma10	46.835	2.374.001	50,7	94,3	56,1	99,5	13,7
cop20k_A	121.192	2.624.331	21,6	59,3	63,7	70,1	21,0
wbp128	16.384	3.933.095	240,1	15,9	14,5	110,5	14,0
dense2	2.000	4.000.000	2.000	0,0	0,0	121,1	14,8
cant	62.451	4.007.383	64,2	13,8	21,9	121,4	31,7
pdb1HYS	36.417	4.344.765	119,3	84,7	26,7	119,5	30,5
consph	83.334	6.010.480	72,1	8,9	26,4	120,0	33,2
shipsec1	140.874	7.813.404	55,5	46,5	20,0	121,8	31,9
pwtk	217.918	11.634.424	53,4	127,6	8,9	128,3	32,8
wbp256	65.536	31.413.932	479,3	32,7	14,7	94,8	12,6

other hand, focussing our interest on the other parameter T, the values of T are divisors of BS, then $T = 2^l < BS$; our experimental results have shown that ELLR-T does not reach the highest performance for $T \geq 16$. Then, the kernel ELLR-T can achieve better performance if $BS = 128, 256, 512$ and $T = 1, 2, 4, 8$.

The performance obtained by SpMV on GPUs is strongly related to the A pattern, since the irregularities of the entries distribution result in a workload unbalance among the threads. Taking account the described details of ELLR-T algorithm in Section 3 and the execution model with streams of threads at warp levels on GPU [5, 10], the unbalance between threads due to the irregular filling of the A rows, $U(rl, T)$, can be estimated as:

$$U(rl,T) = \frac{\sum_{w=0}^{w=\lceil N/32 \rceil - 1} (\sum_{x=0}^{x=(32/T)-1} T * \lceil \frac{M_w}{T} \rceil - rl(32/T * w + x))}{\sum_{k=0}^{k=N-1} rl(k)} \quad (1)$$

where x is the thread identifier belonging to the warp with index w and $M_w = MAX\{rl(32/T * w + x)/0 \leq x < 32/T\}$ represents the maximum row length into the set of rows related to the warp w. Notice that, this unbalance model is only related to rl (the rows filling of A) and T (the number of threads collaborating to compute

 ISBN 13: 978-84-613-5510-5

one element of output vector u). On the other hand, the number of memory accesses (NMA) to read the data structure defined by ELLPACK-R increases when T increases, since T threads have to read the element $rl[i]$ to compute $u[i]$. The experimental results have shown that the values of T, which minimize $U(rl, T) + NMA$ for the optimum value of BS, get a performance which differs 15,85% of the optimum and it exactly matches to the experimental optimum values for 50% of the set of test matrices.

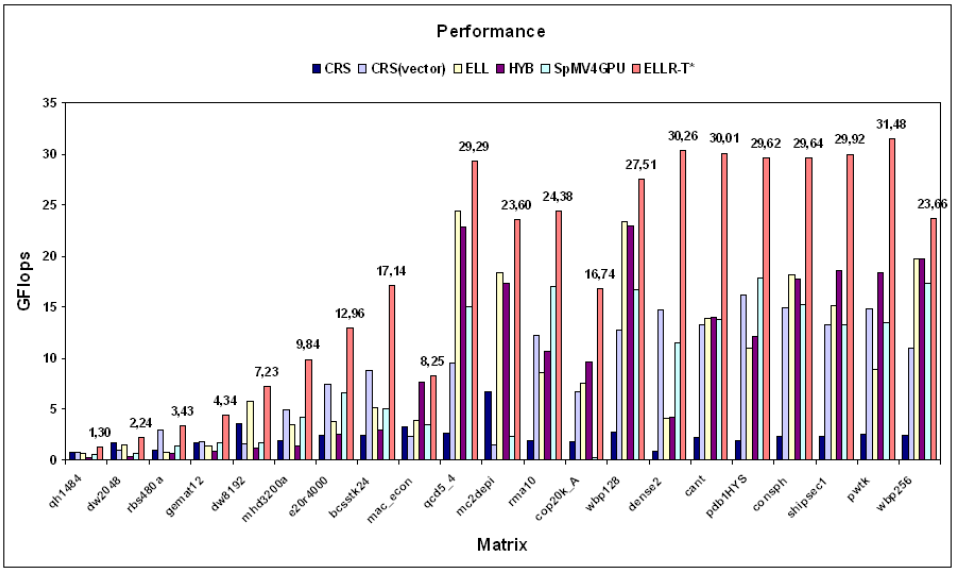


Figure 3: Performance of SpMV based on different formats on GPU GeForce GTX 285 with the set of test matrices, using the texture cache memory.

Figure 3 shows the performance (GFLOPs) of the SpMV kernels based on the formats that have been evaluated: CRS, CRS(vector), SpMV4GPU, ELLPACK, HYB and, moreover, the kernel ELLR-T* with optimal values of BS and T. The results shown in that figure allow us to highlight the following major points: (1) As any parallel implementation of SpMV, the performance obtained by most formats increases with the number of non-zero entries in the matrix, since small matrices do not generate a relevant computational load to reach high parallel performance. Thus, in general, as the dimension of matrices increases, the performance improves. (2) In general, the CRS format yields the poorest performance because the pattern of memory access is not coalescent; (3) The CRS(vector) and SpMV4GPU formats achieve better performance than CRS with most matrices, specially when Av is higher and the distribution of entries is more regular, i.e. $\frac{\sigma}{Av}$ is lower. SpMV4GPU reaches higher performance than CRS(vector) because it better exploits the power of threads, as sixteen threads collaborate to compute every u element, and perform a total coalesced memory access. (4) In general, ELLPACK outperforms both CRS-based formats, however its computation is penalized for some particular matrices, mainly due to the relevance of useless computation of the warps when the matrix histogram includes rows with very uneven length.

 ISBN 13: 978-84-613-5510-5

(5) The performance obtained by HYB is, in general, higher than the four previous formats, but it is remarkable its poorer results for smaller matrices due to the penalty introduced by the call to three different kernels necessary to compute SpMV. Moreover, with specific matrices of higher dimension (qcd5_4, mc2depi, cop20k_A, wbp128, consph, wbp256) it reaches lower or similar performance than ELLPACK, because the percentage of entries stored with ELLPACK format is near to 100 %.(6) Finally, the kernel ELLR-T* based on the format ELLPACK-R clearly achieves the best performance for all matrices considered in this work. In particular, it achieves the highest performance with matrices of high dimensions and higher values of parameters IAv, and $\frac{\sigma}{Av}$.

Memory optimizations are very relevant to maximize the performance of the GPU. The goal is to maximize the use of the hardware by maximizing bandwidth. Table 1 shows the effective bandwidth achieved when SpMV is computed with ELLR-T* on the GPU GeForce GTX 285 for the set of test matrices in Table 1. It is high specially for matrices with large dimension. So, for these matrices the effective bandwidth ranges from 90 to 128 GBps, that is 57-80% of the peak bandwidth (159 GBps) for this card.

In order to estimate the net gain provided by GPUs in the SpMV computation, we have taken the best optimized SpMV implementations for modern processors and for GPUs. For the former, we have considered the MKL implementation of SpMV for a computer based on a state-of-the-art superscalar core, Intel Core 2 Duo E8400, and evaluated the computing times for the set of test matrices. For the GPU GeForce GTX 285, we used the ELLR-T*, which is the best for the GPU according to the results presented above. Table 1 shows the speedup factors obtained by the SpMV operation on the GPU against one superscalar core, for all the test matrices. The results show that the speedup depends on the matrix pattern, though, in general, it increases with the number of non-zero entries. The speedup achieves values higher than 30$\times$ for matrices of large dimensions and higher number of entries. In view of the results related to the effective bandwidth and the speed-up achieved by ELLR-T*, we can conclude that the GPU turns out to be an excellent accelerator of SpMV by means the ELLR-T* algorithm.

4.2 Evaluation of ELLR-T on multi-GPU

Figure 4 shows the speed-up obtained by parallel ELLR-T* using two and three devices against one device of a multi-GPU architecture based on Tesla C1060 with the set of test matrices. It is remarkable that the parallel scalability is strongly related to the total number of entries and the dimension of A matrix. So, for matrices of large dimension and large number of entries the speed-up obtained is nearly the ideal value or even over-linear speed-up is achieved (mc2depi and wbp256). However, the differences of speed-up obtained with different matrices show that the irregularities of the matrix pattern can unbalance the work load of the devices penalizing the performance. For matrices with smaller dimensions the parallel ELLR-T* is not scalable because the work load is low and, consequently, the occupancy levels of the devices are low. Then, if the work-load of $u = Av$ is enough, that is, the dimension and/or the number of A entries

 ISBN 13: 978-84-613-5510-5

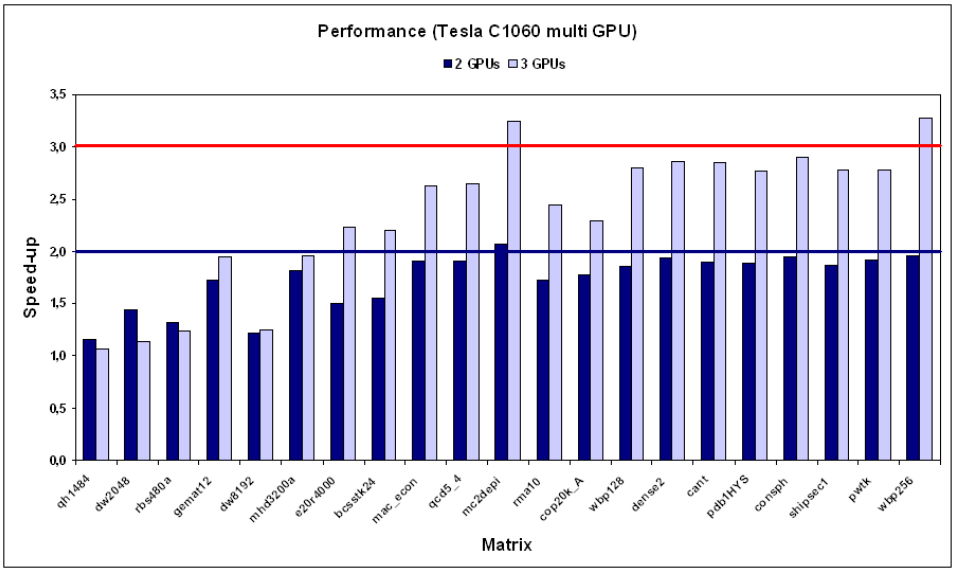


Figure 4: Speed-up of parallel ELLR-T* obtained using two and three devices against one device on the multi-GPU architecture based on Tesla C1060 with the set of test matrices.

are larger, then, the ELLR-T* kernel relevantly accelerate the SpMV on multi-GPU.

5 Conclusions

In this paper a new approach to compute the sparse matrix vector product on multi-GPUs has been proposed and evaluated. Parallel ELLR-T is based on the ELLR-T algorithm to compute SpMV on one GPU device. The comparative evaluation with other proposals has shown that the performance achieved by ELLR-T is the best, after an extensive study on a set of representative test matrices; and a model has been described in order to help to tune the performance of ELLR-T on the GPU architecture with every particular sparse matrix. The evaluation results have shown that the SpMV can be accelerated with multi-GPU architecture and ELLR-T* algorithm if the SpMV has enough work load to warranty that the maximum device occupancy levels are achieved.

Acknowledgment

This work has been funded by grants from the Spanish Ministry of Science and Innovation TIN2008-01117 and Junta de Andalucia (P06-TIC-01426, P08-TIC-3518). Moreover, it has been developed in the framework of the network (CAPAP-H) TIN2009-08058-E.

 ISBN 13: 978-84-613-5510-5

References

[1] Baskaran MM, Bordawekar R. Optimizing Sparse Matrix-Vector Multiplication on GPUs. *IBM Research Report RC24704*. April 2009.

[2] Baskaran MM, Bordawekar R. Sparse Matrix-Vector Multiplication Toolkit for Graphics Processing Units. April, 2009 `http://www.alphaworks.ibm.com/tech/spmv4gpu`

[3] Bell N, Garland M. Implementing Sparse Matrix-Vector Multiplication on Throughput-Oriented Processors Proceedings of SC'09. `http://www.nvidia.com/object/nvidia_research_pub_013.html`

[4] Buatois L, Caumon G, Levy B. Concurrent number cruncher - A GPU implementation of a general sparse linear solver. *International Journal of Parallel, Emergent and Distributed Systems* 2009; **24**(3):205–223

[5] Choi JW, Singh A, Vuduc RW. Model-driven Autotuning of Sparse Matrix-Vector Multiply on GPUs *Proceedings of PPoPP10, 2010*

[6] Kurzak J, Alvaro W, Dongarra J. Optimizing matrix multiplication for a short-vector SIMD architecture - CELL processor. *Parallel Computing* 2009; **35**(3):138–150

[7] Kincaid DR, Oppe TC, Young DM. ITPACKV 2D User's Guide. *CNA-232* 1989. `http://rene.ma.utexas.edu/CNA/ITPACK/manuals/userv2d/`

[8] Intel. Math Kernel Library. Reference Manual
`http://software.intel.com/sites/products/documentation/hpc/mkl/mklman.pdf`

[9] Monakov,A; Lokhmotov, A; and Avetisyan, A. Automatically Tuning Sparse Matrix-Vector Multiplication for GPU Architectures
Proceedings of HiPEAC 2010, LNCS 5952, pp. 111- 125, 2010

[10] NVIDIA, CUDA Programming guide. Version 2.3, August, 2009. `http://developer.download.nvidia.com/compute/cuda/2_3/toolkit/docs/NVIDIA_CUDA_Programming_Guide_2.3.pdf`

[11] Ogielski AT, Aiello W. Sparse matrix computations on parallel processor arrays. *SIAM Journal on Scientific Computing* 1993; **14**:519–530.

[12] Toledo S. Improving the memory-system performance of sparse-matrix vector multiplication. *IBM Journal of Research and Development*. 1997; **41**(6):711–725

[13] Vázquez F, Garzón EM, Martínez JA, Fernández JJ. Accelerating sparse matrix vector product with GPUs. In Proc. 2009 International Conference on Computational and Mathematical Methods in Science and Engineering, vol 2:1081–1092. CMMSE, 2009.

[14] Vázquez F, Garzón EM, Fernández JJ. A matrix approach to tomographic reconstruction and its implementation on GPUs. *Journal of Structural Biology*, 2010; **170**:146–151

[15] Williams S, Oliker L, Vuduc R, Shalf J, Yelick K, Demmel J. Optimization of sparse matrix-vector multiplication on emerging multicore platforms. *Parallel Computing* 2009; **35**(3):178–194

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

GPU computing for 3D EM image classification

F.M. Vázquez-López[1], J.A. Martínez[1], J.J. Fernández[2] and J.R. Bilbao-Castro[1]

[1] *Dept. Arquitectura de Computadores y Electrónica, Universidad de Almería. 04120 Almería. Spain.*

[2] *Centro Nacional de Biotecnología - CSIC, Campus Universidad Autónoma. 28049 Madrid. Spain.*

emails: f.vazquez@ual.es, jmartine@ual.es, jose@ace.ual.es, jrbcast@ace.ual.es

Abstract

Image classification is an essential step in three-dimensional (3D) electron microscopy prior to the calculation of the 3D structure of biological specimens. KerDenSOM is a very well known algorithm in this field and has thus been successfully used in a number of experimental structural studies in molecular biology. One of the main drawbacks of KenDerSOM is the fact that it is an iterative, very time consuming algorithm. Despite its inherently parallel nature, its fine grain of parallelism does not adapt well to standard parallelization techniques such as message-passing and multi-threading computing. In the last few years, graphics processor units (GPUs) have emerged as new computing platforms that offer massive parallelism and that can be exploited for general purpose applications. In this work, we propose and evaluate a GPU-enabled implementation of the KerDenSOM algorithm of image classification. We show that the use of GPU computing for the efficient KerDenSOM execution is indeed feasible. Furthermore, the results we have obtained show that this GPU implementation turns out to be a good parallel alternative to this problem, providing good speed-up figures at a reduced cost.

Key words: parallel computing, 3D reconstruction, GPU computing, three-dimensional electron microscop, single particle electron microscopy, self-organizing map, neural network, SOM

MSC 2000: AMS codes (optional)

1 Introduction

Image classification in single-particle three-dimensional electron microscopy (3D EM) is an essential preprocessing step to reach high resolution 3D reconstructions [1, 2]. In this discipline, a biological sample is imaged with an electron microscope and micrographs of large EM fields are taken where individual particles of the specimen under study are widely dispersed and randomly oriented. These particles are then selected and boxed out from the micrograph. These

 ISBN 13: 978-84-613-5510-5

individual images are affected by a number of factors: low contrast, low signal-to-noise ratio (SNR), the effect of the transfer function of the microscope, and inhomogeneities or distortions due to the preparation techniques used for the biological sample. Image classification aims at obtaining better-quality, representative image projections of the specimen at different views. Individual images are grouped together into classes based on their similarities, considering that similar images can be seen as belonging to the same view. Thus, a unique, representative image of each class can be obtained by two-dimensional (2D) averaging of such images. Averaging of the images in a group yields a single image featuring a better SNR than the original, individual images. These average views are subsequently used to derive the 3D structure of the specimen [1, 2].

There exist different approaches that deal with image classification in 3D EM and that are based on different methodological approaches, such as correspondence analysis, multivariate statistical analysis, fuzzy logic, neural networks, etc. ([3, 4, 5, 6, 7, 8, 9, 10, 11, 12]). Certain image classification algorithms are also able to deal with relatively heterogeneous specimens where different conformations of the specimen are mixed in the biological sample and, in turn, in the image population [13, 14].

This work centers on the ”Kernel Density Self-Organizing Maps” (KerDenSOM) algorithm. The KerDenSOM algorithm [12] is based on a previous work by Marabini [11] who proposed the use of neural networks and Kohonen SOMs [15] for 3D EM image classification. KerDenSOM has been adopted by many groups and can be considered as a broadly used method for image classification in the 3D EM field. Despite its wide adoption, one of the main drawbacks of the KerDenSOM algorithm comes from its computational complexity. When dealing with big experiments (large size and/or number of projections), the algorithm becomes too costly in terms of execution time, thereby negatively impacting biologists' productivity. Its iterative nature and data characteristics are responsible for this major drawback.

Modern computing techniques allow the parallelization of complex applications so their workload can be distributed across multiple computers and/or processing units (CPUs), resulting in an overall lower execution time. High performance computing has turned out to be key to afford some of the computational stages involved in 3D EM [21, 22, 23, 24]. Therefore, these techniques are expected to play an important role to address the computational intensity of KerDenSOM. Nevertheless, the effectiveness of applying a parallelization strategy depends on different factors. One of these factors is known as the ”grain of parallelism”, and reflects the level at which the algorithm is parallelizable [25]. Coarse-grain applications can be parallelized at a high level, with a good rate of computation vs. synchronisations. On the other hand, some applications present a fine-grain parallelism which translates into a low computation vs. synchronizations ratio. In the former case, paralellization through ”classic” paradigms, like message-passing computing, through MPI [26] or shared memory computing through OpenMP [27] or POSIX Threads [28], is straightforward and good speed-ups can be expected. The latter case, however, does not adapt so well to such paradigms and poor speed-up figures should be expected. Despite the apparently parallel nature of self-organizing maps, the underlying algorithm exhibits fine-grain parallelism and the overhead due to synchronization prevents full exploitation of the computational power of parallel systems. Poor speed-up figures are thus obtained when parallelizing the KerDenSOM algorithm through ”classical” MPI and POSIX Threads programming models. So far, most successful parallel implementations

 ISBN 13: 978-84-613-5510-5

of self-organizing maps have been based on specific and/or expensive hardware that requires the algorithm be adapted (e.g transputers, FPGAs), adaptations of the algorithm to make it better suited to parallelization for clusters or parallel systems (e.g [29]), or the adoption of the master-worker paradigm with the master being a bottleneck in the parallelism [30].

Recently, Graphics Processing Units (GPUs) have emerged as new computing platforms that offer massive parallelism and provide incomparable performance-to-cost ratio for scientific computations [31]. The use of GPUs for general purpose applications has exceptionally increased in the last few years thanks to the availability of Application Programming Interfaces (APIs), such as Compute Unified Device Architecture (CUDA^{TM})[32, 33], that greatly facilitate the development of applications targeted at GPUs [34]. In this work we propose and evaluate a high performance implementation of the KerDenSOM algorithm based on CUDA^{TM}that makes use of relatively cheap, broadly available GPUs as the main source of computing power.

2 SOMs and KerDenSOM

KerDenSOM is a variant of the more general SOMs proposed by Kohonen in the 90's [15]. Those were developed for unsupervised data classification purposes. Through SOMs, a correspondence is obtained between the M-Dimensional input data members and a N-Dimensional output map, with N being smaller than M and usually equal to 2. Thus, one of the main applications of SOMs is to get a 2D representation of some multidimensional input data so it can be more easily explored and analysed.

A SOM consists of a matrix of points (neurons), known as "map" or "neural network". Neurons establish relationships with other neighbour neurons following different grid topologies (usually rectangular, hexagonal, toroidal, etc.. (see Fig. 1)).

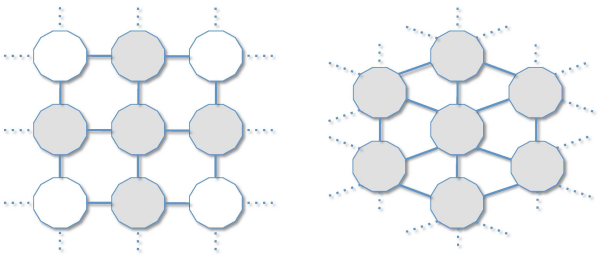


Figure 1: Two different arrangements for the grid of neurons; left, a rectangular grid and right, a hexagonal one.

Each neuron contains a coefficients vector known as "dictionary vector" (DV) which is M-Dimensional as the input data. When an input data member accesses the SOM, a single neuron is activated. That neuron, called the "winning neuron" (WN) will be that whose DV is more similar to the input member. Similarity is usually computed by means of the euclidean distance ($L2$). Given an input member (a vector of values) $x \in \Re^n$, it is compared against all

 ISBN 13: 978-84-613-5510-5

the DVs, m_i, in the SOM. The WN, called c, will be that satisfying:

$$\|x - m_c\| = \min_i\{\|x - m_i\|\}$$

Thus, the x vector is said to be mapped to the neuron c. Prior to this mapping or classification, the SOM must follow a training process that computes the values for m_i. The training process is as follows:

1. DVs (m_i) are initialized, generally with random values.

2. Each input member is mapped to the corresponding WN. The WN is updated and so are the neighbour neurons depending on the selected grid topology. At each training step t, the m_i values are updated according to:

   $$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)]$$

   where t is an integer, discrete value representing time and $h_{ci}(t)$ is the so-called neighbourhood kernel: $h_{ci}(t) = h(\|r_c - r_i\|, t)$, with $r_c \in \Re^2$ and $r_i \in \Re^2$ being the radius vectors of the c and i nodes respectively. The neighbourhood kernel tells whether and how much a neuron's DV will be updated. $h_{ci}(t)$ can adopt different expressions. The simplest kernel expression, known as "bubble", is that using the group of nodes around the c node, named as N_c. This kernel defines $h_{ci}(t) = \alpha(t)$ if $i \in N_c$ and $h_{ci} = 0$ if $i \notin N_c$, being $\alpha(t)$ a function $0 < \alpha(t) < 1$ known as learning rate. Another important kernel type can be expressed by means of a gaussian function:

   $$h_{ci}(t) = \alpha(t) \cdot \exp\left(- \frac{\|r_c - r_i\|^2}{2\sigma^2(t)} \right)$$

   where $\sigma(t)$ represents the kernel width or standard deviation, r_c and r_i are the radius vectors for N_c as commented above. Both $\alpha(t)$ and $\sigma(t)$ are monotonic decreasing time functions.

3. The process is repeated for a number of iterations or when a certain stop condition is reached (e.g. the variability of DVs fall below a certain threshold).

The SOM methods are robust against incoherent input elements which are usually present at all measurement processes. Thus, an atypical input element will affect a single neuron and only slightly to its neighbours, limiting its impact in the classification process. SOMs also allow to detect and discard atypical elements. Nevertheless, SOMs pose also certain limitations:

1. They do not define a density model in the data space but it is based on the $L2$ to determine the WN.

2. The training algorithm does not optimize an objective function.

3. Theoretical convergence is not guaranteed.

4. There is not a robust theoretical explanation on how the algorithm works.

 ISBN 13: 978-84-613-5510-5

KerDenSOM is a variant of the SOM algorithm based on the estimation of the kernel density probability [12]. It does not use the $L2$ to determine the WN but the probability density function (pdf). KerDenSOM was designed to find class representatives whose pdf was as similar as possible to that present on the input data elements. This is, in fact, its main advantage compared to other SOM implementations.

The mathematical formulation of the KerDenSOM is based on:

Given an input data set $X_i \in \Re^{p \cdot 1}$, with $i = 1..n$, treat to find a group of c data $V_j \in \Re^{p \cdot 1}$, with $j = 1..c$, so the pdf:

$$D(X) = \frac{1}{c} \sum_{j=1}^{c} K(X - V_j; \alpha)$$

being K a function of gaussian kernel:

$$K(X - X_i; \alpha) = \frac{1}{(2\Pi\alpha)^{\frac{p}{2}}} \exp\left(- \frac{\|X - X_i\|^2}{2\alpha} \right)$$

as similar as possible to the input data pdf. The α parameter is known as the kernel width and controls the smoothness of the pdf; $\alpha > 0$.

To achieve this, and as developed on [12, 18], the resulting expression representing a cost function l_s must be maximized:

$$l_s = -\frac{np}{2} \ln 2c\pi\alpha + \sum_{i=1}^{n} \left(\sum_{j=1}^{c} \exp\left(- \frac{\|X_i - V_i\|^2}{2\alpha} \right) \right) - \frac{\vartheta}{2\alpha} tr(VCV^T) \qquad (1)$$

where X are the input vectors, V are the DVs and C is the neighbourhood matrix representing relationships between neurons. ϑ is a regularization or smoothness factor that affects the convergence rate. Maximizing Eq. 1 is equivalent to make pdf's for input data and DVs coincide.

The KerDenSOM algorithm is very sensible to the initial data. Therefore, a good approach for the training process consists of applying a deterministic annealing process to the regularization factor ϑ by decrementing its value between a maximum (initial value) of ϑ_0 and a minimum, final value of ϑ_1. The training algorithm will start running with a high ϑ until a point of convergence. Then a new, smaller value is assigned to ϑ and the training is run again. As a result of the training process, we will obtain not only the DVs for the neurons (V matrix) but also the U matrix denoting the membership probability for each input vector/neuron pair.

Those are the steps of the KerDenSOM training process:

1. Initialize $\vartheta_1 > 0$, $\vartheta_0 > 0$, $Iter = 0$ and $MaxIter > 1$ (maximum number of training steps)

2. Initialize matrix U satisfying:

$$\left\{ \begin{array}{c} 0 \leq U_{ji} \leq 1 \\ \sum_{j=1}^{c} U_{ji} = 1, \forall i \end{array} \right\} \qquad (2)$$

 ISBN 13: 978-84-613-5510-5

3. Initialize matrix V following:

$$V_j = \frac{\sum_{i=1}^{n} X_i U_{ji}}{\sum_{i=1}^{n} U_{ji}} \tag{3}$$

4. Deriving and equalling Eq. 1 to zero, calculate α by:

$$\alpha = \frac{1}{np}\left(\sum_{i=1}^{n}\sum_{j=1}^{c} \|X_i - V_j\|^2 U_{ji} + \vartheta \sum_{j=1}^{c}\sum_{k=1}^{c} C_{jk} V_j^T V_k\right) \tag{4}$$

5. Repeat points 6 to 10 $MaxIter$ times.

6. Calculate ϑ for the current iteration following:

$$\vartheta = \exp\left(\ln(\vartheta_1) - (\ln(\vartheta_1) - \ln(\vartheta_0))\frac{Iter}{MaxIter}\right) \tag{5}$$

7. Calculate V_j for $j = 1..c$, as:

$$V_j = \frac{\sum_{i=1}^{n} U_{ji} X_i + \vartheta \overline{V}_j}{\sum_{i=1}^{n} U_{ji} + \vartheta} \tag{6}$$

   until DVs converge:

$$\|V_{jcurrent} - V_{jprevious}\|^2 < \varepsilon \tag{7}$$

8. Calculate α following Eq. 4.

9. Calculate U_{ji} for $i = 1..n$, $j = 1..c$ following equation:

$$U_{ji} = \frac{K(X_i - V_j; \alpha)}{\sum_{k=1}^{c} K(X_i - V_k; \alpha)} \tag{8}$$

10. Go back to point 7 while next expression is not satisfied:

$$\|U_{jicurrent} - U_{jiprevious}\|^2 < \varepsilon \tag{9}$$

Once the map has been trained, similar input members will be mapped to the same neuron in the map.

 ISBN 13: 978-84-613-5510-5

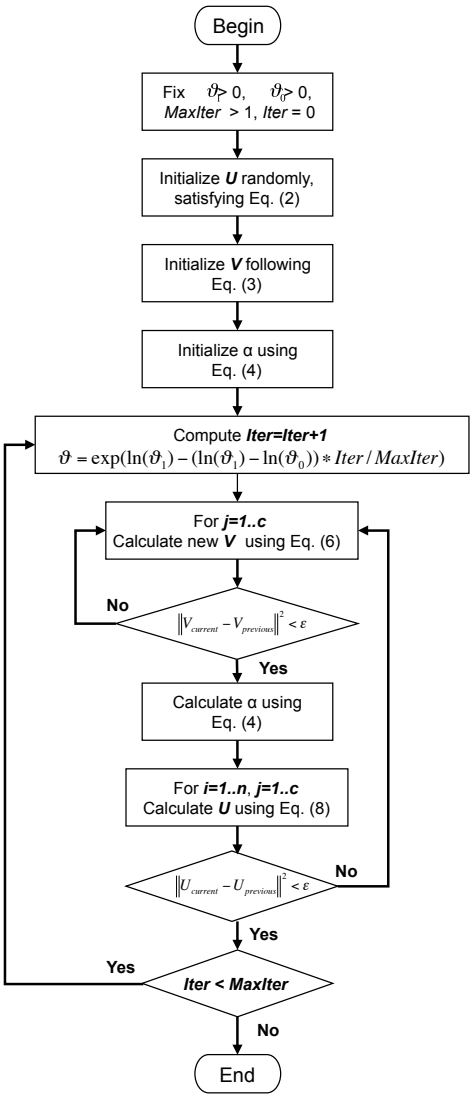


Figure 2: Workflow of the KerDenSOM algorithm. Note its iterative structure with two nested loops. This is one of the main reasons to its high computational costs and a difficulty for parallelization as sequentiality must be preserved.

 ISBN 13: 978-84-613-5510-5

3 GPU computing and CUDA^{TM}architecture

Graphics Processing Units or GPUs have evolved over the years from expensive pieces of hardware devoted to unload CPUs from graphics manipulation to cheap, sophisticated and massively parallel general-purpose computing architectures (GPGPUs). GPGPUs are GPUs whose design has been adapted to general purpose computation needs (i.e. 64-bit processing, not needed for precise graphics representation is now present for other numeric problems). Certain scientific applications that have been ported to GPGPUs have shown impressive speed-up figures which previously needed of powerful supercomputers to be achieved. This has been in part possible thanks to the new APIs given by manufacturers to software developers as the nVIDIA^{TM}'s CUDA^{TM}. Next is explained the CUDA^{TM}architecture present on nVIDIA^{TM}GPUs used for this work (see Fig. 3).

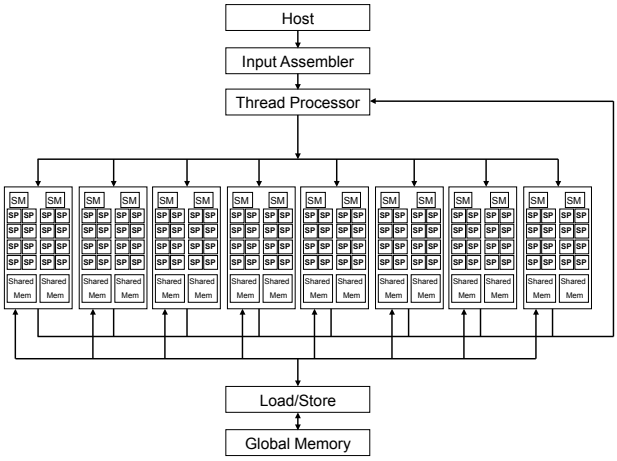


Figure 3: Schematic architecture for a CUDA device. Note memory hierarchy and the high number of processing units (SPs).

A CUDA^{TM}-enabled card is basically an array of processing units known as streaming multiprocessors (SM). Each SM contains 8 scalar processors (SP) or cores. SMs are grouped to form the GPU which is known as device. Specially important is the memory hierarchy present on the device as it has a great impact in performance and demands a great effort from the programmer's point of view. A good memory management by the programmer is essential to obtain good performance figures. Therefore, a profound knowledge of the GPU architecture is strongly advisable. Each SM memory hierarchy is composed of three different levels:

1. Registers: A set of 32 bits registers per SP.

2. Shared Memory: A memory space for reading/writing shared between all the SPs.

3. Read-only Memories: Two different areas (constant memory and texture memory) shared between all SPs from all SMs.

 ISBN 13: 978-84-613-5510-5

Also, a new level of memory hierarchy exists that is shared across all SMs, the device memory.

Apart from the memory hierarchy, the execution model is also central to ensure optimal performance. Each card has a ”Thread Processor” that co-ordinates where (which SPs and SMs) and when instructions are executed. The CUDA execution model is based on the simultaneous execution of a set of threads following a Single Instruction Multiple Threads (SIMT) eschema. SIMT means that different threads are doing the same job (set of instructions) over different pieces of data. Threads can be grouped into the so-called blocks. Also, blocks can be grouped into grids. This grouping strategy is used by the card to assign specific operands to each thread. Each block is assigned to a single SM that will process it until it is completely finished and new blocks can be assigned to the SM.

Model execution and memory hierarchy are interrelated. Each thread owns a private local memory arranged into 32 bits registers. There are a total of 8K (or more depending on the architecture) registers that must be distributed across the threads in a block. All the threads in a block share a portion of read/write, low latency and high bandwidth memory named Shared Memory. All threads in a grid can also share a read-only memory (constant memory and texture memory). Finally, the device memory is available for all threads of the grid.

4 Experiments and Results

To asses the quality and scalability of the parallel algorithm, a workstation was used. This computer had the following characteristics:

- The main host computer characteristics are as follows. Intel Xeon W3520 (45nm) quad-core processor, running at 2.66 Ghz. L1 cache of 256 Kb, L2 cache of 1 MB and L3 cache of 8 MB for each core. The total amount of RAM memory is of 6 GB DDR3 at 1066 Mhz and the Operating system is a Linux Debian distribution.

- The GPU is an nVIDIA Geforce GTX 285 with a compute capability 1.3. There are 30 multiprocessors accounting for a total of 240 scalar processors (or cores). The GPU memory was of 2 GB and the clock frequency was 1,48 Ghz.

Experiments were run both on the original sequential algorithm which just makes use of the CPU and on the CUDA-adapted algorithm that makes use of both the CPU and the GPU processors. For the sequential executions, run on the host machine, the GNU g++ (for C++ code) compiler was used. Optimization flags (-O2, -fomit-frame-pointer, -finline-functions, -funroll-loops) were enabled in order to obtain optimal execution times. The KerDenSOM algorithm present on the Xmipp software package [35] was used for the sequential experiments. Also, the parallel implementation was based on such Xmipp application. The resulting parallel application can be obtained from http://dali.ace.ual.es/∼vazquez/contact.php.

A Ribosome model [36] was used to generate an initial set of 30000 projection images randomly distributed around the specimen. The Ribosome images were classified, using the KerDenSOM algorithm, into different arrangements (groups of classes): 4x4, 8x4 and 16x4 classes. Each arrangement was tested for both rectangular and hexagonal topologies. Data was

 ISBN 13: 978-84-613-5510-5

Table 1: Average Speed-Up figures for the different experiments performed for problems of size 30K (projections) and different mask sizes. DAS stands for Deterministic Annealing Steps. ”Rectangular” and ”Hexagonal” denote the arrangement of the grid of neurons.

		Mask M0		**Mask M1**		**Mask M2**	
Arrang.	**DAS**	**Rect.**	**Hexag.**	**Rect.**	**Hexag.**	**Rect.**	**Hexag.**
4x4	5	6.27	5.13	5.82	5.21	5.48	5.53
	10	5.90	4.96	5.56	5.62	5.56	5.58
	15	5.53	5.01	5.62	5.75	5.52	5.49
8x4	5	8.23	5.53	5.23	5.85	5.79	6.03
	10	6.57	8.08	5.10	6.58	5.92	7.70
	15	6.19	5.68	4.95	6.18	5.59	5.55
16x4	5	4.77	5.95	5.36	5.87	5.44	5.95
	10	5.21	5.61	4.64	5.52	5.09	5.03
	15	5.43	6.46	5.12	5.27	5.10	5.37

masked using three different masks in order to simulate different problem sizes (projections size). Let us name such masks as M0, M1 and M2. M0 means no mask at all, the whole image is converted into an input vector (19600 elements). M1 represents a circular mask centered on the image with a radius equal to half the size of the projection (13205 elements) Finally, M2 represents another circular mask of radius equal to 64 pixels (input vector 12643 elements). Three different number of Deterministic Annealing Steps (DAS) were tested (5,10 and 15) for each mask size, topology and arrangement. For statistical purposes, three repetitions for each experiment where performed and average results were obtained.

Classification runtime varies depending not only on the problem size but also on the initial random values assigned to DVs. In order to stablish comparable metrics between different executions, both the sequential and the parallel algorithms used non-random initialization seeds. This allowed also to compare results between executions and ensure that the parallel implementation could reproduce those results obtained by the sequential version.

A total of 128 experiments were run (54 sequential and 54 parallel). Sequential executions took a total of 608.1 hours while the parallel experiments took a total of 106.2 hours. This means a raw 5,7x speed improvement.

As read from Table 1, the algorithm scales well for different problem sizes. No great speed-up differences can be observed for the different experiments. In general, the speedup factors that have been obtained fluctuate around 5.7x-6.0x. Different input data sizes (masks M0, M1 and M2) do not seem to be affecting the parallel algorithm scalability. It could be expected to obtain better results for bigger masks (resulting in bigger input vectors) as more computation would exist versus synchronisations in the code. Nevertheless this did not happen, which means that a good load balancing has been achieved. No appreciable differences exist for the Rectangular and Hexagonal topologies either. Once more, the number of classes for each topology do not seem to affect scalability of the algorithm. The same applies for the number of Deterministic Annealing Steps used.

 ISBN 13: 978-84-613-5510-5

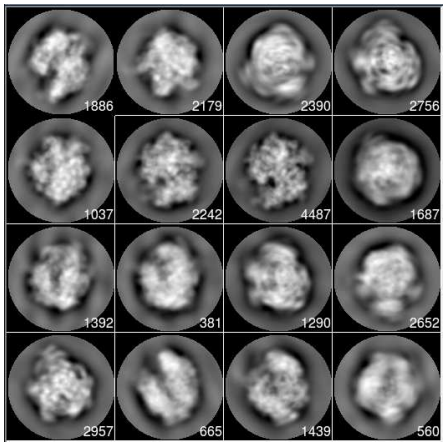


Figure 4: Resulting SOM for a 4x4 classification (16 classes). For each class, the number of input images mapped to it is shown.

5 Conclusions

In this work a parallel implementation of the KerDenSOM algorithm has been implemented and tested on a GPU system. Results show an average 5.7x speed gain for the GPU implementation in comparison with the sequential, original KerDenSOM implementation present in the Xmipp software package. Such speed-up, although discrete in principle, must be examined in relation to the system cost. The GPU used for the experiment had a cost of roughly 300$ which is by no means comparable to an equivalent, theoretical multiprocessor system that could reach such speed-ups (usually more than 6 processors).

The original Xmipp algorithm has shown to adapt well to the SIMD parallelization model imposed by the GPU programming model and has shown a good scalability behaviour under a variety of circumstances.

Acknowledgments

This work has been partially funded by grants from the Spanish MEC (TIN2008-01117), J. Andalucía (P06-TIC01426) and EU (LSHG-CT-2004-502828). During this work, J.R.B.C. was a fellow of the Spanish ”Juan de la Cierva” postdoctoral contracts program, co-financed by the European Social Fund.

References

[1] J. Frank, Three Dimensional Electron Microscopy of Macromolecular Assemblies, Oxford University Press, 2006.

[2] J. J. Fernández, C. O. S. Sorzano, R. Marabini, J. M. Carazo, Image processing and 3-D reconstruction in electron microscopy, IEEE Signal Process. Mag. 23(3) (2006) 84–94.

 ISBN 13: 978-84-613-5510-5

[3] N. Bonnet, Multivariate statistical methods for the analysis of microscope images series: Applications in materials science, Journal of Microscopy 190 (1998) 2–18.

[4] N. Bonnet, Artificial intelligence and pattern recognition techniques in microscope image processing and analysis, Advances in Imaging and Electron Physics 114 (2000) 1–77.

[5] M. V. Heel, J. Frank, Use of multivariate statistics in analyzing the images of biological macromolecules, Ultramicroscopy 6 (1981) 187–194.

[6] M. V. Heel, Multivariate statistical classification of noisy images (randomly oriented biological macromolecules), Ultramicroscopy 13 (1984) 165–184.

[7] J. Frank, M. V. Heel, Correspondence analysis of aligned images of biological particles, Journal of Molecular Biology 161 (1982) 134–137.

[8] J. Frank, J. Betraudiere, J. Carazo, A. Verschoor, T. Wagenknecht, Classification of images of biomolecular assemblies: A study of ribosomes and ribosomal subunits of escherichia coli, J. Microsc. 150 (1988) 99–15.

[9] J. Carazo, F. Rivera, E. Zapata, M. Radermacher, J. Frank, Fuzzy set based classification of em images of biological macromolecules with an application to ribosomal particles, Journal of Microscopy 157 (1990) 187–203.

[10] M. Herbin, N. Bonnet, P. Vautrot, A clustering method based on the estimation of the probability density function and on the skeleton by influence zones: Application to image processing, Pattern Recognition Letters 17 (1996) 1141–1150.

[11] R. Marabini, J. Carazo, Pattern recognition and classification of images of biological macromolecules using artificial neural networks, Biophys. J. 66 (1994) 1804–1814.

[12] A. Pascual-Montano, L. Donate, M. Valle, M. Bárcena, R. Pascual-Marqui, J. Carazo, A novel neural network technique for analysis and classification of em single-particle images, Journal of Structural Biology 133 (2001) 233–245.

[13] S. Scheres, M. Valle, R. Nuñez, C. Sorzano, R. Marabini, G. Herman, J. Carazo, Maximum-likelihood multi-reference refinement for electron microscopy images, Journal of Molecular Biology 348 (2005) 139–149.

[14] S. Scheres, H. Gao, M. Valle, G. H. P. E. J. F. J. Carazo, Disentangling conformational states of macromolecules in 3d-em through likelihood optimization, Nature Methods 27–29 (2007) 4.

[15] T. Kohonen, Self-organizing maps, 2nd.Ed., Springer-Verlag, Berlin, 1997.

[16] M. Gómez-Lorenzo, M. Valle, J. Frank, C. Gruss, C. Sorzano, X. Chen, L. Donate, J. Carazo, Large t antigen on the simian virus 40 origin of replication: a 3d snapshot prior to dna replication, The EMBO Journal 22 (2003) 6205–6213.

 ISBN 13: 978-84-613-5510-5

[17] F. Gubellini, F. Francia, J. Busselez, G. Venturoli, D. Lévy, Functional and structural analysis of the photosynthetic apparatus of rhodobacter veldkampii, Biochemistry 45(35) (2006) 10512–10520.

[18] A. Pascual-Montano, K. Taylor, H. Winkler, R. Pascual-Marqui, J. Carazo, Quantitative self-organizing maps for clustering electron tomograms, Journal of Structural Biology 138 (2002) 114–122.

[19] A. Al-Amoudi, D. Díez, M. Betts, A. Frangakis, The molecular architecture of cadherins in native epidermal desmosomes, Nature 450(7171) (2007) 832–837.

[20] J. Fernández, J. Carazo, Analysis of structural variability within two-dimensional biological crystals by a combination of patch averaging techniques and self organizing maps, Ultramicroscopy 65 (1996) 81–93.

[21] J. R. Bilbao-Castro, J. M. Carazo, I. García, J. J. Fernández, Parallelization of reconstruction algorithms in three-dimensional electron microscopy, Appl. Math. Model. 30 (2006) 688–701.

[22] J. R. Bilbao-Castro, A. Merino, I. García, J. M. Carazo, J. J. Fernández, Parameter optimization in 3d reconstruction on a large scale grid, Parallel Computing 33 (2007) 250–263.

[23] J. R. Bilbao-Castro, R. Marabini, C. O. S. Sorzano, I. García, J. M. Carazo, J. J. Fernández, Exploiting desktop supercomputing for three-dimensional electron microscopy reconstructions using art with blobs, Journal of Structural Biology 265 (2009) 19–26.

[24] J. J. Fernández, High performance computing in structural determination by electron cryomicroscopy, J. Struct. Biol. 164 (2008) 1–6.

[25] B. Wilkinson, M. Allen, Parallel Programming (2nd. ed), Prentice Hall, 2004.

[26] W. Gropp, E. Lusk, A. Skjellum, Using MPI Portable Parallel Programming with the Message-Passing Interface, MIT Press, 1994.

[27] B. Chapman, G. Jost, R. van der Pas, D. J. Kuck, Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation), 2007.

[28] D. R. Butenhof, Programming with POSIX(R) Threads, Addison-Wesley Professional, 1997.

[29] V. Neagoe, A. Ropot, Concurrent self-organizing maps for pattern classification, in: Proceedings of the First IEEE International Conference on Cognitive Informatics, 2002, pp. 304–312.

[30] P. Ozdzynski, A. Lin, M. Liljeholm, B. Jackson, A parallel general implementation of kohonen's self-organizing map algorithm: performance and scalability, Neurocomputing 44(46) (2002) 567–571.

 ISBN 13: 978-84-613-5510-5

[31] M. Pharr, R. Fernando, GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation, Addison-Wesley Professional, 2005.

[32] CUDA architecture. introduction and overview, NVIDIA corporation .

[33] CUDA programming guide, NVIDIA corporation .

[34] J. Nickolls, I. Buck, M. Garland, K. Skadron, Scalable parallel programming with CUDA, ACM Queue 6 (2008) 40–53.

[35] C. O. S. Sorzano, R. Marabini, J. Velázquez-Muriel, J. R. Bilbao-Castro, S. H. Scheres, J. M. Carazo, A. Pascual-Montano, Xmipp: a new generation of an open-source image processing package for electron microscopy, J. Struct. Biol. 148 (2004) 194–204.

[36] W. T. Baxter, R. A. Grassucci, H. Gao, J. Frank, Determination of signal-to-noise ratios and spectral SNRs in cryo-EM low-dose imaging of molecules, Journal of Structural Biology 166(2) (2009) 126–132.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

An ODE solver preserving fixed points and their stability

J. Vigo-Aguiar[1] **and Higinio Ramos**[1]

[1] *Department of Applied Mathematics, University of Salamanca*

emails: jvigo@usal.es, higra@usal.es

Abstract

In this article we provide a predictor-corrector one-step method for numerically solving first-order differential initial-value problems with two fixed-points. The method preserves the stability behaviour of the fixed points which results in an efficient integrator for this kind of problems. Some numerical examples are provided to show the good performance of the method.

1 Introduction

For the general scalar initial-value problem (IVP)

$$y' = f(x, y)\,, \quad y(x_0) = y_0 \tag{1}$$

with $y, f(x, y) \in \mathbb{R}$, and $x \in [x_0, x_N]$, an interval on the real line, a large number of algorithms have been developed to handle with it, since the Runge-Kutta or multistep methods to specific procedures for dealing with particular characteristics.

Recently, particularly from the work of Mickens [1], [2], the application of non-standard finite-difference methods has being increasing for numerically solving the problem in (1). Their use is mainly based on the fact that they are effective in conserving certain qualitative properties of the differential equation such as the preservation of fixed points, the positivity, or the monotonicity of the solutions. Examples of such schemes can be found in Refs. [1], [3], [4]. These discretizations with zero local truncation errors reflect exactly the dynamics of the differential equations. A well-known of such procedures concerns to the logistic equation

$$y' = y(1 - y) \tag{2}$$

for which an exact scheme is given by [1]

$$\frac{y_{n+1} - y_n}{1 - e^{-h}} = y_{n+1}(1 - y_n)\,, \tag{3}$$

 ISBN 13: 978-84-613-5510-5

which may also be rewritten as [4]

$$\frac{y_{n+1}-y_n}{e^h-1} = y_n(1-y_{n+1})\,. \tag{4}$$

Although this example, in general it is not clear how to find an exact scheme for a given IVP. This is not the case if the analytic exact solution is known, from which it can be found easily an exact numerical scheme. Consider the equation in (2) together with an initial value, that is,

$$y' = y(1-y)\,, \qquad y(x_n) = y_n\,. \tag{5}$$

The exact solution for this problem is

$$y(x) = \frac{e^x y_n}{(e^x - e^{x_n})\,y_n + e^{x_n}}$$

from which it is readily deduced the numerical method

$$y_{n+1} = \frac{e^h y_n}{1 + (e^h-1)\,y_n} \tag{6}$$

which is exact for the logistic initial-value problem. This scheme is the same as those in (3) or in (4), but we have it expressed in three different ways. This procedure could be applied to any other IVP for which an explicit exact solution is known.

In this paper we propose a numerical scheme for solving first-order IVPs having two real fixed points. This scheme in particular results to be exact (in the absence of rounding errors) for the problem in (5) or for any IVP whose differential equation is of the form $y' = (y-v_1)(y-v_2)$.

2 Description of the problem

We consider a particular kind of the scalar IVP in (1) with two fixed or equilibrium points, given by

$$y' = (y-v_1)(y-v_2)g(y)\,, \quad y(x_n) = y_n \tag{7}$$

where $v_1 < v_2 \in \mathbb{R}$ and $g(y) \neq 0$ is a bounded real-valued function with continuous derivatives.

Without loss of generality we may consider that the equilibrium points in the above problem are located at $y = 0$ and $y = 1$. To see that, consider the linear transformation given by $y = v_1 + w\,(v_2 - v_1)$. After applying this transformation to the problem in (7) we get

$$w' = w(w-1)\bar{g}(w)\,, \quad w(x_n) = w_n \tag{8}$$

where $\bar{g}(w) \neq 0$ is given by $\bar{g}(w) = (v_2 - v_1)\,g\,(v_1 + w\,(v_2 - v_1))$ and $w_n = (y_n - v_1)/(v_2 - v_1)$. Henceforth we can take in (7) that $v_1 = 0$ and $v_2 = 1$, which results in the problem

$$y' = y(y-1)g(y)\,, \quad y(x_n) = y_n\,. \tag{9}$$

 ISBN 13: 978-84-613-5510-5

3 The finite difference scheme

The numerical scheme for solving the problem in (9) is based on the following proposition.

Proposition 3.1. *Assuming that $y_n \neq 0$, the solution of the problem in (9) may be expressed in the form*

$$\frac{y(x)-1}{y(x)} = \frac{y_n - 1}{y_n} \exp\left(I_n\right) \tag{10}$$

with

$$I_n = \int_{x_n}^{x} g(y(t))\, dt\,.$$

Proof. Taking derivatives on both sides in the above expression, it follows easily the differential equation in (9).

Note that the solution in (10) may be expressed explicitly as

$$y(x) = \frac{y_n}{y_n + (1-y_n)\exp\left(\int_{x_n}^{x} g(y(t))\, dt\right)}\,,$$

from which we readily obtain that $y(x_n) = y_n$. □

Taking $x = x_n + h = x_{n+1}$, where h is a fixed step-size, different numerical schemes may be obtained after approximating the integral I_n in (10). We have consider two one-step formulas:

- an explicit one, obtained using the approximation for the integral $I_n \simeq h\, g(y_n)$ which results in the formula

$$\frac{y_{n+1}-1}{y_{n+1}} = \frac{y_n - 1}{y_n} \exp\left(h\, g(y_n)\right) \tag{11}$$

- an implicit one, obtained using the trapezoidal rule for approximating the integral, $I_n \simeq h\left(g(y_n) + g(y_{n+1})\right)/2$, which results in the formula

$$\frac{y_{n+1}-1}{y_{n+1}} = \frac{y_n - 1}{y_n} \exp\left(\frac{h}{2}\left(g(y_n) + g(y_{n+1})\right)\right). \tag{12}$$

The above methods will be used in a predictor-corrector implementation using the explicit one as a predictor and the implicit one as the corrector. In that case the approximation for $y(x_{n+1})$ obtained with the predictor will be denoted by y^p_{n+1}, and the approximation obtained with the corrector by y^c_{n+1}. The proposed method reads

$$y^p_{n+1} = \frac{y^c_n}{y^c_n + (1-y^c_n)\exp\left(h\, g(y^c_n)\right)} \tag{13}$$

$$\xi_{n+1} = \xi_n \exp\left(\frac{h}{2}\left(g(y^c_n) + g(y^p_{n+1})\right)\right) \tag{14}$$

$$y^c_{n+1} = \frac{1}{1-\xi_{n+1}} \tag{15}$$

 ISBN 13: 978-84-613-5510-5

where the first of the formulas has been obtained from (11) and the other two from (12) setting $\xi_n = \dfrac{y_n^c - 1}{y_n^c}$.

Note that the equation in (2) is of the form in (9), with $g(y) = -1$. In this case the integral in (10) is solved exactly, $I_n = -h$ and thus the methods in (11) or in (12) are the same. In fact, the method is exact for this problem and is given by

$$y_{n+1} = \frac{e^h y_n}{1 + (e^h - 1)\, y_n}\,,$$

that is, the above method in (6).

Remark 3.1. *In (11) for the approximation of the integral I_n we have substituted the function $g(y)$ by the interpolating polynomial passing through $(x_n, g(y_n))$, while in (12) we have substituted $g(y)$ by the interpolating polynomial passing through the points $(x_n, g(y_n))$, $(x_{n+1}, g(y_{n+1}))$, as in the Adams methods. Obviously, we could have consider better approximations to the integral taking interpolating polynomials of higher degrees. But these formulas would have positive and negative coefficients, and could result in a bad performance of the formulas. If $g > 0$ (¡0) then the integral I_n should be $I_n > 0$ ($I_n < 0$), which is not guaranteed if the formulas have positive and negative coefficients.*

Remark 3.2. *We observe that the above scheme may also be used for non-autonomous differential problems of the form $y' = y(y-1)g(x,y)$.*

3.1 Local truncation error

The method in (11) may be written as

$$(y_{n+1} - 1)\, y_n - (y_n - 1)\, y_{n+1} \exp\left(h\, g(y_n)\right) = 0\,. \tag{16}$$

Substituting the approximate values y_n, y_{n+1} for the true values $y(x_n), y(x_n+h)$, having in mind the value of $g(y(x))$ obtained from (9), after expanding by means of the usual Taylor formula we obtain that the local truncation error for the formula in (16) is given by

$$LTE^p(y(x_n); h) = \frac{1}{2}\, y(x_n)(y(x_n) - 1) g'(y(x_n))\, h^2 + \mathcal{O}(h^3)\,, \tag{17}$$

where $g'(y(x_n)) = \dfrac{dg(y(x))}{dx}(x_n)$.

Similarly, the implicit method in (12) may be written as

$$(y_{n+1} - 1)\, y_n - (y_n - 1)\, y_{n+1} \exp\left(\frac{h}{2}\, \left(g(y_n) + g(y_{n+1})\right)\right) = 0\,. \tag{18}$$

Proceeding as before, the local truncation error for the formula in (18) results in

$$LTE^c(y(x_n); h) = \frac{-1}{12}\, y(x_n)(y(x_n) - 1) g''(y(x_n))\, h^3 + \mathcal{O}(h^4)\,,$$

 ISBN 13: 978-84-613-5510-5

where $g''(y(x_n)) = \dfrac{d^2 g(y(x))}{dx^2}(x_n)$.

The analysis of the local error for the method in (13-15) applied in $P(EC)E$ mode is a bit cumbersome since the local truncation error of the corrector will be polluted by that of the predictor. From the predictor method in (13), after expanding in Taylor series we have

$$\begin{aligned} y(x_n+h) &- \frac{y(x_n)}{y(x_n)+(1-y(x_n))\,\exp\left(h\,g(y(x_n))\right)} \\ &= \frac{1}{2}\,y(x_n)(y(x_n)-1)g'(y(x_n))\,h^2+\mathcal{O}(h^3)\,. \end{aligned} \tag{19}$$

Having in mind the localizing assumption, on subtracting the formulas in (13) and (19) we obtain that

$$y(x_n+h)-y^p_{n+1} = \frac{1}{2}\,y(x_n)(y(x_n)-1)g'(y(x_n))\,h^2+\mathcal{O}(h^3)\,. \tag{20}$$

where the principal term coincides with that of the local truncation error in (17), as expected.

From the corrector method in (14), setting $\xi(x) = \dfrac{y(x)-1}{y(x)}$, after expanding in Taylor series we have

$$\begin{aligned} \xi(x_n+h) &= \xi(x_n)\exp\left(\frac{h}{2}\,\left(g(y(x_n))+g(y(x_n+h))\right)\right) \\ &\quad +\frac{-1}{12}\,y(x_n)(y(x_n)-1)g''(y(x_n))\,h^3+\mathcal{O}(h^4)\,. \end{aligned} \tag{21}$$

Using the localizing assumption and the formula in (14) we obtain that

$$\begin{aligned} \xi(x_n+h)-\xi_{n+1} &= \xi(x_n)\,e^{\frac{h}{2}\,g(y(x_n))}\left[e^{\frac{h}{2}\,g(y(x_n+h))}-e^{\frac{h}{2}\,g(y^p_{n+1}))}\right] \\ &\quad +\frac{-1}{12}\,y(x_n)(y(x_n)-1)g''(y(x_n))\,h^3+\mathcal{O}(h^4)\,. \end{aligned}$$

Defining the function $F(y)=\exp\left(\frac{h}{2}\,g(y)\right)$, through the application of the Mean Value Theorem to the bracket in the above formula it results that

$$\begin{aligned} \xi(x_n+h)-\xi_{n+1} &= \xi(x_n)\,e^{\frac{h}{2}\,g(y(x_n))}\,\frac{\partial F}{\partial y}(\eta)\,\left(y(x_n+h)-y^p_{n+1}\right) \\ &\quad +\frac{-1}{12}\,y(x_n)(y(x_n)-1)g''(y(x_n))\,h^3+\mathcal{O}(h^4) \\ &= \xi(x_n)\,e^{\frac{h}{2}\,g(y(x_n))}\,e^{\frac{h}{2}\,g(\eta)}\,\frac{h}{2}\,\frac{dg}{dy}(\eta)\,\left(y(x_n+h)-y^p_{n+1}\right) \\ &\quad +\frac{-1}{12}\,y(x_n)(y(x_n)-1)g''(y(x_n))\,h^3+\mathcal{O}(h^4) \end{aligned}$$

 ISBN 13: 978-84-613-5510-5

where η is an intermediate point between $y(x_n+h)$ and y_{n+1}^p. Introducing (20) in the previous formula, and expanding in Taylor series the exponentials we get

$$\begin{aligned}\xi(x_n+h)-\xi_{n+1} &= \frac{1}{16}\,(y(x_n)-1)^2\,g'(y(x_n))\,\frac{dg}{dy}(\eta)\,h^3\\ &\quad -\frac{1}{12}\,y(x_n)(y(x_n)-1)g''(y(x_n))\,h^3+\mathcal{O}(h^4)\\ &= \left(\frac{1}{16}\,(y(x_n)-1)\,g'(y(x_n))\,\frac{dg}{dy}(\eta)-\frac{1}{12}\,y(x_n)g''(y(x_n))\right)\\ &\quad \times(y(x_n)-1)\,h^3+\mathcal{O}(h^4)\,.\end{aligned}$$

So, in the first correction the order of the $P(EC)E$ mode is that of the corrector. However, the expression of the principal terms of the local truncation errors are different. If we do a second or more corrections, the $P(EC)^mE$ mode (with $m\geq 2$) has the same order and its local truncation error has the same principal part as those of the corrector.

4 Particularization in case of a double fixed point

If the differential equation has a double fixed point the IVP in (7) becomes

$$y'=(y-v)^2\,g(y)\,,\quad y(x_n)=y_n \tag{22}$$

where $v\in\mathbb{R}$ and $g(y)\neq 0$. A procedure for numerically solving the above IVP may be obtained similarly as in previous sections.

As for the case before, without loss of generality we may consider that the equilibrium point in the above problem is located at $y=0$. To see that it is enough to consider the linear transformation given by $y=z+v$. After applying this transformation to the problem in (22) we get

$$z'=z^2\tilde{g}(z)\,,\quad z(x_n)=z_n \tag{23}$$

where $\tilde{g}(z)=g\,(v+z)\neq 0$ and $z_n=y_n-v$.

Henceforth we can take in (22) that $v=0$, and thus we might consider the problem

$$y'=y^2\,g(y)\,,\quad y(x_n)=y_n\,. \tag{24}$$

The numerical scheme for solving the problem in (24) is based on the following proposition.

Proposition 4.1. *The solution of the problem in (24) may be expressed in the form*

$$y(x)=\frac{y_n}{1-y_n\,I_n} \tag{25}$$

with

$$I_n=\int_{x_n}^{x} g(y(t))\,dt\,.$$

 ISBN 13: 978-84-613-5510-5

Proof. Taking derivatives on both sides in the above expression, it follows easily the differential equation in (24).

From the solution in (25) it is straightforward to get that $y(x_n) = y_n$. □

Taking $x = x_n + h = x_{n+1}$, where h is a fixed step-size, different numerical schemes may be obtained after approximating the integral I_n in (25). We have consider two one-step formulas:

- an explicit one, obtained using the approximation for the integral $I_n \simeq h\, g(y_n)$ which results in the formula

$$y_{n+1} = \frac{y_n}{1 - h\, y_n\, g(y_n)} \tag{26}$$

- an implicit one, obtained using the trapezoidal rule for approximating the integral, $I_n \simeq h\,(g(y_n) + g(y_{n+1}))\,/2$, which results in the formula

$$y_{n+1} = \frac{y_n}{1 - \frac{h}{2}\, y_n\, \left(g(y_n) + g(y_{n+1})\right)}\,. \tag{27}$$

We note that the method in (26) may be written as

$$y_{n+1} = \frac{y_n}{1 - y_n\, h\, g(y_n)} = \frac{y_n^2}{y_n - h\, y_n'} = y_n + \frac{h\, y_n\, y_n'}{y_n - h\, y_n'}$$

where we have used that $I_n \simeq h\, g(y_n)$ and in view of (24) $g(y_n) = y_n'/y_n^2$. This method has appeared in [**?**] or [6] as indicated for solving singular IVPs. The local truncation error for this scheme is given by

$$LTE(y(x); h) = \left(\frac{y''(x)}{2} - \frac{y'(x)^2}{y(x)}\right) h^2 + \mathcal{O}\left(h^3\right)\,.$$

The solution of the differential equation resulting from equating to zero the principal term of this LTE is $y(x) = c_2/(x + c_1)$, and eliminating the parameters between this equation and that of the derivative results that the method is exact for differential equations of the form $y'(x) = c_1\, y(x)^2$.

5 Elementary stability

The analysis of linear stability properties is done considering the autonomous differential equation

$$y' = y(y-1)g(y)\,, \tag{28}$$

under the assumption that $g(y) \neq 0$. For this equation numerical instabilities will occur if the linear stability properties of any of the two fixed points for the difference scheme differs from that of the differential equation. Linear stability analysis applied to the fixed points gives the following results [1]:

 ISBN 13: 978-84-613-5510-5

$$If\ g(y) < 0 \implies \begin{cases} y(x) = 0\ is\ linearly\ unstable \\ \\ y(x) = 1\ is\ linearly\ stable\,. \end{cases}$$

$$If\ g(y) > 0 \implies \begin{cases} y(x) = 0\ is\ linearly\ stable \\ \\ y(x) = 1\ is\ linearly\ unstable\,. \end{cases}$$

From (18) it follows immediately that the only two fixed points of the numerical scheme are $y_n = 0$ and $y_n = 1$, which are the same as for the differential equation in (28). According to the method in (13-15) the fixed point $y_n = y_n^c = 1$ corresponds to $\xi_n = 0$, which is the only fixed point of the following difference equation resulting from (14)

$$\xi_{n+1} = \xi_n \exp\left(\frac{h}{2}\left[g\left(\frac{1}{1-\xi_n}\right) + g\left(\frac{1}{1-\xi_n \exp\left(h\, g(1/(1-\xi_n))\right)}\right)\right]\right).$$

After linearizing we get the difference equation

$$\xi_{n+1} = e^{h\, g(1)}\, \xi_n$$

which has the solution

$$\xi_n = \xi_0 \left[e^{h\, g(1)}\right]^n$$

Thus, it follows that for $h > 0$, if $g(1) < 0$ it is $e^{h\, g(1)} < 0$. Therefore, the fixed point $y_n = 1$ is linearly stable for $h > 0$. On the contrary, if $g(1) > 0$ the fixed point $y_n = 1$ is linearly unstable for $h > 0$.

To see the behavior of the fixed point $y_n = y_n^c = 0$ it is better to express the method in (13-15) in the equivalent form

$$\begin{aligned} y_{n+1}^p &= \frac{y_n^c}{y_n^c + (1 - y_n^c)\exp\left(h\, g(y_n^c)\right)} \\ \\ \bar{\xi}_{n+1} &= \bar{\xi}_n \exp\left(-\frac{h}{2}\left(g(y_n^c) + g(y_{n+1}^p)\right)\right) \\ \\ y_{n+1}^c &= \frac{\bar{\xi}_{n+1}}{\bar{\xi}_{n+1} - 1} \end{aligned}$$

where $\bar{\xi} = \dfrac{y_n^c}{y_n^c - 1}$. Proceeding as before we get the linearized difference equation

$$\bar{\xi}_{n+1} = e^{-h\, g(0)}\, \bar{\xi}_n$$

which has the solution

$$\bar{\xi}_n = \bar{\xi}_0 \left[e^{-h\, g(0)}\right]^n .$$

Therefore, for $h > 0$, if $g(0) < 0$ the fixed point $y_n = 0$ is linearly unstable, and if $g(0) > 0$ it is linearly stable. We summarize the above results in the following

 ISBN 13: 978-84-613-5510-5

Theorem 5.1. *The finite-difference scheme in (13-15) has for $h > 0$ the same fixed-points as the differential equation in (28), with the same linear stability properties .*

It is worth to mention here that although the fixed points of the differential equation in (28) are conserved by the numerical scheme in (13-15), but this is not true for any numerical method. As an example, let us consider the Runge-Kutta method given by the Butcher tableau

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 0 & 1 \end{array}$$

Setting $k_1 = f(x_n, y_n)\,, k_2 = f(x_n + h, y_n + h\,k_1)$, this method may be rewritten as

$$y_{n+1} = y_n + h\,f\left(x_n + h, y_n + h\,f(x_n, y_n)\right)\,.$$

When it is applied for solving the problem in (28) with $g(y) = 1$, it results that the fixed points are the solutions of the equation $f(y + h\,f(y)) = 0$, where $f(y) = y(y-1)$. The fixed points are $0, 1, -1/h, 1 - 1/h$, so there are two spurious fixed points of the finite-difference scheme, which could cause the bad behavior of the method.

6 Numerical example

We consider the initial-value problem given by

$$y'(x) = \frac{y(x)^2 - 1}{y(x)^2}\,, \qquad y(0) = y_0 = \frac{99}{100} \tag{29}$$

for which an exact solution in implicit form is given by $G(x, y) = 0$ with

$$G(x, y) = y - x - y_0 + \frac{1}{2}\,\log\left(\left|\frac{(1-y)(y_0+1)}{(y+1)(1-y_0)}\right|\right)\,.$$

For $x \to \infty$ the solution converges to the fixed point $y = 0$, and so it has to cross the horizontal axis, resulting on a singularity for the derivative located at

$$x_s = \frac{1}{2}\left(\log\left(\left|\frac{1+y_0}{1-y_0}\right|\right) - 2y_0\right) \simeq 1.6566524123622457\,.$$

We have solved the problem on the interval $[0, 3.3]$ using a variable stepsize implementation. The strategy considered for changing the step size was that used on multistep codes, [7, 8, 9]: given a tolerance, TOL, the classical stepsize prediction derived from equating this tolerance to the norm of the local truncation error yields a new stepsize h_{new} given by

$$h_{new} \approx \nu h_{old}\left(\frac{TOL}{\delta_n}\right)^{1/(p+1)} \tag{30}$$

 ISBN 13: 978-84-613-5510-5

where p is the order of the method, δ_n is an estimate of the error at each step, and $0 < \nu < 1$ is a safety factor whose purpose is to avoid failed steps.

In predictor-corrector mode the difference between the predicted and corrected values can give an estimate of the error on the current step. The desired solution, $y(x_{n+1})$, can be written in terms of the computed solution with the corrector, and an estimate of the error, δ_n , that is,

$$y(x_{n+1}) = y^c_{n+1} + \mathcal{O}(h_n^3) \simeq y^c_{n+1} + \delta\,,$$

where h_n is the estimate of the stepsize. Using the predictor the solution can be written as

$$y(x_{n+1}) = y^p_{n+1} + \mathcal{O}(h_n^2) \simeq y^p_{n+1} + \frac{\delta}{h_n}\,.$$

Subtracting the above two equations gives

$$0 = y^c_{n+1} - y^p_{n+1} + \delta(1 - \frac{1}{h_n})\,,$$

from which we take as an estimate of the local error on each step

$$\delta \simeq h_n \left| \frac{y^c_{n+1} - y^p_{n+1}}{1 - h_n} \right|\,.$$

Moreover, some restrictions must be considered in order to avoid large fluctuations in stepsize. If $\delta > TOL$ than h_n is decreased by a factor of 2 and the calculations at the current step must be redone. On the other hand, if $\delta < 0.02\,TOL$ than h_n is increased by a factor of 2.

Table 1 shows the data with the method in this article using the above strategy. We have considered the number of steps, the maximum absolute error at the nodal points, the number of rejected steps consisting on doubling or halving it, and the CPU time.

Steps	*MaxErr*	Rejected steps	Time
203	5.1869×10^{-4}	83	0.016
380	3.2633×10^{-5}	57	0.031
773	9.9865×10^{-6}	35	0.062
1643	2.4212×10^{-6}	36	0.140
3514	5.3977×10^{-7}	34	0.328

Table 1: *Numerical results for the problem in (29).*

We have used the NDSolve command of the Mathematica program to solve the problem in (29). After 1984 steps and 0.485 seconds of CPU time a warning appeared

```
NDSolve::ndsz: At x == 1.656652411941456316766932212444752072 46‘24.,
step size is effectively zero; singularity or stiff system suspected.
```

 ISBN 13: 978-84-613-5510-5

indicating that it has not been able to go beyond the singularity at x_s.

Figure 1 shows the numerical solution after joining the points (x_j, y_j) for $TOL = 10^{-8}$. Figure 2 shows the stepsizes needed. We observe that as we approximate to the point x_s the stepsize is smaller and then successively grows. The initial step was taken to be $h_0 = 10^{-2}$.

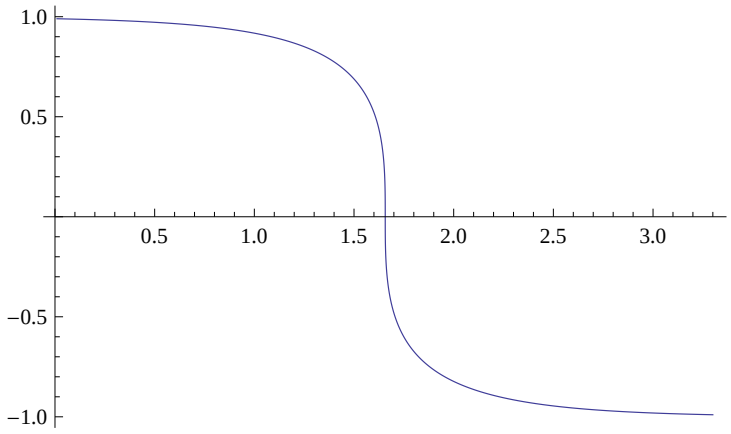


Figure 1: Numerical solution for the problem in (29).

7 Conclusions

A numerical method that preserves the stability of the fixed points is presented for solving initial-value problems with two fixed points. For this kind of problems the method performs adequately even in case of difficult problems whose solution exhibits a singularity or a highly oscillatory behaviour. For certain problems the method results to be an exact scheme where the inaccuracies are due only to round-off errors. Two are the objectives for future research: the formulation in a variable-step mode, and to extend the applicability of the method for solving systems of differential equations.

Acknowledgements

The authors wish to thank JCYL project SA050A08 and MICYT project MTM2008/05489 for financial support.

 ISBN 13: 978-84-613-5510-5

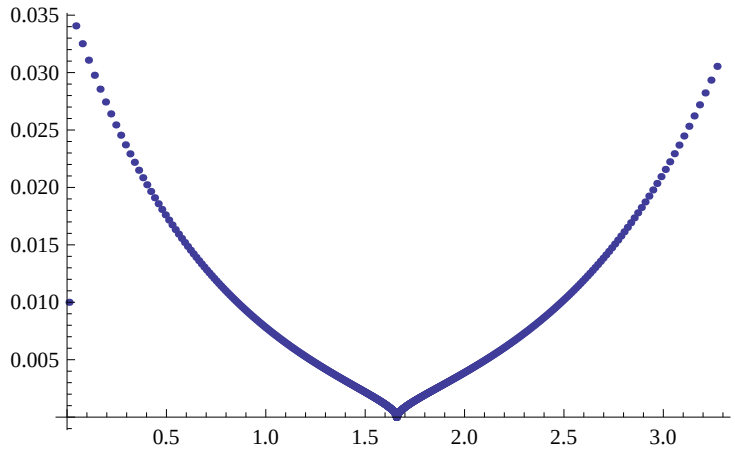


Figure 2: Stepsizes used for solving the problem in (29).

References

[1] R.E. Mickens, *Nonstandard Finite Difference Models of Differential Equations*, World Scientific, Singapore, 1994

[2] R.E. Mickens, *Applications of Nonstandard Finite Difference Schemes*, World Scientific, New Jersey, 2000

[3] K.C. Patidar, *On the use of nonstandard finite difference methods*, J. Differ. Equ. Appl. **11** (2005) 735–758.

[4] L.W. Roeger, R. Mickens, *Exact finite-difference schemes for first order differential equations having three distinct fixed-points*, J. Differ. Equ. Appl. **13** (2007) 1179–1185.

[5] S. O. Fatunla, *Numerical Methods for IVPs in ODEs*, Academic Press Inc., London, 1988.

[6] H. Ramos, *A non-standard explicit integration scheme for initial-value problems*, Applied Mathematics and Computation **189** (2007) 710–718.

[7] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley, England, 1991.

[8] M. Calvo, J.I. Montijano and L. Randez, *the change of step size in multistep codes*, Numer. Alg. **4** (1993) 283–304.

[9] L. F. Shampine and M. K. Gordon, *Computer solution of Ordinary Differential Equations. The initial Value Problem*, Freeman, San Francisco, CA, 1975.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

A Distributed Visual Client for Large-Scale Crowd Simulations

Guillermo Vigueras[1], Miguel Lozano[1], Juan M. Orduña[1] and Yiorgos Chrysanthou[2]

[1] *Departamento de Informática, University of Valencia. Spain*

[2] *Department of Computer Science, University of Cyprus. Cyprus*

emails: guillermo.vigueras@uv.es, miguel.lozano@uv.es, juan.orduna@uv.es, yiorgos@cs.ucy.ac.cy

Abstract

The visualization system of large-scale crowd simulations should scale up with both the number of visuals (views of the virtual world) and the number of agents displayed in each visual. Otherwise, we could have large scale crowd simulations where only a small percentage of the population is displayed. Several approaches have been proposed in order to efficiently render crowds of animated characters. However, these approaches either render crowds animated with simple behaviors or they can only support a few hundreds of user-driven entities. In this paper, we propose a distributed visualization system for large crowds of autonomous agents that allows the visualization of the crowd without adding significant overhead to the simulation servers. The proposed implementation can be hosted on dedicated computers different from the servers, and it takes advantage of the Graphics Processor Unit (GPU) capabilities. As a result, the performance evaluation shows that thousands of agents can be rendered without affecting the performance of the simulation servers. These results suggest that the design of the visual client allows to add multiple visuals for displaying large crowds.

Key words: distributed simulation, parallel rendering, performance evaluation

1 Introduction

Large scale crowd simulations are becoming essential tools for many virtual environment applications in education, training, and entertainment [16, 4]. In order to deal with the computational complexity of large scale simulations, different proposals have been made for achieving both very populated scenes [19] and scalable autonomous behaviors [9, 18]. However, the scalability of autonomous complex agents (crowd simulations) is still an open issue in spite of these efforts.

 ISBN 13: 978-84-613-5510-5

In previous works, we proposed a distributed system architecture that can simulate large crowds of autonomous agents at interactive rates [10, 20, 21], and it can take advantage of the inherent scalability of manycore computer architectures [22]. However, in order to make a truly scalable system for crowd simulations, the visualization system (the module responsible of rendering the images of the virtual world) should also be addressed. The visualization system should scale up with both the number of visuals (cameras focusing on the virtual world) and the number of agents displayed in each camera. Otherwise, we could have large scale crowd simulations where only a small percentage of the population could be rendered (displayed).

In this paper, we propose a distributed visualization system that allows the visualization of the virtual world without adding significant overhead to the simulation servers, regardless of both the number of visuals and the number of agents rendered by each visual. In order to achieve this goal, the visualization system consists of a visual client process (VCP) for each camera, and each VCP is hosted on a computer different from the ones hosting the simulation servers. In this way, the connection of the visual client does not significantly affects the performance of the simulation system. The proposed implementation migrates different rendering tasks of the VCP from the CPU to the Graphics Processor Unit (GPU) of the hosting computer, reducing the CPU workload of the visual client and increasing throughput. Also, we use skinned instancing for reducing the rendering workload. As a result, the performance evaluation shows that thousands of agents can be rendered without affecting the performance of the simulation servers. These results suggest that the design of the visual client allows to add multiple visuals for displaying large crowds.

The rest of the paper is organized as follows: section 2 shows some related work about visual clients for crowd simulations. Section 3 briefly describes the distributed architecture for crowd simulation that was previously proposed, and it shows the scalability problems arising when connecting a VCP to this kind of systems. Next, section 4 describes the proposed implementation for the distributed visual client, and section 5 shows the performance evaluation of the proposed implementation. Finally, section 6 shows some conclusions.

2 Related Work

From the graphics community, several approaches have been proposed in order to efficiently render crowds of animated characters. Image-based [5, 17] and Point-based [23] techniques obtain interactive frame rates when rendering crowded animated scenes by reducing the geometrical complexity of the 3D characters meshes. Other approaches use efficient parallel graphic techniques to provide interactive graphics performance for crowded scenes [14, 2]. Although these graphic-based approaches obtain good frame rates, they are not focused on providing scalable architectures. Other proposals [15] combine parallel architectures with efficient graphic techniques to simulate and to display thousands of individuals. In this case, authors use the Cell processor architecture without considering scalability issues.

 ISBN 13: 978-84-613-5510-5

From the distributed simulation arena, there have been several approaches oriented to handle multiplayer games [3, 11]. Other works use the HLA architecture [24] combining classical scene graphs with simulated federations to provide interactive graphic applications for military and entertainment purposes.

Although these approaches can provide interactive latencies and frame rates, required by multiplayer games, they usually can only support a few hundreds of user-driven entities within a simulation.

3 A Distributed System for Large-Scale Crowd Simulation

In previous works, we proposed an architecture that can simulate large crowds of autonomous agents at interactive rates [10, 20]. In that architecture, the crowd system is composed of many Client Computers, that host agents implemented as threads of a Client Process, and one Action Server (AS), executed in one computer, that is responsible for checking the actions (eg. collision detection) sent by agents [10]. In order to avoid server bottleneck, the simulation world was partitioned into subregions and each one assigned to one parallel AS [20]. A scheme of this architecture is shown in figure 1. This figure shows how the 2D virtual world occupied by agents (black dots) is partitioned into three subregions, and each one managed by one parallel AS (labeled in the figure as AS_x). Each AS is hosted by a different computer. Agents are execution threads of a Client Process (labeled in the figure as CP_x) that is hosted on one Client Computer. The computers hosting client and server processes are interconnected. Each AS process hosts a copy of the Semantic Database. However, each AS exclusively manages the part of the database representing its region. In order to guarantee the consistency of the actions near the border of the different regions (see $agent_k$ in figure 1), the ASs can collect information about the surrounding regions by querying the servers managing the adjacent regions. Additionally, the associated Clients are notified about the changes produced by the agents located near the adjacent regions by the ASs managing those regions.

The architecture shown in Figure 1 allows to simulate large crowds of autonomous agents providing a good scalability. However, it also needs a scalable visualization method in order to render the simulated crowd. The visualization system will be in charge of rendering the simulated world, starting from the information generated by the distributed servers. In order to provide scalability, the visualization system should be designed in a distributed fashion.

A feasible way of implementing a distributed visualization system could be the integration of a rendering module within each Action Server. In this way, each AS could visualize its own region of the virtual world. However, the computational workload resulting from adding a rendering module to each AS could result in a performance degradation of the whole simulation system [12]. Additionally, with this approach the number of cameras would be limited by the number of servers in the system. Instead, we have followed a different approach, where the visualization of the simulation is

 ISBN 13: 978-84-613-5510-5

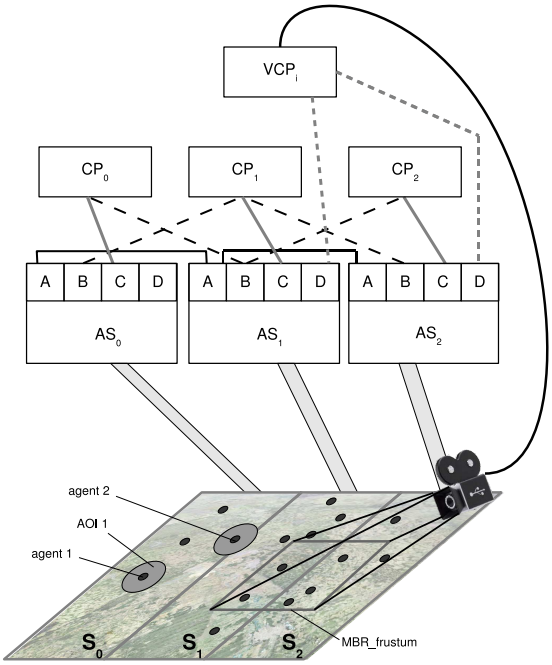


Figure 1: General scheme of the distributed simulation system with a Visual Client Process.

distributed among different processes, each one denoted as Visual Client Process (VCP). Each VCP manages one camera, and it is hosted in a dedicated computer different from the ones hosting either CPs or ASs. A VCP can be connected to several different ASs, depending on the area of the virtual world covered by the camera of the VCP. For example, in Figure 1 the VCP is connected to both AS_1 and AS_2, since the projection of the camera plane (denoted as MBR_Frustum) intersects the regions managed by these ASs.

In order to efficiently designing the rendering module of the VCP, the first step consists of measuring the workload that the information received from the ASs represents for a single VCP. The amount of information sent by the ASs depends on two factors: the number of simulated agents and the acting period of those agents (the period of time between two successive actions requested by an agent). Table 1 shows the percentage of CPU utilization in the computer hosting the VCP when increasing both the number of agents in the MBR_Frustum and the acting period. The results were obtained using up to four servers, each one managing 3000 agents (12000 agents in total for four servers). In these tests, the VCP were connected to the servers and all the agent requests received by the servers were sent to the VCP, i.e. the VCP received updates from 12000 agents when using four servers. Table 1 shows that the VCP workload exceeds the computational bandwidth of the hosting computer when 6000 agents (2 servers) are connected to the VCP, since the percentage of CPU utilization reaches 100%. Also, this table shows that the workload generated by the VCP is inversely

 ISBN 13: 978-84-613-5510-5

related to the acting period, as it could be expected.

Agents	Acting Period (ms.)			
	100	400	700	1000
3000	69	67	65	55
6000	100	100	100	100
9000	100	100	100	100
12000	100	100	100	100

Table 1: CPU utilization (%) of the computer hosting the VCP.

Agents	Acting Period (ms.)			
	100	400	700	1000
3000	0	0	0	0
6000	76207	44729	0	0
9000	205545	238244	61972	0
12000	433716	391644	157606	19137

Table 2: Visualization requests not processed by the VCP.

In order to find how the saturation of the CPU affects the performance of the VCP, we have measured the difference between the number of operations sent by the ASs and the number of operations actually processed by the VCP. Table 2 shows these data, and it shows that for 3000 agents there are no lost operations (the CPU hosting the VCP is not saturated, reaching a maximum CPU utilization of 69% for the lowest acting period (100 ms). However, from 6000 agents up, the VCP does not process all the requests, depending on the agent period. It is worth mention that for the case of 6000 agents the VCP is capable of process all the operations received when using an acting period of 700 and 1000 ms. However, when simulating 12000 agents the VCP cannot process all the operations, regardless of the agent period considered. These results show that the implementation of the VCP should reduce the CPU utilization associated to the graphic tasks as much as possible, in order to increase the VCP throughput.

4 Distributed Rendering of Crowd Simulations

In this section we describe the proposed implementation of the Visual Client Process. The proposed approach for rendering crowd simulation is based on having each VCP connected with one or more Action Servers. Therefore, the first step is to modify the AS scheme proposed in [20] in such a way that the information about the agent actions is also sent to the VCPs. Then, an implementation of the VCP according to that scheme should be developed.

4.1 Modifications to the Action Server

Each AS process [20] contains three basic elements: the Interface module, the Crowd AS Control (CASC) module and the Semantic Data Base (SDB). The Interface module is in charge of communicating the AS with other ASs and CPs. The main module is the Crowd AS Control module, which is responsible for executing the crowd actions. This module contains a configurable number of threads for executing actions (action execution threads). For an action execution thread (AE thread), all messages sent to or received from other ASs and CPs are exchanged asynchronously. This means that the AE threads only may have to wait when accessing the semantic database.

 ISBN 13: 978-84-613-5510-5

The changes required in the Action Server for connecting the VCP exclusively affects the Interface Module. Figure 2 shows the general scheme of an Action Server modified for accepting VCP connections. Two I/O threads are created for each VCP connected to an AS. One of the I/O threads receives, through a socket, the MBR_frustum updates sent by the VCP. It must be noticed that the updates received from the VCP are not passed to the Crowd AS Control Module. In this way, the workload added by each VCP connected to the AS is reduced. The other I/O thread is in charge of forwarding to the VCP the AS replies to agents requests. This thread uses the MBR_frustum updates received from the VCP to filter forwarded replies. It simply checks whether an agent falls within the MBR_frustum or not (see Figure 1). In this way, a VCP can visualize less agents than those that form the crowd simulated by the distributed system.

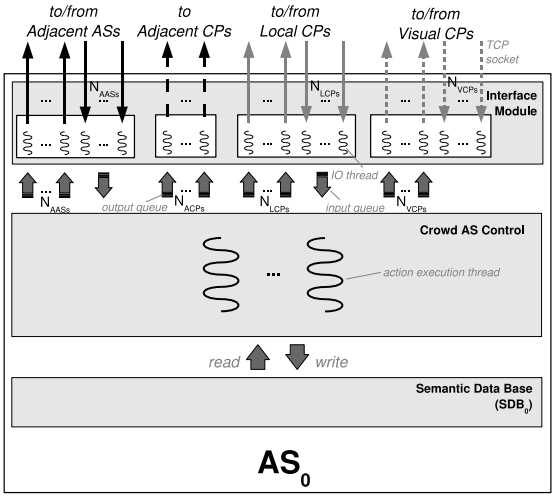


Figure 2: Scheme of an Action Server with VCP connections.

4.2 Implementation of the Visual Client Process

The VCP is mainly composed by two modules: the Interface Module and the Graphic Application Module. Figure 3 shows an schematic view of this process. The Interface Module is in charge of sending updates of the MBR_frustum to the ASs. Also , this module should receive agents updates and pass them to the Graphic Application Module. The Graphic Application Module is in charge of performing the graphic tasks. Some of these tasks are executed on the CPU and other ones are executed on the GPU.

As shown in section 3, it is crucial to reduce as much as possible the percentage of CPU utilization in the computer hosting the VCP in order to minimize the number of visualization updates not processed by the VCP. In order to achieve this goal, the frustum culling and the determination of the Level Of Detail (LOD) are performed by the GPU, like other approaches do [13].

Additionally, we use *Instancing* to efficiently render crowded scenes [1]. When

 ISBN 13: 978-84-613-5510-5

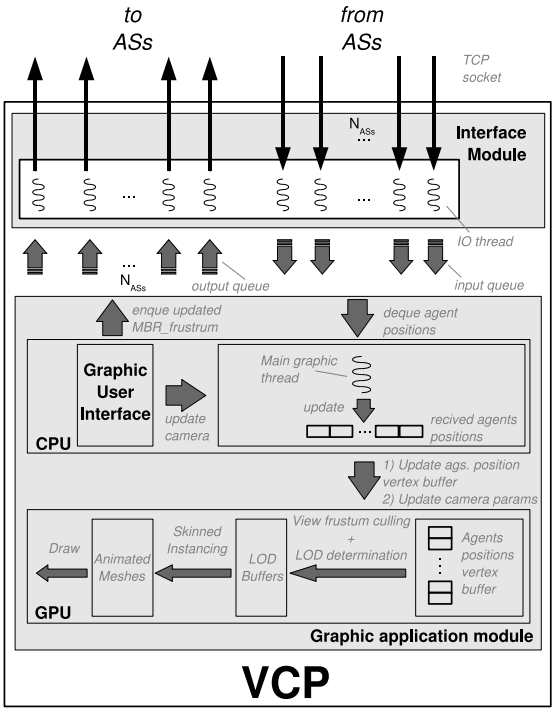


Figure 3: Scheme of the design of the Visual Client Process.

rendering a 3D mesh, typical graphic APIs issue a Draw call to the graphic pipeline. Each API call has associated a fixed-cost overhead for processing a primitive, regardless of the size. Due to this API call overhead, the performance of a graphic application (in terms of Frames Per Second or FPS) is CPU-bounded instead of GPU-bounded. Instancing consists in grouping characters that share a 3D mesh into a batch, generating only one Draw call. However, when using Instancing characters have to share both the mesh and the pose at a given time. Since a crowd is composed of autonomous agents, each one has a different pose at a given time. As a result, the use of Instancing provides non-realistic movements. In order to solve this problem, we have implemented the VCP using the Skinned Instancing technique [7]. This technique takes advantage of the new DirectX graphic API, that allows to perform Instancing but generating an identifier for each instance of the 3D mesh. In this way, each instance can keep its own properties (i.e. translation, rotation and scale) and the GPU skinning method [2] can be independently applied to animate the character meshes.

Figure 3 shows a scheme of the VCP and the different tasks of each part of the Graphic Module. The CPU part of this Module acts as an interface with the user, updating the camera position according to the MBR_frustum. All the MBR_frustum updates are passed to the Interface Module in order to send them to the ASs. Additionally, the CPU processes the agents updates received by the Interface Module and formates this data in order to properly perform the skinned instancing on the GPU.

 ISBN 13: 978-84-613-5510-5

Once the GPU part of the Graphic Module receives agents updates through a vertex buffer, it firstly perform the view frustum culling and the LOD determination. As a result, agents that have passed the culling test are grouped by LOD and one GPU buffer is used to store agents sharing the same LOD. Since we are using three LODs, three buffers are the input of the Skinned Instanced step. Once the instanced meshes are properly animated, they are rendered.

5 Performance Evaluation

In this section, we show the performance evaluation of the proposed distributed visual client. Like other distributed systems, the most important performance measurements in these systems are latency and throughput [6]. Since we are focusing on the visual client performance and how the integration of a VCP affects to the overall system (crowd simulation) performance, we have performed simulations with different servers and we have measured both the response time of the servers and frames rate obtained in the VCP. In order to define an acceptable behavior for the system, we have considered 250 ms. as the maximum acceptable value for the response time, since it is considered as the limit for providing realistic effects to users in DVEs [8]. For the VCP, we have considered about 30 frames per second as an acceptable frame rate.

As a simulation platform, we have used a cluster of computers based on AMD Opteron (2 processors @ 1.56 GHz) with 3.84GB of RAM, executing Linux 2.6.9-1 operating system. The interconnection network in the cluster was a Gigabit Ethernet network. The machine used for the visual client was based on Intel Core 2 Duo @ 2.4 GHz with 4GB of RAM, executing Windows Vista operating system. The graphic card within the VCP was a NVIDIA GForce 9600M GT. We have performed the distributed simulations using up to four computers of the cluster for hosting an AS each. For that case, we used eight cluster nodes (four of them for hosting the servers and four of them for hosting four clients). Using this platform, we have simulated up to twelve thousand agents. The VCP was connected to the servers through the cluster network and the number of agent updates received by the VCP was increased in order to study the VCP performance.

For comparison purposes, we have implemented three different VCPs. The first one (denoted as SDC, for "separate draw calls") does not use Instancing to render the crowd and performs the LOD computation on the CPU. The second version (denoted as iCPU) uses Skinned Instancing and it computes the LODs on the CPU. Finally, and the third kind of VCP (denoted as iGPU) uses Skinned Instancing and performs the LOD calculations on the GPU. Figure 4 shows the CPU load of the different versions of the visual client when the number of rendered agents is increased. In this figure, the X-axis shows the number of rendered characters and the Y-axis shows the percentage of CPU utilization when executing the VCP. This figure shows very much higher CPU utilization load for the SDC version than the one required for the iCPU and iGPU versions. This version of the VCP leads the CPU close to saturation levels for 5000 agents due to the absence of Instancing, while the other two versions only require

 ISBN 13: 978-84-613-5510-5

less than a 50% of CPU utilization. This figure also shows that the iGPU version 3 reduces the percentage of CPU utilization around a 10% respect to the iCPU version. Therefore, the version that provides the best results in terms of visual throughput (the maximum number of characters that can be rendered) is the iGPU implementation of the VCP.

Figure 5 shows the performance of the different versions of the VCP in terms of FPS. In this figure, the X-axis shows the number of rendered agents and the Y-axis shows the frame rate achieved by the VCP. It can be seen that both the iCPU and the iGPU versions clearly outperform the SDC version. The reason for that behavior is that the SDC version saturates the CPU. However, there is no a significant difference between the frame rates obtained by the iCPU and the iGPU versions of the VCP because both of them use Instancing. Therefore, they generate the same number of draw calls. Since the iGPU provides the best throughput and a similar frame rate than the iCPU version, we have used this version as the VCP implementation for the rest of the evaluation.

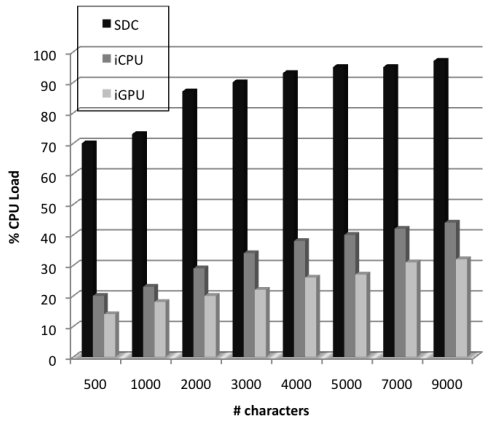


Figure 4: CPU utilization for different implementations of the VCP.

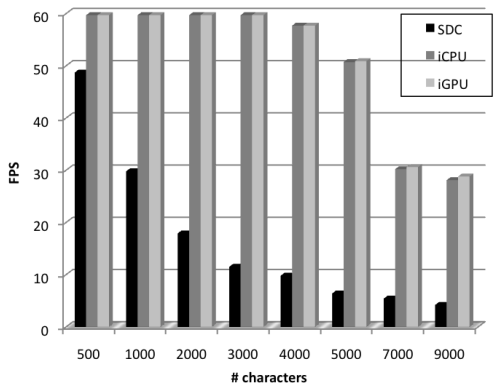


Figure 5: Frame rates for different implementations of the VCP.

Once the version of the VCP that provides the best performance has been selected, the next step has been to study the performance of that VCP when it is connected to the crowd simulation system. Table 3 shows the performance measurements for the VCP when the number of agents updates received from the ASs is increased. It can be seen that when rendering 2000 agents at a good frame rate the CPU utilization is around 87% and no agents updates are lost (all the requests are processed by the VCP). As the number of agent rendering requests increases, so does the CPU utilization, causing a frame rate decrease on the VCP but still above acceptable values (higher than 30 FPS). However, from 5000 agents up to 6000 agents the CPU reaches saturation, resulting in agents updates that are not processed and causing the frame rate to fall below 30 FPS.

Finally, we have studied the performance of the simulation servers when the VCP is connected, in order to show that the latter one does not have a significant effect on the servers performance. Figures 6 and 7 show the performance of the simulation system

 ISBN 13: 978-84-613-5510-5

	Characters rendered				
	2000	3000	4000	5000	6000
% CPU	86,7	89,3	94	95,7	98,8
FPS	55,4	41,6	34,8	28,4	25,4
Lost ops.	0	0	0	34134	57348

Table 3: Performance measurements for the VCP when increasing the number of agents updates received.

when the VCP is connected to a simulation system consisting of four servers and the number of agents updates sent to the VCP is increased. In these figures, the X-axis shows the number of agents updates that the VCP receives. The Y-axis in figure 6 shows the CPU utilization in the system servers, while the Y-axis in figure 7 shows the response time (in ms.) provided to agents (that are executed as threads of the client processes).

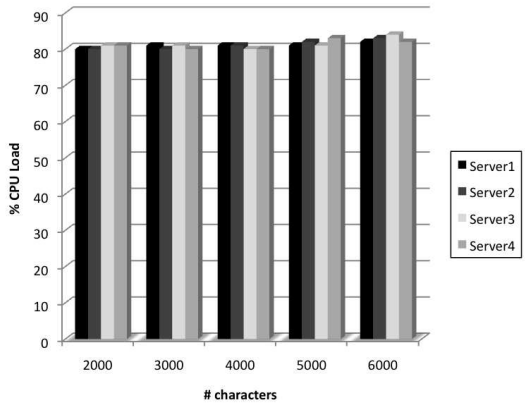


Figure 6: CPU utilization of simulation servers with a connected VCP.

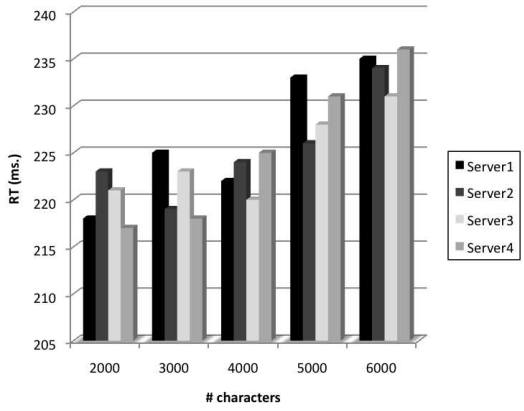


Figure 7: Response time provided by simulation servers with a connected VCP.

Figure 6 shows that the CPU utilization in the system servers remains almost constant as the number of agents rendered by the VCP increases. It starts from a CPU utilization of 80% for 2000 agents and for 6000 agents it does not reach 90%. Since the VCP does not lead the system servers to saturation, figure 7 shows that they provide an acceptable response time (shorter than 250 ms.) up to 6000 agents. These results show that the VCP does no have significant effects on the performance of the system servers.

 ISBN 13: 978-84-613-5510-5

6 Conclusions and Future Work

In this paper, we have proposed a distributed visualization system that allows the visualization of the virtual world without adding significant overhead to the simulation servers. The proposed implementation can be hosted on dedicated computers different from the servers, and it migrates different rendering tasks of the VCP from the CPU to the GPU of the hosting computer. Also, we use skinned instancing for reducing the rendering workload. As a result, the performance evaluation shows that thousands of agents can be rendered without affecting the performance of the simulation servers. These results suggest that the design of the visual client allows to add multiple visuals for displaying large crowds.

As a future work to be done, we plan to study the scalability of the proposed visualization system when a lot of visuals of the same simulation are needed.

Acknowledgements

This work has been jointly supported by the Spanish MICINN, the European Commission FEDER funds, and the University of Valencia under grants Consolider-Ingenio CSD2006-00046 and TIN2009-14475-C04-04.

References

[1] T. Akenine-Möller, E. Haines, and N. Hoffman. *Real-Time Rendering 3rd Edition*. A. K. Peters, Ltd., Natick, MA, USA, 2008.

[2] G. Ashraf and J. Zhou. Hardware accelerated skin deformation for animated crowds. In *13th International Multimedia Modeling Conference, MMM*, pages 226–237. Springer, 2007.

[3] A. Bharambe, J. Pang, and S. Seshan. Colyseus: a distributed architecture for online multiplayer games. In *NSDI'06: Proceedings of the 3rd conference on Networked Systems Design & Implementation*, pages 12–12, Berkeley, CA, USA, 2006. USENIX Association.

[4] D. Chen, G. K. Theodoropoulos, S. J. Turner, W. Cai, R. Minson, and Y. Zhang. Large scale agent-based simulation on the grid. *Future Generation Computer Systems*, 24(7):658 – 671, 2008.

[5] S. Dobbyn, J. Hamill, K. O'Conor, and C. O'Sullivan. Geopostors: a real-time geometry/impostor crowd rendering system. *ACM Trans. Graph.*, 24(3):933–933, 2005.

[6] J. Duato, S. Yalamanchili, and L. Ni. *Interconnection Networks: An Engineering Approach*. IEEE Computer Society Press, 1997.

[7] B. Dudash. Skinned instancing. Technical report, NVIDIA Corp., February 2007.

[8] T. Henderson and S. Bhatti. Networked games: a qos-sensitive application for qos-insensitive users? In *Proceedings of the ACM SIGCOMM 2003*, pages 141–147. ACM Press / ACM SIGCOMM, 2003.

[9] A. Iglesias and F. Luengo. New goal selection scheme for behavioral animation of intelligent virtual agents. *IEICE Transactions on Information and Systems, Special Issue on 'CyberWorlds'*, E88-D(5):865–871, 2005.

[10] M. Lozano, P. Morillo, J. M. Orduña, V. Cavero, and G. Vigueras. A new system architecture for crowd simulation. *J. Netw. Comput. Appl.*, 32(2):474–482, 2009.

 ISBN 13: 978-84-613-5510-5

[11] A. V. Martinez, H. R. Orozco, F. F. R. Corchado, and M. Siller. A peer-to-peer architecture for real-time distributed visualization of 3d collaborative virtual environments. In *13th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications*, pages 251–254, 2009.

[12] P. Morillo, J. M. Orduña, M. Fernández, and J. Duato. Improving the performance of distributed virtual environment systems. *IEEE Transactions on Parallel and Distributed Systems*, 16(7):637–649, 2005.

[13] H. Park and J. Han. Fast rendering of large crowds using GPU. In *ICEC '08: Proceedings of the 7th International Conference on Entertainment Computing*, pages 197–202, Berlin, Heidelberg, 2009. Springer-Verlag.

[14] M. Pharr and R. Fernando. *GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation (Gpu Gems)*. Addison-Wesley Professional, 2005.

[15] C. Reynolds. Big fast crowds on ps3. In *sandbox '06: Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames*, pages 113–121, New York, NY, USA, 2006. ACM.

[16] A. Shendarkar, K. Vasudevan, S. Lee, and Y.-J. Son. Crowd simulation for emergency response using bdi agent based on virtual reality. In *WSC '06: Proceedings of the 38th conference on Winter simulation*, pages 545–553. Winter Simulation Conference, 2006.

[17] F. Tecchia, C. Loscos, and Y. Chrysathou. Visualizing crowds in real time. *Computer Graphics Forum*, 21, 2002.

[18] H. Tianfield, J. Tian, and X. Yao. On the architectures of complex multi-agent systems. In *Proc. of the IEEE/WIC International Conference on Web Intelligence / Intelligent Agent Technology,*, pages 195–206. IEEE Press, 2003.

[19] A. Treuille, S. Cooper, and Z. Popovic. Continuum crowds. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 1160–1168. ACM, 2006.

[20] G. Vigueras, M. Lozano, C. Pérez, and J. Orduña. A scalable architecture for crowd simulation: Implementing a parallel action server. In *Proceedings of International Conference on Parallel Processing (ICPP)*, pages 430–437, Los alamitos, CA, USA, 2008. IEEE Computer Society.

[21] G. Vigueras, M. C. Lozano, J. Orduña, and F. Grimaldo. A comparative study of partitioning methods for crowd simulations. *Appl. Soft Comput.*, 10(1):225–235, 2010.

[22] G. Vigueras, J. M. Orduña, and M. Lozano. *Advances in Practical Applications of Agents and Multiagent Systems*, chapter A GPU-Based Multi-agent System for Real-Time Simulations, pages 15–24. Springer Berlin / Heidelberg, 2010.

[23] M. Wand and W. Straßer. Multi-resolution rendering of complex animated scenes. *Comput. Graph. Forum*, 21(3), 2002.

[24] H. Xiong, Z. Wang, X. Jiang, and J. Shi. Building high performance DVR via HLA, scene graph and parallel rendering. In *VRST '07: Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, pages 141–144, New York, NY, USA, 2007. ACM.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2010 27–30 June 2010.

Transformed exponential order for neutral functional differential equations with infinite delay

Víctor M. Villarragut[1] **and Rafael Obaya**[1]

[1] *Departamento de Matemática Aplicada, Escuela de Ingenierías Industriales, Universidad de Valladolid*

emails: vicmun@wmatem.eis.uva.es, rafoba@wmatem.eis.uva.es

Abstract

This work is devoted to the establishment of the 1-covering property of the omega-limit sets of a family of monotone non-autonomous neutral functional differential equations with infinite delay and autonomous D-operator.

Key words: NFDEs, infinite delay, exponential order
MSC 2000: 37B55, 34K40, 34K14

1 Introduction

The aim of this work is to study the long term behavior of the solutions of a family of monotone non-autonomous neutral functional differential equations with infinite delay and autonomous D-operator. Following the ideas presented in Muñoz-Villarragut, Novo and Obaya [3] and taking advantage of the results in Novo, Obaya and Villarragut [4], we are able to give some new conditions under which the 1-covering property of the omega-limit sets of this family.

In order to do this, we define a convolution operator associated to D as was done in [3]. This operator defines a change of variables which transforms our equation into one with no neutral part.

Then, we consider a new order relation which is given by means of a linear operator D, in the line of what was done in [3], and use the exponential order introduced by Smith and Thieme [6, 7]. This order relation together with the 1-covering property obtained for functional differential equations in [4] yield the desired result.

 ISBN 13: 978-84-613-5510-5

2 Linear autonomous D-operators

We consider the Fréchet space $X = C((-\infty, 0], \mathbb{R})$ endowed with the compact-open topology, that is, the topology of uniform convergence over compact subsets, which is a metric space for the distance

$$\mathsf{d}(x, y) = \sum_{n=1}^{\infty} \frac{1}{2^n} \frac{\|x - y\|_n}{1 + \|x - y\|_n}, \quad x, y \in X,$$

where $\|x\|_n = \sup_{s\in[-n,0]} |x(s)|$.

Let $BU \subset X$ be the Banach space

$$BU = \{x \in X : x \text{ is bounded and uniformly continuous}\}$$

with the supremum norm $\|x\|_\infty = \sup_{s\in(-\infty,0]} |x(s)|$. Given $r > 0$, we will denote

$$B_r = \{x \in BU : \|x\|_\infty \le r\}.$$

As usual, given $I = (-\infty, a] \subset \mathbb{R}$, $t \in I$ and a continuous function $x : I \to \mathbb{R}$, x_t will denote the element of X defined by $x_t(s) = x(t + s)$ for $s \in (-\infty, 0]$.

Let $D\colon BU \to \mathbb{R}$ be a linear operator satisfying the hypotheses:

(D1) D is linear and continuous for the norm.

(D2) For each $r > 0$, $D\colon B_r \to \mathbb{R}$ is continuous when we take the restriction of the compact-open topology to B_r.

(D3) D is atomic at 0 (see definition in Hale [1] or Hale and Verduyn Lunel [2]).

(D4) D is *stable*, i.e. there is a continuous function $c \in C([0, \infty), \mathbb{R}^+)$ with $\lim_{t\to\infty} c(t) = 0$ such that, for each $\varphi \in BU$ with $D\varphi = 0$, the solution of

$$\begin{cases} Dx_t = 0, & t \ge 0 \\ x_0 = \varphi, \end{cases}$$

satisfies $|x(t)| \le c(t)\, \|\varphi\|_\infty$ for each $t \ge 0$.

Following [3], we define the linear operator

$$\begin{array}{rccl} \widehat{D}\colon & BU & \longrightarrow & BU \\ & x & \mapsto & \widehat{D}x\colon (-\infty, 0] \to \mathbb{R} \\ & & & \qquad\quad s \mapsto Dx_s. \end{array} \tag{1}$$

As seen in [3] and [4], from (D4), we get the following result.

Theorem 2.1. *$\widehat{D}$ is invertible, $\widehat{D}^{-1}$ is bounded for the norm and uniformly continuous when we take the restriction of the compact-open topology to B_r, i.e. given $\varepsilon > 0$ there is a $\delta(r) > 0$ such that $\mathsf{d}(\widehat{D}^{-1}h_1, \widehat{D}^{-1}h_2) < \varepsilon$ for all h_1, $h_2 \in B_r$ with $\mathsf{d}(h_1, h_2) < \delta(r)$.*

As a consequence, the linear operator $T\colon BU \to \mathbb{R}$, $x \mapsto (\widehat{D}^{-1}x)(0)$, also satisfies (D1)-(D4).

 ISBN 13: 978-84-613-5510-5

3 Transformed exponential order and structure of omega-limit sets

Let $\mathbb{R}^+ = [0, \infty)$. We recall two basic definitions of topological dynamics.

Definition 3.1. Let (Ω, d) be a compact metric space. A real *continuous flow* $(\Omega, \sigma, \mathbb{R})$ is defined by a continuous map $\sigma : \mathbb{R} \times \Omega \to \Omega$, satisfying

(i) $\sigma(0, \cdot) = \mathrm{Id}$,

(ii) $\sigma(t + s, \cdot) = \sigma(t, \cdot) \circ \sigma(s, \cdot)$ for each s, $t \in \mathbb{R}$.

Definition 3.2. Let E be a complete metric space. A *semiflow* $(E, \Phi, \mathbb{R}^+)$ is determined by a continuous map $\Phi : \mathbb{R}^+ \times E \to E$, $(t, x) \mapsto \Phi(t, x)$ which satisfies

(i) $\Phi(0, \cdot) = \mathrm{Id}$,

(ii) $\Phi(t + s, \cdot) = \Phi(t, \cdot) \circ \Phi(s, \cdot)$ for all t, $s \in \mathbb{R}^+$.

Given $x \in E$, the set $\{\Phi_t(x) : t \geq 0\}$ is the *trajectory* of the point x.

Let $F : \Omega \times BU \to \mathbb{R}$, let (Ω, d) be a compact metric space and let $\sigma : \mathbb{R} \times \Omega \to \Omega$ be a minimal real flow on Ω. Denote $\omega{\cdot}t = \sigma(t, \omega)$ for all $\omega \in \Omega$ and $t \in \mathbb{R}$. Let $D : BU \to \mathbb{R}$ be an operator satisfying hypotheses (D1)-(D4).

Let us consider the family of equations

$$\frac{d}{dt} D\, z_t = F(\omega{\cdot}t, z_t), \quad t \geq 0,\ \omega \in \Omega. \tag{3.2}_\omega$$

Let ρ, $\mu > 0$; let us recall the definitions of exponential ordering on BU given in [4]. If x, $y \in BU$, then

$$x \leq_\mu y \iff x \leq y \text{ and } y(t) - x(t) \geq e^{-\mu(t-s)}(y(s) - x(s)), -\rho \leq s \leq t \leq 0,$$
$$x <_\mu y \iff x \leq_\mu y \quad \text{and} \quad x \neq y.$$

The aforementioned relation defines a positive cone in BU, $BU^+_\mu = \{x \in BU : x \geq_\mu 0\}$, in the sense that it is a closed subset of BU and satisfies $BU^+_\mu + BU^+_\mu \subset BU^+_\mu$, $\mathbb{R}^+ BU^+_\mu \subset BU^+_\mu$ and $BU^+_\mu \cap (-BU^+_\mu) = \{0\}$.

Now, we define the following *transformed exponential order* relation: if x, $y \in BU$, then

$$x \leq_{D,\mu} y \iff \widehat{D}x \leq_\mu \widehat{D}y.$$

Let us assume the following hypotheses:

(F1) $F : \Omega \times BU \to \mathbb{R}$ is continuous on $\Omega \times BU$ and its restriction to $\Omega \times B_r$ is Lipschitz continuous in its second variable when the norm is considered on B_r for all $r > 0$.

(F2) $F(\Omega \times B_r)$ is a bounded subset of $\mathbb{R}$ for all $r > 0$.

(F3) The restriction of F to $\Omega \times B_r$ is continuous when the compact-open topology is considered on B_r, for $r > 0$.

 ISBN 13: 978-84-613-5510-5

(F4) If (ω, x), $(\omega, y) \in \Omega \times BU$ and $x \leq_{D,\mu} y$, then $F(\omega, y) - F(\omega, x) \geq -\mu D(y - x)$.

From (F1), for each $\omega \in \Omega$, it follows that we have local existence and uniqueness of the solutions of equation $(3.2)_\omega$ (see e.g. Wang and Wu [8] and Wu [9]). Moreover, given $(\omega, x) \in \Omega \times BU$, if $z(\cdot, \omega, x)$ is the solution of equation $(3.2)_\omega$ with initial datum x, then the map $u(t, \omega, x) : (-\infty, 0] \to \mathbb{R}$, $s \mapsto z(t + s, \omega, x)$ belongs to BU for all $t \geq 0$ where the solution is defined.Consequently, we can define a local skew-product semiflow on $\Omega \times BU$ as follows:

$$\begin{array}{rccc} \tau: & \mathcal{U} \subset \mathbb{R}^+ \times \Omega \times BU & \longrightarrow & \Omega \times BU \\ & (t, \omega, x) & \mapsto & (\omega{\cdot}t, u(t, \omega, x)). \end{array}$$

Now, the omega-limit set of (ω_0, x_0) can be defined as

$$\mathcal{O}(\omega_0, x_0) = \{(\omega, x) \in \Omega \times BU : \exists t_n \uparrow \infty \text{ with } \omega_0{\cdot}t_n \to \omega,\ u(t_n, \omega_0, x_0) \xrightarrow{\mathsf{d}} x\}.$$

Definition 3.3. Given $r > 0$, a forward orbit $\{\widehat{\tau}(t, \omega_0, x_0) : t \geq 0\}$ of the transformed skew-product semiflow $\widehat{\tau}$ is said to be *uniformly stable for the order* $\leq_\mu$ *in* B_r if, for every $\varepsilon > 0$, there is a $\delta > 0$ such that, if $s \geq 0$ and $\mathsf{d}(u(s, \omega_0, x_0), x) \leq \delta$ for certain $x \in B_r$ with $x \leq_\mu u(s, \omega_0, x_0)$ or $u(s, \omega_0, x_0) \leq_\mu x$, then for each $t \geq 0$,

$$\mathsf{d}(u(t + s, \omega_0, x_0), u(t, \omega_0{\cdot}s, x)) = \mathsf{d}(u(t, \omega_0{\cdot}s, u(s, \omega_0, x_0)), u(t, \omega_0{\cdot}s, x)) \leq \varepsilon.$$

We will assume two more hypotheses.

(F5) There exists $r_0 > 0$ such that all the trajectories for with a Lipschitz continuous initial datum within $B_{\widehat{r}_0}$ are relatively compact for the product metric topology and uniformly stable for the order $\leq_{D,\mu}$ in bounded sets, where

$$\widehat{r}_0 = \frac{1}{\mu} \|\widehat{D}^{-1}\| \sup\{|F(\omega, x)| : (\omega, x) \in \Omega \times B_{r_0}\} + r_0.$$

(F6) If (ω, x), $(\omega, y) \in \Omega \times BU$ admit a backward orbit extension (see Shen and Yi [5]), $x \leq_{D,\mu} y$ and $\widehat{D}x(s) < \widehat{D}y(s)$ for all $s \leq 0$, then $F(\omega, y) - F(\omega, x) + \mu D(y - x) > 0$.

Theorem 3.4. *Assume that conditions* (D1)-(D4) *are satisfied; furthermore, assume conditions* (F1)-(F6). *Fix* $(\omega_0, x_0) \in \Omega \times B_{r_0}$ *such that* $\{\widehat{\tau}(t, \widehat{D}(\omega_0, x_0)) : t \geq 0\}$ *is relatively compact for the product metric topology and uniformly stable for* $\leq_\mu$ *in bounded sets, and such that* $K = \mathcal{O}(\omega_0, x_0) \subset \Omega \times B_{r_0}$. *Then* $K = \{(\omega, c(\omega)) : \omega \in \Omega\}$ *and*

$$\lim_{t \to \infty} \mathsf{d}(u(t, \omega_0, x_0), c(\omega_0{\cdot}t)) = 0,$$

where $c : \Omega \to BU$ *is a continuous map, considering the compact-open topology on* BU.

 ISBN 13: 978-84-613-5510-5

References

[1] J.K. Hale, *Theory of Functional Differential Equations*, Applied Mathematical Sciences **3**, Springer-Verlag, Berlin, Heidelberg, New York 1977.

[2] J.K. Hale, S.M. Verduyn Lunel, *Introduction to Functional Differential Equations*, Applied Mathematical Sciences **99**, Springer-Verlag, Berlin, Heidelberg, New York 1993.

[3] V. Muñoz-Villarragut, S. Novo, R. Obaya, *Neutral functional differential equations with applications to compartmental systems*, SIAM J. Math. Anal. **40** No. 3 (2008), 1003–1028.

[4] S. Novo, R. Obaya, V.M. Villarragut, *Exponential ordering for nonautonomous neutral functional differential equations*, SIAM J. Math. Anal. **41** No. 3 (2009), 1025–1053.

[5] W. Shen, Y. Yi, *Almost Automorphic and Almost Periodic Dynamics in Skew-Product Semiflows*, Mem. Amer. Math. Soc. **647**, Amer. Math. Soc., Providence 1998.

[6] H.L. Smith, H.R. Thieme, *Monotone semiflows in scalar non-quasi monotone functional differential equations*, J. Math. Anal. Appl. **150** (1990), 289–306.

[7] H.L. Smith, H.R. Thieme, *Strongly order preserving semiflows generated by functional differential equations*, J. Differential Equations **93** (1991), 332–363.

[8] Z. Wang, J. Wu, *Neutral Functional Differential Equations with Infinite delay*, Funkcial. Ekvac. **28** (1985), 157–170.

[9] J. Wu, *Unified treatment of local theory of NFDEs with infinite delay*, Tamkang J. Math. No. 1 (1991), 51–72.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

ESRKN methods for Hamiltonian Systems

Xinyuan Wu[1], Bin Wang[1] and Jianlin Xia[2]

[1] *State Key Laboratory for Novel Software Technology at Nanjing University; Department of Mathematics, Nanjing University, Nanjing, P.R.China*

[2] *Department of Mathematics , Purdue University West Lafayette, IN 47907, USA*

emails: xywu@nju.edu.cn, wangbin@smail.nju.edu.cn, xiaj@math.purdue.edu

Abstract

This work devotes to developing the ESRKN methods for systems of oscillatory second-order differential equations $q'' + Mq = f$ with $M \in R^{m\times m}$, a symmetric positive semi-definite matrix, and the perturbing function f depending only on q, based on ERKN methods proposed by Yang et al. The symplectic conditions for the systems of oscillatory second-order differential equations with Hamiltonian $H(p,q) = \frac{1}{2}p^T p + \frac{1}{2}q^T Mq + V(q)$ are presented.

Key words: SERKN methods; Symplectic conditions; Oscillatory systems; Hamiltonian systems.

MSC 2000: AMS codes (Primary 65L05, 65L06, 65M20, 65N40)

1 Introduction

The basic idea of structure-preserving algorithms is that *numerical algorithms should conserve as much as possible the qualitative behavior of the original systems.* A good theoretical foundation of structure-preserving algorithms for ordinary differential equations can be found in Hairer et al. [1], Sanz-Serna et al. [2] and for the review articles we refer to references Petzold [3] and Lubich et al. [4].

It is well-known that Hamiltonian systems are differential equations of the form:

$$\begin{cases} q' = -\nabla H_q(p,q), \\ p' = \nabla H_p(p,q), \end{cases} \tag{1}$$

where $H : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, and the dimension d is the number of degrees of freedom. Lubich et al. [4] claimed to develop numerical methods for problems with Hamiltonian

$$H(p,q) = \frac{1}{2}P^T R^{-1} P + \frac{1}{2}q^T Mq + V(q),$$

 ISBN 13: 978-84-613-5510-5

where M is a positive semi-definite constant stiffness matrix, R is a positive definite constant mass matrix (in the sequel of this paper taken as the identity matrix for convenience) and V is a smooth potential having moderately bounded derivatives. Here we are concerned with the Hamiltonian systems in the following form

$$\begin{cases} q' = p, \\ p' = -Mq - \nabla V(q), \\ q(t_0) = q_0,\ p(t_0) = p_0, \end{cases} \tag{2}$$

where $q : \mathbb{R} \to \mathbb{R}^d$ and $p : \mathbb{R} \to \mathbb{R}^d$ are generalized positions and generalized momenta, respectively, M is a $d \times d$ symmetric and positive semi–definite matrix containing implicitly the frequencies of the problem and $\nabla V(q)$ is the gradient of a real-valued function $V(q)$ whose second derivatives are continuous. The Hamiltonian of systems (2) is given by

$$H(p, q) = \frac{1}{2}p^T p + \frac{1}{2}q^T Mq + V(q).$$

It is easy to see that the Hamiltonian system (2) is equivalent to the following system of second order differential equations

$$\begin{cases} q''(t) + Mq(t) = f(q(t)), \qquad t \in [t_0, T], \\ q(t_0) = q_0, \quad q'(t_0) = q'_0. \end{cases} \tag{3}$$

with $f(q) = -\nabla V(q)$, which can be thought of as the perturbing force.

Early work in the scientific literature on the numerical integration of the system (3) were mostly concerned with the case where the frequency matrix $M = w^2 I$ is a diagonal matrix. Recently, new approaches to constructing RKN-type methods for (3) have been proposed (see [5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18] for examples).

2 Multidimensional ERKN methods

Yang et al. proposed the ERKN methods [17]. However, the ERKN methods in that paper only consider for the special case where M is a diagonal matrix with nonnegative entries. This section extends the ERKN methods to the general case with $M \in R^{d\times d}$, and the perturbing function f depends only on q.

Define

$$\phi_0(V) := \sum_{k=0}^{\infty} \frac{(-1)^k V^k}{(2k)!}, \quad \phi_1(V) := \sum_{k=0}^{\infty} \frac{(-1)^k V^k}{(2k+1)!}. \tag{4}$$

From the variation-of-constant formula, the solutions of the systems (3) at $\xi = t_n + hz$ satisfy the following integral equations:

$$\begin{cases} q(t_n + \mu h) = \phi_0(\mu^2 V)q(t_n) + \mu h\phi_1(\mu^2 V)q'(t_n) + h^2 \int_0^\mu (\mu - z)\phi_1\big((\mu - z)^2 V\big)\hat{f}(t_n + hz)dz, \\ q'(t_n + \mu h) = -\mu hM\phi_1(\mu^2 V)q(t_n) + \phi_0(\mu^2 V)q'(t_n) + h\int_0^\mu \phi_0\big((\mu - z)^2 V\big)\hat{f}(t_n + hz)dz. \end{cases} \tag{5}$$

where $V = h^2 M$, $\hat{f}(\xi) = f(q(\xi))$.

In order to obtain a numerical method for the system (3) one has to approximate the integrals in (5) with some higher order quadrature formulas. This leads to the following scheme for (3).

 ISBN 13: 978-84-613-5510-5

Definition 2.1. *An s-stage ERKN method for the numerical integration of the oscillatory system (3) is defined as*

$$\begin{cases} Y_i &= \phi_0(c_i^2V)q_n + hc_i\phi_1(c_i^2V)q'_n + h^2\sum\limits_{j=1}^{s} a_{ij}(V)f(Y_j), \qquad i = 1,\cdots,s, \\ q_{n+1} &= \phi_0(V)q_n + h\phi_1(V)q'_n + h^2\sum\limits_{i=1}^{s} \bar{b}_i(V)f(Y_i), \\ q'_{n+1} &= -hM\phi_1(V)q_n + \phi_0(V)q'_n + h\sum\limits_{i=1}^{s} b_i(V)f(Y_i), \end{cases} \tag{6}$$

where, b_i, $\bar{b}_i, i = 1,\cdots,s$ and $a_{ij}, i, j = 1,\cdots,s$ are matrix-valued functions of $V = h^2M$.

It is very important to notice that when $M = 0$, the scheme (6) reduces to the classical RKN methods.

3 Order conditions for multidimensional ERKN methods

An ERKN method (6) for the system of second order differential equations (3) has *order* r, if for sufficiently smooth (3) the conditions

$$e_{n+1} := q_{n+1} - q(t_n + h) = \mathcal{O}(h^{r+1}) \quad \text{and} \quad e'_{n+1} := q'_{n+1} - q'(t_n + h) = \mathcal{O}(h^{r+1}) \tag{7}$$

are satisfied simultaneously, where $q(t_n + h)$ and $q'(t_n + h)$ are the exact solution of (3) and its derivative at $t_n + h$, respectively, and q_{n+1} and q'_{n+1} are the one step numerical results obtained by the method from the exact starting values $q_n = q(t_n)$ and $q'_n = q'(t_n)$ (the local assumptions).

After introducing two functions ϕ_0 and ϕ_1 in(4), we now define

$$\phi_j(M) := \sum_{k=0}^{\infty} \frac{(-1)^k M^k}{(2k+j)!}, \quad j = 2, 3, \cdots. \tag{8}$$

Then the asymptotic expansions of the true solution to the problem (3) and its derivative in powers of h are given, respectively, by

$$\begin{aligned} q(t_n + h) &= \phi_0(V)q_n + h\phi_1(V)q'_n + \sum_{j=0}^{\infty} h^{j+2}\phi_{j+2}(V)\hat{f}_n^{(j)}, \\ q'(t_n + h) &= \phi_0(V)q'_n - hM\phi_1(V)q_n + \sum_{j=0}^{\infty} h^{j+1}\phi_{j+1}(V)\hat{f}_n^{(j)}, \end{aligned}$$

where $\hat{f}_n^{(j)} = \frac{d^j}{dt^j}\hat{f}(t)\Big|_{t=t_n}$ is the jth derivative of $\hat{f}(z)$ at $z = t_n$.

Theorem 3.1. *The necessary and sufficient conditions for an s-stage ERKN method to be of order r are given by*

$$\begin{cases} \bar{b}^T(V)\Phi(\tau) = \frac{\rho(\tau)!}{\gamma(\tau)}\phi_{\rho(\tau)+1}(V) + \mathcal{O}(h^{r-\rho(\tau)}), & \rho(\tau) = 1,\cdots,r-1, \\ b^T(V)\Phi(\tau) = \frac{\rho(\tau)!}{\gamma(\tau)}\phi_{\rho(\tau)}(V) + \mathcal{O}(h^{r+1-\rho(\tau)}), & \rho(\tau) = 1,\cdots,r, \end{cases} \tag{9}$$

where τ is the EN-tree associated with an elementary differential $\mathcal{F}(\tau)(q_n, q'_n)$ of the function $f(q, q')$ at (q_n, q'_n). $\rho(\tau)$, $\alpha(\tau)$ and $\gamma(\tau)$ are exactly the same as those in [17].

 ISBN 13: 978-84-613-5510-5

4 Symplectic conditions for ERKN methods

This section only considers the situation when M is a symmetric positive semi-definite matrix. The symplectic conditions for Runge-Kutta methods are obtained by Sanz-Serna [19] and for RKN methods are derived by Suris [20]. We now state the symplectic conditions for the ERKN methods.

Theorem 4.1. *If the coefficients of an s-stage ERKN method satisfy the following conditions*

$$\begin{aligned}
&b_i(V)\phi_0(V)+\bar{b}_i(V)V\phi_1(V)=d_i\phi_0(c_i^2V),\ d_i\ \in\ \mathbb{R},\ i=1,2,\cdots,s,\\
&\bar{b}_i(V)(\phi_0(V)+c_iV\phi_1(V)\phi_0^{-1}(c_i^2V)\phi_1(c_i^2V))\\
&=b_i(V)(\phi_1(V)-c_i\phi_0(V)\phi_0^{-1}(c_i^2V)\phi_1(c_i^2V)),\ i=1,2,\cdots,s,\\
&b_i(V)(\bar{b_j}(V)-\phi_0(V)\phi_0^{-1}(c_i^2V)a_{ij}(V))-\bar{b_i}(V)V\phi_1(V)\phi_0^{-1}(c_i^2V)a_{ij}(V)\\
&=b_j(V)(\bar{b_i}(V)-\phi_0(V)\phi_0^{-1}(c_j^2V)a_{ji}(V))-\bar{b_j}(V)V\phi_1(V)\phi_0^{-1}(c_j^2V)a_{ji}(V),\ i,\ j=1,2,\cdots,s,
\end{aligned}\tag{10}$$

then the ERKN method is symplectic.

Acknowledgements

The research is supported by the Natural Science Foundation of China under Grant 10771099.

References

[1] E. Hairer, C. Lubich and G. Wanner, Geometric Numerical Integration: Structure-Preserving Algorithms, Springer-Verlag, Berlin, Heidelberg, 2002.

[2] Sanz-Serna and M. P. Calvo, Numerical Hmiltonian Problem, in Applied Mathematics and Matematical Computation, vol. 7, Chapman & Hall, London, 1994

[3] L. R. Petzold, L. O. Jay & J. Yen, Numerical solution of highly oscillatory ordinary differential equations, Acta Numerica (1997), 437-483.

[4] David Cohen, Tobias Jahnke, Katina Lorenz, Christian Lubich, Numerical methods for highly oscillatory Hamiltonian systems: a review, in Analysis, Modeling and Simulation of Multiscale Problems (A. Mielke, ed.), Springer, Berlin, 2006, 553-576.

[5] A. García, P. Martín, A.B. González, New methods for oscillatory problems based on classical codes, Appl. Numer. Math. 42 (2002) 141-157.

[6] J. Vigo-Aguiar, T.E. Simos, J.M. Ferrándiz, Controlling the error growth in long-term numerical integration of perturbed oscillations in one or more frequencies, Proc. Roy. Soc. London Ser. A, 460 (2004) 561-567.

 ISBN 13: 978-84-613-5510-5

[7] A.B. González, P. Martín, J.M. Farto, A new family of Runge-Kutta type methods for the numerical integration of perturbed oscillators, Numer. Math. 82 (1999) 635-646.

[8] S. Stavroyiannis, T.E. Simos, Optimization as a function of the phase-lag order of two-step P-stable method for linear periodic IVPs, Applied Numerical Mathematics 59 (2009) 2467-2474.

[9] M. Hochbruck, Ch. Lubich, A Gautschi-type method for oscillatory second-order differential equations, Numer. Math. 83 (1999) 403-426

[10] J.M. Franco, Runge-Kutta-Nyström methods adapted to the numerical integration of perturbed oscillators, Comput. Phys. Comm. 147 (2002) 770-787.

[11] J.M. Franco, A 5(3) pair of explicit ARKN methods for the numerical integration of perturbed oscillators, J. Comput. Appl. Math. 161 (2003) 283-293.

[12] H. Van de Vyver, Stability and phase-lag analysis of explicit Runge-Kutta methods with variable coefficients for oscillatory problems, Comput. Phys. Comm. 173 (2005) 115-130.

[13] J.M. Franco, New methods for oscillatory systems based on ARKN methods, Appl. Numer. Math. 56 (2006) 1040-1053.

[14] Y.L. Fang, X.Y. Wu, A new pair of explicit ARKN methods for the numerical integration of general perturbed oscillators, Appl. Numer. Math. 57 (2007) 166-175.

[15] H.L. Yang, X.Y. Wu, Trigonometrically-fitted ARKN methods for perturbed oscillators, Appl. Numer. Math. 58 (2008) 1375-1395.

[16] X. Wu, X. You, J. Li, Note on derivation of order conditions for ARKN methods for perturbed oscillators, Comput. Phys. Comm. 180 (2009) 1545-1549.

[17] H.L. Yang, X.Y. Wu, Xiong You and Yonglei Fang, Extended RKN-type methods for numerical integration of perturbed oscillators, Comput. Phys. Comm., 180 (2009) 1777-1794.

[18] X. Wu, X. You, J. Xia, Order conditions for ARKN methods solving oscillatory systems, Comput. Phys. Comm. 180 (2009) 2250-2257.

[19] Sanz-Serna, Runge-Kutta schems for Hamiltonian systems, BIT 28 (1988) 877-883.

[20] Y.B. Suris, The canonicity of mapping generated by Runge-Kutta type methods when integrating the systems $\ddot{x} = -\frac{\partial U}{\partial x}$, Zh. Vychisl. Mat. i Mat. Fiz., 29(1989), 202-211. (In Russian) Translation, U.S.S.S. Comput. Maths. and Math. Phys., 29 (1989) 138-144.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

New Characterization and Reconstruction Formula for Bandlimited Functions in Higher Dimensions

Ahmed I. Zayed[1]

[1] *Department of Mathematical Sciences, DePaul University, Chicago, IL 60614*

emails: azayed@depaul.edu

Abstract

The class of bandlimited functions on the real line can be characterized in a number of different ways. Some of these characterizations can be extended to higher dimensions. In this talk we give a new characterization of bandlimited functions on the real line and then extend it to higher dimensions. This new characterization is based on the notion of chromatic derivatives and chromatic series expansions. Although chromatic derivative are linear combinations of the ordinary derivatives, chromatic series have better approximation properties than Taylor series.

Key words: Bandlimited functions, chromatic derivatives, chromatic expansions
MSC 2000: Primary 41A58 ; Secondary 94A20

1 Introduction

The space of bandlimited functions plays an important role in communication engineering. In fact, the term 'bandlimited' signals came from electrical engineering where it means that the frequency content of a signal is limited by certain bounds from below and above. The classical space of bandlimited functions, which is also called the Paley-Wiener space, PW_π, of functions bandlimited to $[-\pi, \pi]$, consists of square integrable functions whose Fourier transforms, $\hat{f}$ have supports in $[-\pi, \pi]$.

One of the fundamental characterizations of the space PW_π, is given by the following theorem of Paley and Wiener

Theorem 1.1 (Paley-Wiener) *A function f is bandlimited to $[-\sigma, \sigma]$, i.e., $f \in PW_\sigma$ if and only if*

$$f(t) = \int_{-\sigma}^{\sigma} e^{-i\omega t} g(\omega)\, d\omega \qquad (t \in I\!R)\,,$$

for some function $g \in L^2(-\sigma, \sigma)$ and if and only if f is an entire function of exponential type that is square integrable on the real line, i.e., f is an entire function such that

$$|f(z)| \leq \sup_{x \in I\!R} |f(x)| \exp(\sigma\,|y|), \quad z = x + iy,$$

 ISBN 13: 978-84-613-5510-5

and

$$\int_{\mathbb{R}} |f(x)|^2 \, dx < \infty.$$

Another important property of the space PW_σ is given by the Whittaker-Shannon-Koteln'nikov (WSK) sampling theorem, which can be stated as follows [10]:

Theorem 1.2 *If $f \in PW_\sigma$, then f can be reconstructed from its samples, $f(t_k)$, where $t_k = k\pi/\sigma$ via the formula*

$$f(t) = \sum_{k=-\infty}^{\infty} f(t_k) \frac{\sin \sigma(t - t_k)}{\sigma(t - t_k)} \qquad (t \in \mathbb{R}), \tag{1}$$

with the series being absolutely and uniformly convergent on $\mathbb{R}$.

One of the earliest generalizations of the Paley-Wiener space is the Bernstein space. Let $\sigma > 0$ and $1 \leq p \leq \infty$. The Bernstein space B_σ^p is a Banach space consisting of all entire functions f of exponential type with type at most σ that belong to $L^p(\mathbb{R})$ when restricted to the real line. It is known [1, p. 98] that $f \in B_\sigma^p$ if and only if f is an entire function satisfying

$$\|f(x + iy)\|_p \leq \|f\|_p \exp(\sigma|y|), \quad z = x + iy,$$

where the norm on the left is taken with respect to x for any fixed y and

$$\|f\|_p = \left(\int_{-\infty}^{\infty} |f(x)|^p \, dx \right)^{1/p} < \infty, \quad \text{if } 1 \leq p < \infty$$

and

$$\|f\|_\infty = \text{ess.sup}_{x \in \mathbb{R}} |f(x)| < \infty, \text{ if } p = \infty.$$

Unlike the spaces $L^p(\mathbb{R})$, the spaces B_σ^p are closed under differentiation and the differentiation operator plays a vital role in their characterization. The Bernstein spaces have been characterized in a number of different ways and one can prove that the following are equivalent:

A) A function $f \in L^p(\mathbb{R})$ belongs to B_σ^p if and only if its Fourier transform has support $[-\sigma, \sigma]$. For $2 < p$, $\hat{f}$ is understood to be in the sense of distributions.

B) Let $f \in C^\infty(\mathbb{R})$ be such that $f^{(n)} \in L^p(\mathbb{R})$ for all $n = 0, 1, \cdots$, and some $1 \leq p \leq \infty$, then $f \in B_\sigma^p$ if and only if f satisfies the Bernstein's inequality [**?**, p. 116]

$$\left\| f^{(n)} \right\|_p \leq \sigma^n \|f\|_p, \quad n = 0, 1, 2, \cdots; 1 \leq p \leq \infty. \tag{2}$$

C) Let $f \in C^\infty(\mathbb{R})$ be such that $f^{(n)} \in L^p(\mathbb{R})$ for all $n = 0, 1, \cdots$, and some $1 \leq p \leq \infty$. Then

$$\lim_{n \to \infty} \left\| f^{(n)} \right\|_p^{1/n} \leq \infty, \quad \text{exists}$$

and $f \in B_\sigma^p$ if and only if $\lim_{n \to \infty} \left\| f^{(n)} \right\|_p^{1/n} = \sigma < \infty$.

 ISBN 13: 978-84-613-5510-5

D) Let $f \in C^\infty(\mathbb{R})$ be such that $f \in L^p(\mathbb{R})$ for some $1 \leq p \leq \infty$. Then $f \in B_\sigma^p$ if and only if it satisfies the Riesz interpolation formula

$$f^{(1)}(x) = \frac{\sigma}{\pi^2} \sum_{k \in \mathbb{Z}} \frac{(-1)^{k-1}}{(k-1/2)^2} f\left(x + \frac{\pi}{\sigma}(k-1/2)\right) \tag{3}$$

where the series converges in $L^p(\mathbb{R})$.

In higher dimensions, the situation becomes slightly more complicated. Let Ω be a fixed compact set in $\mathbb{R}^d$, and define $B^p(\Omega), 1 \leq p \leq \infty$ as the space of functions bandlimited to Ω, i.e.,

$$B^p(\Omega) = \left\{ f \in L^p(\mathbb{R}^d) : \text{supp}\hat{f} \subset \Omega \right\},$$

where $\text{supp}\hat{f}$ is the support of the Fourier transform of f. For $2 < p$, $\hat{f}$ is understood to be in the sense of distributions. Some of the aforementioned characterizations (A) -(D) of bandlimited functions in one variable are still valid in higher dimensions.

In this article we will give a new characterization of the space $B^2(\Omega)$ that employs the notion of chromatic derivatives.

2 Chromatic Derivatives and Series Expansions

The Whittaker-Shannon-Kotel'nikov (WSK) [10] sampling series (1) may be viewed as a global expansion because it uses the function values at infinitely many points uniformly distributed on the real line. On the other hand, as an entire function, f has a Taylor series expansion of the form $f(t) = \sum_{n=0}^{\infty} \left(f^{(n)}(0)/n!\right) t^n$, which may be viewed as a local expansion since it uses the values of f and all its derivatives at a single point. Unlike the sampling series, which plays an important role is signal processing, the Taylor series has very limited applications because, among other reasons, a truncated Taylor series is a polynomial and not bandlimited.

Chromatic derivatives and series expansions have recently been introduced by A. Ignjatovic in [2] as an alternative representation to Taylor series and they have been shown to be more useful in practical applications than Taylor series; see [3, 4, 5, 6, 7, 8].

The n-th chromatic derivative $K^n[f](t_0)$ of an analytic function $f(t)$, at t_0 is a linear combination of the ordinary derivatives $f^{(k)}(t_0)$, $0 \leq k \leq n$, where the coefficients of the combination are based on a system of orthogonal polynomials.

For the reader's convenience, we will briefly describe how chromatic series are constructed. Fix a real-valued weight function $\rho(\omega) \geq 0$, $\omega \in \mathbb{R}$, with $\int_{-\infty}^{\infty} \rho(\omega) d\omega = 2\pi$. Assume that ρ has finite moments. Then there exists a complete family of orthogonal polynomials $\{p_k\}_{k=0}^{\infty}$ with respect to ρ

Definition 2.1 *The n-th chromatic derivative of f at $t = t_0$ is defined as*

$$K^n[f](t_0) = p_n(jD)[f](t_0) = \mathcal{F}^{-1}(\hat{f} p_n)(t_0),$$

 ISBN 13: 978-84-613-5510-5

in particular,

$$K^n[f](0) = p_n\left(jD\right)[f](0) = \frac{1}{2\pi} < \hat{f}, p_n >, \tag{4}$$

where $D = d/dt$, $j = \sqrt{-1}$, $p_n\left(jD\right)$ *is a polynomial in the operator* jD *and* $\mathcal{F}^{-1}$ *denotes the inverse Fourier transformation.*

The chromatic series expansion of f *is given by*

$$f(t) \sim \sum_{n=0}^{\infty} K^n[f](0)\psi_n(t) = \sum_{n=0}^{\infty} K^n[f](0)K^n[\tilde{\psi}](t), \tag{5}$$

where $\psi_n(t) = K^n[\tilde{\psi}](t) = \mathcal{F}^{-1}(p_n\rho)(t)$, *and* $\tilde{\psi} = \mathcal{F}^{-1}(\rho)(t)$.

It has been shown [5] that when $\rho(\omega) = \chi_{(-1,1)}$, the characteristic function of $(-1, 1)$, the chromatic series associated with the Legendre polynomials converge in the whole complex plane, $\mathbb{C}$, to entire functions and the set of entire functions for which $\sum_{n=0}^{\infty} |K^n(f)(0)|^2$ converges is precisely the set of L^2 functions whose Fourier transforms are finitely supported, i.e., the set of bandlimited functions. For such functions the chromatic expansions converge uniformly on $\mathbb{R}$, and their truncations are themselves bandlimited.

In this talk we extend some of these results to higher dimensions.

References

[1] R. Boas, *Entire Functions,* Academic Press, New York (1954).

[2] A. Ignjatovic and N. Carlin, *Method and a system of acquiring local signal behavior parameters for representing and processing a signal,* US Patent 6313778, filled July 1999, issued November 6, (2001).

[3] A. Ignjatovic,, *Local approximations based on orthogonal differential operators*, J. Fourier Anal. and Appls. **13** (2007),pp. 661–692.

[4] A. Ignjatovic, *Chromatic derivatives and local approximations,* IEEE Transactions on Signal Process., **57, No. 8**, (2009).

[5] A. Ignjatovic, *Chromatic derivatives, chromatic expansions and associated function spaces,* East Journal on Approximations, **15, No. 2**, (2009), pp. 263-302.

[6] M. J. Narasimha, A. Ignjatovic and P. P. Vaidyanathan, *Chromatic Derivative Filter Banks,* IEEE Signal Processing Letters, **9, No. 7**, (2002).

[7] P. P. Vaidyanathan, A. Ignjatovic and S. Narasimha,*New Sampling Expansions of Band Limited Signals Based on Chromatic Derivatives,* Proc. 35th Asilomar Conference on Signals, Systems, and Computers, Monterey, California, (2001).

 ISBN 13: 978-84-613-5510-5

[8] G. Walter and X. Shen, *A Sampling Expansion for non Bandlimited Signals in Chromatic Derivatives,* IEEE Transactions on Signal Processing, **53**, (2005).

[9] G. Walter, *Chromatic Series and Prolate Spheroidal Wave Functions,* Journal of Integral Equations and Appls, **20, No. 2**, (2006).

[10] A. I. Zayed, *Advances in Shannon's Sampling Theory,* CRC Press, Boca Raton, Florida, 1993.

[11] A. I. Zayed, *Generalizations of Chromatic Derivatives and Series Expansions,* IEEE Transactions of Signal Processing, **58, No. 3** (2010), pp. 1638-1647.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Parallel Implementation of a Semi-Implicit 3-D Lake Hydrodynamic Model

Mario C. Acosta [1,2], Mancia Anguita[2], Francisco J. Rueda[1,3] and F. Javier Fernández-Baldomero [2]

[1] *Water Institute, University of Granada*
[2] *Department of Computer Architecture and Computer Technology, CITIC-UGR, University of Granada*
[3] *Department of Civil Engineering, University of Granada*

emails: marioa@correo.ugr.es, manguita@ugr.es, fjrueda@ugr.es, jfernand@ugr.es

Abstract

The parallel implementation of a three-dimensional (3-D) lake hydrodynamic model in a small commodity cluster of three multi-core nodes is presented. The parallel program uses the three nodes in the cluster (by using the message passing standard MPI) and the four cores in a node (by using the shared memory standard OpenMP). This work analyzes the influence in performance of using different platform configurations, several workload distributions, several parallel implementations, and block-driven processing.

Key words: Parallel processing, Shared memory systems, Distributed memory systems, Hydrodynamics.

1. Introduction

High performance computations are being increasingly demanded in water sciences to get detailed descriptions of the flow fields that develop in natural ecosystems within reasonable lengths of time. It has been through these detailed descriptions of the flow fields, obtained either by means of simulations conducted with three-dimensional (3D) numerical algorithms solving the governing equations of fluid motion (Navier-Stokes equations), or through field observations collected with high-resolution experimental techniques that water scientists have gained, in the last few years, some understanding of transport processes in natural lakes and reservoirs [1], [2]. This understanding, however, is still far from complete.

 ISBN 13: 978-84-613-5510-5

Many of the 3D hydrodynamic models currently used in lake research are based on the solution of a simplified form of the Navier-Stokes equations, referred to as the shallow water equations (SWE), in which the vertical pressure gradients are assumed hydrostatic. The main state variables in the SWE are the spatially-varying horizontal velocities (u, v) and the water surface elevation (η). Being based on a simplified set of equations, SWE models have a moderate computational cost. SWE models, however, are still time and memory consuming when high density spatial grids are used or when they are used to simulate the long-term behavior of natural water systems. High-resolution grids, for example, are needed in order to resolve flow features of small spatial scales, such as near-shore currents, which are important to understand the physical and biogeochemical behavior of large-scale natural water bodies. Long simulation times are unacceptable when the SWE models are part of decision support systems. Model results in these cases are needed much faster than real time so that they can be used to develop and test management strategies aimed at minimizing the effects of natural disasters, such as floods or the introduction of invasive species.

The models in decision support systems are typically run repeatedly, each time with a different set of parameters and/or perturbed boundary conditions, in order to provide predictions of the future state of the flow field with an appropriate degree of uncertainty (see, for example, [3]). Most efforts in the field of environmental fluid modeling, hence, are being directed towards improving the execution time of existing numerical models, especially in inexpensive commodity platforms, so that (1) environmental scientists can build a rigorous and detailed understanding of the physical processes of transport and mixing occurring in inland water bodies and (2) water managers can get accurate predictions of the response of flow systems to external perturbations (e.g. pollution) and management strategies in a timely manner. The goal of this work is to present a parallel implementation of 3D SWE model in a small commodity cluster of three multi-core nodes.

The paper is organized as follows: Section 2 briefly describes the 3-D hydrodynamic model here implemented in parallel; Section 3 deals with related works; Section 4 compares several parallel implementations and data domain decompositions; Section 5 presents the computational platform and discusses some experimental results. Finally, Section 6 summarizes conclusions.

2. Hydrodynamic Model

This work evaluates a parallel implementation of a 3-D SWE model (SI3D, [4]), which has been extensively validated, both against analytical solutions and field data sets collected in a wide range of lake environments [5]. SI3D is based on the numerical solution of the continuity equation for incompressible fluids, the Reynolds-averaged and shallow water form of the Navier-Stokes equations for momentum, the transport equation for temperature, and an equation of state relating temperature to fluid density. The governing equations are first posed in

 ISBN 13: 978-84-613-5510-5

layer-averaged form by integrating over the height of a series of horizontal layers separated by level planes. The layer-averaged form of the equations is discretized using a semi-implicit, three-level, iterative leapfrog-trapezoidal finite difference algorithm on a staggered Cartesian grid, which introduces little numerical diffusion [4]. The semi-implicit approach is based on treating the gravity wave and vertical diffusion terms implicitly to avoid time-step limitations due to gravity wave Courant–Friedrich–Levy conditions, and to guarantee stability of the method [6]. All other terms (including advection) are treated explicitly. Laplacian operators are used to represent mixing. Constant mixing coefficients are used to parameterize the effect of horizontal eddies. A two-equation turbulence model calculates the vertical eddy coefficients of mixing [7]. Computations in each iteration proceed on a water column-by-water column basis, to assemble a five-diagonal system of equations for water surface elevation η, which is solved using a preconditioned-conjugate gradient solver [4]. Horizontal velocities are recovered from the updated values of η.

3. Related Works

Several SWE models have been implemented in parallel that take advantage of their data parallelism. Implementations that use the message passing paradigm with MPI for both 2-D ([8], with one and also two layers in [9], [10], [11]) and 3-D models ([11]) can be found. The implementation of [10] parallelizes a 3-D lattice Boltzmann model using the shared memory paradigm with OpenMP. These MPI and OpenMP implementations use domain decomposition to divide the workload among processes or threads. Performance can also be increased using SSE instructions either explicitly (manually) or through libraries, or, alternatively, using GPUs. [12], for example, presents results of a SSE optimized implementation of a 2-D SWE-model. [13], in turn, solves a 2-D SWE-model using the Intel Integrated Performance Primitives library. An implementation of a 2-D SWE-model in several GPUs supporting CUDA programming toolkit is presented in [14].

Three-dimensional models, like SI3D, manage larger amounts of data and require higher computer performance. Moreover, distributing workload evenly is more difficult because, to the irregular horizontal dimensions or layer dimensions (first and second dimension), the irregular vertical dimension or depth (third dimension) is added.

Some parallel implementations of a semi-implicit 3-D hydrodynamic model, SI3D, are presented and evaluated here. The parallel implementations of the 3-D model combine both message passing (with MPI) and shared memory paradigms (with OpenMP). Implementations with redundant operations (workload overlapping) are compared to non-redundant implementations. Workload overlapping increases the number of operations and decreases communications. This work also analyzes the influence of different platform configurations (such as simultaneous multithreading, Intel SpeedStep and Turbo Mode technologies, and prefetching hardware), and different domain decompositions have on code

 ISBN 13: 978-84-613-5510-5

performance. Different compiler optimization options and a block-driven processing implementation were also tested.

4. Implementation

Several parallel SI3D versions have been implemented. The speedup achieved in a parallel implementation of SI3D in which OpenMP construct !$OMP PARALLEL DO- END PARALLEL DO is used to locate parallelism was 1.22 with the four cores of a processor. The performance has been increased when the programmer has also done explicitly these tasks: assign jobs to threads; create and destroy threads; communicate and synchronize threads. Moreover, several parallel versions have been implemented in order to compare implementations with redundant operations, which avoid communications and synchronizations (C/S), with non-redundant operation implementations. The results show that redundant operations improve the MPI version performance of SI3D but the OpenMP version increases performance when the redundant operations are reduced by adding some extra synchronization.

Figure 1 shows a flow diagram of the best SI3D parallel implementation. The stage 2 was not parallelized because it is a 2% of the stages 1 and 3 together in the SI3D sequential version. C/S occur several times per iteration. Process 0 assembles and solves the penta-diagonal matrix for free surface elevation η (stage 2), and scatters the values of η among processes. It also occurs at the end of each iteration, where the processes interchange some data of u, v, and η. In the non-redundant MPI version there are additional data interchanges between processes: four in stage 1 and five in stage 3.

The parallel implementations of SI3D use domain decomposition to divide the workload among processes and threads, as is usually done in Computational Fluid Dynamics applications. Domain decomposition is done previous to the start of the simulations (Figure 1). The criteria followed in determining the best domain decomposition is that all the sub-domains should have the same o similar number of wet cells and that the sub-domain data must be stored in contiguous memory positions.

The overhead of the parallel implementations of SI3D, like in other related applications, is mainly affected by:

- Load unbalance. The irregular grid dimensions make difficult to obtain an even distribution.
- Communication time. It depends both on the number of communications and on the amount of data being transferred in each communication. In the data interchanges between processes, both of them depend on the domain decomposition approach used (most of the C/S are border interchanges between processes).
- Extra operations due to sub-domain overlapping. The number of communications can be reduced by overlapping sub-domains. In these overlapping regions, computations are redundant. The overhead that results

 ISBN 13: 978-84-613-5510-5

from redundant calculations depends on the extent of overlapping regions, and this, in turn, depends on the particular domain decomposition approach used.

Therefore, domain decomposition affect performance, several approaches are possible in 3-D models. Either horizontal-cut or vertical-cut (depth) decomposition can be applied in these cases. Horizontal-cut decomposition distributes layers among sub-domains, i.e. among processors/cores. The degree of parallelism in this case equals the number of layers and communication depends on the horizontal resolution and the horizontal extent of the lake. Given that large differences exist between the horizontal and vertical dimensions of large-scale geophysical systems, the degree of parallelism in the horizontal-cut

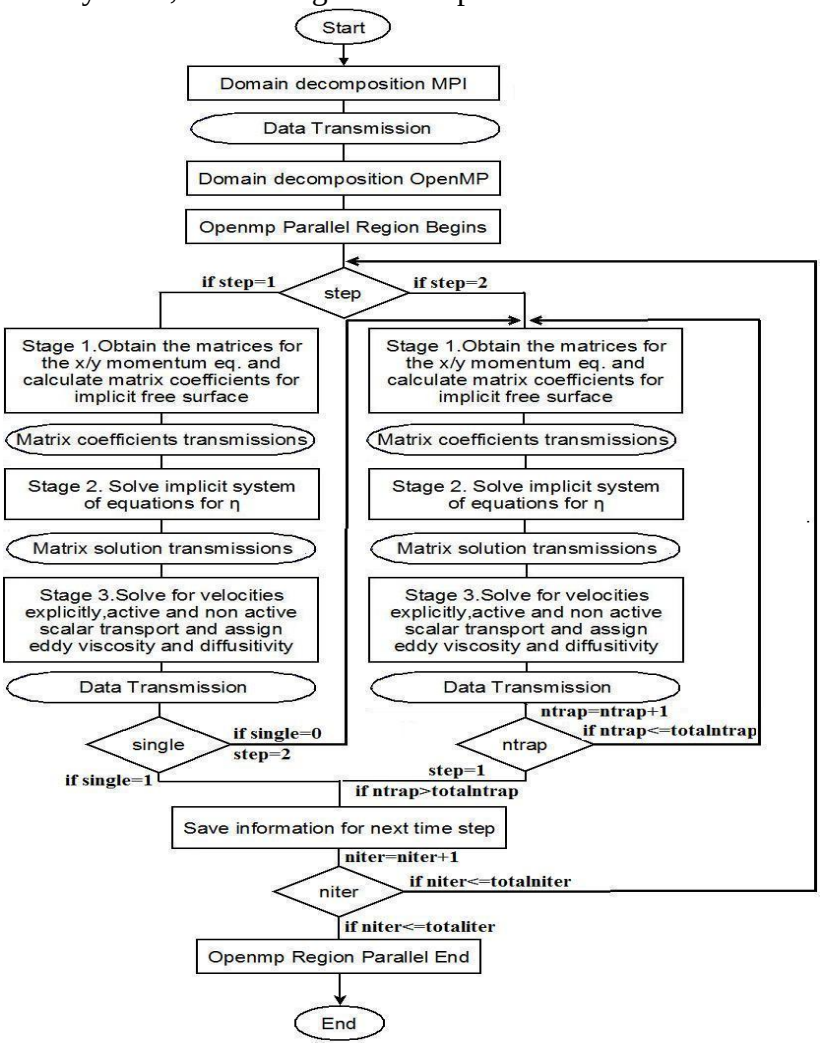
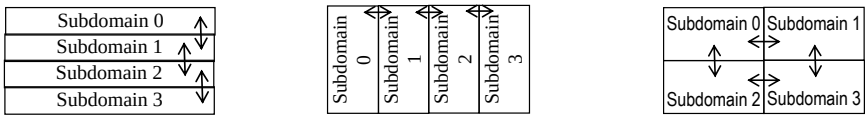


Figure 1. Flow diagram of the best parallel algorithm. The diagram includes the MPI transmission points.

decomposition tend to be lower than in a vertical-cut decomposition.

 ISBN 13: 978-84-613-5510-5

Three types of vertical-cut decomposition (of a river, lake, etc.) are possible (Figure 2). The data interchange between the sub-domains is indicated by the arrows in the Figure 2. The length of the boundaries between any two given sub-domains reflects the amount of data exchanged between them. It is also indicative of the amount of redundant calculations if the number of communications is reduced by overlapping sub-domains. The total length of the sub-domain boundaries will depend on the particular geometry of the water body being simulated, and on how the domain is partitioned among processes. The number of interchange communications is larger if one uses the *two-direction* cut distribution, as shown in Figure 2(c). The total amount of data exchanged among processes and the number of redundant calculations, though, could be less than in the other two distributions, depending on the particular geometry and number of sub-domains. With this distribution a process can both send to and receive from more than two processes. Both *narrow* (Figure 2(b)) and *wide* (Figure 2(a)) cut distributions have the same number of interchange communication operations. A process will send to and receive from just one or two processes. Larger amounts of data are exchanged and more redundant operations are done in the distribution shown in Figure 2(a) (wide-cut distribution). [15] compares the alternatives in Figure 2(b) and Figure 2(c) using MPI in a cluster of four AMD Opteron 2.2 GHz nodes (2 cores each) connected through Gigabit Ethernet. The results for different grid sizes show that better performances are achieved if the decomposition is done using narrow cut distribution compared to the two-direction distribution. [8] compares the alternatives in Figure 2(a) and Figure 2(b) using MPI in a CC-NUMA HP/Convex Exemplar X-Class (SPP2200) with 64 processors distributed in four hyper-nodes. The eight nodes of a hyper-node are connected through a network (switch) of 960 MB/s bandwidth in each link direction [16] (the network of the Exemplar is an implementation of the standard SCI). The results show that narrow-cut distribution reduces execution time compared to wide-cut distribution. Here some tests (Section 5) compare wide and narrow-cut distributions with both message passing and shared memory paradigms in SI3D. Note that the alternative in the Figure 2(c) has less data locality compared to the alternatives in (Figure 2(b)) and (Figure 2(a)). The data of a sub-domain in the wide and narrow distribution were stored in disk and memory in contiguous positions in order to improve locality. The lack of locality decreases performance, especially in shared memory implementations.



(a) (b) (c)

Figure 2. Three domain decomposition alternatives with vertical cut: (a) wide cut distribution, (b) narrow cut distribution, (c) two-direction cut distribution. Arrows show the communication needed among sub-domains in this kind of applications

 ISBN 13: 978-84-613-5510-5

O2: inline expansion and cloning of functions, classical optimization (loop unrolling, constant and copy propagation, strength reduction, variable renaming, dead store elimination, global instruction scheduling and control speculation …) and vectorization (this tries to generate MMX, SSE, SSE2 instructions). O2 is the generally recommended optimization level for reducing execution time.
O3: O2 optimizations plus more aggressive optimizations, such as prefetching, scalar replacement to reduce memory references, and loop and memory access transformations.
ipo: multifile interprocedural optimization (this, for instance, allows inline expansion and cloning for calls to functions defined in separate files).
openmp: this option enables the parallelizer to generate multi-threaded code based on the OpenMP directives included by the programmer.
xSSE4.2 (architecture-specific optimization): this tries to generate MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1 and SSE4.2 instructions (vectorization) and can optimize for the Intel Core i7 processor family.
prof_gen and prof_use (Profile Guided Optimization or PGO): PGO allows optimization by taking into account real benchmark data instead of heuristic data.

Table 1. Optimization options (Intel C compiler 11.1)

A block-driven processing approach was also tested as in the shared memory implementation in [10]. Extra communication and block-driven implementation are also suitable in a process-level parallel implementation when the memory of the processing node is not enough for the application ([17],[9]).

5. Test Results

Platform

The results have been obtained in a small commodity cluster of three nodes connected through a Gigabits Ethernet switch. Each node has 6 GB of memory and a Core i7 CPU 920 (launch date: fourth quarter of 2008). The Core i7 920 has four cores of 2.667 GHz (two threads per core if Hyper-Threading is active), L3 cache of 8MB shared by all the cores, and QuickPath of 4.8 GT/s. The cluster price was of 3,000 € (first quarter of 2009) approximately with all the components, including the cabinet. It runs Linux Fedora 10 (kernel 2.6.27.41). Cluster communication system has a bandwidth of 113 MB/s, near to the theoretic 125 MB/s.

The program is compiled using Intel Fortran 11.1 compiler. The OpenMP of this compiler is used for the shared memory implementation and MPICH-1.3 for the MPI message passing implementation. The source-code versions implemented were compiled using options that drive classic optimizations and vectorizations. Table 1 summarizes the optimization options checked. Similar execution times are obtained with O2 and O3. When the options ipo and/or SSE4.2 are added to O2 or O3 performance does not improve. PGO does not improve the execution time compared to a version with the same optimization options but without PGO. The executables used in this section have been obtained with O2 and openmp compiler options.

Practical Application

The test application is a simulation of the currents in Lake Tahoe. The ultimate goal of these simulations is to characterize the pathways of transport of young life stages of an invasive species (the bivalve *Corbicula fluminea*, or Asian clam)

 ISBN 13: 978-84-613-5510-5

from the existing near-shore beds to other sites in the lake and the environmental conditions they would be exposed to en route. Given that $O(10^2)$ m (hundreds) features of the velocity fields, characteristics of nearshore regions, should be resolved in Lake Tahoe, the computational grid cells should have horizontal dimensions of at least O(10) m (tens). Simulating a lake of the size of Lake Tahoe (roughly 20 km x 30 km) with O(10)m horizontal size cell columns, poses a serious computational problem which can only be addressed through the use of parallel computers. For example, the ratio of real to computational time in simulations conducted with 50 m wide grid cells in a single core of the cluster is approximately 1/1. The simulations presented here are conducted in grids with 95 layers of variable thickness and squared columns of 50 m x 50 m in the horizontal. The grid includes 14,654,639 computational cells in 197,781 columns.

Performance of different platform configurations

This work analyzes the influence in performance of the multiple cores in a node, the prefetching hardware, the Intel Hyper-Threading technology, and the Intel SpeedStep and Turbo Mode technology.

Hardware prefetcher monitors data access patterns and prefetches data automatically into processor caches. Core i7 cores can track 16 forward streams and 4 backward streams each. Simultaneous multithreading allows the execution of multiple threads in a core; in particular, two threads with Intel Hyper-Threading. Intel SpeedStepTechnology allows the operating system to control the core speed. Intel Turbo Mode Technology allows processor cores to run faster than the assigned frequency under specific conditions.

Table 2 shows the seconds per iteration obtained for different platform configurations and different number of processes and threads. The narrow-direction distribution and the MPI redundant operation version have been used. The communication time due to data distribution or collection is not included because it does not depend on the number of iterations. Up to four threads are used to each node; a higher number of threads makes performance worst despite of Hyper-Threading being enabled. The column HSTP shows the results for the default configuration. In the default configuration the BIOS and the operating system have enabled Hyper-Threading (H), SpeedSteep (S) and Turbo Mode (T), and prefetching hardware (P). In particular, *ondemand* is the default *CPUfreq governor* of the cluster operating system, which means the governor sets the frequency depending on the current usage, between a minimum of 1.6 GHz and a maximum of 2.667 GHz, last one can increase due to Turbo Mode. The time in the default configuration is less reproducible due to the thread distribution of the operating system among the eight logical cores of a node. If Hyper-Threading is disabled (column -STP) performance improves, but if either SpeedStep/Turbo Mode (column ---P) or Prefetching (column -ST-) are also disabled, time increases slightly. The results in the columns -STP and ---P show an increment in the clock frequency due to the Turbo Mode. The results in the columns -STP and -ST- suggest that the prefetching hardware is being weakly used.

 ISBN 13: 978-84-613-5510-5

Tahoe 50mx50m		Platform configurations			
No. Processes	No. Threads	HSTP	-STP	---P	-ST-
1	1	21.1	**21.15**	21.92	22.54
1	2	11.07	11.1	11.56	11.79
1	3	7.77	**7.73**	8.07	8.22
1	4	9.75	**6.12**	6.12	6.51
2	1	10.96	10.96	11.45	11.65
2	2	5.96	6.05	6.21	6.38
2	3	6.75	**4.27**	4.48	4.56
2	4	5.15	**3.42**	3.57	3.68
3	1	7.62	**7.65**	7.9	8.04
3	2	4.23	**4.26**	4.4	4.48
3	3	4.81	**3.1**	3.21	3.29
3	4	3.65	**2.51**	2.68	2.71

Table 2. Performance of different platform configurations (seconds per iteration). In HSTP, H means Hyper-Threading enable, S means SpeedStep enable, T means Turbo enable, and P mean Prefeching enable. " -" means Disable

Block-driven processing was added to try to reduce cache miss by facilitating data locality. It reduces the execution time by 4% with horizontal cell size of 100 m x 100 m and one process with four threads. Block processing improves only marginally this implementation's performance, although it never makes performance worst as it was observed in the block processing implementation of [10]. The results in [10] are obtained in a platform of IBM with Power5+ 1.9 GHz. For a grid of 1024x1024x10 (=10,485,760 cells), from 1 to 8 processors block-driven processing makes performance worst but from 12 to 16, the maximum number of processors tested, the block-driven implementation improves performance [10].

The results presented in the next subsections are obtained with the configuration –STP and *ondemand* as the *CPUfreq governor.*

Comparison of wide-direction and narrow-direction distributions in both MPI versions, with and without redundant operations

Both, wide-direction and narrow-direction distributions, have the same number of communication in both MPI versions, but they are of different sizes. Also, in the MPI version with redundant operations, the wide-direction distribution has more redundant operations than the narrow-direction approach (because it has larger border length).

Table 3 shows the execution time per iteration and speedup for both wide and narrow-direction distribution and both the MPI implementation with redundant operations (R) and the MPI approach with non-redundant operations (NR). As can be observed narrow-cut distribution also improves sequential execution time. The best approach is to use the MPI implementation with redundant operations and the narrow-cut distribution. Speedup improves more with the narrow-cut approach because this approach has lesser border length than the wide-cut approach; the border size is decreased a 25% approximately.

 ISBN 13: 978-84-613-5510-5

6. Conclusion

This work discusses the performance of several thread- and process- level implementations of a semi-implicit 3-D lake hydrodynamic model (SI3D) and the influence of different platform configurations and domain decompositions. It has been found that:

Tahoe 50m		Sec./iteration				Speedup			
No. Pr.	No. Th	Narrow		Wide		Narrow		Wide	
		R	NR	R	NR	R	NR	R	NR
1	1	21.1	21.1	21.32	21.32	1	1	1	1
1	2	11.1	11.1	11.34	11.34	1.9	1.9	1.88	1.88
1	3	7.73	7.73	7.97	7.97	2.73	2.73	2.68	2.68
1	4	6.12	6.12	6.29	6.29	3.45	3.45	3.39	3.39
2	1	**10.96**	11.18	11.21	11.51	**1.93**	1.89	1.9	1.85
2	2	**6.05**	6.21	6.19	6.43	**3.49**	3.41	3.44	3.32
2	3	**4.27**	4.41	4.51	4.76	**4.94**	4.8	4.73	4.48
2	4	**3.42**	3.58	3.61	3.79	**6.17**	5.91	5.91	5.63
3	1	**7.65**	7.85	7.88	8	**2.76**	2.69	2.71	2.67
3	2	**4.26**	4.48	4.43	4.66	**4.95**	4.72	4.81	4.58
3	3	**3.1**	3.31	3.34	3.52	**6.81**	6.39	6.38	6.06
3	4	**2.51**	2.69	2.77	3.02	**8.41**	7.86	7.7	7.06

Table 3. Wide and narrow distributions in both MPI versions: with (R) and without (NR) redundant operations

- The program makes a weak use of the prefetching hardware (prefetching decreases execution time by between %5 to %8) and obtains little improvements by using block-driven processing (%4 improvement approximately).
- Intel® Turbo Mode Technology decreases slightly the execution time (by between %3 to 7%).
- Performance is worse if the default BIOS and operating system configuration is used (time increases by between %40 to 60%, depending on the number of processes and threads). This is due to the thread distribution of the operating system among the eight logical cores of a node when Hyper-Threading is enabled. Thread affinity could be used to avoid this problem instead of disable Hyper-Threading.
- Block-driven processing reduces execution time too slightly.
- Process level implementation reduces execution time using overlapping sub-domains (redundant operations).
- With the best parallel implementation and performance configuration, and with narrow-cut domain decomposition the simulation of 24 hours with 50mx50m cell columns in a core of the cluster requires proximately 6 hours

 ISBN 13: 978-84-613-5510-5

with one processor (4 threads) instead of 20 hours and 30 minutes (with 1 threads) and approximately 2 hours and 30 minutes with the three processors (12 threads).

Acknowledgment

This work was partially funded by the project 'Risk Assessment of Asian clam expansion and potential environmental impacts to Lake Tahoe. Task: Larval Transport Modelling' funded by Pacific Southwest Research Station, USDA Forest Service.

References

[1] B. R. Hodges, J. Imberger, A. Saggio and K. B. Winters, "Modeling Basin-Scale Internal Waves in a Stratified Lake," *Limnology and Oceanography,* vol. 45, pp. 1603-1620, Nov., 2000, 2000.

[2] F. J. Rueda, S. G. Schladow and S. Ó. Pálmarsson, "Basin-scale internal wave dynamics during a winter cooling period in a large lake," *J. Geophys. Res.,* vol. 108, pp. 3097, 03/27, 2003.

[3] F. Rueda, J. Vidal and G. Schladow, "Modeling the effect of size reduction on the stratification of a large wind-driven lake using an uncertainty-based approach," *Water Resour. Res.,* vol. 45, pp. W03411, 03/14, 2009.

[4] P. E. Smith, *A Semi-Implicit, Three-Dimensional Model of Estuarine Circulation.* Sacramento, California: 2006.

[5] F. J. Rueda and E. A. Cowen, "The residence time of a freshwater embayment connected to a large lake," *Limnol. Oceanogr.,* vol. 50, pp. 1638-1653, 2005.

[6] V. Casulli and R. T. Cheng, "Semi-implicit finite difference methods for three-dimensional shallow water flow," *Int. J. Numer. Methods Fluids,* vol. 15, pp. 629-648, 1992.

[7] L. H. Kantha and C. A. Clayson, "An improved mixed layer model for geophysical applications," *J. Geophys. Res.,* vol. 99, pp. 25235-25266, 1994.

[8] P. Rao, "A parallel hydrodynamic model for shallow water equations," *Applied Mathematics and Computation,* vol. 150, pp. 291-302, 2/27, 2004.

[9] M. J. Castro, J. A. García-Rodríguez, J. M. González-Vida and C. Parés, "A parallel 2d finite volume scheme for solving systems of balance laws with nonconservative products: Application to shallow flows," *Comput. Methods Appl. Mech. Eng.,* vol. 195, pp. 2788-2815, 4/1, 2006.

 ISBN 13: 978-84-613-5510-5

[10] K. R. Tubbs and F. T. C. Tsai, "Multilayer shallow water flow using lattice Boltzmann method with high performance computing," *Adv. Water Resour.*, vol. 32, pp. 1767-1776, 12, 2009.

[11] O. Nesterov, "A simple parallelization technique with MPI for ocean circulation models," *Journal of Parallel and Distributed Computing*, vol. 70, pp. 35-44, 1, 2010.

[12] D. van Dyk, M. Geveler, S. Mallach, D. Ribbrock, D. Göddeke and C. Gutwenger, "HONEI: A collection of libraries for numerical computations targeting multiple processor architectures," *Comput. Phys. Commun.*, vol. 180, pp. 2534-2543, 12, 2009.

[13] M. J. Castro, J. A. García-Rodríguez, J. M. González-Vida and C. Parés, "Solving shallow-water systems in 2D domains using Finite Volume methods and multimedia SSE instructions," *J. Comput. Appl. Math.*, vol. 221, pp. 16-32, 2008.

[14] M. de la Asunci, J. Mantas and M. Castro, "Simulation of one-layer shallow water systems on multicore and CUDA architectures," *The Journal of Supercomputing*, online 10 March 2010, 2010.

[15] O. Nesterov, "A simple parallelization technique with MPI for ocean circulation models," *Journal of Parallel and Distributed Computing*, vol. 70, pp. 35-44, 1, 2010.

[16] P. Messina, D. Culler, W. Pfeiffer, W. Martin, J. T. Oden and G. Smith, "Architecture," *Commun ACM*, vol. 41, pp. 36-44, 1998.

[17] L. Paglieri, D. Ambrosi, L. Formaggia, A. Quarteroni and A. L. Scheinine, "Parallel computation for shallow water flow: A domain decomposition approach," *Parallel Computing*, vol. 23, pp. 1261-1277, 9, 1997.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE2010 Almeria, Spain, 27-30 June 2010

On the Visualization of Honeypot Data through Projection Techniques

Álvaro Alonso[1], Santiago Porras[1], Iñaki Garitano[2], Ignacio Arenaza[2], Roberto Uribeetxeberria[2], Urko Zurutuza[2], Álvaro Herrero[1] and Emilio Corchado[3]

[1] *Department of Civil Engineering, University of Burgos, Burgos (Spain)*

[2] *Electronics and Computing Department, Mondragon University, Arrasate-Mondragon (Spain)*

[3] *Departamento de Informática y Automática, Universidad de Salamanca, Salamanca (Spain)*

emails: aad0038@alu.ubu.es, spa0001@alu.ubu.es, igaritano@eps.mondragon.edu, iarenaza@eps.mondragon.edu, ruruibeetxeberria@eps.mondragon.edu, uzurutuza@eps.mondragon.edu, escorchado@usal.es, ahcosio@ubu.es

Abstract

A crucial aspect in network monitoring for security purposes is the visual inspection of traffic patterns, which chiefly provides the network manager with a synthetic and intuitive representation of the current situation. In keeping with this idea, neural projection techniques can adaptively map high-dimensional data into a low-dimensional space, for the user-friendly visualization of data collected by different security tools. Different projection methods for the visual inspection of honeypot data are applied in this study, which may be seen as a complementary network security tool that sheds light on internal data structures through visual inspection. Empirical verification of the proposed projection methods was performed in an experimental domain where 1-month data sets were captured and stored for analysis. Experiments showed that whereas an Intrusion Detection System may only identify a low percentage of the malicious traffic, a deeper understanding of attack patterns could easily be gained by means of visual inspections.

 ISBN 13: 978-84-613-5510-5

Keywords: Projection Models, Artificial Neural Networks, Unsupervised Learning, Network & Computer Security, Intrusion Detection, Honeypots.

1. Introduction

A network attack or intrusion will inevitably violate one of the three computer security principles -availability, integrity and confidentiality- by exploiting certain vulnerabilities such as Denial of Service, Modification and Destruction [1]. One of the most harmful issues of attacks and intrusions, which increases the difficulty of protecting computer systems, is precisely the ever-changing nature of attack technologies and strategies.

For that reason alone, among others, Intrusion Detection Systems (IDSs) have become a very necessary asset in addition to the computer security infrastructure of most organizations. In the context of computer networks, an IDS can roughly be defined as a tool designed to detect suspicious patterns that may be related to a network or system attack. Intrusion Detection (ID) is therefore a field that focuses on the identification of attempted or ongoing attacks on a computer system (Host IDS - HIDS) or network (Network IDS - NIDS).

Visual inspection of traffic patterns is an alternative and crucial aspect in network monitoring [2]. Visualization is a critical issue in the computer network defence environment, which chiefly serves to generate a synthetic and intuitive representation of the current situation for the network manager; as a result, several research initiatives have recently applied information visualization to this challenging task [3] [4] [5] [6]. Visualization techniques typically aim to make the available statistics supplied by traffic-monitoring systems more understandable in an interactive way. They therefore focus on traffic data as well as on network topology. Regardless of their specific characteristics, these methods all map high-dimensional feature data into a low-dimensional space for presentation purposes. The baseline of the research presented in this study is that Artificial Neural Networks (ANNs), in general, and unsupervised connectionist models [7, 8], in particular, can prove quite adequate for the purpose of network data visualization through dimensionality reduction. As a result, unsupervised projection models are applied in the present research for the visualization and subsequent analysis of Honeypot data.

The remaining five sections of this study are structured as follows: section 2 contains a brief description of Intrusion Detection (mainly visualization-based). Section 3 presents the approach proposed for ID and the neural projection techniques applied in this work. Some experimental results are presented and described in section 4; the conclusions of this study are discussed in section 5, as well as future work.

 ISBN 13: 978-84-613-5510-5

2. Intrusion Detection and Honeynets

The accurate detection in real-time of computer and network system intrusions has always been an interesting and intriguing problem for system administrators and information security researchers. It may be attributed on the whole to the dynamic nature of systems and networks, the creativity of attackers, the wide range of computer hardware and operating systems and so on. Such complexity arises when dealing with distributed network-based systems and insecure networks such as the Internet.

A honeypot has no authorised function or productive value within the corporate network other than to be explored, attacked or compromised [9]. Thus, a honeypot should not receive any traffic at all. Any connection attempt with a honeypot is then an attack or attempt to compromise the device or services that it is offering-is by default illegitimate traffic. From the security point of view, there is a great deal that may be learnt from a honeypot about a hacker's tools and methods in order to improve the protection of information systems.

One of the most extended classifications of honeypots takes into account their level of interaction. Low interaction honeypots offer limited interaction with attackers and the most common ones only simulate services and operating systems. High interaction honeypots follow a different strategy: instead of using simulated services and operating systems, real systems and applications are used, usually running in virtual machines.

Somewhere between the two are medium interaction honeypots, which also emulate vulnerable services, but leave the operating system to manage the connections with their network protocol stack. Recently, a new type of honeypot has been proposed as a response to the behavioural change observed in the attackers. Instead of waiting for the attackers to reach traditional honeypots, client side honeypots, also known as honeyclients, scan communication channels looking for malware.

In a honeynet, all the traffic received by the sensors is suspicious by default. Thus every packet should be considered as an attack or at least as a piece of a multi-step attack. Numerous studies propose the use of honeypots to detect automatic large scale attacks; honeyd [10] and nepenthes [11] among others. The first Internet traffic monitors known as Network Telescopes, Black Holes or Internet Sinks were presented by Moore *et al*. [12].

3. A Visualization-based Approach

This work proposes the application of projection models for the visualization of Honeypot data. Visualisation techniques have been applied to massive datasets, such as those generated by honeynets, for many years. These techniques are considered a viable approach to information seeking, as humans are able to recognize different features and to detect anomalies by inspecting graphs [13]. The underlying operational assumption of the proposed approach is mainly grounded in the ability to render the high-dimensional traffic data in a consistent

 ISBN 13: 978-84-613-5510-5

yet low-dimensional representation. So, security visualisation tools have to map high-dimensional feature data into a low-dimensional space for presentation. One of the main assumptions of the research presented in this paper is that neural projection models will prove themselves to be satisfactory for the purpose of security data visualisation through dimensionality reduction.

This problem of identifying patterns that exist across dimensional boundaries in high dimensional datasets is a challenging task. Such patterns may become visible if changes are made to the spatial coordinates. However, an *a priori* decision as to which parameters will reveal most patterns requires prior knowledge of unknown patterns.

Projection methods project high-dimensional data points onto a lower dimensional space in order to identify "interesting" directions in terms of any specific index or projection. Having identified the most interesting projections, the data are then projected onto a lower dimensional subspace plotted in two or three dimensions, which makes it possible to examine the structure with the naked eye. Projection methods can be smart compression tools that map raw, high-dimensional data onto two or three dimensional spaces for subsequent graphical display. By doing so, the structure that is identified through a multivariable dataset may be visually analysed with greater ease.

Visualisation tools can therefore support security tasks in the following way:

- Visualisation tools may be understood intuitively (even by inexperienced staff) and require less configuration time than more conventional tools.
- Providing an intuitive visualisation of data allows inexperienced security staff to learn more about standard network behaviour, which is a key issue in ID [14]. The monitoring task can be then assigned to less experienced security staff.
- As stated in [3], "*visualizations that depict patterns in massive amounts of data, and methods for interacting with those visualizations can help analysts prepare for unforeseen events*". Hence, such tools can also be used in security training.
- They can work in unison with some other security tools in a complementary way.

As with other machine learning paradigms, an interesting facet of ANN learning is not just that the input patterns may be precisely learned/classified/identified, but that this learning can be generalised. Whereas learning takes place within a set of training patterns, an important property of the learning process is that the network can generalise its results on a set of test patterns that were not previously learnt. The identification of unknown patterns fits the 0-day attack [15] detection.

Due to the aforementioned reasons, the present study approaches the analysis of honeynet data from a visualization standpoint. That is, some neural projection techniques are applied for the visualization of such data. The different projection models applied in this study are described in the following sections.

 ISBN 13: 978-84-613-5510-5

3.1 Principal Component Analysis

Principal Component Analysis (PCA) is a statistical model, introduced in [16] and independently in [17], that describes the variation in a set of multivariate data in terms of a set of uncorrelated variables each, of which is a linear combination of the original variables.

Its goal is to derive new variables, in decreasing order of importance, that are linear combinations of the original variables and are uncorrelated with each other. From a geometrical point of view, this goal mainly consists of a rotation of the axes of the original coordinate system to a new set of orthogonal axes that are ordered in terms of the amount of variance of the original data they account for. The optimal projection given by PCA from an N-dimensional to an M-dimensional space is the subspace spanned by the M eigenvectors with the largest eigenvalues.

According to [18], it is possible to describe PCA as a mapping of vectors $\mathbf{x^d}$ in an N-dimensional input space $(x_1,...,x_N)$ onto vectors $\mathbf{y^d}$ in an M-dimensional output space $(y_1,...,y_M)$, where $M \leq N$. $\mathbf{x}$ may be represented as a linear combination of a set of N orthonormal vectors W_i:

$$\mathbf{x} = \sum_{i=1}^{N} y_i W_i \qquad (1)$$

Vectors W_i satisfy the orthonormality relation:

$$W_i^t W_j = \delta_{ij} \qquad (2)$$

where δ_{ij} is the Kronecker delta.

Making use of equation (1), the coefficients y_i may be given by

$$y_i = W_i^T \mathbf{x} \qquad (3)$$

which can be regarded as a simple rotation of the co-ordinate system from the original $\mathbf{x}$ values to a new set of co-ordinates given by the $\mathbf{y}$ values. If only one subset $M < N$ of the basis vectors, W_i, is retained so that only M coefficients y_i are used, and having replaced the remaining coefficients by constants b_i, then each $\mathbf{x}$ vector may be approximated by the following expression:

$$\tilde{\mathbf{x}} = \sum_{i=1}^{M} y_i W_i + \sum_{i=M+1}^{N} b_i W_i \qquad (4)$$

Consider the whole dataset of D vectors, $\mathbf{x^d}$ where $d = 1,...,D$.

PCA can be performed by means of ANNs or connectionist models such as [19, 20, 21, 22, 23]. It should be noted that even if we are able to characterize the data with a few variables, it does not follow that an interpretation will ensue.

 ISBN 13: 978-84-613-5510-5

3.2 Cooperative Maximum Likelihood Hebbian Learning

The Cooperative Maximum Likelihood Hebbian Learning (CMLHL) model [24] extends the Maximum Likelihood Hebbian Learning (MLHL) [25] model, which is based on Exploratory Projection Pursuit (EPP) [26]. The statistical method of EPP was designed for solving the complex problem of identifying structure in high dimensional data by projecting it onto a lower dimensional subspace in which its structure is searched for by eye. To that end, an "index" must be defined to measure the varying degrees of interest associated with each projection. Subsequently, the data is transformed by maximizing the index and the associated interest. From a statistical point of view the most interesting directions are those that are as non-Gaussian as possible.

The MLHL model is based on the Negative Feedback Network and, as the AABP model; it associates an input vector, $\boldsymbol{x} \in \Re^D$, with an output vector, $\boldsymbol{y} \in \Re^Q$. In this case, the output of the network ($\boldsymbol{y}$) is computed as:

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \qquad (\mathbf{5})$$

where, W_{ij} is the weight linking input j to output i.

Once the output of the network has been calculated, the activation (e_j) is fed back through the same weights and subtracted from the input:

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j \qquad (\mathbf{6})$$

Finally, the learning rule determines the way in which the weights are updated:

$$\Delta W_{ij} = \eta . y_i . sign(e_j) | e_j |^{p-1} \qquad (\mathbf{7})$$

where, η is the learning rate and p is a parameter related to the energy function.

The main difference between the basic MLHL model and its Cooperative version is the introduction of lateral connections. After the Feed forward step (Eq. 5) and before the Feed back step (Eq. 6), lateral connections between the output neurons are applied as follows:

$$y_i(t+1) = \left[y_i(t) + \tau(b - Ay) \right]^+ \qquad (\mathbf{8})$$

where, τ is the "strength" of the lateral connections, b is the bias parameter and A is a symmetric matrix used to modify the response to the data. Its effect is based on the relation between the distances among the output neurons.

4. Experiments and Results

The Euskalert project [26] has deployed a network of honeypots in the Basque Country (northern Spain) where eight companies and institutions have installed

 ISBN 13: 978-84-613-5510-5

one of the project's sensors behind the firewalls of their corporate networks. The honeypot sensor transmits all the traffic received to a database via a secure communication channel. These partners can consult information relative to their sensor (after a login process) as well as general statistics in the project's website. Once the system is fully established, the information available can be used to analyse attacks suffered by the honeynet at network and application level. Euskalert is a distributed honeypot network based on a Honeynet GenIII architecture [26].

This honeypot system receives 4000 packets a day on average. All the traffic is analyzed by the Snort IDS, and an alert is launched whenever the packet matches a known attack signature. For this experiment, we have analysed the logs coming from Euskalert and Snort gathered during February 2010. Fig. 1 shows the traffic volume in terms of number of packets received for that period of time.

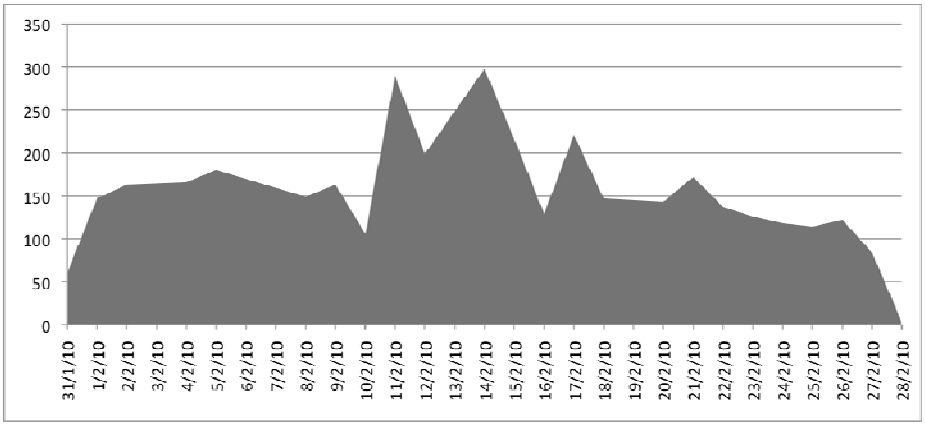


Fig. 1. Temporal distribution of the traffic volume in terms of number of packets captured by Euskalert in February, 2010.

The February 2010 dataset contains a total of 3798 packets, including TCP, UDP and ICMP traffic received by the distributed honeypot sensors. The characterization of the traffic in the dataset is shown in Table 1. The table shows which alerts have been triggered in that period of time and their percentage. Those signatures starting with "Wormledge" are automatically generated and not present in the default signature database.

From this dataset, it may be said that a misuse detection-based IDS such as Snort is only capable of identifying about 10.38% of bad-intentioned traffic. Furthermore, it was demonstrated that only 2% of the unsolicited traffic was identified by the IDS when automatically generated signatures were included from a previous work [27]. Thus, a deeper analysis of the data is needed in order to discover the internal structure of the remaining 90% of the traffic. Explaining the behaviour of the unknown traffic is a difficult task that must be performed to better protect computer networks and systems.

 ISBN 13: 978-84-613-5510-5

Signature	# Packets	%
Unknown Traffic	3404	89,62
BLEEDING-EDGE POLICY Reserved IP Space Traffic - Bogon Nets 2	127	3,34
BLEEDING-EDGE WORM Allaple ICMP Sweep Ping Inbound	58	1,52
ICMP PING	75	1,97
Wormledge, microsoft-ds, smb directory packet (port 445). SMBr...PC NETWORK PROGRAM 1.0...LANMAN1.0...Windows for Workgroups 3.1a...LM1.2X002...LANMAN2.1...NT LM 0.12 . Created on 2007-08-07	34	0,89
Wormledge, KRPC Protocol (Kademlia RPC), BitTorrent information exchange:ping query. Created on 2007-08-07	11	0,28
Wormledge, NetBios Session Service (port 139). Payload CKFDENECFDEFFCFGAAAAAAAAAAAAAAAAA. Created on 2007-08-07	7	0,18
Wormledge, NetBios Name Query (udp port 137). Payload CKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA. Created on 2007-08-07	7	0,18
Wormledge, Microsoft RPC Service, dce endpoint resoluction (port 135). Created on 2007-08-07	7	0,18
WEB-IIS view source via translate header	6	0,15
BLEEDING-EDGE SCAN LibSSH Based SSH Connection - Often used as a BruteForce Tool	5	0,13

Table 1. Characterization of data traffic captured by Euskalert, in February, 2010.

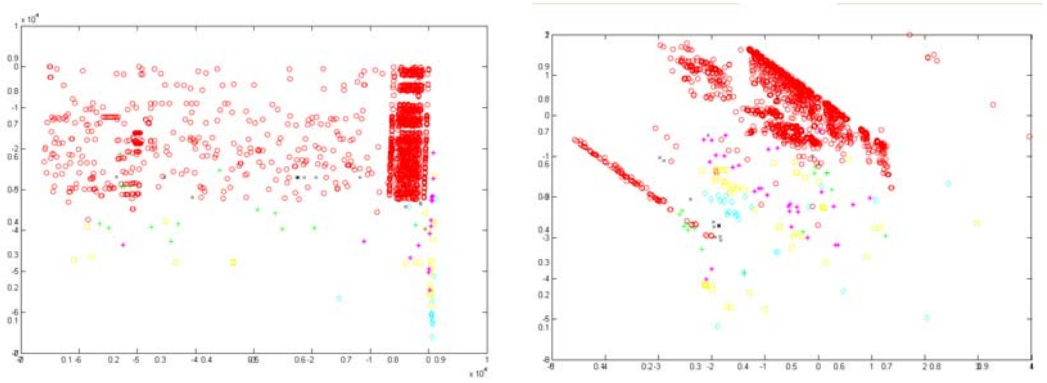
The following features were extracted from each one of the records in the dataset:

- **Time**: the time when the attack was detected. Difference in relation to the first attack in the dataset (in minutes).
- **Protocol**: whether TCP, UDP or ICMP (codified as three binary features).
- **Ip_len**: number of bytes in the packet.
- **Source Port**: number of the port from which the source host sent the packet. In ICMP protocol, this represents the ICMP type field.
- **Destination Port**: destination host port number to which the packet is sent. In the ICMP protocol, this represents the ICMP type field.
- **Flags**: Control bits of a TCP packet, which contains 8 1 bit values.

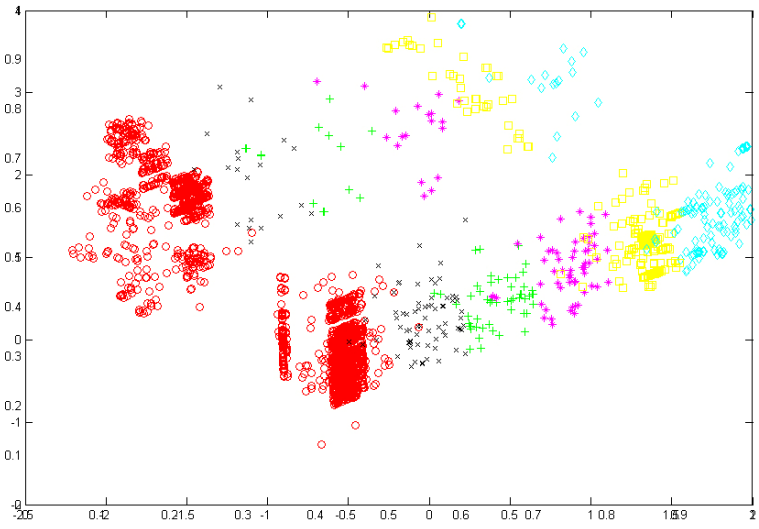
The previously introduced projection techniques were applied to this dataset, generating the projections shown in Fig. 2. In these projections, the data are

 ISBN 13: 978-84-613-5510-5

depicted with different colors and shapes, taking into account the destination port; from 3 to 10371: red circles, from 10371 to 20739: black crosses, from 20739 to 31107: green pluses, from 31107 to 41475: magenta stars, from 41475 to 51843: yellow squares, and from 51843 to 62205: cyan diamonds.



a) PCA projection b) MLHL projection



c) CMLHL projection

Fig. 2. Projections of data traffic captured by Euskalert, in February, 2010.

5. Conclusions and Future Work

From the projections in Fig. 2 we can conclude that CMLHL provides a more sparse representation that the other two methods. This enables the intuitive visualization of the honeynet, where the general structure of these data can be seen. After getting a general idea of the dataset structure, an in-deep analysis was

 ISBN 13: 978-84-613-5510-5

carried out to comprehensively analysed each one of the points in the groups identified by CMLHL. As a result, the following conclusions can be stated for each one of the destination ports in the analysed dataset:

- 8: ICMP ping, used for probing the Internet, looking for victim hosts.
- 22: SSH. It seems to be a traffic flow with many packets coming from one source to one of the honeypot. They correspond to connection attempts by attackers or infected machines.
- 80: HTTP. Attackers try different vulnerabilities against web applications.
- 135: DCE endpoint resolution, used by Microsoft for Remote Procedure Call protocol. It has always been and still is one of the most exploited services by virus and worms.
- 139: NETBIOS Session Service. Plenty of attacks to this Microsoft Windows service can be found.
- 443: HTTP protocol over TLS SSL connection attempts.
- 445: SMB directly over IP. As most of the traffic in the biggest group identified by CMLHL is aimed at this destination port, we can conclude that this is a widely exploited service.
- 1433: Microsoft-SQL-Server, used by the old SQL Slammer worm.
- 1521: Oracle TNS Listener. It seems that attackers try to connect to the honeypot via Oracle service.
- 2967: Symantec System Center. Vulnerabilities have been found on Symantec service, and it is being expiated in the wild.
- 3128: Proxy Server // Reverse WWW Tunnel Backdoor, where the MyDoom worm operates.
- 3389: MS Terminal Services, used for Remote Desktop.
- 4444: This port is a common return port for the rpc dcom.c buffer overflow vulnerability and for the msblast rpc worm.
- 4899: Remote Administrator default port. There is a known remote exploitable vulnerability in radmin server versions 2.0 and 2.1 that allows code execution.
- 5061: SIP-TLS. Used for VoIP communications.
- 5900: Virtual Network Computer or VNC, used also as a remote desktop solution.
- Port 8080: HTTP Alternate, used as an HTTP proxy.
- Port 19765: Used in Kademlia (Bittorrent protocol).

Future work will combine the honeypot data with the output of a signature-based IDS, such as Snort, in the same visualization. This will validate the proposed approach as a complementary tool that can be combined with some other security tools or IDSs.

Acknowledgments

This research has been partially supported through the Regional Government of Castilla y León under Project BU006A08, the Department of Research, Education and Universities of the Basque Government; and the Spanish Ministry of Science

 ISBN 13: 978-84-613-5510-5

and Innovation (MICINN) under projects CIT-020000-2008-2 and CIT-020000-2009-12. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolin Ingenieria S.A., within the framework of the MAGNO2008 – 1028.- CENIT Project also funded by the MICINN.

References

[1] Myerson, J.M., Identifying Enterprise Network Vulnerabilities. International Journal of Network Management 12-3 (2002) 135-144.

[2] Becker, R.A., Eick, S.G., Wilks, A.R., Visualizing Network Data. IEEE Transactions on Visualization and Computer Graphics 1-1 (1995) 16-28.

[3] D'Amico, A.D., Goodall, J.R., Tesone, D.R., Kopylec, J.K., Visual Discovery in Computer Network Defense. IEEE Computer Graphics and Applications 27-5 (2007) 20-27.

[4] Goodall, J.R., Lutters, W.G., Rheingans, P., Komlodi, A., Focusing on Context in Network Traffic Analysis. IEEE Computer Graphics and Applications 26-2 (2006) 72-80.

[5] Itoh, T., Takakura, H., Sawada, A., Koyamada, K., Hierarchical Visualization of Network Intrusion Detection Data. IEEE Computer Graphics and Applications 26-2 (2006) 40-47.

[6] Livnat, Y., Agutter, J., Moon, S., Erbacher, R.F., Foresti, S., A Visualization Paradigm for Network Intrusion Detection. Proceedings of the Sixth Annual IEEE SMC Information Assurance Workshop, 2005. IAW '05 - (2005) 92-99.

[7] Herrero, Á., Corchado, E., Gastaldo, P., Zunino, R., Neural Projection Techniques for the Visual Inspection of Network Traffic. Neurocomputing 72-16-18 (2009) 3649-3658.

[8] Herrero, Á., Corchado, E., Pellicer, M.A., Abraham, A., MOVIH-IDS: A Mobile-Visualization Hybrid Intrusion Detection System. Neurocomputing 72-13-15 (2009) 2775-2784.

[9] Charles, K.A., Decoy Systems: A New Player in Network Security and Computer Incident Response. International Journal of Digital Evidence 2-3 (2004)

[10] Provos, N., A Virtual Honeypot Framework. Proceedings of the 13th USENIX Security Symposium 132 - (2004)

[11] Baecher, P., Koetter, M., Holz, T., Dornseif, M., Freiling, F., The Nepenthes Platform: An Efficient Approach to Collect Malware. Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection (RAID 2006). LNCS 4219 -. Springer Berlin / Heidelberg (2006) 165-184.

[12] Moore, D., Shannon, C., Brown, D.J., Voelker, G.M., Savage, S., Inferring Internet Denial-of-service Activity. ACM Transactions on Computer Systems 24-2 (2006) 115-139.

[13] Ahlberg, C., Shneiderman, B., Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In: Readings

in Information Visualization: using Vision to Think, pp. 244-250. Morgan Kaufmann Publishers Inc. (1999).

[14] Goodall, J.R., Lutters, W.G., Rheingans, P., Komlodi, A., Preserving the Big Picture: Visual Network Traffic Analysis with TNV. Proceedings of the IEEE Workshop on Visualization for Computer Security (VizSEC 05) -. IEEE Computer Society (2005) 47-54.

[15] Laskov, P., Dussel, P., Schafer, C., Rieck, K., Learning Intrusion Detection: Supervised or Unsupervised? Proceedings of the 13th International Conference on Image Analysis and Processing (ICIAP 2005). LNCS 3617 -. Springer, Heidelberg (2005) 50-57.

[16] Pearson, K., On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine 2-6 (1901) 559-572.

[17] Hotelling, H., Analysis of a Complex of Statistical Variables into Principal Components. Journal of Education Psychology 24- (1933) 417-444.

[18] Bishop, C.M., Neural Networks for Pattern Recognition. Oxford University Press, 1996.

[19] Oja, E., Neural Networks, Principal Components, and Subspaces. International Journal of Neural Systems 1- (1989) 61-68.

[20] Oja, E., Principal Components, Minor Components, and Linear Neural Networks. Neural Networks 5-6 (1992) 927-935.

[21] Oja, E., A Simplified Neuron Model as a Principal Component Analyzer. Journal of Mathematical Biology 15-3 (1982) 267-273.

[22] Sanger, D., Contribution Analysis: a Technique for Assigning Responsibilities to Hidden Units in Connectionist Networks. Connection Science 1-2 (1989) 115-138.

[23] Fyfe, C., A Neural Network for PCA and Beyond. Neural Processing Letters 6-1-2 (1997) 33-41.

[24] Corchado, E., Fyfe, C., Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. International Journal of Pattern Recognition and Artificial Intelligence 17-8 (2003) 1447-1466.

[25] Fyfe, C., Corchado, E., Maximum Likelihood Hebbian Rules. Proceedings of the 10th European Symposium on Artificial Neural Networks (ESANN 2002) - (2002) 143-148.

[26] Friedman, J.H., Tukey, J.W., A Projection Pursuit Algorithm for Exploratory Data-Analysis. IEEE Transactions on Computers 23-9 (1974) 881-890.

[27] Zurutuza, U., Uribeetxeberria, R., Zamboni, D., A Data Mining Approach for Analysis of Worm Activity through Automatic Signature Generation. Proceedings of the 1st ACM Workshop on AISec -. ACM (2008) 61-70.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE2010 Almeria, Spain, 27-30 June 2010

A Study of Meteorological Conditions by means of Soft Computing Models

Ángel Arroyo[1], Verónica Tricio[2], Álvaro Alonso[1], Santiago Porras[1] and Emilio Corchado[3]

[1] *Department of Civil Engineering, University of Burgos, Burgos, Spain*
[2]*Department of Physics, University of Burgos, Burgos, Spain*
[3]*Department of Computer Science and Automatic, University of Salamanca, Salamanca, Spain*

emails: aarroyop@ubu.es, vtricio@ubu.es, aad0038@alu.ubu.es, spa0001@alu.ubu.es, escorchado@usal.es

Abstract

In this interdisciplinary study, soft computing models are used to identify typical days in terms of their meteorological conditions. Meteorological and pollution data were taken from a pollution measurement station in the Spanish Autonomous Region of Castile-Leon. In this case, six meteorological variables are considered for the second half of 2006. The relation between the variables and the evolution of its values throughout the day is shown through the application of statistical and soft computing models. Two case studies are analyzed, in an attempt to identify a 'Typical' day in Summer and Autumn, 2006. Differences between the various methods are discussed and comparisons drawn with the results of similar studies in other periods.

Keywords: artificial neural networks; soft computing; meteorology; atmospheric pollution

1. Introduction

In recent years, our knowledge of atmospheric pollution and our understanding of its effects have advanced greatly. It has now been accepted for some years that air pollution not only represents a health risk, but that it also reduces, for example, food production and vegetative growth due to its effects on photosynthesis. Other serious consequences may be mentioned such as acid rain, corrosion, climate change and global warming. Thus, all efforts that are directed towards studying

 ISBN 13: 978-84-613-5510-5

these phenomena may improve our understanding and help to prevent the serious problematic nature of atmospheric pollution.

Finding solutions to current environmental problems constitutes a fundamental step towards life with a sense of sustainability. Fulfilling such a wish is to a great extent determined by the preservation of a clean atmosphere given its impact on the dynamics of the biosphere.

Systematic measurements in Spain, which are usually taken within large cities, are fundamental due to the health risks caused by high levels of atmospheric pollution. Recent trends point to the benefits of continuing to extend the network of atmospheric pollution measurement stations.

The basis of this study is the application of a series of statistical and soft computing models to identify what may be called 'Typical Days' in terms of previously selected meteorological variables.

The rest of this study is organized as follows. Section 2 presents the statistical and soft computing methods applied throughout this research. Section 3 details the various case studies and Section 4 describes the experiments and results. Finally, Section 5 sets out the conclusions and future lines of work

2. Statistical and Soft Computing Models

Several statistical and soft computing models are used in this study, although the results are only shown of those that offer the best performance.

1. Principal Components Analysis (PCA)

PCA [1] gives the best linear compression of the data in terms of least mean square error and can be implemented by several artificial neural networks [2, 3]. The basic PCA network [4] applied in this study is described by the next three equations (Eq.(1) to Eq.(3)): an N-dimensional input vector at time t, $x(t)$, and an M-dimensional output vector, y, with W_{ij} being the weight linking input j to output i, and η being the learning rate. Its activation and learning may be described as follows:

Feedforward step, "Eq. (1)":

$$y_i = \sum_{j=1}^{N} W_{ij} x_j, \forall i \tag{1}$$

Feedback step, "Eq. (2)":

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i \tag{2}$$

Change weights, "Eq. (3)":

 ISBN 13: 978-84-613-5510-5

$$\Delta W_{ij} = \eta e_j y_i \tag{3}$$

2. An Exploratory Projection Pursuit Neural Model (EPP)

EPP [2, 3] projects the data onto a low dimensional subspace which allows its structure to be examined by eye. This is done by means of an index that measures the "interestingness" of a given projection, the data for which is then represented by projections that maximize the most "interesting" vectors. "Interesting" structure is usually defined with respect to the fact that most projections of high-dimensional data onto arbitrary lines through most multi-dimensional data give almost Gaussian distributions [2, 5]. Therefore to identify "interesting" features in data, it is important to look for those directions onto which the data-projections are as far from the Gaussian as possible.

3. Cooperative Maximum Likelihood Hebbian Learning (CMLHL)

CMLHL [6, 7] is an extended version of MLHL [6, 8] adding lateral connections which have been derived from the Rectified Gaussian Distribution [9]. The resultant net can find the independent factors of a data set but does so in a way that captures some type of global ordering in the data set.

Consider an N-dimensional input vector x, an M-dimensional output vector y and a weight matrix W, where the element W_{ij} represents the relationship between input x_j and output y_i, then as is shown in [6, 10], the CMLHL can be carried out as a four-step procedure:

Feed-forward step, outputs are calculated "Eq. (4)":

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \tag{4}$$

Lateral activation passing step, "Eq. (5)":

$$y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+ \tag{5}$$

Feedback step, "Eq. (6)":

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j \tag{6}$$

Weights update step, learn the neural network, "Eq. (7)":

$$\Delta W_{ij} = \eta . y_i . sign(e_j) | e_j |^{p-1} \tag{7}$$

Where t represents an instant, $[\ ]^+$ is necessary to ensure that the y-values remain in the positive quadrant, η is the learning rate, τ is the "strength" of the lateral connections, b the bias parameter, p a parameter related to the energy function, and A is a symmetric matrix used to modify the response to the data. The effect of this matrix is based on the relation between the distances separating the output neurons.

 ISBN 13: 978-84-613-5510-5

3. Case of Study. Identifying the Typical Day in Summer and Autumn

This study presents interesting results related to the evolution of different meteorological parameters using the records of an air quality control station (made available by the Department of the Environment-Directorate of Environmental Quality of the Government of the Spanish Autonomous Region of Castile-Leon) [11, 12]. The aforementioned station is situated in the urban area of the Spanish city of Burgos. The study was conducted over approximately half a year in 2006.

In this study, the following variables were analyzed: wind direction (degrees), wind speed (m/s), dry temperature (C^o), relative humidity (%), atmospheric pressure (mbar) and solar radiation (W/m^2).

The general characteristics of the site where the measurement station used in the study is situated are as follows: Burgos, a city in the north-centre of Spain with a population of around 170,000 inhabitants and a total municipal area of approximately 107 km^2. The city of Burgos is 854 masl (meters above sea level) at latitude (N) 42º20' and longitude (W) 3º42'. The measurement station is located within the city and may be classified as an urban station.

The aim of the present study is to identify the existence of 'Typical' meteorological days or at least to find some kind of associated patterns, for which purpose several statistical and soft computing methods were used. Only the results of applying PCA and CMLHL are shown. This is because PCA (Section 2.1) is the statistical method which offers a vision of the internal structure of the information and CMLHL (Section 2.3) is a Soft Computing Model which provides the best results in terms of identifying internal structure.

4. Experiments and Results

As stated, the aim of this study is to identify the 'Typical Day' in Summer and the 'Typical Day' in Autumn for 2006.

The study, which forms part of a more ambitious project [13, 14], is based on a file containing meteorological and pollution data sets recorded at fifteen-minute intervals: a daily total of 96 records for the second part of 2006, referring to six variables, as explained in Section 3.

The information represented at each point is visually labelled from Fig. 1 to Fig. 2, which shows the record number, (from record numbered as 1.-0:00 AM, to record numbered as 96.-23:45 PM). All data was normalized for the study.

1. Typical Day in Summer

The graphical results obtained in this study for a Typical Day in Summer are presented (Figure 1) and analyzed as follows.

 ISBN 13: 978-84-613-5510-5

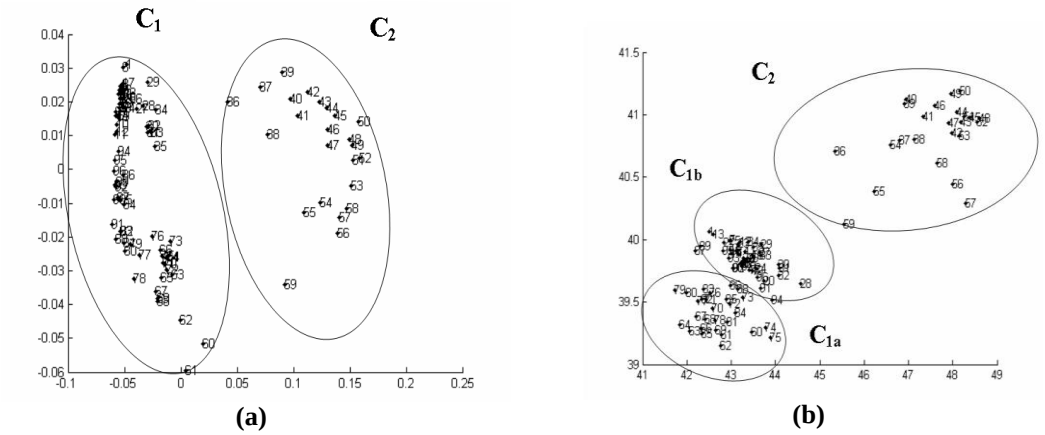


Figure 1. Typical Day in Summer. (a) PCA Projections. (b) CMLHL Projections

A Typical Day in Burgos, Summer 2006, according to the results of applying PCA to the meteorological variables, is shown in Figure 1(a). Two data clusters are identified. C_2 is related to samples with the highest values of solar radiation and temperature which correspond to the records taken around midday and the early afternoon, from 12:00PM to 16:00PM approximately. Cluster C_1 is related to samples with the lowest values which correspond to the rest of the day. Cluster C_2 contains fewer samples than C_1. These are the general characteristics of a Typical Day in Summer: variations between the different Typical Days are explained by the lowest values of the most representative variables -temperature and solar radiation- for the day being found among the earliest or the latest records of the day. But this is not enough, it is necessary to study the samples contained into the cluster C_1, for this reason it is important to apply soft computing models in order to obtain finer a response.

Figure 1(b) is obtained by applying CMLHL to the data set. Cluster C_2 contains the same samples as in Figure 1(a). In this case CMLHL is able to identify three clusters instead of two, achieving a sparser representation. Cluster C_1 in Figure 1(a) contains the same samples as clusters (C_{1a} and C_{1b}). Cluster C_{1a} contains samples from late evening, just before sunset, and cluster C_{1b} contains the samples belonging to the night-time, at which time solar radiation is almost null.

2. Typical Day in Autumn

The graphical results obtained in this study for a Typical Day in Autumn are presented (Figure 2) and analyzed below.

 ISBN 13: 978-84-613-5510-5

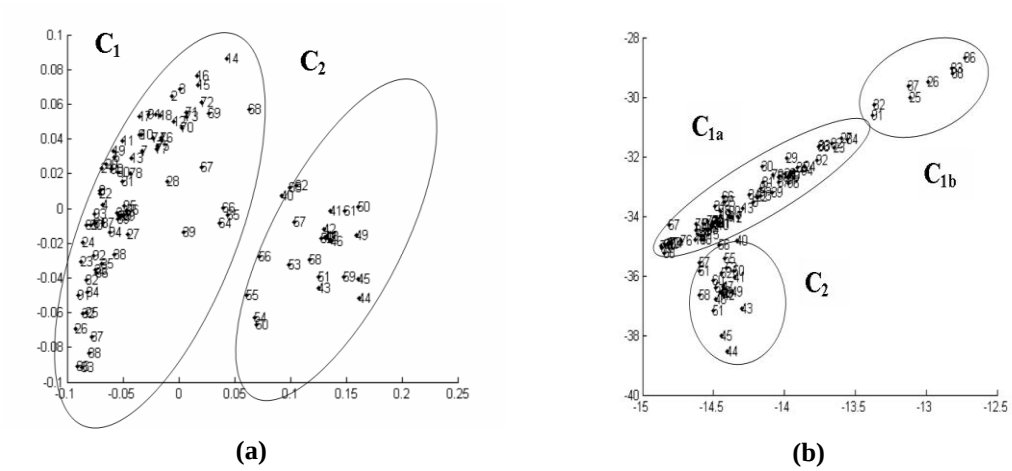


Figure 2. Typical Day in Autumn. (a) PCA Projections (b) CMLHL Projections

Figure 2(a) shows the results of applying a PCA model to identify a Typical Day in Autumn, which once again highlights two clusters (C_1, and C_2) in a similar way to Figure 1. Cluster C_2 is again related to samples with the highest values of solar radiation and temperature, which correspond to the records taken around midday and the early afternoon. In this case, C_2 in Figure 2 contains fewer samples than it does in Figure 1. This is because sunset is earlier in the day. A further difference is that the C_2 cluster is closer to cluster C_1 in Figure 2 than it is in Figure 1, which is because there is not so much variability in autumn data values throughout the day as they are in the Summer period.

Figure 2(b) shows the graphical results of applying CMLHL. Cluster C_2 contains the same samples as in Figure 1(a), but in a more grouped form. Again, in this case, CMLHL is able to identify three clusters instead of two. Cluster C_1 in Figure 1(a) contains the same samples as in Figure 2 (C_{1a} and C_{1b}). Cluster C_{1a} contains most of the samples, and cluster C_{1b} contains few samples belonging to the night, where a significant variability of the wind direction is observed. The detection of this effect is an interesting example of how a model like CMLHL can help us to analyze complex data sets.

5. Conclusions and Future Works

In this study it has been possible to demonstrate the validity of soft computing models for the identification of the so called "Typical Day" of Summer and Autumn in 2006, as it was possible in 2007 [15].

PCA provides a first approximation to the internal structure of the data, but other soft computing models provide a top response, discovering new clusters of information which represent extra information. Several soft computing models were applied, and in this case only the results of PCA and CMLHL are shown.

 ISBN 13: 978-84-613-5510-5

The Typical Day in Summer in the city of Burgos is characterized by sudden changes in temperature and in solar radiation. Due to these factors, it is easy to identify a cluster corresponding to the central hours of the day, as temperature decreases very quickly at sunset.

In contrast, the variability of solar radiation and temperature is smoother in the Typical Day in Autumn in the city of Burgos. The clusters are not so clearly identified due to the evolution of the data values throughout the day.

In [15], a similar study was undertaken in 2007. The results of both studies are consistent. In subsequent studies, other seasons and annual periods will be analyzed, and a meticulous comparison will be made between those studies and public information on atmospheric pollution and meteorological conditions.

Acknowledgments. This research has been partially supported through projects BU006A08 and BU035A08, both of the JCyL, and project CIT-020000-2008-2 of the Spanish Ministry of Education and Innovation. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolín Ingeniería, S.A., within the framework of the project MAGNO2008 - 1028.- CENIT Project funded by the Spanish Ministry

6. References

[1] H. HOTELLING, *analysis of a complex of statistical variables into principal components*, Journal of Educational Psychology, 24:417-444. 1933.

[2] A. HYVÄRINEN, *Complexity Pursuit: Separating interesting components from time series*, Neural Computation 13 (2001), pp. 898-883.

[3] A. HYVÄRINEN, J. KARHUNEN AND E. OJA, *Independent Component Analysis*, Wiley, (2002).

[4] C. FYFE AND R. BADDELEY, *Non-linear data structure extraction using simple Hebbian networks*, Biological Cybernetics 72(6) (1995), pp. 541-533

[5] S. SEUNG, N.D. SOCCI AND D. LEE, *The Rectified Gaussian Distribution*, Advances in Neural Information Processing Systems, 10 (1998), pp. 350-356.

[6] E. CORCHADO, C. FYFE, *Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors*, IJPRAI 17(8): 1447-1466 (2003).

[7] E. CORCHADO, Y. HAN AND C. FYFE, *Structuring Global Responses of Local Filters Using Lateral Connections*, Journal of Experimental & Theoretical Artificial Intelligence, 15(4) (2003), pp. 473-487.

[8] EMILIO CORCHADO, DONALD MACDONALD AND COLIN FYFE, *Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit*, Data Min. Knowl. Discov. 8,203-225 (2004)

 ISBN 13: 978-84-613-5510-5

[9] P. L. LAI, D. CHARLES AND C. FYFE, *Seeking Independence using Biologically Inspired Artificial Neural Networks*, in: Developments in Artificial Neural Network Theory: Independent Component Analysis and Blind Source Separation, M. A. Girolami (ed.), Springer Verlag, 2000.

[10] E. OJA, H. OGAWA AND J. WANGVIWATTANA, *Principal Components Analysis by Homogeneous Neural Networks*, part 1, The Weighted Subspace Criterion, IEICE Transaction on Information and Systems E75D (1992), pp. 375-366.

[11] V. TRICIO, R. VILORIA, A. MINGUITO, *Evolución del ozono en Burgos y provincia a partir de los datos de la red de medida de contaminación atmosférica. Los retos del desarrollo sostenible en España*, Informe CONAMA 2006. 31 PAGES. (2006) http://www.conama8.org/modulodocumentos/documentos/CTs/CT86.pdf.

[12] V. TRICIO, R. VILORIA, A. MINGUITO, *Ozone Measurements in Urban and Semi-Rural Sites at Burgos (Spain)*, In Geophysical Research Abstracts, Volume 5, 2003. EGS-AGU-EUG Joint Assembly. EAE03-A-14249. ISSN: 1029-7006.

[13] ANGEL ARROYO, EMILIO CORCHADO AND VERÓNICA TRICIO, *Computational Methods for Immision Analysis of Urban Atmospheric Pollution*, Proceedings of the International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE2009, pp. 1169-1176, ISBN: 978-84-612-9727.

[14] ANGEL ARROYO, EMILIO CORCHADO AND VERÓNICA TRICIO, *Atmospheric Pollution Analysis by Unsupervised Learning*, Intelligent Data Engineering and Automated Learning - IDEAL 2009. Springer - Lecture Notes in Computer Science, vol. 5788: 767 – 773.

[15] EMILIO CORCHADO, ÁNGEL ARROYO AND VERÓNICA TRICIO, *Soft Computing Models to Identify Typical Meteorological Days*, Logic Journal of the IGPL, Oxford Journals, Press Online ISSN 1368-9894 - Print ISSN 1367-0751.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Orthogonal Zero-Interpolants: *Properties & Applications*

M. A. Bokhari

Department of Mathematics & Statistics, KFUPM, Dhahran, Saudi Arabia

emails: mbokhari@kfupm.edu.sa

Abstract

This paper deals with the notion of "Orthogonal Zero Interpolants" (*OZI*) which have several properties similar to the classical orthogonal polynomials. *OZI* are constructed in such a way that they interpolate the "Zero Function" at a finite number of pre-assigned nodes, even of multiple orders, in the sense of Hermite. These polynomials are also determined by the 3-term recurrence relation. We shall discuss structure and some properties of *OZI* alongwith their applications to certain approximation and boundary value problems.

Key words: L_2-approximation, Hermite interpolation, Orthogonal Zero Interpolants, Erdos-Turan theorem, 3-term recurrence relation, Two-point boundary value problem.

1. Introduction

Solution of several approximation problems and quite a few numerical solutions of boundary value problems (BVP) are based on orthogonal polynomials with respect to Jacobi weight functions $\omega_{\alpha,\beta}(x) := (1-x)^{\alpha}(1+x)^{\beta}$, $\alpha, \beta > -1$ over $[-1,1]$. These polynomials, in general, are not suitable for determining solution of the problems subject to constraints. Here, we present some problems and discuss their solutions by a specific class of polynomials which we shall refer to as orthogonal zero interpolants. In order to avoid repitition, we list below some notations which will be frequently used in this paper:

$\omega(t) :=$ Positive weight function defined on $[c,d]$

$\mathbb{L}^2_{\omega}[c,d] :=$ Class of functions f with $\|f\|^2_{\omega,[c,d]} := \int_c^d f(t)\omega(t)dt < \infty$

 ISBN 13: 978-84-613-5510-5

$\pi_n :=$ Class of all polynomils of degree $\le n$

$\langle g, h \rangle_{\omega,[c,d]}$: $\int_c^d g(t)h(t)\omega(t)dt$

$I[g, \omega]$: $\int_c^d g(t)\omega(t)dt$

$L_n(., Z_{n+1}, f)$: Lagrange polynomial of degree n which interpolates f at the $n+1$ points of Z_{n+1}

$\langle f_0, f_1, ..., f_n \rangle$: Vector space generated by $f_0, f_1, ..., f_n$

With the notations given above, we look into the following problems:

A. Least squares approximation problem [1]

Given $f \in \mathbb{L}^2_\omega[c,d]$ *and a finite data* $\{(x_i, y_{i,j})\}_{i=1, j=0}^{k \quad n_i-1}$ *with distinct* x_i*'s, can we find* $p* \in \pi_m$ *that minimizes the error* $\|p - f\|_{\omega,[c,d]}$ *over all* $p \in \pi_m$ *subject to the constraints*: $p^{(j)}(x_i) = y_{i,j}$, $i = 1, ..., k$; $j = 0, 1, ..., n_i - 1$?

B. Lagrange interpolants in Erdos-Turan theorem

A result due to Erdos-Turan [4] *states that* $\lim_{n \to 0} \|L_n(., Z_{n+1}, f) - f\|_{\omega,[c,d]} = 0$ *where the set* Z_{n+1} *consists of the n + 1 zeros of an (n + 1) degree orthogonal polynomial with respect to* $\omega(x)$ *over* [*c,d*]. *An extension of this problem may be posed as follows: Given a finite data* $\{(x_i, f^{(j)}(x_i))\}_{i=1, j=0}^{k \quad n_i-1}$ *with distinct* x_i*'s lying outside the interval* (*c,d*), *can we modify the interpolating polynomial* $L_n(., Z_{n+1}, f)$ *to one which interpolates the additional data and preserves the convergence property over the interval* [*c,d*]*?*

C. Numerical Solution of BVP by collocation method

When determining a numerical solution of boundary value problems by collocation method, the Gaussian nodes are usually regarded as a best choice [6]. *Can we determine their appropriate replacement by certain orthogonal points with partial freedom of choice without compromising the quality of numerical solution?*

To answer these questions, we slightly change the structure of classical orthogonal polynomials by appending a finite number of pre-assigned zeros. The modified polynomials will be referred to as *OZI*. The suggested polynomials convert certain constrained approximating problems to unconstrained ones by

 ISBN 13: 978-84-613-5510-5

modifying their approximating set. In addition, the zeros of *OZI* in some cases have some advantage over the Gaussian nodes in collocation methods.

2. Orthogonal Zero Interpolants (*OZI*)

The structure of *OZI*, in general, is based on a given data $\{(x_i, n_i)\}_{i=1}^{k}$ where x_i's are distinct real numbers and n_i's are positive integers. These interpolants arise from a sequence of polynomials $\psi_j(x)$, $j = 0,1,2,.....$, which is constructed by the 3-term recurrence relation [5] as follows:

$$\psi_{j+1}(x) = (x - \alpha_j)\psi_j(x) - \beta_j \psi_j(x), \quad j = 1,2,\ldots, \qquad \textbf{(1)}$$

with $\psi_0(x) = \prod_{i=1}^{k}(x - x_i)^{n_i}$ and $\psi_1(x) = (x - \alpha_0)\psi_0(x)$. The recursion coefficients in (1) are given by

$$\alpha_j = \frac{I[x\psi_j^2, \omega]}{I[\psi_j^2, \omega]},\ j = 0,1,\ldots; \quad \beta_j = \frac{I[\psi_j^2, \omega]}{I[\psi_{j-1}^2, \omega]}, \quad j = 1,2,\ldots. \qquad \textbf{(2)}$$

where the notation $I[h, \omega]$ stands for $\int_c^d h(x)\omega(x)dx$.

Definition 1. The polynomials $\psi_j(x), j = 0,1,2,.....$, generated from relation (1) will be refered to as orthogonal zero interpolants relevant to the data $\{(x_i, n_i)\}_{i=1}^{k}$.

Some properties of *OZI*. It is obvious that the *OZI* $\psi_n(x)$ is a polynomial of degree $(n+N)$ with $N = \sum_{i=1}^{k} n_i$ and that $\psi_0(x) = \prod_{i=1}^{k}(x - x_i)^{n_i}$ is a factor of this polynomial. Also, $\psi_j^{(l)}(x_i) = 0$ for $i = 1,.....,k;\ l = 0,1,2,....,n_i - 1$. The polynomials $\psi_n(x)$, i = 0,1,,2,….. are monic and mutually orthogonal *w.r.t.* the weight function $\omega(x)$ over the interval $[c, d]$, i.e., $I[\psi_j \psi_l, \omega] = 0$ for $j \neq l$. Besides the fixed zeros each $\psi_n(x)$ has exactly n real and distinct zeros in the open interval (c, d). We shall denote these zeros by $z_{i,n}$, $i = 1,2,...,n$, in the sequel. Thus, $\psi_n(x) = \psi_0(x)Q_n(x)$ where

$$Q_n(x) := \prod_{i=1}^{n}(x - z_{i,n}). \qquad \textbf{(3)}$$

Also, the coefficients β_k in (1) are positive. As a custom, we set $\beta_0 = I[\psi_0^2, \omega]$.

 ISBN 13: 978-84-613-5510-5

Remark 1. The vector space $\left\langle x^i\psi_0 : i = 0,1,...,n \right\rangle$ which will be denoted by $\pi_n(\psi_0)$ is an $(n+1)$-dimensional subspace of π_{n+N} where $N = \sum_{i=1}^{k} n_i$. Also, $\{\psi_0, \psi_1, \psi_2, ..., \psi_n\}$ is an orthogonal bases $\pi_n(\psi_0)$. Thus, $\psi_n \perp \pi_r(\psi_0)$ for $0 \le r \le n-1$.

3. Problem A: *Formulation, solution and convergence*

We reformulate Problem (A) as follows [1]: *Find a polynomial* $p_m^* \in \pi_m$ *which minimizes* $\|p - f\|_{\omega,[c,d]}$ *over all* $p \in \pi_m$ *satisfying* $p^{(j)}(x_i) = y_{i,j}$, $i = 1,2,\ldots,k$, $j = 0,1,\ldots,n_i - 1$.

Because of the number of interpolatory conditions, *m* can not be less than $N-1$. We solve the reformulated problem by converting it to an unconstrained minimization problem. To do so, first we set

$$f_H(x) := f(x) - H_{N-1}(x, Y), \qquad \textbf{(4)}$$

where $H_{N-1}(x,Y)$ is the polynomial of degree $N-1$ satisfying *N* interpolation conditions: $H_{N-1}^{(j)}(x_i, Y) = y_{i,j}$, $i = 1,...,k$; $j = 0,1,...,n_i$. In terms of $f_H(x)$ and $\pi_m(\psi_0)$, we state an equivalent form of Problem (A):

Problem (A*): Find a polynomial $\phi_m \in \pi_m(\psi_0)$ which solves the problem:

$$\min_{\phi \in \pi_m(\psi_0)} \|\phi - f_H\|_{\omega,[c,d]}. \qquad \textbf{(5)}$$

Solution of Problem (A): By Riesz-Fischer Theorem, we note that the solution of (4) is given by $\phi_m(x) = \sum_{i=0}^{m} \frac{\langle f_H, \psi_i \rangle}{\langle \psi_i, \psi_i \rangle} \psi_i(x)$. Thus, $p_m^*(x) := H_{N-1}(x, Y) + \phi_m(x)$ is the solution of Problem (A).

Convergence. If f is at least N^*-times continuously differentiable where $N^* = \max_{0 \le i \le k}(n_i - 1)$, we can prove:

Theorem 1 [1]**.** *If* $y_{i,j} = f^{(j)}(x_i)$, $i = 1,...,k$; $j = 0,1,...,n_i$ *in the set-up of Problem (A), then* $\lim_{m \to 0} \|p_m^* - f\|_{\omega,[c,d]} = 0$.

 ISBN 13: 978-84-613-5510-5

Remark 2. The *OZI's* considered in the solution of Problem (A) satisfy the Parseval equality for all $f \in C^{N*}[a,b]$, *i.e.*, $\langle f_H, f_H \rangle = \sum_{i=0}^{\infty} \langle f_H, \psi_i \rangle / \langle \psi_i, \psi_i \rangle$.

4. Problem B: *Formulation, solution and convergence*

Problem (B) can be elaborated as follows [3]: *Let* $f:[a,b] \to \Re$ *with* $[c,d] \subseteq [a,b]$. *Assuming that* $\Delta_k = \{x_1, x_2, ..., x_k\} \subseteq [a,b] \setminus (c,d)$, *find a polynomial* $P_{n,k} \in \pi_{n+k-1}$ *satisfying the following properties*:

(i) *The polynomial* $P_{n,k}$ *emerges from an interpolation scheme based on distinct* $(n+1)$ *zeros of certain orthogonal polynomial with respect to* ω *over* [c,d],

(ii) $P_{n,k}^{(j)}(x_i) = f^{(j)}(x_i),\ x_i \in \Delta_k,\ j = 0,1,\ldots n_i - 1,$

(iii) $\lim_{n \to 0} \left\| P_{n,k} - f \right\|_{\omega,[c,d]} = 0.$

Like Problem (A), this problem is also based on the data $\left\{(x_i, f^j(x_i)\right\}_{i=1, j=0}^{k,\ n_1-1}$. In order to construct $P_{n,k}$, we find an orthogonal polynomial which has $n+1$ free nodes in the open interval (c,d) and k pre-assigned nodes x_i of multiplicity n_i, $i = 1,2,\ldots,k$, lying outside (c,d).

Choice of orthogonal polynomials. The requirements (i) and (ii) are met by the *OZI* ψ_{n+1} which can be determined by the 3-term recurrence relation (1) with $\psi_0(x) = \prod_{i=1}^{k} (x - x_i)^{n_i}$. Thus, (cf (3)):

$$\psi_{n+1}(x) = \psi_0(x) Q_{n+1}(x). \qquad \textbf{(6)}$$

As before, we set $f_H(x) = f(x) - H_{N-1}(x, f)$ where $H_{N-1}(x, f)$ is the polynomial of degree $N-1$ which interpolates f in the sense of Hermit at k pre-assigned nodes x_i of multiplicity n_i, $i = 1,2,\ldots,k$. With $N := \sum_{i=1}^{k} n_i$, Problem (B) is reformulate in terms of $f_H(x)$ and $\pi_{n+1}(\psi_0)$:

Problem B*. Find a polynomials $P_n^* \in \pi_n(\psi_0)$ which interpolate f_H at the $N+n+1$ zeros of ψ_{n+1}. Moreover, $\lim_{n \to 0} \left\| P_n^* - f_H \right\|_{\omega,[c,d]} = 0$.

 ISBN 13: 978-84-613-5510-5

Existence of P_n^*. Set $\Delta_k := \{x_1, x_2, \ldots, x_k\}$ and define $f_{H,\Delta_k} : [a,b] \to \Re$ as follows:

$$f_{H,\Delta_k}(x) = \begin{cases} \dfrac{f_H(x)}{\psi_0(x)}, & \text{if } x \notin \Delta_k \\ \lim\limits_{t \to x} \dfrac{f_H(t)}{\psi_0(x)}, & \text{if } x \in \Delta_k. \end{cases}$$

With $Z_{n+1} = \{z_{1,n+1}, \ldots . z_{n+1,n+1}\}$ (cf (3)) define $P_n^*(x) = \psi_0(x) L_n(x, Z_{n+1}, f_{H,\Delta_k})$. It may be noted that $P_n^*(x) = f_{H,\Delta_k}(x)$ if $x \in Z_{n+1}$, and $P_n^{*(j)}(x_i) = 0 = f_H^{(j)}(x_i)$ for $i = 1, \ldots k$, $j = 0, 1, \ldots, n_i - 1$.

Convergence. Set $p_{n,N}(x) := P_n^*(x) - H_{N-1}(x, f)$. Then with suitable differentiability conditions on f we have

Theorem 2 [3]**:** $\lim\limits_{n \to \infty} \| p_{n,N} - f \|_{\omega,[c,d]} = 0$.

Remark 3. Theorem 2 is an extension of a result due to Erdos and Turan [4].

5. Problem C: *Formulation, solution and computational aspect* [2]

Problem C addresses the nature of collocation points required in the collocation solution of boundary value problems (*BVP*). Here, we shall notice certain advantage of the zeros of *OZI*'s over the Gaussian points when used as collocation points. This phenonenon is explained by a two-point linear *BVP*

$$y'' + \alpha(t) y' + \beta(t) y = f(t) : y(a) = 0,\ y(b) = 0, \qquad \textbf{(7)}$$

with $\alpha(t) = 0,\ \beta(t) = -1,\ f(t) = \cos t$ and $a = 0,\ b = 1$. The exact solution of (7) is given by

$$y(t) = \frac{e^{-1} - \cos(1)}{2(e^{-1} - e)} e^t + \frac{\cos(1) - e}{2(e^{-1} - e)} e^{-t} - \frac{1}{2} \cos t\,.$$

We shall compute collocation solutions of this problem by using a three dimensional solution space $S_3 = \langle \phi_1, \phi_2, \phi_3 \rangle$ where each polynomial ϕ_i vanishes at 0 and 1. The collocation solution in this se up will be of the form [6] $\varsigma = c_1\phi_1 + c_3\phi_2 + c_3\phi_3$ where the unknowns are retrieved by solving the linear system

 ISBN 13: 978-84-613-5510-5

$$\begin{bmatrix} L\phi_1(t_1) & L\phi_2(t_1) & L\phi_3(t_1) \\ L\phi_1(t_2) & L\phi_2(t_2) & L\phi_3(t_2) \\ L\phi_1(t_3) & L\phi_2(t_3) & L\phi_3(t_3) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} f(t_1) \\ f(t_2) \\ f(t_3) \end{bmatrix}, \qquad (8)$$

Here, $L := D^2 + a(x)D + b(x)$. If ϕ_1, ϕ_2, ϕ_3 are linearly independent and the points t_1, t_2, t_3 are distinct, then the coefficient matrix $\left[L\phi_i(t_j)\right]_{i,j=1}^{3}$ is always non-singular. Therefore, a unique solution of (8) does exist.

Our aim is discuss the collocation solutions based on different sets $\{t_1, t_2, t_3\}$ which consist of either the Gaussian nodes or the zeros of *OZI's*, and then compare the resulting errors.

We have selected two different basis polynomials for S_3:

B-1: $\phi_1(t) = t(1-t), \quad \phi_2(t) = t^2(1-t), \quad \phi_3(t) = t^2(1-t)^2,$

B-2: $\phi_1(t) = t(1-t), \quad \phi_2(t) = t^2(1-t), \quad \phi_3(t) = t^3(1-t).$

For collocation points, we have considered different sets consisting of zeros of *OZI*'s of the form "$\psi_2(t) = \psi_0(t)Q_2(t)$" where $Q_2 \in \pi_2$ and $\psi_0(t) = t - t_1$. Having fixed different values of $t_1 \in [0,1]$, the remaining two zeros, t_2 and t_3, are determined from $Q_2(t)$ (cf (3)). We have constructed eight sets of collocation points (Cpt) with different choice of $t_1 \in [0,1]$. These are

Cpt-1: $t_1 = 0$ $t_2 = 4.5585 \times 10^{-1}$ $t_3 = 8.7749 \times 10^{-1}$
Cpt-2: $t_1 = 1.5000 \times 10^{-1}$ $t_2 = 4.8928 \times 10^{-1}$ $t_3 = 8.8543 \times 10^{-1}$
Cpt-3: $t_1 = 3.0000 \times 10^{-1}$ $t_2 = 2.3246 \times 10^{-1}$ $t_3 = 8.7315 \times 10^{-1}$
Cpt-4: $t_1 = 4.5000 \times 10^{-1}$ $t_2 = 1.1607 \times 10^{-1}$ $t_3 = 8.8529 \times 10^{-1}$
Cpt-5: $t_1 = 6.0000 \times 10^{-1}$ $t_2 = 1.1897 \times 10^{-1}$ $t_3 = 8.6954 \times 10^{-1}$
Cpt-6: $t_1 = 7.5000 \times 10^{-1}$ $t_2 = 1.2662 \times 10^{-1}$ $t_3 = 6.6825 \times 10^{-1}$
Cpt-7: $t_1 = 9.0000 \times 10^{-1}$ $t_2 = 1.1290 \times 10^{-1}$ $t_3 = 5.0097 \times 10^{-1}$
Cpt-8: $t_1 = 1.0000\text{e+}000$ $t_2 = 1.2251 \times 10^{-1}$ $t_3 = 5.4415 \times 10^{-1}$

Two collocation solutions based on B-1 and B-2 each with three Gaussian points Gpt: $t_1 = 5.0000 \times 10^{-1}$ $t_2 = \left(1-\sqrt{3/5}\right)/2$ $t_3 = \left(1+\sqrt{3/5}\right)/2$ are also computed for the sake of comparison.

Accuracy. The level of accuracy of the resulting collocation solutions is determined by considering maximum error 'M-Err $= \max_{1 \le i \le n} \left| y(t_i) - \varsigma_{(K,L)}(t_i) \right|$', and

 ISBN 13: 978-84-613-5510-5

the root mean squared error**:** '$\text{RMS} = \sqrt{\frac{\sum_{i=1}^{n} \left| y(t_i) - \varsigma_{(K,L)}(t_i) \right|^2}{n}}$. These errors are computed over n = 101 uniform mesh points of $[-1,1]$. Here, $\varsigma_{(K,L)}$ denotes the collocation solution corresponding to basis B-K, K =1, 2 and collocation points Cpt-L, L=1,2,…8. The errors corresponding to each solution are tabulated below.

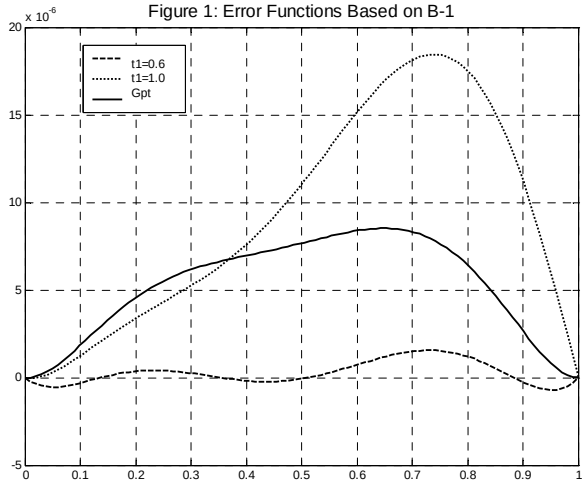
Table 1: Comparison of errors based on B-1 polynomials

	$\varsigma_{(1,1)}$ t_1= 0	$\varsigma_{(1,2)}$ t_1= .15	$\varsigma_{(1,3)}$ t_1= .3	$\varsigma_{(1,4)}$ t_1= .45	$\varsigma_{(1,5)}$ t_1= .6	$\varsigma_{(1,6)}$ t_1= .75	$\varsigma_{(1,7)}$ t_1= .9	$\varsigma_{(1,8)}$ t_1= 1	$\varsigma_{(1,Gpt)}$
M-Err	1.6248×10^{-5}	7.8463×10^{-6}	4.8222×10^{-6}	8.8012×10^{-6}	1.5613×10^{-6}	5.7026×10^{-6}	9.8077×10^{-6}	1.8457×10^{-5}	8.5202×10^{-6}
RMS-Err	1.0982×10^{-5}	4.9540×10^{-6}	2.6670×10^{-6}	6.1676×10^{-6}	7.0045×10^{-7}	3.7685×10^{-6}	6.7537×10^{-6}	1.1005×10^{-5}	5.9790×10^{-6}

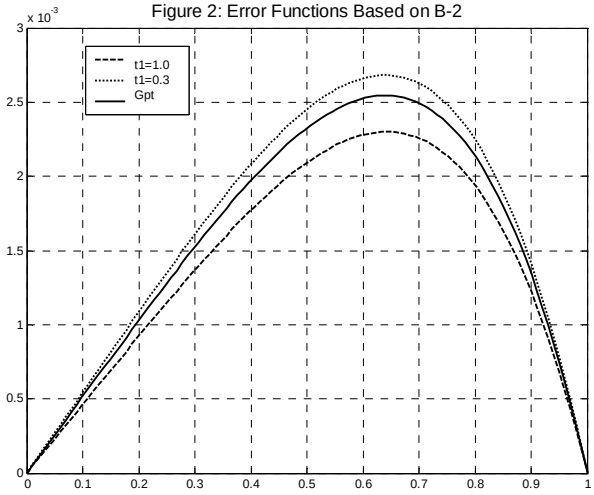
Table 2: Comparison of errors based on B-2 polynomials

	$\varsigma_{(2,1)}$ t_1= 0	$\varsigma_{(2,2)}$ t_1= .15	$\varsigma_{(2,3)}$ t_1= .3	$\varsigma_{(2,4)}$ t_1= .45	$\varsigma_{(2,5)}$ t_1= .6	$\varsigma_{(2,6)}$ t_1= .75	$\varsigma_{(2,7)}$ t_1= .9	$\varsigma_{(2,8)}$ t_1= 1	$\varsigma_{(2,Gpt)}$
M-Err	2.5382×10^{-3}	2.5583×10^{-3}	2.6814×10^{-3}	2.5888×10^{-3}	2.4589×10^{-3}	2.5547×10^{-3}	2.5248×10^{-3}	2.2997×10^{-3}	2.5449×10^{-3}
RMS-Err	1.7835×10^{-3}	1.7960×10^{-3}	1.8829×10^{-3}	1.8177×10^{-3}	1.7253×10^{-3}	1.7933×10^{-3}	1.7722×10^{-3}	1.6125×10^{-3}	1.7865×10^{-3}

The graphs of error functions for each basis polynomials are given in Figures 1 and 2. Each figure involves three error functions corresponding to Gaussian points and the sets of the zeros of two different *OZI*'s. In case of *OZI*'s, we have selected one set resulting to a superior solution and the other one resulting to inferior when compared with the solution based on Gaussian nodes.



 ISBN 13: 978-84-613-5510-5



Conclusions. The performance of collocation methods, as already known, depends on the choice of collocation points and the nature of basis functions of the solution space. From the tables, we note that the zeros of several *OZI*'s provide a better solution to the one based on the Gaussian points. However, we could not figure out any criterion that determines a better choice of *OZI* for a specific basis of a collocation solution space.

Acknowledgement. The author acknowledges the research facilities availed at King Fahd University of Petroleum & Minerals during the preparation of this paper.

6. References

[1] M. A. BOKHARI, *On Hermite interpolating L_2-approximants*, Dynamical Systems and Applications, Vol. 16, pp. 203-216 (2007).

[2] M. A. BOKHARI AND H. AL-ATTAS, *Combination of Orthogonal Collocation Points with Pre-assigned Knots*, Project # FT060002, KFUPM, Saudi Arabia, 2008, http://faculty.kfupm.edu.sa/math/mbokhari/ (Projects)

[3] M. A. BOKHARI AND H. AL-ATTAS, *Interpolation outside the Interval of Convergence*, Project # FT080008, KFUPM, Saudi Arabia, 2009, http://faculty.kfupm.edu.sa/math/mbokhari/ (Projects)

[4] P. ERDOS AND P. TURAN, *On Interpolation*, AM 38, 142-155 (1937).

[5] W. GAUTSCHI, Orthogonal Polynomials: Computations and Applications, Claredon Press, 2004

[6] P. M. PRENTER, *Splines and variational methods*, John Wiley & Sons 1975.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Fitting a straight line to a Normal Q-Q Plot. R Script

Castillo-Gutiérrez, Sonia[1] and Lozano-Aguilera, Emilio[1]
[1] *Department of Statistics and Operation Research, University of Jaén*

emails: socasti@ujaen.es, elozano@ujaen.es

Abstract

In this contribution we present different proposals of straight lines and develop an R Script to fit these lines to a Normal Quantile-Quantile Plot (Q-Q Plot), which can be chosen among six possibilities.

Key words: Normal Q-Q Plot, R Script, straight line.

1. Introduction

In Statistics, there are many studies in which it is necessary to verify if the data set comes from a Normal distribution.

The Normal Quantile-Quantile Plot (Q-Q Plot) is a popular and useful tool for assessing the normality of a data set. This plot compares the ordered distribution of a sample with the quantiles of the Standard Normal distribution indicated by the straight line. If the sample is normally distributed, the points will lie along this line.

Given a set of ordered observations $x_{(1)}, x_{(2)}, \ldots, x_{(n)}$, a Normal Q-Q Plot is constructed by plotting the pairs $\left(\Phi^{-1}(p_i), x_{(i)}\right)$, where Φ represents the standard normal cumulative distribution function (with zero mean and unit variance) [1] and $p_1, p_2, \ldots, p_n$ are appropriate plotting positions [2]. In this paper, we will use the definition of plotting position proposed by Hazen [3] in 1930, which is defined as:

$$p_i = \frac{i - 0.5}{n} \qquad i = 1, 2, \ldots, n$$

 ISBN 13: 978-84-613-5510-5

2. Fitting a straight line to a Normal Q-Q Plot

The Normal Q-Q Plot graphically compares the distribution of a given variable with the Normal distribution, represented by a straight line, though not necessarily the straight line y=x.

The value of the straight line on the point of abscissas zero, will provide estimation of the population average, and the value of the straight line on the point of abscissas one, will show the value of the sum of the average and the standard deviation. We may use this to provide estimation of the population mean and the standard deviation.

Moreover, what adjusts of the straight line to the Normal Q-Q Plot is an important aspect to consider. In this paper we develop an R script [4] through which we can choose a desired straight line (among six possible alternatives).

3. Possible straight lines in a Normal Q-Q Plot

In a Normal Q-Q Plot there exists the possibility of representing diverse straight lines for these pairs of points, since for each of the straight lines we will obtain different population parameters for the average and the standard deviation.

We will study the following straight lines:

1. Straight line that passes through the first and third quartiles.
2. Straight line that passes through the 10 and 90 percentiles.
3. Straight line fitted by the method of least squares.
4. Tukey's resistant line [5].
5. Theil's line [6].
6. Straight line with slope 's' and constant the average of the data set.

4. R Script

The basic content of the R script that we propose to select the different straight lines in a Normal Q-Q Plot, is the following:

```
lines <- function(x)
{       x <- sort(x)
        n <- length(x)
        phazen <- c(((1:n)-0.5)/n)
        ejex <- qnorm(phazen)
        print("To key in definition's number of straight line, where:")
```

 ISBN 13: 978-84-613-5510-5

```
print("1:'First and Third quartiles' ; 2:'10 and 90 percentiles' ; 3:'Method of
least squares' ; 4:'Tukey's resistant line' ; 5:'Theil's line' ; 6:'Slope 's' and
constant average'")
line <- scan(file="", what=integer(0), n=1, quiet=TRUE)
if (line<1 || line>6) print("You must key a number between 1 and 6") else
qqnorm(x, xlab="Quantiles N(0,1)", ylab="Observations")

if (line==1) qqline(x)

if (line==2)
{       p10x <- quantile(ejex,0.1)
        p90x <- quantile(ejex,0.9)
        p10y <- quantile(x,0.1)
        p90y <- quantile(x,0.9)
        pte <- (p90y-p10y)/(p90x-p10x)
        cte <- p10y-(pte*p10x)
        abline(a=cte,b=pte)
}

if (line==3) abline(lm(x~ejex))

if (line==4)
{       library("LearnEDA")
        Tukey <- rline(ejex,x)
        abline(a=Tukey$a, b=Tukey$b)
}

if (line==5)
{       library("mblm")
        Theil <- mblm(x~ejex)
        abline(a=Theil$coefficients[1], b=Theil$coefficients[2])
}

if (line==6) abline(a=mean(x), b=sd(x))
}
```

5. Example

An application of the previous R script is represented in the following example, where we use simulated observations of a Chi-Square distribution.

 ISBN 13: 978-84-613-5510-5

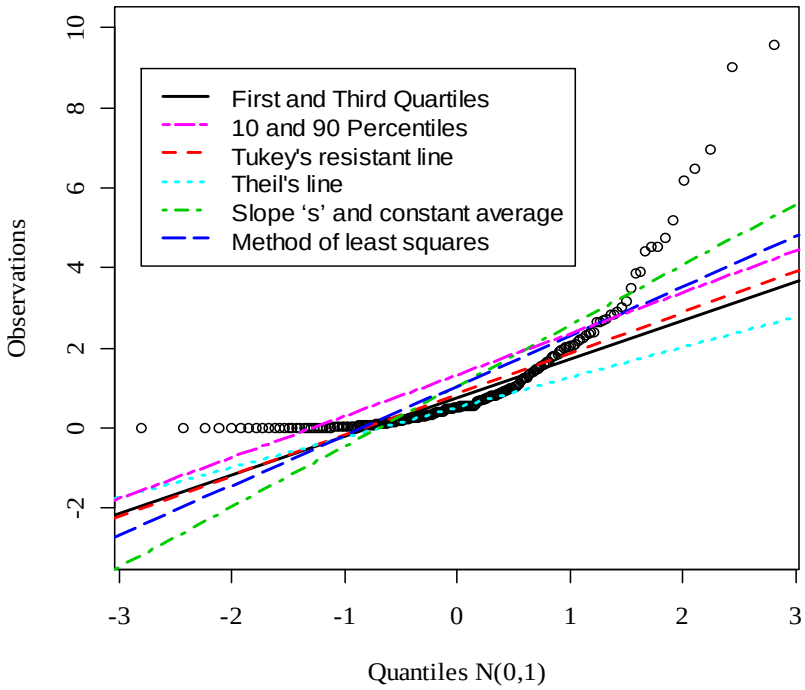


Figure 1: Normal Q-Q Plot with six different straight lines.

In Figure 1 we can observe how the choice of the adjustment method of a straight line is fundamental in a Normal Q-Q Plot, since there are differences in the straight lines represented in the plot.

6. References

[1] E. D. LOZANO AGUILERA, *Aportaciones a las técnicas gráficas para el estudio de normalidad y las causas de su pérdida*. Tesis Doctoral, Spain, 1995.

[2] S. CASTILLO GUTIERREZ AND E. D. LOZANO AGUILERA, *Q-Q Plot Normal. Los puntos de posición gráfica*, Revista electrónica ‘Iniciación a la Investigación’, 2:a9, (2007).

[3] A. HAZEN, *Flood Flows. A Study of Frequencies and Magnitudes*, Wiley, New York, 1930.

[4] R DEVELOPMENT CORE TEAM, R: *A Language and Environment for Statistical Computing*, Viena, Austria, 2008

[5] J. W. TUKEY, *Exploratory Data Analysis*, Addison-Wesley Publishing Company, 1977.

[6] H. THEIL, *A rank invariant method for linear and polynomial regression analysis I, II, III*, Proc. Kon. Ned. Akad. Wetensch. Ser. A **53** (1950) 386–392.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Evolutionary strategies of thermal adaptation to parasite load in a heterogeneous habitat

Jaime Combadão[1]

[1] *Centro de Matemática e Aplicações Fundamentais, Universidade de Lisboa, Avenida Prof. Gama Pinto 2, 1649-003 Lisboa*

emails: combadao@ptmat.fc.ul.pt

Abstract

By evolutionary game theory, we can study the pressure of biotic factors on strategies adopted by predators and prey. Here, we show an analysis of a thermal game between predators and prey, where the habitat is composed of hotter and colder areas. In these areas, predators choose patches based on prey density and preys choose based on operative temperature and on the parasite load.

Key words: game theory, thermoregulation, temperature, parasite load

1. Introduction

Mathematical models about the behavioural regulation of temperature were pioneered by Huey and Slatkin [6]. In these models we formulate how phenotipes are dependent on thermal environments. This means that we evaluate the cost and benefit to be in a specific thermal environment, finding the optimal solutions. Importantly, these models generate hypotheses that can be investigated experimentally, leading to a possible cycle of improvements. Models of interacting predators and preys, with pressure from abiotic factors, are still lacking, but, more recently, effort has been done to further expand the field [3,9].

There are evidences of interaction between biotic factors and thermoregulation. For example: different preference for microclimates in the presence of competitors [5, 4]; reduced time of basking, in ectotherms, in situations of higher risk of predation [7, 8], avoidance of thermal favourable waters [1].

 ISBN 13: 978-84-613-5510-5

2. Thermal game

In this model we use a thermal game, where the prey has the ability to choose the patch that confers a specific energetic gain and a specific risk of predation. Besides that, we simulate a natural situation, where the patch with higher temperature will boost the immune system, being more advantageous for preys with some parasite load. If the prey chooses the patch with higher energetic gain, most of the time, it will lead to a distribution of predators around that some patch, and vice-versa. So, some trade-off must occur, where the need for a high energetic gain is counterbalanced by an evolutionary strategy that minimizes the risk of predation. However, in our model, a broader distribution of the prey will lead to more time spent in less favourable patches for the immune system functions.

3. Model

Our model has a habitat that is divided in two patches, a hotter and a colder, where the hotter one gives a higher energetic gain (e) for the prey. This higher energetic gain will lead to a faster growth rate, and a faster rate of reproduction and, consequently, to a higher fitness (G) of the prey. The fraction of time spent by the prey, in the colder patch will be t_1 and in the hotter one t_2. Similarly, for the predator, the fraction of time spent in the colder patch will be u_1 and u_2 for the hotter patch.
The fitness of the prey is defined as the product of fecundity (F) and probability to survive to maturity (S) [3, 2]:

$$G = FS \tag{1}$$

where we assume that fecundity increases monotonically with the energetic state of the prey (X).
The energetic state of the prey will be defined as the energetic gain of each patch multiplied by the fraction of time spent on it and taking into account the parasite load (I). We assume that the parasite load (that it also is the fraction of the population infected) only affects the energetic gain on the colder patch, while on the hotter patch the boosted immune system lowers the parasite load to minimal levels. So, it follows that:

$$X(t_1, t_2, I) = [e_1 t_1 (1 - I) + e_2 t_2] \tag{2}$$

where I is the parasite load.
For the survival of the prey, we assume that it declines exponentially over time [brown], as a function of rate of predation. This rate depends on the encounters of the prey with predators and on the lethality of that encounter (l). So, the fitness of the prey can be described as:

 ISBN 13: 978-84-613-5510-5

$$G(t_1, t_2, u_1, u_2) = F[e_1 t_1 (1 - I) + e_2 t_2].e^{-l_p(u_1 t_1 + u_2 t_2) - l_i I(u_2 t_2)} \quad (3)$$

where l_p is the lethality due to normal predation and l_i is the increased lethality that an infected prey has due to different behavior when infected. For example, infected ectotherms tend to increase their basking time to maintain a higher body temperature, which leads to an increased probability of being spotted by predators.
The predator's fitness depends on the number of preys that it consumes the efficiency of conversion of the preys and on the time it allocates to each patch. In this model we assume that there is no significant difference for the energetic gain of the predator.

$$H(t_1, t_2, u_1, u_2, N) = \pi N \left[1 - e^{-l_p(u_1 t_1 + u_2 t_2) - l_i I(u_2 t_2)}\right] \quad (4)$$

In equation 4, N is the number of preys in the range of the predator, at the beginning of the season and pi is the efficiency to which the predator converts prey to offspring.

4. Equilibrium

For a strategy of allocation of time to a patch by the prey, there will be an optimum allocation of time for the predator, in order to optimize its fitness. Likewise, for a given allocation of time by the predator, there will be a best strategy of allocation for the prey. The intersection of these set of responses gives us the combination of strategies that the predator and prey can not change unilaterally so that they can achieve greater fitness.
To find the set of strategies above, we study the limit cases for t1, t2 and u1,u2, when their values are 0 or 1, and evaluate the rate of fitness gain ($\partial H/\partial u_i$ or $\partial G/\partial t_i$, where i is 1 or 2). Besides the limit cases, we also evaluate on the range between these limits. If the higher rate of fitness gain is not found in one of the limits above, there will be an equilibrium, where the fitness gain, by changing t_1 (u_1), will be the same as the fitness gain by changing t_2 (u_2), on the opposite direction, as follows:

$$\left.\frac{\partial H}{\partial u_1}\right|_{u_1^*} = \left.\frac{\partial H}{\partial u_2}\right|_{u_2^*} \quad (5)$$

where u_1^* and u_2^* are the values at equilibrium.
For the predator, the patches confer equal rates of fitness gain if the prey spends time in the patches, following the equation:

 ISBN 13: 978-84-613-5510-5

$$t_1 = (1 + \frac{l_i}{l_p} I) t_2 \qquad (6)$$

As t_2 can be evaluated knowing that t_2=1-t_1, by equation 6, we see that the allocation of time giving equal fitness gain, to the predator, in the two patches, is only dependent on the parasite load, the risk of predation and the increased risk of predation by the parasite load, on the hotter patch. It does not depend on the decrease on the energetic state, by the prey, due to the parasite load on the colder patch.
For the prey fitness gain equilibrium, we have a more complex formulation:

$$u_1 = \frac{e_1(1-I) - e_2}{(2l_p + l_i I)X} + \frac{l_p + l_i I}{2l_p + l_i I} \qquad \mathbf{(7)}$$

From equation 7, the fitness gain equilibrium of the prey is shown to depend on t_1,t_2,e_1,e_2,l_p.l_i and I.
If we use equation 6 and equation 7 to draw the curves (lines) that define the equilibrium for the predator and for the prey, we can find the behavioural Nash equilibrium, where these two curves intercept. In figure 1 we can evaluate the impact of raising the parasite load, from left to right, from 0 to 0.5. As the parasite load rises, predators spend more time on the warm patch. The preys start to spend more time on the colder patch (figure 1b), but as I increases, they allocate more of their time to the warm patch.

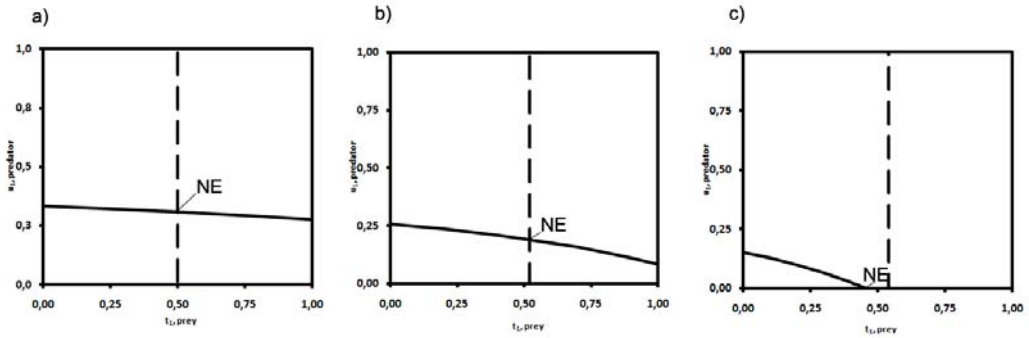


Figure 1. The allocation of time of the prey in the colder patch (t_1) and of the predator (t_2). In a) I=0, I=0.25 in b) and I=0.5 in c). For all graphs, e_1=0.75, e_2=1.0, l_i=0.25 and l_p=0.75. NE is the behavioural Nash equilibrium, where the two curves intercept.

Increasing the lethality of predation, for preys with or without parasite load, leads to more time allocated to the colder patch, for the prey and for the predator, as can be seen in figure 2.

 ISBN 13: 978-84-613-5510-5

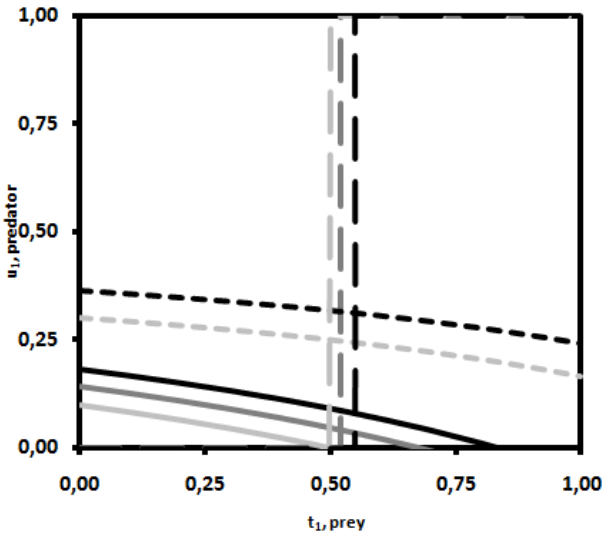


Figure 2. The allocation of time of the prey in the colder patch (t_1) and of the predator (t_2). From the bottom, for the curves: l_i=0, l_p=0.5; l_i=0.25, l_p=0.5; l_i=0.5, l_p=0.5; l_i=0, l_p=1; l_i=1.0, l_p=1.0. The vertical lines correspond to the allocation of time for the prey, for the corresponding curves of the same color. From left to right, t_1=0.5, t_1=0.52 and t_1=0.55. For all, I=0.2, e_1=0.75, e_2=1.0.

These results reinforce that it is necessary to take into account biotic factors to understand the thermoregulation. Particularly, parasite load, in this model, was responsible for a shift in the behaviour of preys. For higher values of parasite load, the prey allocates more time to the warm patch, opposing the strategy of allocating more time to the colder patch when the parasite load is not so severe. The increase in the lethality of predation, due to an increased parasite load, leads to an increase in the time allocated to the colder patch.

References

[1] G.W. Gerald and L.C. Spezzano, *The influence of chemical cues and conspecific density on the temperature selection of a freshwater snail (Melanoides tuberculata)*, Journal of Thermal Biology **30** (2005) 237-245

[2] J.S. BROWN, *Vigilance, patch use and habitat selection: foraging under predation risk*, Evolutionary Ecology Research **1** (1999) 49-71

[3] N.F. HUGHES AND T.C. GRAND, *Physiological ecology meets the ideal-free distribution: predicting the distribution of size-structured fish populations across temperature gradients*, Environmental Biology of Fishes, **59** (2000) 285-298

 ISBN 13: 978-84-613-5510-5

[4] P.A. MEDVICK, J.J. MAGNUSON AND S. SHARR, *Behavioral thermoregulation and social interaction of bluegills, Lepomis macrochirus*, Copeia **1981** (1981) 9-13
[5] P.J. REGAL, *Long term studies with operant conditioning techniques of temperature regulation patterns in reptiles*, Journal of Physiology **63** (1971) 403-406
[6] R.B. HUEY AND M. SLATKIN, *Cost and benefits of lizard thermoregulation*, Quartely Review of Biology **51** (1976) 363-382.
[7] S. DOWNES, *Trading heat and food for safety: costs of predator avoidance in a lizard*, Ecology **82** (2001) 2870-2881
[8] V. Polo, P. López and J. Martin, *Balancing the thermal costs and benefits of refuge use to cope with persistent attacks from predators: a model and an experiment with an alpine lizard*, Evolutionary Ecology Research, **7** (2005) 23-35
[9] W.A. Mitchel and M.J. Angilleta Jr, *Thermal games: frequency-dependent models of thermal adaptation*, Functional Ecology **23** (2009) 510-520

 ISBN 13: 978-84-613-5510-5

Modeling a P-FAIMS with COMSOL

Raquel Cumeras, Isabel Gràcia, Eduard Figueras, Luis Fonseca, Joaquin Santander, Carlos Calaza, Neus Sabaté and Carles Cané

Department of Micro and Nano Systems, Instituto de Microelectrónica de Barcelona, IMB-CNM (CSIC)

emails: raquel.cumeras@imb-cnm.csic.es, isabel.gracia@imb-cnm.csic.es, eduard.figueras@imb-cnm.csic.es, luis.fonseca@imb-cnm.csic.es, joaquin.santander@imb-cnm.csic.es, carlos.calaza@imb-cnm.csic.es, neus.sabaté@imb-cnm.csic.es and carles.cane@imb-cnm.csic.es

Abstract

A micro Planar high-Field Asymmetric waveform Ion Mobility Spectrometer (P-FAIMS) has been simulated in air at ambient pressure using COMSOL Multiphysics software. Targeted analytes used in simulations are vapor phase compounds for security applications. In P-FAIMS target ions are discriminated by the application of the proper separation voltages to the electrodes of the system. By modeling, optimum voltages for achieving the proper sensitivity have been obtained and dual detection is achieved for ions with opposite charges.

Key words: FAIMS; FEM Gas Simulation; COMSOL

1. Introduction

Ion Mobility Spectrometry (IMS) is an analytical technique based on ion separation in gaseous phase due to an electric field. The IMS technology has fundamental advantages: high resolution (~ppb) and fast measurements (~ms). Also, ionization and characterization of the sample in IMS instruments occurs at ambient pressure[1], allowing a smaller analytical unit, lower power requirements, lighter weight and easier use for field applications. IMS instruments have a typical minimum volume of 40cm^3, but due to the trend toward miniaturization of ion mobility spectrometer, smaller volumes (~mm^3) are being explored [2-4]. These advantages make IMS a rapidly advancing technique with a wide spectrum of applications, including detection of chemical warfare agents' and explosives [5].

 ISBN 13: 978-84-613-5510-5

2. P-FAIMS working principle

In presence of an electric field, ions with different collision cross-sections temporally separate based on the frequency of ion-neutral interactions. The continual micro-scale acceleration and scattering collisions deceleration of ions results in a constant average velocity, the drift velocity v_d (m/s); that is directly proportional to the magnitude of the applied electric field strength E (V/cm) [1]:

$$v_d = K \cdot E \qquad (1)$$

where K(cm^2/Vs) is the ion mobility coefficient. This parameter is characteristic of each ion and each medium, and is the basis for its identification. The mobility of a given ion at constant temperature and pressure with gas density N (m^{-3}) through a drift gas under the influence of a high electric field can be expressed by [6]:

$$K(E/N) = K(0) \cdot [1 + \alpha(E/N)] = K(0) \cdot [1 + \alpha_2 \cdot (E/N)^2 + \alpha_4 \cdot (E/N)^4 + ...] \qquad (2)$$

where $K(0)=K(E/N)_{|E=0}$ is the ion mobility at low electric field at N; and $\alpha(E/N)$ describes ion mobility dependence on the electric field at a constant density of drift gas at atmospheric pressure and constant temperature. E/N is the electric field in Townsend (1 Td = 10^{-17} Vcm^2) units. Equation 2 is a convenient mathematical expression for the alpha function [7]. $K(0)$, α_2, and α_4, are characteristic of each ion and are obtained experimentally. When electric field exceeds 10000 V/cm ($E/N \sim 40$ Td) the mobility of some ions increase, decrease or remains unchanged.

A high radio frequency (RF) asymmetric electric field is applied to a narrow gap between two parallel plates, while ions are carried between them by a gas flow and undergo oscillations, i.e. perpendicular to gas flow as shown in Fig. 1. One of the plates is grounded and the other is biased at high voltage with an asymmetric waveform, V_{RF}(t), satisfying that its integration over a period has to be zero. While all ions interact with the applied RF field and are drawn towards the drift channel walls, selected ions can be kept in the flowing gas by applying particular low DC voltage or compensation voltage (V_C), prevents the ion migration towards either electrode. Thus, a selected ion passes through the filter electrodes and reaches the detector being this V_C voltage a characteristic of each ion species.

3. Two Dimensional Modeling Planar-FAIMS

COMSOL Multiphysics software is used to simulate the behavior of three different vapor ions in a P-FAIMS. The software takes into account nonlinear combined effects of different forces and concentrations fields. Created model combines fluid dynamics and electric field which have been found to be the most significant effects. Other effects such as electric repulsion in ion cloud due to space charge have been found to be considerably less significant (for the low concentration level simulated, 1ppm) and thus were not included in the simulations presented [8]. A 2D schema of the drift channel model used in the simulations is shown in Fig. 1.

 ISBN 13: 978-84-613-5510-5

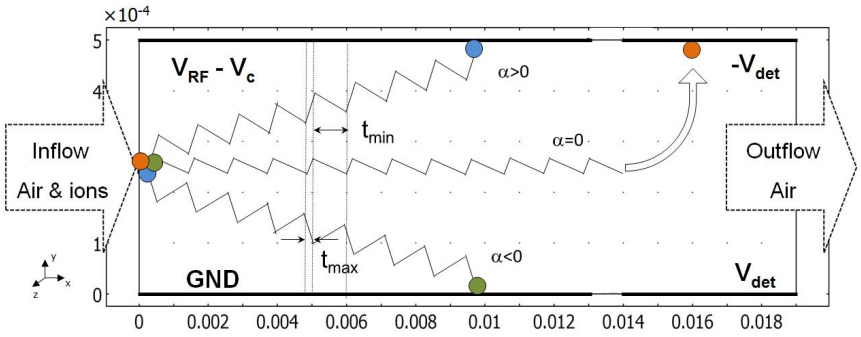


Fig. 1. Schematic of a drift channel defined by P-FAIMS and detector electrodes. Ion paths are schematized under the influence of RF and DC fields for the filtering region and the detector fields for detection region.

The P-FAIMS electrodes of 13×5 mm^2 separated by 0.5 mm gap, are used to produce the alternating electric field in the gap. A two-harmonics asymmetric waveform is applied to the top electrode while the bottom electrode is grounded. The VC is applied to the top electrode too. Detector electrodes (charge collectors) of 5×5 mm^2 are placed after P-FAIMS electrodes to collect ions and generate the V_C spectrum. Drift gas and ions enter the P-FAIMS from the left, passes through the P-FAIMS electrodes and only 'selected' ions reach the detector electrodes. Ions are introduced from the centre of the channel high at the beginning of the P-FAIMS electrodes with a spatial distribution specified as Δy = 0.02 mm, while air gas flows over all the channel height.

Model assumptions are resumed in Fig. 2. Drift gas velocity in the P-FAIMS gap has been calculated using the Navier-Stokes module for air. Electric potentials applied to the P-FAIMS and detector electrodes are calculated using the conductive media DC module and, the movement of ions is calculated with electro kinetic flow module, which takes into account of ions behavior.

The modeled ions correspond to a vapor phase compounds: 1) positive and negative reactant ions in purified air identified as protonated water clusters $H^+(H_2O)_n$ and $O_2^-(H_2O)_n$, 2) a chemical warfare agent simulant positive ion monomer: $DMMPH^+$ that emulates gas sarin. Ions modeled are listed in Table 1 and have been selected because their main properties: K_0, α_2 and α_4 are available in the literature [7].

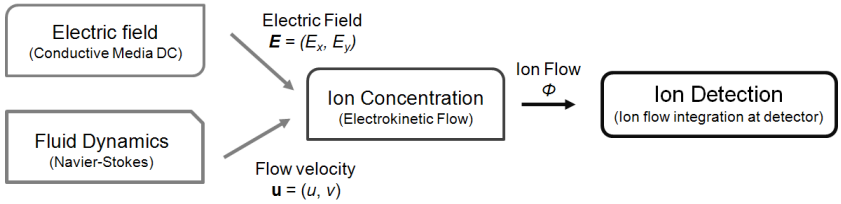


Fig. 2. Block diagram of key computational steps involved in modeling P-FAIMS with COMSOL Multiphysics software. Straight squares indicate main modules and dashed squares indicate variables needed for the modules.

 ISBN 13: 978-84-613-5510-5

Table 1: Parameters used in simulations for the studied compounds [7].

Chemical	Ion Acronym	K_0 (10^{-4} $m^2/V{\cdot}s$)	α_2 (Td^{-2})	α_4 (Td^{-4})
Positive Reactant Ion	$H^+(H_2O)_n$	2.34	$1.78{\cdot}10^{-5}$	$-4.91{\cdot}10^{-10}$
Negative Reactant Ion	$O_2^-(H_2O)_n$	2.13	$1.93{\cdot}10^{-5}$	$-4.30{\cdot}10^{-10}$
Dimethyl methylphosphonate (DMMP)	$DMMPH^+$	1.94	$5.09{\cdot}10^{-6}$	$-1.58{\cdot}10^{-10}$

To simplify numerical simulations, following assumptions are made: 1) All ions are singly charged 2) Ions are assumed to be free from clusters -from water vapor and nitrogen in the ionization process- 3) Ions do not interact with one another, so that interactions resulting from space charging do not occur 4) Ions are created immediately upon entering the analyzer 5) The type of ionization is not considered 6) Reactant ions and product ions are not present in the system 7) Ions do not have dipolar moment.

4. Results and Discussion

For low RF electric fields (E/N < 40 Td) there is no dependence of mobility with electric field, therefore V_C = 0 V for all ions. Increasing electric field they can be separated due to their differences on mobility coefficients.

In Fig. 3, concentrations of the $H^+(H_2O)_n$ and $DMMPH^+$ ions are presented for an E/N of 60Td, showing that with the proper selection of the applied compensation voltage V_C it is possible to obtain a good separation. As could be seen in this case of $DMMPH^+$, ion reach the detector for a V_C=-1.35V. For the same compensation voltage, the positive reactant ion peak $H^+(H_2O)_n$ ion dose not reach the detector. Differentiation is achieved.

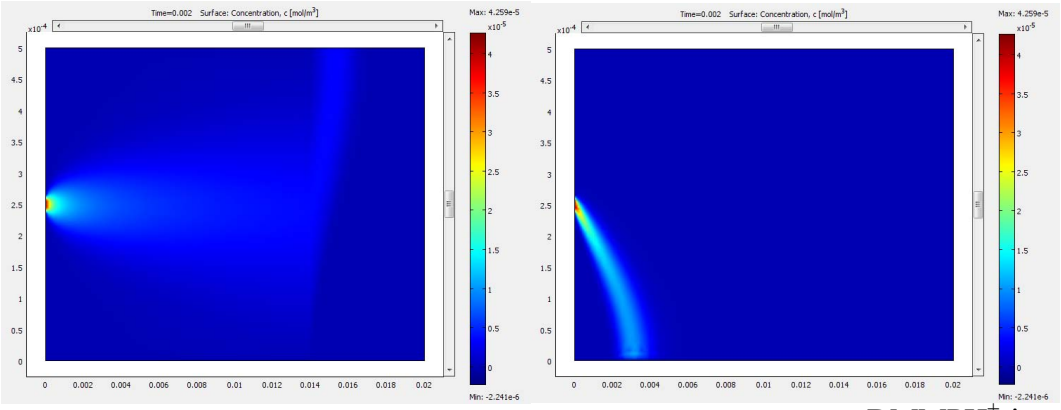


Fig. 3. Concentrations for a separation field of E/N = 60 Td, of LEFT) $DMMPH^+$ ions and RIGHT) $H^+(H_2O)_n$ ions; showing that for the same V_C=-1,35V only the $DMMPH^+$ ion reaches the detector electrode. Differentiation is achieved.

For an E/N=60Td showed, the positive reactant ion $H^+(H_2O)_n$ is detected for V_C=-4.6V, and the negative reactant ion $O_2^-(H_2O)_n$ is detected for V_C=-5.5V. Therefore, differentiation is also obtained for the three compounds studied.

 ISBN 13: 978-84-613-5510-5

Results obtained from simulations showed that ion detection could be achieved with COMSOL software. Obtained intensities for initial concentrations of 1ppm, are in all cases of the order of nA.

5. Conclusions and Prospect

Simulations of a P-FAIMS have been done with COMSOL Multiphysics software for three compounds in vapor phase that could be considered representative for security applications.

Reactant ion peaks has been shown that can be separated from ions of dimethyl methylphosphonate positive monomer ion $DMMPH^+$ applying a determinate V_C that makes ions pass through the drift channel and reach the detector.

From the good simulation results, the fabrication of the P-FAIMS instrument device will be addressed using micro-electro-mechanical systems fabrication techniques.

Acknowledgment

This work has been financially supported by the Spanish Ministry of Education and Science MEC-TEC2007-67962-C04-01 project.

References

[1] G.A. EICEMAN, Z. KARPAS, *Ion Mobility Spectrometry*, second ed, CRC Press, Boca Raton, 1993.

[2] R.A. MILLER, G.A. EICEMAN, E.G. NAZAROV, A.T. KING, *A novel micromachined high-field asymmetric waveform-ion mobility spectrometer*, Sens. Actuator B-Chem. **67** 3 (2000) 300-306.

[3] R. GUEVREMONT, R.W. PURVES, *Atmospheric pressure ion focusing in a high-field asymmetric waveform ion mobility spectrometer*, Rev. Sci. Instrum. **70** 2 (1999) 1370-1383.

[4] M. SALLERAS, A. KALMS, A. KRENKOW, M. KESSLER, J. GOEBEL, G. MULLER, S. MARCO, *Electrostatic shutter design for a miniaturized ion mobility spectrometer*, Sens. Actuator B-Chem. **118** 1-2 (2006) 338-342.

[5] B.M. KOLAKOWSKI, Z. MESTER, *Review of applications of high-field asymmetric waveform ion mobility spectrometry (FAIMS) and differential mobility spectrometry (DMS)*, Analyst **132** 9 (2007) 842-864.

[6] A.A. SHVARTSBURG, *Differential Ion Mobility Spectrometry: Nonlinear Ion Transport and Fundamentals of FAIMS*, first ed, CRC Press, Boca Raton, FL, 2009.

[7] E.V. KRYLOV, E.G. NAZAROV, R.A. MILLER, *Differential mobility spectrometer: Model of operation*, Int. J. Mass Spectrom. **266** 1-3 (2007) 76-85.

[8] D.A. DAHL, T.R. MCJUNKIN, J.R. SCOTT, *Comparison of ion trajectories in vacuum and viscous environments using SIMION: Insights for instrument design*, Int. J. Mass Spectrom. **266** 1-3 (2007) 156-165.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Mathematical model for a temporal-bounded classifier in security environments

Juan F. De Paz[1], Martí Navarro[2], Cristian I. Pinzón[3], Vicente Julián[2], Javier Bajo[1] and Juan. M. Corchado[1]

[1] *Departamento Informática y Automática Universidad de Salamanca*
[2] *Departamento de Sistemas Informaticos y Computacion Universidad Politécnica de Valencia*
[3] *Universidad Tecnológica de Panamá*

emails: fcofds@usal.es, mnavarro@dsic.upv.es, cristian_ivanp@usal.es, vinglada@dsic.upv.es, jbajope@usal.es, corchado@usal.es

Abstract

Security is a major concern when web applications are implemented. This has led to the proposal of a variety of specifications and approaches to provide the necessary security for these environments. SQL Injection attacks on web applications have become one of the most important information security concerns over the past few years. The purpose of this article is to present an adaptive and intelligent mechanism that can handle SQL injection attacks and real time. Our approach is based on a real time classifier agent that incorporates a mixture of experts to choose a specific classification technique depending on the feature of the attack and the time available to solve the classification. This research presents a case study to evaluate the effectiveness of the approach and also presents the preliminary results obtained with an initial prototype.

Key words: Case-Based Reasoning, Support Vector Machine, Artificial Neural Network, SQL Injection, Intrusion Detection

1. Introduction

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments

In recent years, Internet attacks have increased due to the large number of information systems connected to the Internet. One of the most serious security threats around Web application and databases has been the SQL Injection attack [1]. This attack takes place at the database layer when a user request that has been sent through an HTTP request is executed without prior validation. Confidentiality, integrity and availability are the main objectives of any information security model [2]. Various approaches have attempted to deal with the problem of SQL injections [3] [4] [5] [6] [7]. However, the biggest inconvenience of these solutions is their inability to adapt to the rapid changes in attack patterns, which renders them a bit inefficient in the long term. More complex SQL attacks are characterized by the various techniques used for remaining undetected by existing security solutions. Finally, none of these approaches consider the limitations or restrictions in response time.

Response time is a critical aspect in the majority of Internet security systems. With systems requiring a response to be given before a specific deadline, as determined by the system needs, it is essential that the execution time for each of the tasks carried out by the system is predictable and capable of guaranteeing correct execution within the time needed for the given response. Furthermore, the system providing the service and the security analysis must both have the necessary mechanisms for executing tasks in a predictable framework, that is, the agent must be prepared for its execution in a real time environment. A Real-Time Agent may have its interactions bounded; a modification that will affect all the communication processes in the Multi-Agent system where the Real-Time Agent is located. Some examples of real time agents are: The ARTIS agent specifically designed to develop Real-Time Systems [8] [9] [10], The ObjectAgent Architecture developed by Pinceton Satellites in 2001[9] and time-aware agents proposed by Prouskas et al. in 2002 [10].

This study presents a new agent model with a novel perspective for analyzing and classifying SQL injections in real time. The agent's internal structure, an integrated mixture of an Artificial Neural Network (ANN) and a Support Vector Machine (SVM) is used as a classification mechanism. By using this mixture, it is possible to exploit the advantages of both strategies in order to classify the SQL queries in a more reliable way. The internal structure of the agent is based on the Case-Base Reasoning (CBR) model, with the main difference being that the different CBR phases are time-bounded, thus enabling its use in real time. CBR can be very suitable for application in agent reasoning, where similar problems should have similar solutions. However, few of the existing approaches cope with the problem of applying CBR as deliberative engine for agents in MAS with real-time constraints. Additionally, the adaptation phase in the CBR system integrated in the agent proposes a new analysis classification model that is carried out by a mixture of experts. The concept of a mixture of experts was first proposed by[11]. It involves a system that contains a series of input data that is distributed over a set of expert classifiers. Depending on the time available for performing classification, a set of experts is selected to perform the different analyses. The experts are selected with a multiple method model [12]. Finally the different

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments selected experts generate the predictions and the outputs are fused to generate a new unique result [13] [25].

The paper is structured as follows: Section 2 presents the problem that has prompted most of this research work. Section 3 describes the SQL attack problem. Section 4 shows a general view of the temporal bounded CBR used as deliberative mechanism in the classifier agent. Section 5 describes a set of tests to evaluate our proposal.

2. Real Time Agent and Case-Based Reasoning (CBR)

A real time agent is one that is able to support tasks that should be performed within a restricted period of time [14]. This characteristic justifies its use in real time systems. In this type of environment, the validity of the solution is determined not only by its correct execution, but by its ability to be carried out within the allotted time frame [15].

The main problem in the architecture of a Real Time Agent (RTA) is with the deliberation process. This process may use Artificial Intelligence (AI) techniques as problem-solving methods to compute more intelligent actions. If this is the case, it is difficult to know the time required, because it can either be unbounded or have a high variability. If the agent has to operate in a real-time environment, the agent complexity required to achieve any or all of these features is greatly increased. Thus a RTA requires an efficient integration of high-level, deliberative processes within reactive processes. When using AI methods, it is necessary to provide techniques that allow their response times to be bounded. These techniques are mainly based on well-known Real-Time Artificial Intelligence System (RTAIS) techniques[16][17].

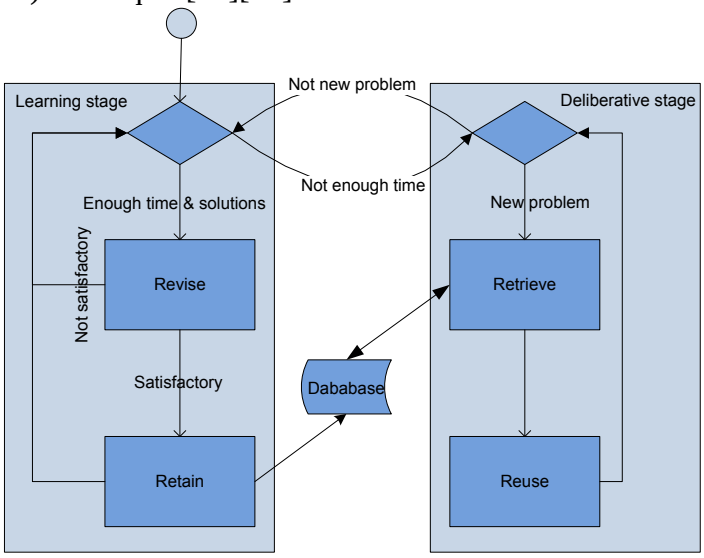


Figure 1. TB-CBR cycle

 ISBN 13: 978-84-613-5510-5

Figure 1 shows the reasoning cycle for a TB-CBR system. The TB-CBR cycle starts at the learning stage, where it checks to see if there are previous cases waiting to be revised and possibly stored in the case-base. In our model, the plans provided at the end of the deliberative stage will be stored in a solution list while feedback about their utility is received. When each new TB-CBR cycle begins, this list is accessed. If there is enough time, the learning stage is implemented for those cases whose solution feedback has been recently received. If the list is empty, this process is omitted.
The next stage to be implemented is the deliberative stage. The retrieval algorithm is used to search the case-base and retrieve a case that is similar to the current case (i.e. one that characterizes the problem to be solved). Each time a similar case is found, it is sent to the reuse phase where it is transformed into a suitable plan for the current problem by using a reuse algorithm. Therefore, at the end of each iteration of the deliberative stage, the TB-CBR method is able to provide a plan for the problem at hand, although this plan can be improved in subsequent iterations if the deliberative stage has enough time to perform them.
Hence, the temporal cost of executing the cognitive task is greater than or equal to the sum of the execution times of the learning and deliberative stages (as shown in equation 1):

$$\begin{aligned} t_{cognitiveTask} &\geq t_{learning} + t_{deliberative} \\ t_{learning} &\geq (t_{revise} + t_{retain}) * n \\ t_{deliberative} &\geq) t_{retrieve} + t_{reuse}) * m \end{aligned} \qquad (1)$$

where $t_{cognitiveTask}$ is the maximum time available for the agent to provide a response , $t_{learning}$ and $t_{deliberative}$ are the total execution time of the learning and deliberative stages; t_x is the execution time of the phase x and n and m are the number of iterations of the learning and deliberative stages respectively.

This algorithm can be launched when the real-time agent considers it appropriate and there is enough time for it to be executed. The real-time agent indicates to the TB-CBR the maximum time (tmax, where $t_{\max} \geq t_{cognitiveTask}$) that is available to complete its execution cycle. The time tmax must be divided between the learning and the deliberative stages to guarantee the execution of each stage. The designer can assign more time to the learning stage if it desires a real-time agent with a greater capacity to learn.
The anytime behaviour of the TB-CBR is achieved through the use of two loop control sequences. The loop condition is built using the $enoughTime$ function, which determines if a new iteration is possible according to the total time that the TB-CBR has to complete each stage.
The first phase of the algorithm executes the learning stage. This stage is executed only if the agent has the solutions from previous executions stored in the

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments solutionQueue. The solutions are stored just after the end of the deliberative stage. The deliberative stage is only launched if the agent has a problem to solve in the problemQueue. This configuration allows the agent to launch the TB-CBR in order to only learn (no solution is needed and the agent has enough time to reason previous decisions), only deliberate (there are no previous solutions to consider and there is a new problem to solve) or both.

2. SQL attack

A SQL injection attack takes place when a hacker changes the semantic or syntactic logic of a SQL text string by inserting SQL keywords or special symbols within the original SQL command that will be executed at the database layer of an application [18]. A SQL injection attack can cause serious damage to an organization, including financial loss, breach of trust with clients, among others.
There have been many proposed solutions for SQL injection attacks, including some Artificial intelligence techniques. One of the approaches is WAVES (Web Application Vulnerability and Error Scaner)[22]. This solution is based on a black-box technique. WAVES is a web crawler that identifies vulnerable points, and then builds attacks that target those points based on a list of patterns and attack techniques. WAVES monitors the response from the application and uses a machine learning technique to improve the attack methodology. WAVES cannot check all the vulnerable points like the traditional penetration testing. The strategy used by the intrusion detection systems has also even been implemented to deal with some SQL injection attacks. Valeur [21] presents an IDS approach that uses a machine learning technique based on a dataset of legal transactions. These are used during the training phase prior to monitoring and classifying malicious accesses. Generally, IDS systems depend on the quality of the training set; a poor training set would result in a large number of false positives and negatives. Skaruz [19] proposes the use of a recurrent neural network (RNN). The detection problem becomes a time serial prediction problem. The main problem with this approach is the large number of false positives
and false negatives.
Other strategies based on string analysis techniques and the generation of dynamic models have been proposed as solutions to SQL injection attacks. Halfond and Orso [18] propose AMNESIA (Analysis and Monitoring for Neutralizing SQL Injection Attacks). Kosuga et al. proposes SANIA (Syntactic and Semantic Analysis for Automated Testing against SQL Injection) [20]. With only slight variations of accuracy in the models, these strategies have as drawback their meaningful rate of false positives and negatives.

2. SQL-TB-CBR agent classifier

In this section the new SQL-TB-CBR agent is presented, with special attention paid to its internal structure and the classification mechanism of SQL attacks. This mechanism combines the advantages of CBR systems, such as learning and adaptation, real time, with the predictive capabilities of ANNs and SVMs.

Mathematical model for a temporal-bounded classifier in security environments In terms of CBR, the case is composed of elements of the analysed SQL Query described as follows:

- Problem: describes the initial information available for generating a plan. The problem description consists of: case identification, user session and SQL query elements.
- Solution: states the action carried out in order to solve the problem. In this case, the applied prediction models.
- Final State: describes the state achieved after that the solution has been applied.

The fields defining a case are as follows: *IdCase, Session, User, IP_Address, Query_SQL, Affected_table, Affected_field, Command_type, Word_GroupBy, Word_Having, Word_OrderBy, Numer_And, Numer_Or, Number_literals, Number_LOL, Length_SQL_String, Start_Time_Execution, End_Time_Execution,* and *Query_Category*. Additionally, the information related to the prediction models used is stored as well.

In Fig. 1, the different stages applied in the reasoning cycle can be seen.

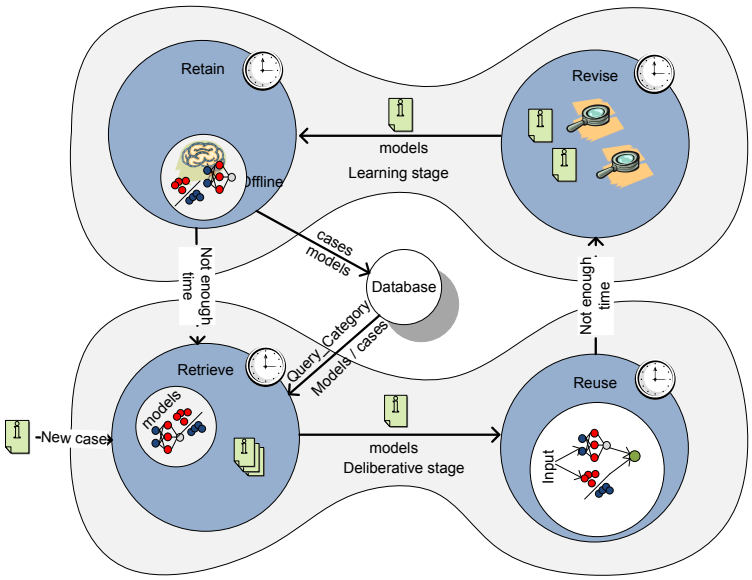


Figure 2. TB-CBR cycle and classification mechanism of the SQL-TB-CBR agent

In the retrieval stage, there is a selection of queries sorted by type and by the memory's classification models. In the reuse phase, as seen in Fig. 1, a Multilayer Perceptron (MLP) and/or an SVM are applied to carry out the prediction of the new query. Subsequently, a new inspection is performed which can be done automatically or by a human expert. In the case of the query resulting as suspicious, further inspection will be carried out manually by a human expert. During learning, memory information regarding the cases and models will be

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments updated. Below, the different stages of the CBR reasoning cycle associated with the system are described in more detail.

Retrieve

In the Retrieve phase, the real time agent recovers the cases that it will use to perform classification. The time needed to recover the different cases to be used is clearly defined and temporally bounded. The retrieval time for the cases depends on the number of cases in the case base. If the number is known, it is easy to predict how much execution time will be used to recover the cases. The asyntotic cost is linear (O(n)).

The retrieve phase is broken down into two phases; case retrieval and model retrieval. Case retrieval is performed by using the Query_Category attribute which retrieves queries from the case memory (Cr) which were used for a similar query in accordance with attributes of the new case c_n. Subsequently, the models for the multilayer perceptron and/or SVM associated with the recovered cases are retrieved. The recovery of these memory models allows the improvement of the system's performance so that the time necessary for the creation of models will be considerably reduced, mainly in the case of the ANN training.

Reuse

The SQL injection in our proposal can be analyzed by two different techniques. Execution time in both cases is known, since previous stored models as used. The first is known as the Light technique Support Vector Machine (SVM) and is usually a detection algorithm with a low temporal cost, but of low quality as well. Using the Heavy technique, Multiplayer Perceptron, the result of the analysis is much more exact, but it requires a much higher amount of execution time. The inputs of the MLP are: Query_SQL, Affected_table, Affected_field, Command_type, Word_GroupBy, Word_Having, Word_OrderBy, Numer_And, Numer_Or, Number_literals, Number_LOL, and Length_SQL_String. The number of neurons in the hidden layer is 2n+1, where n is the number of neurons in the input layer. Finally, there is one neuron in the output layer. The activation function selected for the different layers has been the sigmoid. Taking into account the activation function f_j, the calculation of output values are given by the following expression

$$y_j^p = f_j\left(\sum_{i=1}^{N} w_{ji}(t)\, x_i^p(t) + \theta_j\right) \qquad (2)$$

The outputs correspond to x_r. As the neurons exiting from the hidden layer of the neural network contain sigmoidal neurons with values between [0, 1], the incoming variables are redefined so that their range falls between [0.2, 0.8]. This transformation is necessary because the network does not deal with values that fall outside of this range. The outgoing values are similarly limited to the range of [0.2, 0.8] with the value of 0.2 corresponding to a non-attack and the value of 0.8

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments corresponding to an attack. The network training is carried out through the error Backpropagation Algorithm [23].

The light algorithm SVM represents an extension of nonlinear models [24]. SVM also allows the separation of element classes which are not linearly separable. To do so, the space of initial coordinates is mapped in a high dimensionality space through the use of functions. Due to the fact that the dimensionality of the new space can be very high, it is not feasible to calculate hyperplanes that allow the production of linear separability. For this reason, a series of non-linear functions called kernels is used.

Let us consider a set of patterns $T = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$ where xi is a vector of the dimension n. The idea is to convert the elements xi in a space of high dimensionality through the application of a function, in such a way that the set of original patterns is converted into the following set $\Phi(T) = \{(\Phi(x_1), y_1), (\Phi(x_2), y_2), ..., (\Phi(x_m), y_m)\}$ that, depending on the selected function $\Phi(x)$, could be linearly separable. To carry out the classification, this equation sign is studied [16]:

$$class(x_k) = sign\left(\sum_{i=1}^{m} \lambda_i y_i \Phi(x_i)\Phi(x_k) + b\right) \quad (3)$$

The selected kernel function in this problem was polynomial. The values used for the estimation are dominated by decision values and are related to the distance from the points to the hyperplane.

If there is enough time for carried out both techniques, the mixture is performed in such a way that the higher value is selected for both methods. This is done in order to avoid false negatives.

Revise and Retain

The revise phase can be manual or automatic depending on the output values. The automatic review is given for non-suspicious cases during the estimation obtained for the reuse phase. For cases detected as suspicious, with output values determined experimentally in the interval [0.35, 0.6]), a review by a human expert is performed.

The learning phase updates the information of the new classified case and reconstructs the classifiers offline to leave the system available for new classifications. The ANN classifier is reconstructed only when an erroneous classification is produced. In the case of a reference to inspection of suspicious queries, information and classifiers are updated when the expert updates the information.

 ISBN 13: 978-84-613-5510-5

2. Results and Conclusions

This article has presented a novel proposal for detecting SQL injections in real time. The article proposes a new vision in which each attack mechanism is individually analyzed. It also makes it possible to obtain better classification results with regard to both the effectiveness of the classification process and the response time, since all classification mechanism tasks are temporally bounded.

In order to validate the initial prototype, we proposed a benchmark case study that contains 705 SQL queries (437 legal queries and 268 attacks). The tests were conducted with a simple web application with database access, MySQL 5.0. The entries were automated by using the SQLMap 0.6.3 tool, with which an initial case base was established for training the SQL-TB-CBRClassifier.

Prior to initiating the tests, the attack classification mechanisms were analyzed for each use of a Light or Heavy technique and other classifiers. To analyze the successful rates, a test of the classification of queries was conducted, taking into account the following classifiers: Bayesian Network, Naive Bayes, AdaBoost M1, Bagging, DecisionStump, J48, JRIP, LMT, Logistic, LogitBoost, MultiBoosting AdaBoost, OneR, SMO, light planner, heavy planner. The different classifiers were applied to 705 previously classified queries.

Table 1. Total number of hits for the different classifiers.

Method		Method		Method	
BayesNet	638	Naive Bayes	666	AdaBoostM1	665
Bagging	684	DecisionStump	598	J48	689
JRIP	692	LMT	693	Logistic	688
LogitBoost	680	MultiBoostAB	666	OneR	622
SMO	685	light	696	heavi	**702**

The analysis demonstrated that the use of Heavy techniques provided a better classification, but with a greater temporal cost. The average execution time for the queries, and the worst time used for the ligth and heavy techniques were respectively 0.013/0.051 and 0.28/1.07 ms.

For the second test a set of 50 queries were selected and then classified according to different pre-determined deadlines. The number of executions and errors obtained for each of the classifiers are shown in Figure 3. The x axis represents the average time between queries and the deadline, while the y axis represents the number of queries executed. As can be seen, the number of executions for the mixture increased as the execution time between queries increased.

 ISBN 13: 978-84-613-5510-5

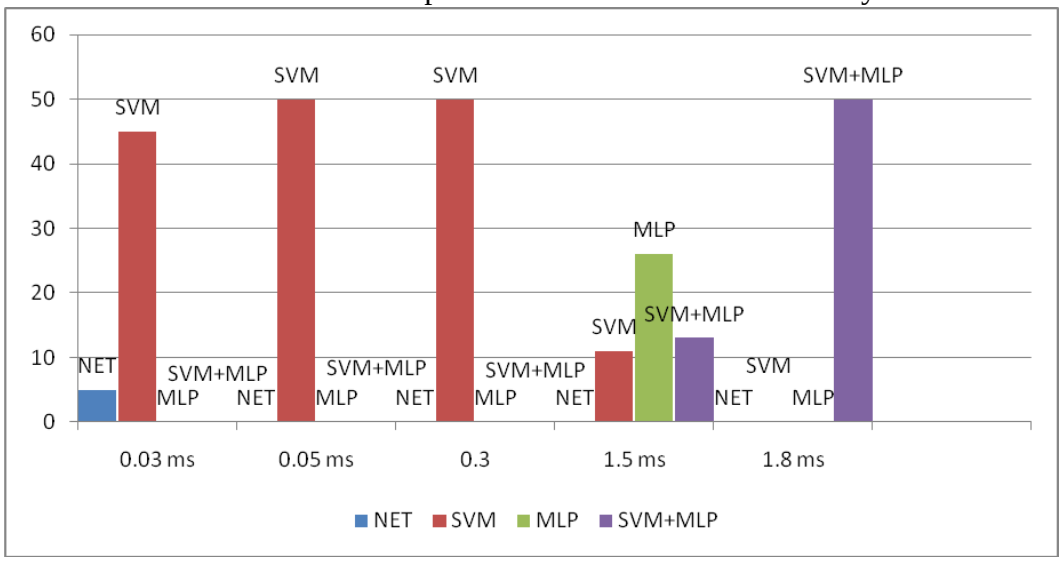


Figure 3. Queries made for each combination according to time.

The proposed SQL-TB-CBR agent is capable of detecting SQL injections with low error rates compared with other existing techniques, as shown in table 1. Moreover, it is possible to provide a real-time classifier mechanism, with a high level of confidence to identify legal queries and attacks. The combination of different Artificial Intelligence paradigms allows the development of a hybrid intelligent system with characteristics such as the capacity for learning and reasoning, flexibility and robustness which make the detection of SQL injection attacks possible

Acknowledgements

This work has been supported by the MICINN TIN 2009-13839-C03-03 projectand and the Professional Excellence Program 2006-2010 IFARHU-SENACYT-Panama.

References

[1] W. G. J. Halfond, et al. A Classification of SQL-Injection Attacks and Countermeasures, in Proceedings of the IEEE International Symposium on Secure Software Engineering, Arlington, VA, USA, 2006.

[2] M. Stamp, Information Security: Principles and Practice, Wiley InterScience, 2006.

[3] W. Halfond and A. Orso AMNESIA: Analysis and Monitoring for Neutralizing SQLinjection Attacks. In: 20th IEEE/ACM international Conference on Automated software engineering, (2005) 174-183.

 ISBN 13: 978-84-613-5510-5

[4] J. SKARUZ AND F. SEREDYNSKI RECURRENT NEURAL NETWORKS TOWARDS DETECTION OF SQL ATTACKS. IN: 21TH INTERNATIONAL PARALLEL AND DISTRIBUTED PROCESSING SYMPOSIUM, IEEE INTERNATIONAL, (2007) 1-8

[5] Y. KOSUGA, K. KONO, M. HANAOKA, M. HISHIYAMA AND Y. TAKAHAMA SANIA: SYNTACTIC AND SEMANTIC ANALYSIS FOR AUTOMATED TESTING AGAINST SQL INJECTION. IN: 23RD ANNUAL COMPUTER SECURITY APPLICATIONS CONFERENCE, IEEE COMPUTER SOCIETY (2007) 107-117

[6] F. VALEUR, D. MUTZ AND G. VIGNA A LEARNING-BASED APPROACH TO THE DETECTION OF SQL ATTACKS. IN: PROCEEDINGS OF THE CONFERENCE ON DETECTION OF INTRUSIONS AND MALWARE AND VULNERABILITY ASSESSMENT (2005) 123-140

[7] Y. HUANG, S. HUANG, T. LIN AND C. TSAI WEB APPLICATION SECURITY ASSESSMENT BY FAULT INJECTION AND BEHAVIOR MONITORING. IN: 12TH INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, ACM, USA (2003) 148-159

[8] C. CARRASCOSA, A. TERRASA, F.A. GARCÍA, A. ESPINOSA, V.J. BOTTI, A META-REASONING MODEL FOR HARD REAL-TIME AGENTS, IN: CAEPIA, (2005) 42-51.

[9] D.M. SURKA, M.C. BRITO, C.G. HARVEY, THE REAL-TIME OBJECT AGENT SOFTWARE ARCHITECTURE FOR DISTRIBUTED SATELLITE SYSTEMS, IN: PROCEEDINGS OF THE IEEE AEROSPACE CONFERENCE, (2001) 2731-2741.

[10] K. PROUSKAS, J. PITT, TOWARDS A REAL-TIME ARCHITECTURE FOR TIME-AWARE AGENTS, IN: FIRST INTERNATIONAL JOINT CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS (AAMAS '02), ACM, NEW YORK, NY, USA, (2002) 92-93.

[11] R.A. JACOBS, M.I. JORDAN, S.J. NOWLAN, G.E. HINTON, ADAPTIVE MIXTURES OF LOCAL EXPERTS, NEURAL COMPUTING, 3 (1991) 79-87.

[12] A.J. GARVEY, V.R. LESSER, DESIGN-TO-TIME REAL-TIME SCHEDULING, SYSTEMS, MAN AND CYBERNETICS, IEEE TRANSACTIONS ON, 23 (1993) 1491-1502.

[13] A. SUBASI, EEG SIGNAL CLASSIFICATION USING WAVELET FEATURE EXTRACTION AND A MIXTURE OF EXPERT MODEL, EXPERT SYSTEMS WITH APPLICATIONS, 32 (2007) 1084-1093.

[14] V. JULIAN, V. BOTTI, DEVELOPING REAL-TIME MULTI-AGENT SYSTEMS, INTEGRATED COMPUTER-AIDED ENGINEERING, 11 (2004) 135-149.

[15] J.A. STANKOVIC, MISCONCEPTIONS ABOUT REAL-TIME COMPUTING: A SERIOUS PROBLEM FOR NEXT-GENERATION SYSTEMS, IEEE COMPUTER, 21 (1988) 10-19.

[16] A. GARVEY, V. LESSER, A SURVEY OF RESEARCH IN DELIBERATIVE REAL-TIME ARTIFICIAL INTELLIGENCE, REAL-TIME SYSTEMS, 6 (1994) 317-347.

[17] V.J. BOTTI, C. CARRASCOSA, J. VICENTE, J. SOLER, MODELLING AGENTS IN HARD REAL-TIME ENVIRONMENTS, IN: 9TH EUROPEAN WORKSHOP ON MODELLING AUTONOMOUS AGENTS IN A MULTI-AGENT WORLD (MAAMAW '99), SPRINGER-VERLAG, LONDON, UK, (1999) 63-76.

 ISBN 13: 978-84-613-5510-5

[18] W. HALFOND AND A. ORSO AMNESIA: ANALYSIS AND MONITORING FOR NEUTRALIZING SQLINJECTION ATTACKS. IN: 20TH IEEE/ACM INTERNATIONAL CONFERENCE ON AUTOMATED SOFTWARE ENGINEERING, (2005) 174-183.

[19] J. SKARUZ, AND F. SEREDYNSKI RECURRENT NEURAL NETWORKS TOWARDS DETECTION OF SQL ATTACKS. IN: 21TH INTERNATIONAL PARALLEL AND DISTRIBUTED PROCESSING SYMPOSIUM, IEEE INTERNATIONAL, (2007) 1-8.

[20] Y. KOSUGA, K. KONO, M. HANAOKA, M. HISHIYAMA AND Y. TAKAHAMA SANIA: SYNTACTIC AND SEMANTIC ANALYSIS FOR AUTOMATED TESTING AGAINST SQL INJECTION. IN: 23RD ANNUAL COMPUTER SECURITY APPLICATIONS CONFERENCE, IEEE COMPUTER SOCIETY, (2007) 107-117

[21] F. VALEUR, D. MUTZ AND G. VIGNA A LEARNING-BASED APPROACH TO THE DETECTION OF SQL ATTACKS. IN: PROCEEDINGS OF THE CONFERENCE ON DETECTION OF INTRUSIONS AND MALWARE AND VULNERABILITY ASSESSMENT, (2005) 123-140.

[22] Y. HUANG, S. HUANG, T. LIN AND C. TSAI WEB APPLICATION SECURITY ASSESSMENT BY FAULT INJECTION AND BEHAVIOR MONITORING. IN: 12TH INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, ACM, (2003) 148-159.

[23] Y. LECUN, ET AL., EFFICIENT BACKPROP, IN NEURAL NETWORKS: TRICKS OF THE TRADE, (1998) 546.

[24] E. CORCHADO AND C. FYFE, CONNECTIONIST TECHNIQUES FOR THE IDENTIFICATION AND SUPPRESSION OF INTERFERING UNDERLYING FACTORS INTERNATIONAL JOURNAL OF PATTERN RECOGNITION AND ARTIFICIAL INTELLIGENCE, VOL. 17, (2003) 1447-1466.

[25] A. HERRERO, E. CORCHADO, L. SAIZ AND A. ABRAHAM. DIPIP: A NEURAL KNOWLEDGE MANAGEMENT MODEL FRAMEWORK FOR DECISION SUPPORT, COMPUTATIONAL INTELLIGENCE 26 (1) (2010) 26-56.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Implementation in Chimère of a conservative solver for the advection equation

Luis Gavete[1], Marta García Vivanco[2], Pedro Molina[1], M. Lucía Gavete[3], Francisco Ureña[4] y Juan José Benito[5]
[1]*Universidad Politécnica de Madrid, Spain*
[2] *C.I.E.M.A.T. Madrid, Spain*
[3] *Universidad Rey Juan Carlos, Madrid, Spain*
[4]*Universidad Castilla-la Mancha,Ciudad Real, Spain*
[5]*U.N.E.D., Madrid, Spain*

emails: lu.gavete@upm.es
m.garcia@ciemat.es, p.molina@upm.es, lucia.gavete@urjc.es
francisco.urena@uclm.es, jbenito@ind.uned.es

Abstract

Extensive research has been performed to solve the advection equation and different numerical methods have been proposed. Most part of these methods including semi-lagrangian methods are not conservative. In this paper we present the implementation in the European scale Eulerian chemistry transport model CHIMERE of an exactly conservative method for the advection equation. The results of the method are compared with a set of observation sites in the area of Madrid(Spain).

Key words: advection equation, conservative scheme, rational interpolation
MSC2000: AMS Codes (optional)

1. Introduction

Accurate numerical simulation of tropospheric air pollution phenomena has become a major challenge in atmospheric science. The advection equation is important for the study of the dynamics of flows, as well for the development of new numerical schemes that are applied to more complex models. Semi-

 ISBN 13: 978-84-613-5510-5

Lagrangian schemes have gathered a wide acceptance for solving advection dominated problems, especially in the atmospheric sciences [1]. Most part of these methods are not conservative. Apart from the corrective methods different approaches have been developed to generate an inherently conservative solution of advection equation. The conservative methods are used in meteorological simulations as very accurate methods for explicitly computing the transport of rain water.

Our main goal of this research is to improve the transport module in the European scale Eulerian chemistry transport model CHIMERE by implementing a conservative rational scheme. The results of the method are compared with a set of observation sites in the area of Madrid (Spain).

Section 2 introduces the conservative methods. Section 3 gives a description of the rational interpolation and Section 4 introduces the conservative formulation. In section 5 we introduce the European-scale chemistry-transport model (CHIMERE). The comparison of observed and modeled data is given in Section 6, and finally some conclusions are given in Section 7.

2. Conservative methods

One of the main drawbacks of the semi-Lagrangian method is the lack of conservation properties in its original formulation. Therefore, several authors presented stable semi-Lagrangian schemes that were modified such that conservations properties hold [2, 3]

Some authors have recently succeeded in developing new conservative semi-Lagrangian schemes. The schemes conserve the mass using an additional constraint of the value integrated over neighbouring two grid points, this value is introduced as a new model variable that is updated for the advection equation by a flux-form formulation. Then the mass can be exactly conserved. One of these conservative methods, developed by Xiao and Pen [4], is based in rational interpolation. In this scheme a constraint of the conservation relation for cell-integrated average is imposed at the stage to determine the piecewise rational interpolation function. The method conserves exactly the cell-integrated average of the transported field. In this paper we have implemented this conservative method in the European-scale chemistry transport model (CHIMERE). We have adopted the dimensional splitting to extend the scheme to multi-dimensions.

We describe the numerical formulation for the scalar conservative advection transport equation as follows:

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = 0 \tag{1}$$

where $f(x,t)$ is the advected quantity, and u is the advecting current. This is a linear, first-order, partial differential equation with a constant coefficient namely

 ISBN 13: 978-84-613-5510-5

u (velocity). When the velocity is constant, the solution of equation (1) gives a simple translational motion of field *f* with velocity u.
The value of f in (n+1) step is readily obtained by shifting the profile by $u_i \Delta t$, so that:

$$f(x_{i+1}, t+\Delta t) = f(x_{i+1} - u\Delta x, t) = F_i^n(x_{i+1} - u\Delta x) \tag{2}$$

where F is an interpolation function which depends on the conservative algorithm we are using.

3. Rational conservative interpolation

Normally the interpolation function is based on polynomials. The main disadvantage of the polynomial interpolation is that can be unstable on the most common grid – equidistant grid. The rational interpolation consists of the representation of a given function as the quotient of two polynomials. The rational interpolation is an alternative for the polynomial interpolation. Its advantages are the high accuracy and absence of the problems which are typical for polynomial interpolation, such as the typical oscillations. However new difficulties can appear in the rational interpolation due to the existence of the poles.
The conservative interpolation is based on the concept of the conservation of the integral of the function, using an additional constraint of the value integrated over neighbouring two grid points. Both properties can be used at the same time in the piecewise rational conservative interpolation.
The rational interpolation function F, is expressed in a generic mesh element with boundaries $x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$ and considering the velocity u<0 as

$$F_i(x) = \frac{a_i + 2b_i(x - x_{i-1/2}) + \beta_i b_i (x - x_{i-1/2})^2}{1 + \beta_i (x - x_{i-1/2})^2} \tag{3}$$

where a, b, and β are the coefficients of the interpolation function which are obtained by using the constraint conditions, where the constraint conditions are given in $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ by

$$\begin{aligned} F_i(x_{i-\frac{1}{2}}) &= f^n_{i-\frac{1}{2}} \\ F_i(x_{i+\frac{1}{2}}) &= f^n_{i+\frac{1}{2}} \\ \frac{1}{\Delta x_i} \int_{x_i-\frac{1}{2}}^{x_i+\frac{1}{2}} F_i(x)dx &= \rho_i^n \quad \text{where} \quad \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \end{aligned} \tag{4}$$

 ISBN 13: 978-84-613-5510-5

Then by solving the three equations we obtain the coefficients of the rational interpolation.

$$a = f^{n}_{i-\frac{1}{2}}$$

$$b_i = \beta_i \rho_i^n + \frac{1}{\Delta x_i}(\rho_i^n - f^{n}_{i-1/2}) \tag{5}$$

$$\beta_i = -\Delta x_i^{-1}\left[\frac{\left|f^{n}_{i-1/2} - \rho_i^n\right|}{\left|\rho_i^n - f^{n}_{i+1/2}\right|} - 1\right]$$

This last expression is corrected according with Xiao and Peng[4], to avoid division by zero as follows

$$\beta_i = -\Delta x_i^{-1}\left[\frac{\left|f^{n}_{i-1/2} - \rho_i^n\right| + 10^{-20}}{\left|\rho_i^n - \rho^{n}_{i+1/2}\right| + 10^{-20}} - 1\right] \tag{6}$$

4. Conservative Formulation.

For the pollution problem, it is appropiate to use finite volume method. We divide the spatial domain in cells called finite or control volumes C_i, this corresponds in one dimension to a partition of a bounded domain by intervals, see Fig. 1.

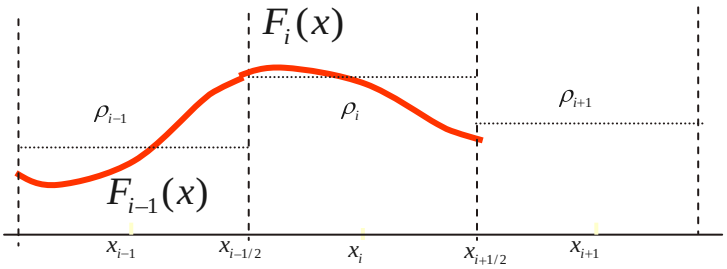


Fig. 1 Control volumes

On the other hand one built discrete equations from the integral form of the advection equation. The integral form of the conservation law is given by

$$\frac{d}{dt}\int_{C_i} f(x,t)dx = g(f(x_{i-(1/2)},t) - g(f(x_{i+(1/2)},t) \tag{7}$$

 ISBN 13: 978-84-613-5510-5

being $g(f(x_{i-(1/2)},t)$, $g(f(x_{i+(1/2)},t)$ the fluxes in the cell.

By integrating in the time

$$\int_{C_i} f(x,t+\Delta t)dx - \int_{C_i} f(x,t)dx = \int_t^{t+\Delta t} g(f(x_{i-(1/2)},t)dt - \int_t^{t+\Delta t} g(f(x_{i+(1/2)},t)dt$$

(8)

$$\frac{1}{\Delta x}\int_{C_i} f(x,t+\Delta t)dx = \frac{1}{\Delta x}\int_{C_i} f(x,t)dx - \frac{1}{\Delta x}(\int_t^{t+\Delta t} g(f(x_{i+(1/2)},t)dt - \int_t^{t+\Delta t} g(f(x_{i-(1/2)},t)dt)$$

we can put this expression as

$$\rho_i^{n+1} = \rho_i^n - (g_{i+\frac{1}{2}} - g_{i-\frac{1}{2}})/\Delta x_i \tag{9}$$

where

$g_{i+\frac{1}{2}} = -\frac{a_{i+1}\xi + b_{i+1}\xi^2}{1+\beta_{i+1}\xi}$ = Flux across boundary $x = x_{i+\frac{1}{2}}$ during $t^{n+1} - t^n$

Finally we shall need to interpolate in the next time step to determine interface values as a function of the cell averages.

$$f_{i+\frac{1}{2}} = \frac{1}{2}(\rho_i + \rho_{i+1}) - \frac{1}{6}(\bar{\delta} f_i - \bar{\delta} f_{i-1}) \tag{10}$$

For the rational function we calculate the average slope in a cell $\bar{\delta} f_i$ as

$$\bar{\delta} f_i = \begin{cases} \min(|\delta f_i|, 3|\rho_{i+1} - \rho_i|, 3|\rho_i - \rho_{i-1}|)\operatorname{sgn}(\delta f_i), & \text{si } (\rho_{i+1} - \rho_i)(\rho_i - \rho_{i-1}) > 0 \\ 0, \quad \textit{otherwise} \end{cases} \tag{11}$$

being $\delta f_i = (\rho_{i+1} - \rho_{i-1})/2$

By using time-splitting the method can be easily extended to solve advection equation in two and three dimensions. For example in two dimensions the time splitting is equivalent to do the transport of particles to the direction (Ox) and then according to the other direction (Oy).

5. Model description

 ISBN 13: 978-84-613-5510-5

Chimère is based on the mass continuity equation for the concentrations of chemical species in every box of a given grid:

$$\frac{\partial f}{\partial t} + \nabla \cdot uf = \nabla \cdot k\nabla f + P - L \quad (12)$$

In this equation, characteristic for the Eulerian approach, f is a vector containing the concentrations of all model species for every grid box, u is the three dimensional wind vector, k the tensor of eddy diffusivity and P and L represent production and loss terms due to chemical reactions, emissions and deposition.

The numerical method for the temporal solution of the stiff system of partial differential equations (12) is adapted from the second-order TWO-STEP algorithm originally proposed by [5] for gas phase chemistry only. It is based on the application of a Gauss-Seidel iteration scheme to the 2-step implicit backward differentiation (BDF2) formula:

$$f^{n+1} = \frac{4}{3} f^n - \frac{1}{3} f^{n-1} + \frac{2}{3} \Delta t R \quad f^{n+1} \quad (13)$$

With f^n being the vector of chemical concentrations at time t^n, At the time step leading from time t^n to t^{n+1} and R(f) = $P_{(f)}$ - $L_{(f)}$ the temporal evolution of the concentrations due to chemical production and emissions (P) and chemical loss and deposition (L). Note that L is a diagonal matrix here. After rearranging and introducing the production and loss terms this equation reads

$$f^{n+1} = \left(I + \frac{2}{3} \Delta t L \quad f^{n+1} \right)^{-1} \left(\frac{4}{3} f^n - \frac{1}{3} f^{n-1} + \frac{2}{3} \Delta t P \quad f^{n+1} \right) \quad (14)$$

The implicit nonlinear system obtained in this scheme can be solved pertinently with a Gauss-Seidel method [5].

We can find a more complete description and evaluation of the Chimère model designed for seasonal simulations and real time forecasts without the use of super-computers in [6], where details about the implementation and evaluations of the modeling are given.

6. Numerical results

Simulations of photochemical compounds were carried out using the regional V200603par-rc1 version of the CHIMERE model for August 2003. This version calculates the concentration of 44 gaseous species and both inorganic and organic aerosols of primary and secondary origin, including primary particulate matter, mineral dust, sulfate, nitrate, ammonium, secondary organic species and water. Numerical resolution scheme was analyzed for a domain centred on Madrid (MAD in Figure 2), with a similar set up to that used in [7]. This area is one of the most populated areas in Spain, with more than 6 millions of inhabitants. High ozone level episodes are quite frequent over this area. This domain at a horizontal resolution of 0.07 degrees and 14 vertical sigma-pressure levels extending up to 500 hPa, was nested to a coarser one, covering the Iberian Peninsula (SP in Figure

 ISBN 13: 978-84-613-5510-5

1) at a 0.2 degree resolution. This second domain was also nested to a European scale domain (EUR1 in Figure 2), ranging from 10.5W to 22.5E and from 35N to 57.5 N and a 0.5 degree horizontal resolution. A one-way nesting procedure was used; coarse-grid simulations forced the fine-grid ones at the boundaries without feedback.

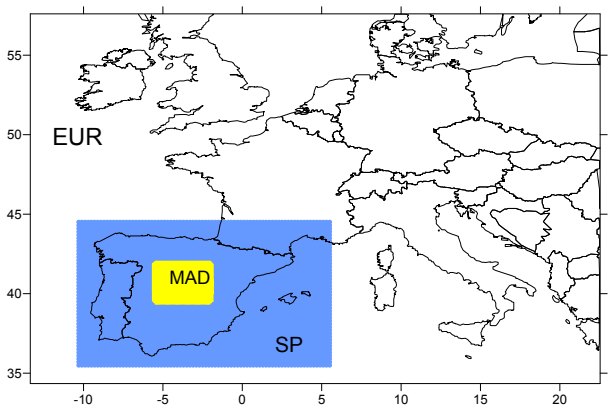


Fig. 2. Simulation domains

Boundary conditions for the coarsest domain were provided from monthly 2003 climatology from LMDz-INCA model [8] for gases concentrations and from monthly 2004 GOCART model [9] for particulate species, as described in [10].

Emissions for all the simulations were derived from the annual totals of the EMEP database for 2004 [11]. Original EMEP emissions were disaggregated taking into account land use information (Global Land Cover Facility, GLCF, http://change.gsfc.nasa.gov/create.html) in order to get higher resolution emission data. For each SNAP activity sector, the total NMVOC emission was split into emissions of 227 real individual NMVOC according to the AEAT speciation [12]. These species were then aggregated into the CHIMERE model ones.

The MM5 model was used to obtain the meteorological input fields. The simulations were carried out also for three domains, with respective resolutions of 36 Km, 19 Km and 7 km. The two coarsest MM5 simulations were forced by the National Centres for Environmental Prediction model (GFS) analyses. The finest domain was nested to a 21 km resolution MM5 simulation.

The quality of model predictions obtained with the implementation of the rational conservative formulation (RCF) was analyzed by comparing it to observations at the monitoring sites. Figure 3 shows the location of the NO_2, SO_2 and O_3 monitoring stations located inside the Madrid domain. Figure 4 gives the location of a common monitoring station 280974 and figures 5, 6 and 7 present the numerical results of ozone, NO_2, and SO_2 in this monitoring station. Also the corresponding observed concentrations with RCF numerical model between 1st august 2003 and 5th august 2003 for this common monitoring station are presented in Figure 5, 6 and 7.

 ISBN 13: 978-84-613-5510-5

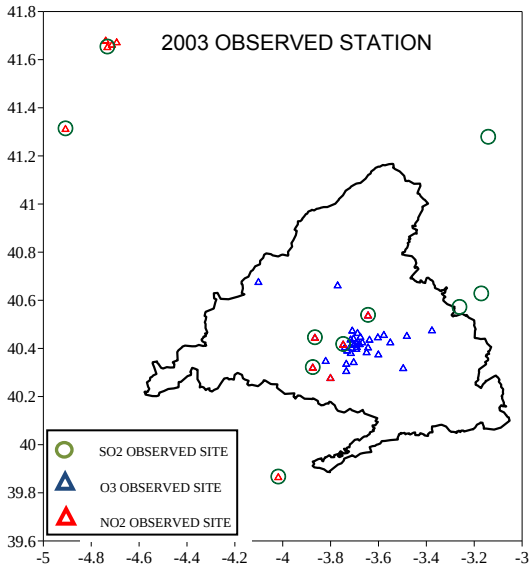


Fig.3. Distribution map of NO_2, SO_2 and O_3 monitoring stations in the area of Madrid (Spain).

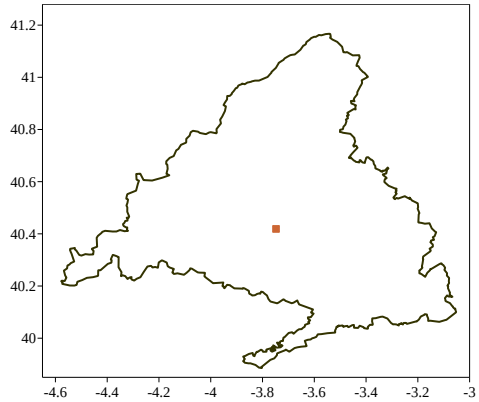


Fig.4: Location of the monitoring station 280974

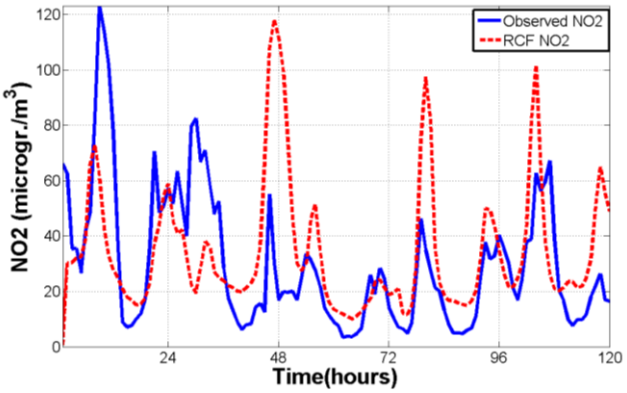


Fig.5: Observed and simulated concentration of NO_2 at station 2807924

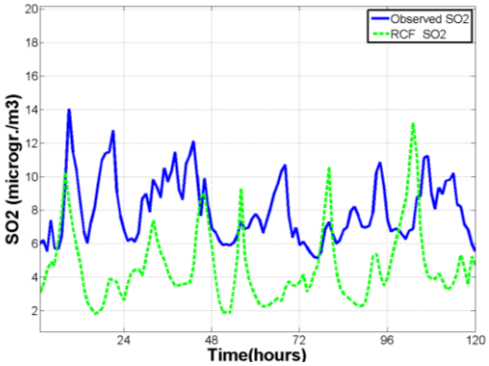


Fig.6: Observed and simulated concentration of SO_2 at station 2807924

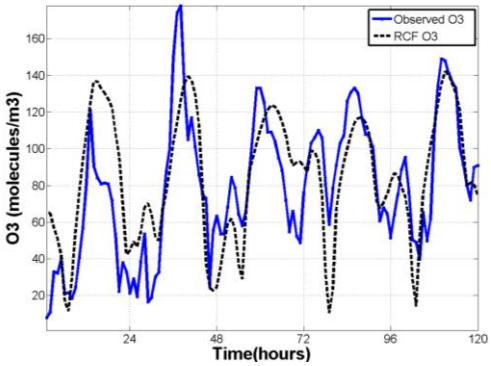


Fig.7: Observed and simulated concentration of O_3 at station 2807924

 ISBN 13: 978-84-613-5510-5

In order to evaluate the performance of the CHIMERE model with RCF some statistics were calculated. Table 1 presents the metrics used and their definition. Parameters such as mean bias (B_{MB}), mean normalized bias (B_{MNB}), mean normalized absolute error (E_{MNAE}), root mean square error (E_{RMSE}) and root mean normalized square error (E_{RMNSE}) were estimated for O_3, NO_2 and SO_2. Regarding ozone, only statistics for moderate-to-high ozone concentration cases (more important for human health protection) were considered by selecting predicted-observed value pairs when hourly observations were equal to or greater than the cutoff of 80 μgm^{-3}. For NO_2 and SO_2 a cutoff value of $5\mu gm^{-3}$ was used. 39 air quality sites were taken into account to estimate ozone statistics. For NO_2 and SO_2 evaluation information from 10 stations was considered.

Table 1. Definition of the metrics used in the evaluation of the CHIMERE model performance

Mean bias	$B_{MB} = \frac{1}{N}\sum(M_i - O_i) = \overline{M} - \overline{O}$
Mean normalized bias	$B_{MNB} = \frac{1}{N}\sum\left(\frac{M_i - O_i}{O_i}\right) = \left(\frac{1}{N}\sum\frac{M_i}{O_i} - 1\right)$
Mean normalized absolute error	$E_{MNAE} = \frac{1}{N}\sum\left(\frac{\lvert M_i - O_i\rvert}{O_i}\right)$
Root mean square error	$E_{RMSE} = \left[\frac{1}{N}\sum(M_i - O_i)^2\right]^{\frac{1}{2}}$
Root mean normalized square error	$E_{RMNSE} = \left[\frac{1}{N}\sum\left(\frac{M_i - O_i}{O_i}\right)^2\right]^{\frac{1}{2}}$

N: pairs of modeled and observed concentrations *Mi* and *Oi*. The index *i* is over time series and over all the locations in the domain. $* \overline{M} = \frac{1}{N}\sum M_i \overline{O} = \frac{1}{N}\sum O_i$

Statistical results for ozone, nitrogen dioxide and sulfur dioxide are presented in Table 2.

 ISBN 13: 978-84-613-5510-5

Table 2. Statistics for ozone, nitrogene dioxide and sulfur dioxide evaluation for august 2003. Based on hourly values higher than 80 μgm^{-3}, 5 μgm^{-3} and 5 μgm^{-3} respectively.

Statistical measure	O_3	NO_2	SO_2
Mean bias ($\mu g\ m^{-3}$)	-11.2043	-2.6061	-6.7785
Mean normalized bias	-0.0839	-0.3402	-0.6879
Mean normalized absolute error	0.1844	0.6285	0.7054
Root mean square error ($\mu g\ m^{-3}$)	25.486	5.8845	8.3290
Root mean normalized square error	0.2277	0.7728	0.7322

The plots showing the mean normalized absolute error for the individual stations and for the three contaminants are presented in Figures 8, 9 and 10.

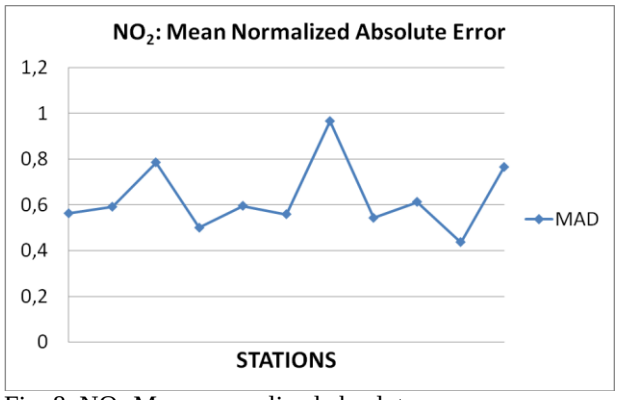


Fig. 8. NO_2 Mean normalized absolute error

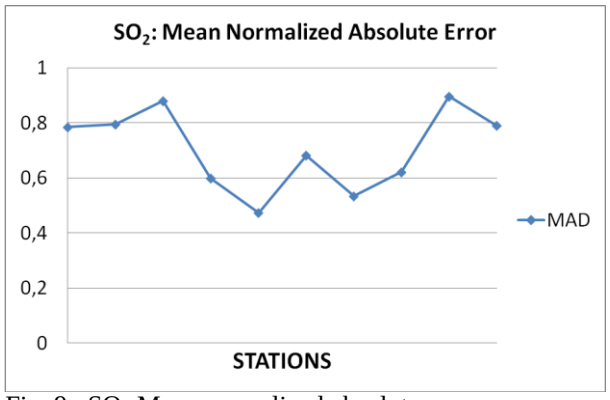


Fig. 9. SO_2 Mean normalized absolute error

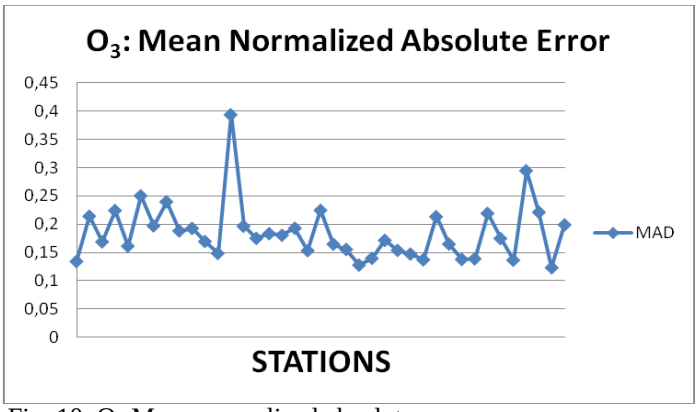


Fig. 10. O_3 Mean normalized absolute error

 ISBN 13: 978-84-613-5510-5

7. Conclusions

In this paper we have presented the implementation in the European scale Eulerian chemistry transport model CHIMERE of an exactly conservative method for the advection equation. The advantage of these methods is that the cell-integrated average is predicted via a flux formulation, thus the mass is exactly conserved. The results of the method for ozone, nitrogen dioxide and sulfur dioxide statistics have been compared with a set of observation sites in the area of Madrid (Spain). The mean normalized bias and the mean normalized absolute error present values, that are inside the range to consider an accurate model performance.

Acknowledgements

This study was supported by Ministerio de Ciencia y Tecnología of Spain under the project CGL2008-1757/CLI.

References

[1] A. SATANIFORTH AND J.COTE, *Semi-Langrangian integration scheme for atmospheric model. A review*, Mon. Weather Rev. **119**, 2206 (1991).

[2] J.S. SCROGGS AND F.H.M. SEMAZZI. *A conservative semi.Lagrangian method for multidimensional fluid dynamics applications*, Numerical Methods for Partial Differential Equations, **11**, (1995), 445-452.

[3] F.XIAO, *A class of single-cell high-irder semi-Lagrangian adventions schemes*, Mon. Wea. Rev, **128** (2000), 1165-1176.

[4] F. XIAO, X. PEN, *A convexity preserving scheme for conservative advection transport*, J. Comput. Phys. **198** (2004) 389-402.

[5] J.G, VERWER, *A Gauss-Seidel iteration for stiff ODEs from chemical kinetics*, SIAM J. Scientific Computing. **15** (1994) 1243-1250.

[6] H. SCHMIDT, C. DEROGNAT, R. VAUTARD, M. BEEKMANN, *A comparasion of simulated and observed ozone mixing ratios for the summer of 1998 in Western Europe*. Atmospheric Environment, **35** (2001) 6277-6297.

[7] M.G. VIVANCO, I. PALOMINO, R. VAUTARD, B. BESSAGNET, F. MARTÍN, L. MENUT, S. JIMÉNEZ. *Multi-year assessment of photochemical air quality simulation over SPAIN*, Environmental Modelling & Software, **24** (2009), 63-73

[8] D. A. HAUGLUSTAINE, F. HOURDIN, L. JOURDAIN, M.A. FILIBERTI, S. WALTERS, J.F. LAMARQUE, , AND E. A. HOLLAND. *Interactive chemistry in the Laboratoire de Meteorologie Dynamique general circulation*

 ISBN 13: 978-84-613-5510-5

model: Description and background tropospheric chemistry evaluation, J. Geophys. Res., **109**, (2004), doi:10.1029/2003JD003957.

[9] M. CHIN, P. GINOUX, S. KINNE, B. N. HOLBEN, B. N. DUNCAN, R. V. MARTIN, J. A. LOGAN, A. HIGURASHI, AND T. NAKAJIMA. *Tropospheric aerosol optical thickness fromt he GOCART model and comparisons with satellite and sunphotometer measurements*, J. Atmos. Sci. **59**, (2002) 461-483.

[10] R. VAUTARD, B. BESSSAGNET, M. CHIN, AND L. MENUT, *On the contrigution of natural Aeolian soruces to particulate matter concentrations in Europe: Testing hypotheses with a modelling approach,* Atmospheric Environment., **39**, (2005), 3291-3303.

[11] V. VESTRENG, K. BREIVIK, M. ADAMS, A. WAGENER, J. GOODWIN, O. ROZOVSKKAYA, J. M. PACYNA. *Inventory Review 2005, Emission Data reported to LRTAP Convention and NEC Directive, Initial review of HMs and POPs*, Technical report MSC-W **1/2005**, (2005) ISSN 0804-2446.

[12] N.R. PASSANT, *Speciation of UK Emissions of Non-methane Volatile Organic Compounds*. AEAT/ENV/R/0545, issue **1** (2000).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Modelling of the advection-diffusion equation with a meshless method without numerical diffusion

Luis Gavete[1], Francisco Ureña[2], Juan José Benito[3] and María Lucía Gavete[4]

[1] *Universidad Politécnica de Madrid, Spain*
[2] *Universidad Castilla-la Mancha, Ciudad Real, Spain*
[3] *U.N.E.D., Madrid, Spain*
[4] *Universidad Rey Juan Carlos, Madrid, Spain*

emails: lu.gavete@upm.es, francisco.urena@uclm.es, jbenito@ind.uned.es, lucia.gavete@urjc.es

Abstract

A comprehensive study is presented regarding the stability of the forward explicit integration technique with generalized finite difference spatial discretizations, free of numerical diffusion, applied to the advection-diffusion equation. The modified equivalent partial differential equation approach is used to demonstrate that the approximation is free of numerical diffusion. Two-dimensional results are obtained using the von Neumann method of stability analysis. Numerical results are presented showing the accuracy obtained.

Key words: advection-diffusion, generalized finite difference
MSC2000: AMS Codes (optional)

1. Introduction

With the development of modern industry, various pollulants discharge in the air rivers, lakes and oceans. The changes of pollulants in the air or in the water consist of the physical, chemical and biochemical process and so on. The physical changes of pollution involve two main important processes, that is, advection and diffusion. The mathematical model describing these two processes is the well known advection-diffusion equation. In two dimensions this equation is as follows

 ISBN 13: 978-84-613-5510-5

$$\frac{\partial U}{\partial t}+\beta_x\frac{\partial U}{\partial x}+\beta_y\frac{\partial U}{\partial y} = \alpha_x\frac{\partial^2 U}{\partial x^2}+\alpha_y\frac{\partial^2 U}{\partial y^2} \quad ;t>0, \quad \mathbf{x}=\ \mathrm{x,\ y}\ ^T\in\Omega$$

with the initial condition:

$$U\ \mathbf{x},0\ = f\ \mathbf{x} \tag{1}$$

and the boundary conditions:

$$aU\ \mathbf{x}_0,\mathrm{t}\ +b\frac{\partial U\ \mathbf{x}_0,\mathrm{t}}{\partial n}=g\ \ t \quad in\ \Gamma$$

being $f\ \mathbf{x}$ and $g(t)$ two known functions, *a*, *b* are constants , Γ is the boundary of Ω, $U(x,y,t)$ is a transported (advected and diffused) scalar variable, $\beta_x>0$, $\beta_y>0$ are constant speeds of advection and $\alpha_x>0$, $\alpha_y>0$ are constant diffusivities in the x-, and y- direction respectively.

Various numerical techniques can be used to solve this partial differential equation with the associated initial and boundary conditions [1]. An active field of research is the use of meshless methods. An evolution of the method of finite differences has been the development of generalized finite difference method (GFDM) that can be applied as a meshless or meshfreee method to irregular grids or clouds of points. Benito, Ureña and Gavete have made interesting contributions to the development of this method [2-7]. This paper shows the application of the generalized finite difference method to solve the advection-diffusion equation by an explicit method.

The paper is structured in six sections. In section 2, we describe briefly the GFDM. In section 3 we describe the explicit scheme used to approximate the advection-diffusion equation. In section 4 we study the truncation error and the stability. In Section 5 an error analysis is done comparing with a test case. Section 6 contains concluding remarks.

2. The Generalized finite difference method

In the GFDM the intention is to obtain explicit linear expressions for the approximation of partial derivatives in the points of a domain. First of all, an irregular grid or cloud of points is generated in the domain $\Omega\cup\Gamma$. On defining the central node with a set of nodes surrounding that node, the star of nodes then refers to a group of established nodes in relation to a central node. Each node in the domain has an associated star assigned to it.

We define the following function based in the approximation of second order in Taylor series

$$B\ u\ =\sum_{j=1}^{N}\left[\left(u_0-u_j+h_j\frac{\partial U_0}{\partial x}+k_j\frac{\partial U_0}{\partial y}+\frac{1}{2}\left(h_j\frac{\partial U_0}{\partial x}+k_j\frac{\partial U_0}{\partial y}\right)^2\right)w_j\ h_j,k_j\right]^2$$

 ISBN 13: 978-84-613-5510-5

(2)

where u_0 is the approximated value of the function at the central node of the star, (x_0, y_0), u_j are the function values of the rest of the nodes, $h_j = x_j - x_0$, $k_j = y_j - y_0$ and $w(h_j, k_j)$ is the denominated weight function.

If the function (2) is minimized with respect to the partial derivatives, the following linear equation system is obtained

$$\mathbf{A}\,\mathbf{D_u} = \mathbf{b} = \left\{ \sum_{j=1}^{N} \left(u_0 + u_j \right) h_j w^2 \ \ldots \sum_{j=1}^{N} \left(u_0 + u_j \right) \frac{h_j^2}{2} w^2 \ \ldots \sum_{j=1}^{N} \left(u_0 + u_j \right) h_j k_j w^2 \ \ldots \right\}^T \quad (3)$$

$$\mathbf{D}_u = \left\{ \frac{\partial U_0}{\partial x}, \frac{\partial U_0}{\partial y}, \frac{\partial^2 U_0}{\partial x^2}, \frac{\partial^2 U_0}{\partial y^2}, \frac{\partial^2 U_0}{\partial x \partial y} \right\}^T$$

on solving the system (3) the explicit finite difference formulae are obtained.

$$\begin{cases} \mathbf{Y}\ k\ = -u_0 \sum_{\mathrm{i}=1}^{\mathrm{p}} \mathrm{M}_{\mathrm{ki}} \mathrm{c}_{\mathrm{i}} + \sum_{j=1}^{N} u_j \left(\sum_{i=1}^{p} M_{ki} d_{ji} \right), \quad k = 1,..,5 \\ \mathbf{D}_u\ k\ = \frac{1}{q_{kk}} \left(\mathbf{Y}\ k\ - \sum_{i=1}^{p-k} q_{\,k+1\ k} \mathbf{D}_u\ k+i\ \right), \quad k = 1,..,5 \end{cases} \quad (4)$$

where

$$M_{ij} = \begin{cases} -1^{\,1-\delta_{ij}} \sum_{k=j}^{i-1} q\ i,k\ M\ k,j & for: j < i, \quad i = 1,..,5 \quad j = 1,..,5 \\ \frac{1}{q_{ij}} & for: j = i, \quad i = 1,..,5 \quad j = 1,..,5 \\ 0 & for: j > i, \quad i = 1,..,5 \quad j = 1,..,5 \end{cases}$$

with δ_{ij} the Kronecker delta function, and:

$$\mathrm{c}_{\mathrm{i}} = \sum_{j=1}^{N} d_{ji}$$

$$d_{j1} = h_j w^2;\ d_{j2} = k_j w^2;\ d_{j4} = \frac{h_j^2}{2} w^2;\ d_{j5} = \frac{k_j^2}{2} w^2;\ d_{j6} = h_j k_j w^2$$

3. The advection-diffusion GFDM explicit scheme

On including the explicit expressions for the values of partial derivatives (5) in the differential equation of problem, we obtain the star equation (explicit difference scheme)(6):

 ISBN 13: 978-84-613-5510-5

$$\frac{\partial U_0}{\partial \mathrm{t}} = \frac{u_0^{n+1} - u_0^n}{\Delta t}$$
$$\frac{\partial U_0}{\partial \mathrm{x}} = -\lambda_0 u_0^n + \sum_{j=1}^{N} \lambda_j u_j^n \; ; \quad \frac{\partial U_0}{\partial \mathrm{y}} = -\mu_0 u_0^n + \sum_{j=1}^{N} \mu_j u_j^n \; ; \tag{5}$$
$$\frac{\partial^2 U_0}{\partial \mathrm{x}^2} = -m_0 u_0^n + \sum_{j=1}^{N} m_j u_j^n \; ; \quad \frac{\partial^2 U_0}{\partial \mathrm{y}^2} = -\eta_0 u_0^n + \sum_{j=1}^{N} \eta_j u_j^n$$

$$u_0^{n+1} = u_0^n - \Delta t \left[\beta_x \left(-\lambda_0 u_0^n + \sum_{j=1}^{N} \lambda_j u_j^n \right) + \beta_y \left(-\mu_0 u_0^n + \sum_{j=1}^{N} \mu_j u_j^n \right) \right] +$$
$$\Delta t \left[\alpha_x \left(-m_0 u_0^n + \sum_{j=1}^{N} m_j u_j^n \right) + \alpha_y \left(-\eta_0 u_0^n + \sum_{j=1}^{N} \eta_j u_j^n \right) \right] \tag{6}$$
$$with: \;\; \lambda_0 = \sum_{j=1}^{N} \lambda_j \; ; \quad \mu_0 = \sum_{j=1}^{N} \mu_j \; ; \quad m_0 = \sum_{j=1}^{N} m_j ; \quad \eta_0 = \sum_{j=1}^{N} \eta_j$$

This scheme uses the forward-difference form for the time derivative and generalized finite difference forms for all spatial derivatives.
By using the modified equivalent partial differential equation approach of Warming and Hyett [8], we obtain the following expansion equation

$$\frac{\partial U}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 U}{\partial t^2} + \frac{(\Delta t)^2}{6} \frac{\partial^3 U}{\partial t^3} + \frac{(\Delta t)^3}{24} \frac{\partial^4 U}{\partial t^4} + \beta_x \left(\frac{\partial U}{\partial x} + \sum_{j=1}^{N} \gamma_{1,j} \;\; h_j, k_j \;\; \frac{\partial^3 U}{\partial x^3} + \ldots \right)$$
$$+ \beta_y \left(\frac{\partial U}{\partial y} + \sum_{j=1}^{N} \gamma_{2,j} \;\; h_j, k_j \;\; \frac{\partial^3 U}{\partial y^3} + \ldots \right) - \alpha_x \left(\frac{\partial^2 U}{\partial x^2} + \sum_{j=1}^{N} \gamma_{3,j} \;\; h_j, k_j \;\; \frac{\partial^3 U}{\partial x^3} + \ldots \right) \tag{7}$$
$$- \alpha_y \left(\frac{\partial^2 U}{\partial y^2} + \sum_{j=1}^{N} \gamma_{4,j} \;\; h_j, k_j \;\; \frac{\partial^3 U}{\partial x^3} + \ldots \right) = 0$$

and the modified equation

$$\frac{\partial U}{\partial t} + \beta_x \frac{\partial U}{\partial x} + \beta_y \frac{\partial U}{\partial y} - \left(\alpha_x - \frac{\beta_x^2}{2} \Delta t \right) \frac{\partial^2 U}{\partial x^2} - \left(\alpha_y - \frac{\beta_y^2}{2} \Delta t \right) \frac{\partial^2 U}{\partial y^2} + \ldots\ldots = 0 \tag{8}$$

This method incorporates numerical diffusion.
A new GFD scheme free of numerical diffusion can be created as follows

 ISBN 13: 978-84-613-5510-5

$$u_0^{n+1} = u_0^n - \Delta t\left[\beta_x\left(-\lambda_0 u_0^n + \sum_{j=1}^{N}\lambda_j u_j^n\right) + \beta_y\left(-\mu_0 u_0^n + \sum_{j=1}^{N}\mu_j u_j^n\right)\right] +$$

$$\Delta t\left[\left(\alpha_x + \frac{\beta_x^2}{2}\Delta t\right)\left(-m_0 u_0^n + \sum_{j=1}^{N} m_j u_j^n\right) + \left(\alpha_y + \frac{\beta_y^2}{2}\Delta t\right)\left(-\eta_0 u_0^n + \sum_{j=1}^{N}\eta_j u_j^n\right)\right] \quad (9)$$

$$with:\ \lambda_0 = \sum_{j=1}^{N}\lambda_j;\quad \mu_0 = \sum_{j=1}^{N}\mu_j;\quad m_0 = \sum_{j=1}^{N} m_j;\quad \eta_0 = \sum_{j=1}^{N}\eta_j$$

Then by using the modified equivalent partial differential equation approach of Warming and Hyett [8] we obtain the following expansion equation

$$\frac{\partial U}{\partial t} + \frac{\Delta t}{2}\frac{\partial^2 U}{\partial t^2} + \frac{(\Delta t)^2}{6}\frac{\partial^3 U}{\partial t^3} + \frac{(\Delta t)^3}{24}\frac{\partial^4 U}{\partial t^4} + \beta_x\left(\frac{\partial U}{\partial x} + \sum_{j=1}^{N}\gamma_{1,j}\ \ h_j, k_j\ \ \frac{\partial^3 U}{\partial x^3} + \ldots\right) +$$

$$\beta_y\left(\frac{\partial U}{\partial y} + \sum_{j=1}^{N}\gamma_{2,j}\ \ h_j, k_j\ \ \frac{\partial^3 U}{\partial y^3} + \ldots\right) - (\alpha_x + \frac{\beta_x^2\Delta t}{2})\left(\frac{\partial^2 U}{\partial x^2} + \sum_{j=1}^{N}\gamma_{3,j}\ \ h_j, k_j\ \ \frac{\partial^3 U}{\partial x^3} + \ldots\right) - \quad (10)$$

$$(\alpha_y + \frac{\beta_y^2\Delta t}{2})\left(\frac{\partial^2 U}{\partial y^2} + \sum_{j=1}^{N}\gamma_{4,j}\ \ h_j, k_j\ \ \frac{\partial^3 U}{\partial x^3} + \ldots\right) = 0$$

and the modified equation

$$\frac{\partial U}{\partial t} + \beta_x\frac{\partial U}{\partial x} + \beta_y\frac{\partial U}{\partial y} - \alpha_x\frac{\partial^2 U}{\partial x^2} - \alpha_y\frac{\partial^2 U}{\partial y^2} + \ldots\ldots = 0 \quad (11)$$

The modified equivalent partial differential equation of this method shows that this GFD formula (9) is free of numerical diffusion.

4. Convergence

According to Lax's equivalence theorem, if the consistency condition is satisfied, stability is the necessary and sufficient condition for convergence. In this section we study firstly the truncation error of the advection-diffusion equation, and secondly consistency and stability.

We split the truncation error (TTE) in time derivative error (TE_t) and space derivatives error ($TE_{\mathbf{x}}$). As the first order time derivative is given by

$$\frac{\partial U\ \mathbf{x_0}, t}{\partial t} = \frac{U\ \mathbf{x_0}, t + \Delta t\ - U\ \mathbf{x_0}, t}{\Delta t} - \frac{\Delta t}{2}\frac{\partial^2 U\ \mathbf{x_0}, t_1}{\partial t^2} + O\ \ \Delta t\ ^2 \qquad \forall\ t < t_1 < t + \Delta t$$

 ISBN 13: 978-84-613-5510-5

then the truncation time error is given by

$$TE_t = -\frac{\Delta t}{2}\frac{\partial^2 u(x_0, y_0, t_1)}{\partial t^2} + \Theta((\Delta t)^2), \quad t < t_1 < t + \Delta t \tag{12}$$

In order to obtain the truncation error for space GFD derivatives, Taylor's series expansion including higher order derivatives is used and then higher order function $B^*(u)$ is obtained

$$B*(u) = \sum_{j=1}^{N}\left[\begin{array}{l} u_0 - u_i + h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y} + \frac{1}{2}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{2)} + \\ \frac{1}{6}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{3)} + \frac{1}{24}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{4)} + \ldots \end{array}\right] w(h_j, k_j)^2$$

If $B^*(u)$ is minimized with respect to the partial derivatives up to second order, the following linear equation system is defined

$$\mathbf{A}\,\mathbf{D_u} = \left(\sum_{j=1}^{N}\Xi h_j \quad \sum_{j=1}^{N}\Xi k_j \quad \sum_{j=1}^{N}\Xi\frac{h_j^2}{2} \quad \sum_{j=1}^{N}\Xi\frac{k_j^2}{2} \quad \sum_{j=1}^{N}\Xi h_j k_j\right)^T \tag{13}$$

where

$$\Xi = \left(u_0 - u_i - \frac{1}{6}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{3)} - \frac{1}{24}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{4)} - \ldots\right) w(h_j, k_j)^2$$

with N=8, and then

$$TE_{(x,y)} = \mathbf{C}\mathbf{A}^{-1}\left(\sum_{j=1}^{N}\Upsilon h_j \quad \sum_{j=1}^{N}\Upsilon h_j \quad \sum_{j=1}^{N}\Upsilon\frac{h_j^2}{2} \quad \sum_{j=1}^{N}\Upsilon\frac{k_j^2}{2} \quad \sum_{j=1}^{N}\Upsilon h_j k_j\right)^T \tag{14}$$

where

$$\Upsilon = \left(-\frac{1}{6}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{3)} - \frac{1}{24}\left(h_j\frac{\partial U_0}{\partial x} + k_j\frac{\partial U_0}{\partial y}\right)^{4)} - \ldots\right) w(h_j, k_j)^2$$

 ISBN 13: 978-84-613-5510-5

$$\mathbf{A}=\begin{pmatrix} \sum_{j=1}^{N} h_j^2 w^2 & \sum_{j=1}^{N} h_j k_j w^2 & \sum_{j=1}^{N} \frac{h_j^3}{2} w^2 & \sum_{j=1}^{N} \frac{h_j k_j^2}{2} w^2 & \sum_{j=1}^{N} h_j^2 k_j w^2 \\ & \sum_{j=1}^{N} k_j^2 w^2 & \sum_{j=1}^{N} \frac{h_j^2 k_j}{2} w^2 & \sum_{j=1}^{N} \frac{k_j^3}{2} w^2 & \sum_{j=1}^{N} h_j k_j^2 w^2 \\ & & \sum_{j=1}^{N} \frac{h_j^4}{4} w^2 & \sum_{j=1}^{N} \frac{h_j^2 k_j^2}{4} w^2 & \sum_{j=1}^{N} \frac{h_j^3 k_j}{2} w^2 \\ & & & \sum_{j=1}^{N} \frac{k_j^4}{4} w^2 & \sum_{j=1}^{N} \frac{h_j k_j^3}{2} w^2 \\ & & & & \sum_{j=1}^{N} h_j^2 k_j^2 w^2 \end{pmatrix}$$

$$\mathbf{C}=\begin{pmatrix} -\beta_x & -\beta_y & -\left(\alpha_x+\frac{\beta_x^2\Delta t}{2}\right) & -\left(\alpha_y+\frac{\beta_y^2\Delta t}{2}\right) & 0 \end{pmatrix}$$

where

$$w^2 = w(h_j,k_j)^2$$

Then by solving (14), we obtain

$$TE_{(x,y)} = -\beta_x\left[\sum_{j=1}^{N}\left(\psi_{1,j}\frac{\partial^3 U}{\partial x^3}+\psi_{2,j}\frac{\partial^3 U}{\partial x^2\partial y}+\psi_{3,j}\frac{\partial^3 U}{\partial x\partial y^2}+\psi_{4,j}\frac{\partial^3 U}{\partial y^3}+....\right)\right]-$$
$$\beta_y\left[\sum_{j=1}^{N}\left(\psi_{5,j}\frac{\partial^3 U}{\partial x^3}+\psi_{6,j}\frac{\partial^3 U}{\partial x^2\partial y}+\psi_{7,j}\frac{\partial^3 U}{\partial x\partial y^2}+\psi_{8,j}\frac{\partial^3 U}{\partial y^3}+....\right)\right]-$$
$$\left(\alpha_x+\frac{\beta_x^2\Delta t}{2}\right)\left[\sum_{j=1}^{N}\left(\psi_{9,j}\frac{\partial^3 U}{\partial x^3}+\psi_{10,j}\frac{\partial^3 U}{\partial x^2\partial y}+\psi_{11,j}\frac{\partial^3 U}{\partial x\partial y^2}+\psi_{12,j}\frac{\partial^3 U}{\partial y^3}+....\right)\right]-$$
$$\left(\alpha_y+\frac{\beta_y^2\Delta t}{2}\right)\left[\sum_{j=1}^{N}\left(\psi_{13,j}\frac{\partial^3 U}{\partial x^3}+\psi_{14,j}\frac{\partial^3 U}{\partial x^2\partial y}+\psi_{15,j}\frac{\partial^3 U}{\partial x\partial y^2}+\psi_{16,j}\frac{\partial^3 U}{\partial y^3}+....\right)\right]+\Theta\left(h_j,k_j\right) \quad (15)$$

where $\psi_{i,j}(h_j,k_j)$ are second-order rational functions and $\Theta\ h_j,k_j$ is a series of third- and higher-order functions.

The total truncation error for advection-diffusion equation is

$$TTE = TE_t + TE(x,y) \quad (16)$$

By considering bounded derivatives in *TTE*, we have consistency

$$\lim_{\Delta t,\, h_j,k_j \to 0,\,0,\,0} TTE \to 0 \quad (17)$$

 ISBN 13: 978-84-613-5510-5

"Boundary conditions are neglected by the von Neumann method which applies in theory only to pure initial value problems with periodic initial data. It does however provide necessary conditions for stability of constant coefficient problems regardless of the type of boundary condition" [9].
From the previously obtained formula (9)

$$u_0^{n+1} = \left(1+\Delta t\left[\beta_x\lambda_0+\beta_y\mu_0-(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_0-(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_0\right]\right)u_0^n -$$
$$\Delta t\left[\beta_x\sum_{j=1}^{N}\lambda_j u_j^n+\beta_y\sum_{j=1}^{N}\mu_j u_j^n-(\alpha_x+\beta_x^2\frac{\Delta t}{2})\sum_{j=1}^{N}m_j u_j^n-(\alpha_y+\beta_y^2\frac{\Delta t}{2})\sum_{j=1}^{N}\eta_j u_j^n\right] \quad (18)$$

Fort the stability analysis a harmonic decomposition is made of the approximate solution at grid points at a given time level *n*

$$u_0^n=\xi^n e^{i(\kappa_x x_0+\kappa_y y_0)};\ u_j^n=\xi^n e^{i(\kappa_x(x_0+h_j)+\kappa_y\ y_0+k_j\ )}$$

where (x_0,y_0) are the coordinates in the central node of the star, and (h_j,k_j) are the coordinates of the other nodes of the star with respect to the central node.

$$\xi=\left(1-\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]\ 1-\cos(\kappa_x h_j+\kappa_y k_j)\ \right)$$
$$+i\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]sen(\kappa_x h_j+\kappa_y k_j)$$

Taking into account that the stability condition is $\|\xi\|\le 1$, then

a) $|\text{Real}(\xi)|<1$

$$\left(1-\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]\ 1-\cos(\kappa_x h_j+\kappa_y k_j)\ \right)<1\Rightarrow$$
$$-1<\left(1-\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]\ 1-\cos(\kappa_x h_j+\kappa_y k_j)\ \right)<1\Rightarrow$$
$$\Delta t\left[-\beta_x\lambda_0-\beta_y\mu_0+(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_0+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_0\right]\le 1 \quad (19)$$

b) $\|\xi\|\le 1$

 ISBN 13: 978-84-613-5510-5

$$\|\xi\|^2 = \left(1-\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]1-\cos(\kappa_x h_j+\kappa_y k_j)\right)^2$$

$$+\left(\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]sen(\kappa_x h_j+\kappa_y k_j)\right)^2\le 1$$

$$\Delta t\left(\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]sen(\kappa_x h_j+\kappa_y k_j)\right)^2\le$$

$$\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]1-\cos(\kappa_x h_j+\kappa_y k_j)\ \times$$

$$\left[2-\Delta t\sum_{j=1}^{N}\left[(\alpha_x+\beta_x^2\frac{\Delta t}{2})m_j+(\alpha_y+\beta_y^2\frac{\Delta t}{2})\eta_j-\beta_x\lambda_j-\beta_y\mu_j\right]\left(1-\cos(\kappa_x h_j+\kappa_y k_j)\right)\right]$$

$$\Delta t\left[\frac{\beta_x^2\left(\sum_{j=1}^{N}|\lambda_j|\right)^2}{\alpha_x m_0+\frac{5}{6}\beta_x^2\left(\sum_{j=1}^{N}|\lambda_j|\right)^2\Delta t}+\frac{\beta_y^2\left(\sum_{j=1}^{N}|\mu_j|\right)^2}{\alpha_y\eta_0+\frac{5}{6}\beta_y^2\left(\sum_{j=1}^{N}|\mu_j|\right)^2\Delta t}\right]\le 1 \qquad (20)$$

Both formulae (19) and (20) give us the conditions for the stability.

5. Numerical results

In order to illustrate the application of the numerical explicit GFD scheme developed previously, a problem for which an exact solution is available is required so that approximate results obtained can be compared with an exact solution. The problem to be solved is

$$\frac{\partial U(\mathbf{x},t)}{\partial t}+\frac{\partial U(\mathbf{x},t)}{\partial x}+\frac{\partial U(\mathbf{x},t)}{\partial y}=0.1\left(\frac{\partial U^2(\mathbf{x},t)}{\partial x^2}+\frac{\partial U^2(\mathbf{x},t)}{\partial y^2}\right) \qquad (21)$$

$$t>0,\quad 9<x^2+y^2<25$$

The exact solution is

 ISBN 13: 978-84-613-5510-5

$$U(x, y, t) = e^{-1.8t+x+y}$$

$$\begin{cases} \text{Initial conditions: } U(\mathbf{x}, 0) \\ \textit{Dirichlet} \text{ boundary conditions at } 9 = x^2 + y^2 = 25 \end{cases}$$

$$\text{The weight function is: } w(x_j, y_j) = \frac{1}{\sqrt{x_j^2 + y_j^2}^3} \tag{22}$$

The global error is evaluated in the last step considered, using the following formula

$$\text{global error} = \frac{\sqrt{\sum_1^M (sol(i) - exac(i))^2}}{|exac|_{max}} \tag{23}$$

where sol(i) is the GFDM solution at the node i, exac(i) is the exact value of solution at the node (i), $|exac_{max}|$ is the maximum value of the exact solution in the cloud of nodes considered and M is the total number of nodes of the domain.
In this problem we consider different irregular clouds of points as given in Fig. 1.

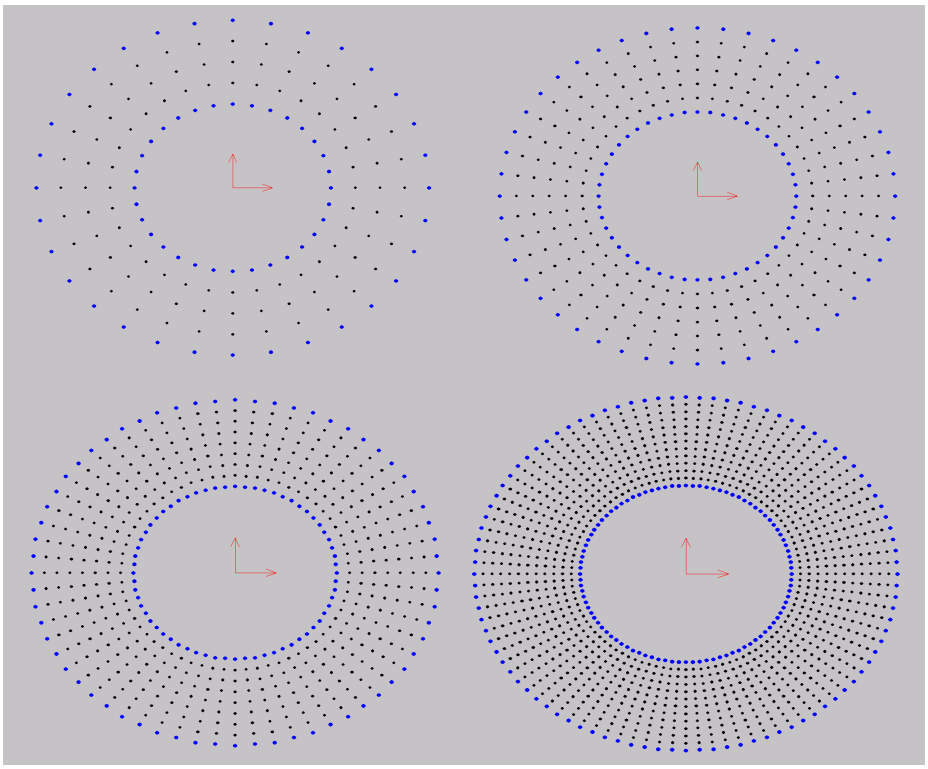


Fig. 1. Different irregular clouds of points.

 ISBN 13: 978-84-613-5510-5

The influence on global error of using different number of nodes is given in Fig. 2.

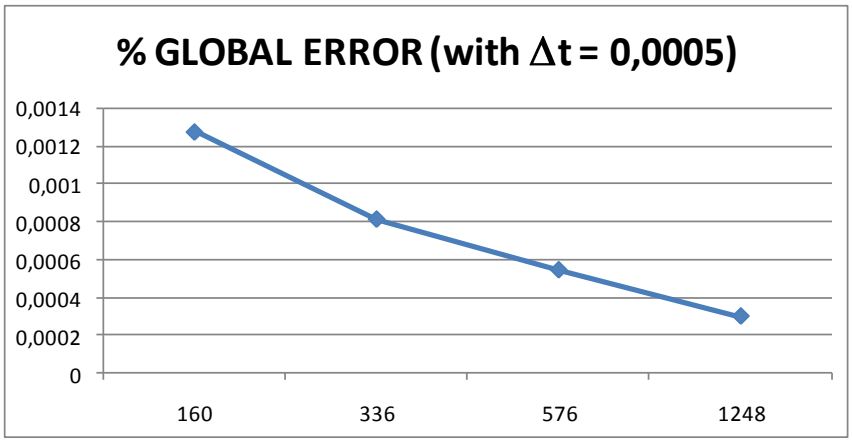


Fig. 2 Global error versus the number of nodes.

Also we consider for the irregular cloud of 1248 nodes the influence on global error versus different values of time increment in Fig.3.

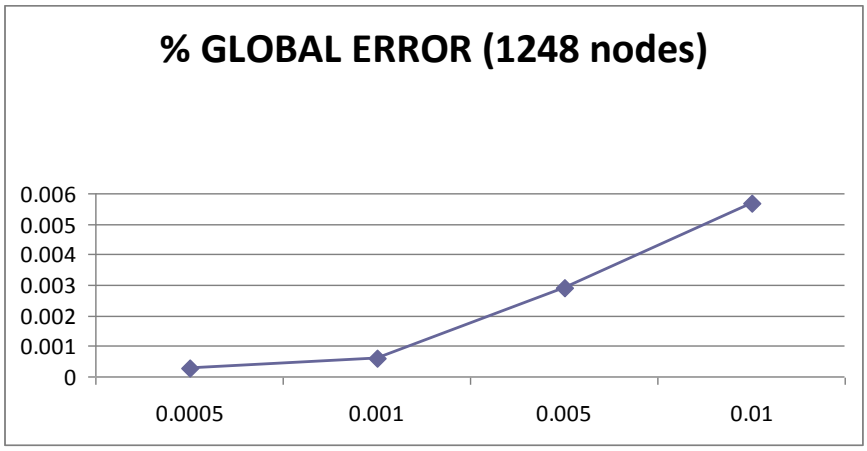


Fig. 3 Global error versus the time increment.

As it is shown in Fig.2 and 3, the global error decreases by increasing the number of nodes or decreasing the time increment.

6. Conclusions

An explicit solution of advection-diffusion equation has been presented for the case of using the GFDM over irregular grids. We have defined the truncation

 ISBN 13: 978-84-613-5510-5

error of the scheme in the case of irregular grids of nodes. Then, we have established the consistency and stability (following the von Neumann stability analysis) criteria for this scheme. An academic test has been presented to illustrate the application of this method.

The fully explicit generalized finite difference schemes are simple to implement and economical to use. They are very efficient and very quick. They are conditionally stable.

The modified equivalent partial differential equation approach of Warming and Hyett[8] has been employed which permits to demonstrate that the GFD scheme is free of numerical diffusion.

Acknowledgements

The authors acknowledge the support fron Ministerio de Ciencia e Innovación of Spain, research project CGL2008-01757/CLI.

References

[1] M. DEHGHAN, *Numerical solution of the three-dimensional advection-diffusion equation*, Appl. Math. Comput. **150** (2004) 5-19.

[2] J.J. BENITO, F. UREÑA, L. GAVETE, *Influence of several factors in the generalized finite difference method*, Applied Mathematical Modelling **25** (2001) 1039-1053.

[3] J.J. BENITO, F. UREÑA, L. GAVETE, R. ALVAREZ, *An h-adaptive method in the generalized finite difference method*. Computer methods in Applied Mechanics and Engineering **192**, (2003) 735-759.

[4] J.J. BENITO, F. UREÑA, L. GAVETE, *Solving parabolic and hyperbolic equations by Generalized Finite Difference Method*. Journal of Computational and Applied Mathematics **209**, Issue 2, (2007) 208-233.

[5] J.J. BENITO, F. UREÑA, L. GAVETE, *A posteriori error estimator and indicator in Generalized Finite Differences. Application to improve the approximated solution of elliptic pdes*. International Journal of Computer Mathematics **85** (2008) 359-370.

[6] J.J. BENITO, F. UREÑA , L. GAVETE, *Application of the Generalized Finite Difference Method to improve the approximated solution of elliptic pdes*. Computer Modelling in Engineering & Sciences **38** (2009) 39-58.

[7] J.J. BENITO, F. UREÑA, L. GAVETE, *Leading-edge Applied Mathematical Modelling Research (chapter 7)*. Nova Science Publishers, New York, 2008.

[8] R.F.WARMING AND B.J.HYETT, *The Modified Equation Approach to the Stability and Accuracy Analysis of Finite-Difference Methods*, Journal on Computational Physics **14** (1974) 159-179.

[9] A.R. MITCHELL, D.F. GRIFFITHS, *The Finite Difference Method in Partial Differential Equations*, John Wiley & Sons, New York, 1980.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Wireless teleoperation system for vehicles based on automaton and secure communications

J.A. Gázquez[1], N. Novas[1] and J.A. López-Ramos[2]
[1] *Department of Architecture of Computers and Electronics, University of Almeria, 04120 Almería, Spain*
[2] *Department of Algebra and Analysis, University of Almería, 04120 Almería, Spain*

emails: jgazquez@ual.es, nnovas@ual.es, jlopez@ual.es

Abstract

We present a teleoperation system via radio for vehicles of combustion engine or electric, designed for usings in which human presence is not adequate, such as access to dangerous sites after catastrophes, spraying tasks in farming environments, road rollers, etc. The system has been developed by means of a radio modem of specific purpose that interchanges digital information of commands from the console to the vehicle and states from the vehicle to the console. A security system prevents against non-desired functioning in case of loss of communication or outsider intromissions. This system has been implemented and successfully tested in a prototype of vehicle used for spraying tasks.

Key words: Telecontrol, finite automaton, secure communications

1. Introduction

There exist several control systems of vehicles for several usings. On one hand we have autonomous functioning vehicles such as those described in [1] and [2]. This type of vehicles does not need an operator and have autonomy for the followed route. They are based on a navigation schedule. Therefore autonomous systems present as main problems the necessity of information a priori about the work environment and possible changes in such environment, taking decisions in non-expected situations and a big complexity of the sensorial system that allows the autonomous functioning, including a local computer with high capability of

 ISBN 13: 978-84-613-5510-5

calculus and information storing, usually by means of a hard disk. By the precedent, autonomous systems need a complex and delicate control system.

On the other hand, well-known classical telecontrolled systems require a permanent attention and direct observation of the behaviour of the vehicle by the operator. The solution of events such as loss of communication may cause serious consequences in these systems or in the environment. The system that we introduce and have implemented is based on that the operation console interprets what is the desired action to be developed and, by means of a dialogue of operative commands and state responses, the vehicle carries out the desired tasks, solving, automatically, possible problems of the classical case such as loss of communication and decreases the bandwidth of the communication channel. Moreover, our system allows taking pre-programmed decisions in case of non-expected events, such as automatic detention in case of loss of communication or proximity to an obstacle in its trajectory. This system is adequate for controlling vehicles that can be directly watched by the operator or monitored by video cameras in the vehicle. Security of communications is also important depending on the purposes of the vehicle. We have used a novel key selection method for real time communications based on a Linear Feedback Shift Register which allows selecting keys from a list in a pseudorandom manner.

1.1 Technology

The digital communication system has been implemented by means of bidirectional narrow band radio-modems, in the band of 400-470 MHz. The modulation is 4L-DFSK with a speed of 9600 bits/s, which is enough since the control system in the vehicle is continuously attended by a local control and the communication system only transfers operative commands from the operation console to the local control. The vehicle has incorporated as local control a system based on a microcontroller of the family MCS-51 with the embedded program in flash memory. The system has digital outputs by means of drivers of solid state, relay outputs and analogical power outputs with PWM technology to act on the elements of control of the vehicle.
The operation console has a joystick in order to input the information about motion, stop or motion directions by the operator. It also has lightning indicators that inform the operator about the state of the vehicle in what concerns to actions developed by the vehicle in real time. The console system has been implemented by means of an embedded microcontroller, taking into account a low consume in order to be supplied with batteries

 ISBN 13: 978-84-613-5510-5

2. System Design

Our system, Figure 1, is formed by two interconnected subsystems via the wireless communication device (radiomodem). The first one is constituted by the operation console and the other one is located in the vehicle and corresponds to the control system of drivers of the engines, direction control and additional functionalities, such as a spraying system control, cameras, etc.

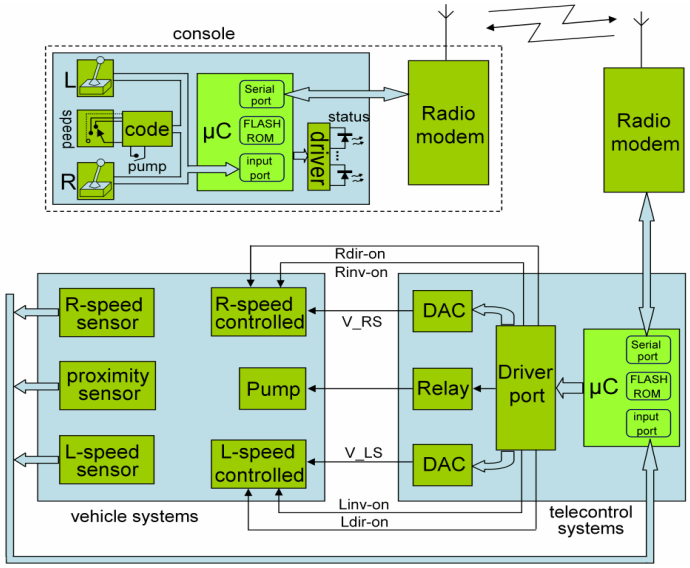


Fig. 1 *Block diagram of the system*

2.1 Console

The console is formed by a couple of joysticks, a speed selector, interrupters for activation of mechanisms in the vehicle such as spraying functions in the case of a vehicle with farming purposes, and lightning indicators of the state of the vehicle, that inform about the state of communications and actions that are being carried out by the vehicle. The system in charge of the reading of the console controls is based on a microcontroller of the family MS51 whose functioning is as follows. A finite automaton, Figure 2 implemented in the microcontroller makes the functions of reading and encodes as input all possible combinations of the console controls, that jointly with the information of the previous state, allow interpreting the action that the operator wants to carry out. This action is encoded as the output of the following state. The outputs of the states are transmitted to the

 ISBN 13: 978-84-613-5510-5

control system of the vehicle by means of the communication system (narrowband radiomodem) as packets with a ratio of two packets per second. After emitting each packet, the console receives a packet coming from the vehicle with information about its state. This state is shown on the lighting indicators. This ratio is adequate for tracked vehicles since they manoeuvre slowly. For other kind of vehicles with more speed and that require a faster response time this system allows to increase what needed the packets ratio for a right functioning.

The codification system that is described corresponds to a model implemented on a tracked vehicle with spraying purposes. This vehicle moves at slow or moderated speeds and it was checked that 4 speeds (vslow, slow, middle, high) and the stop state are enough for its manoeuvre. However, from this model we can extrapolate applications to other types of vehicles considering minor modifications.
The codification of the states constitutes a finite automaton, whose entries correspond with the steering levers, the speed selector and the interrupter of the trigger sprayer pump.

The entry alphabet S possesses three subsets: the first one, with respect to the direction Sd={d11, d10, d01, d00, d0-1, d-10, d-1-1, d1-1, d-11} in function of the state of action of the drive state of steering levers. In this way the entry d11 indicates the two levers forward, d00 stop, d0-1 left lever stop y right lever backward, etc; the second one with respect to speed Ss={s00, s01, s10, s11}
s00 -> vslow, s01->slow, s->10 medium and s11->high and finally with respect to the actuator Sa={a0, a1} ao -> no action, a1 -> action.

The subset of direction entries produces a subset of states which are dependent on the direction according with the following table:

dL	dR	state	Effect
1	1	S1	Straight
1	0	S2	front-right
0	1	S3	front-left
0	0	S0	Stop
0	-1	S6	reverse-left
-1	0	S5	reverse-right
-1	-1	S4	reverse-straight
1	-1	S7	turn on clockwise direction
-1	1	S8	turn on anti-clockwise direction

 ISBN 13: 978-84-613-5510-5

Not all possible transitions between states are possible for a right functioning. The next diagram defines an automaton with the set of states reduced direction of the movement.

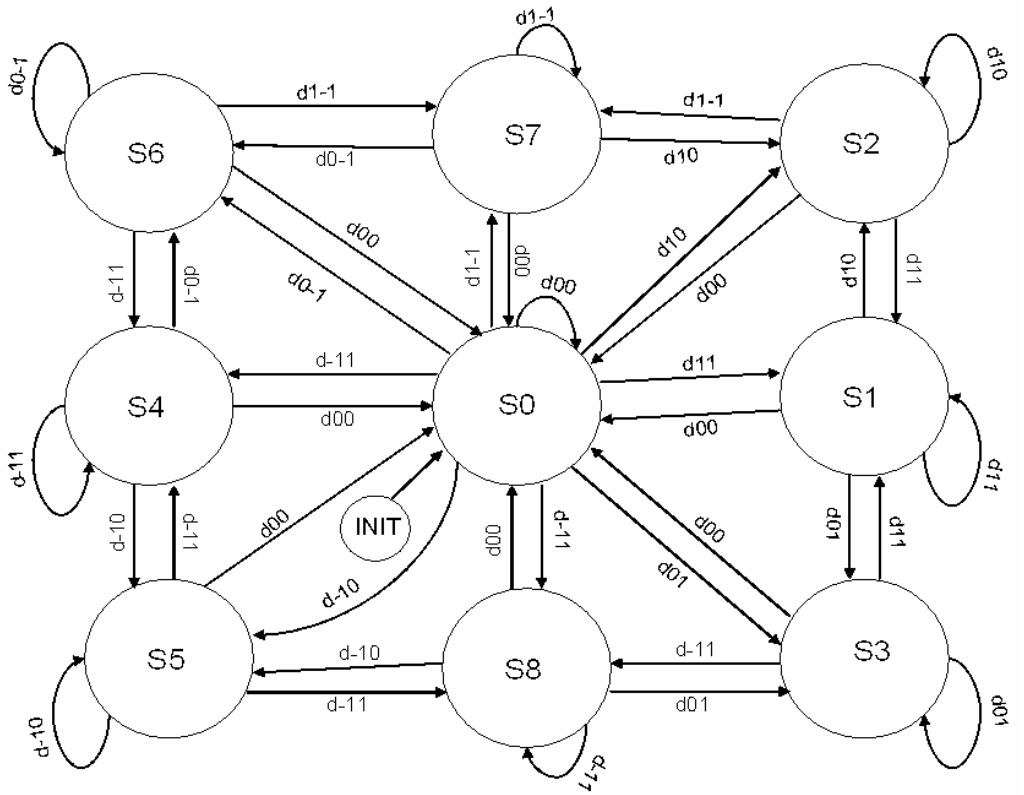


Fig. 2 *Automaton representing states and transitions*

We have taken into account that we cannot pass from entries 1 to -1 and viceversa without passing through entry 0.

For each of the states of this subset we have as a new subset the speed and the actuator system, which produces a total number of 72 possible states, although not all of them will be considered in practice because of the following exceptions:
- rotational states on its axis S8 and S7 only admit the lowest speed, ignoring other speed entries.
- in the states which imply a turn S2, S3, S5 and S6, the band corresponding to stop, what it is carried out in practice is to decrease one degree in the speed and it will only be stop when the speed selector is in vslow.

 ISBN 13: 978-84-613-5510-5

2.2 Vehicle control

The controller of the actuators of the engine of the vehicle has been implemented in a microcontroller of the same type of the console by means of another automaton. This automaton receives as inputs the packets that are sent by the console and jointly the sensorial information about direction, speed and proximity to obstacles of the vehicle, determine its next state that contains the outputs towards the actuators and the state information to be sent back to the console.
Proximity detectors in the vehicle generate the entries Sp = {p0, p1}, where p0 indicates free way and p1 that an obstacle has been detected and detention is needed.
Communications state generates the entries Sf = {f0, f1}, where f0 indicates that the last two packets have been recieved with a right checksum and f1 indicates that two consecutive errors of checksum in the packets o that they have not been received (see Figure 5). In case that the vehicle does not receive two packets of orders consecutively (fail transmission), the automaton goes to an inactivity state.
Figure 3 represents transitions between states of speed that are mapped on the states of direction Sj, depending on the codified selector on the console Ss, the proximity detectors Sp and the state of communications Sf.

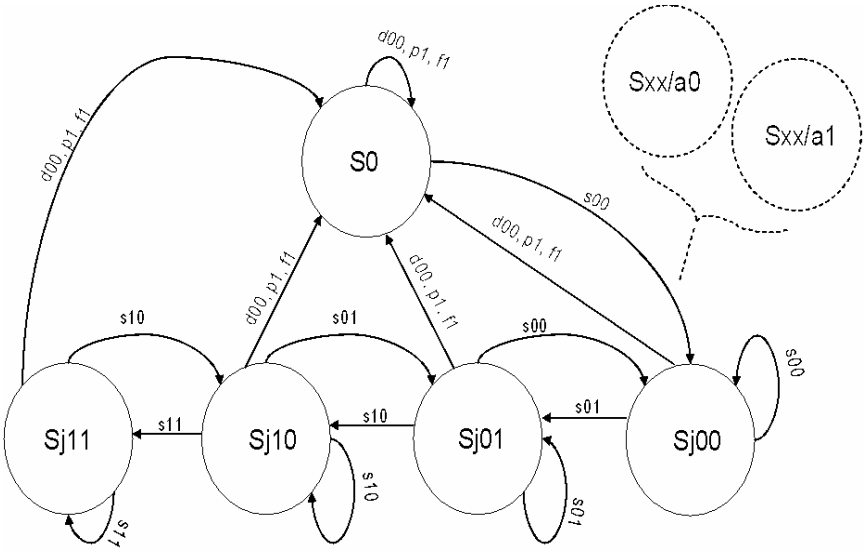


Fig. 3 Transition of states in the vehicle corresponding to speed

Each state Smn is duplicated in function of the estate of the actuator a0 to a1. The diagram of states of the vehicle does not shows differences of speed in each rolling band that is inherent to changes of direction indicated in the console automaton. We are assuming that such information is codified in each state Sj

 ISBN 13: 978-84-613-5510-5

3. Communication System

We have used a method that allows encoding the actions that at any time the vehicle can carry out using four ASCII characters. Figure 4 shows an example of the encoding used for some states in the case of tracking vehicles with a spraying system. Other vehicles with a more complex functionality can also use this system by adding some characters to the states encoding. This system of states encoding allows notorious saving in band width with respect to classical telecontrol systems and so a bit-rate of 4800 b/s is enough to the bidirectional communication (orders sending/receiving of states). Narrow band communication systems by radio are the most adequate for this type of telecontrol operations by the high ratio distance/power with low consumption, their stability and reliability and a lot of operation channels to be used. We have used a radiomodem with bit-rate 9600 b/s and F.E.C. ¾ in the UHF common used band 433 MHz.. The power of 250 mW are enough to a long distance control of the vehicle. Other communication systems such as Bluetooth or Zigbee can be used to implement this system for telecontrol in short distance, substituting the narrowband radiomodem, but with possible collisions with other users given the more extended use of these types of communications. Moreover this system increases the reliability of the control of vehicle. In case of a packet is mismatched in the communication process and F.E.C. does not detect and correct the error, this does not produces any action unless it coincides with some of the expected packets related with the state of the automaton at that time.

4bytes	1bit	1bit	1bit	1bit	1bit	2bytes	
COM	Rdir	Rinv	Ldir	Linv	pump	speed	action
0000	0	0	0	0	0	00	stop
AA01	1	0	1	0	0	01	run forward slow speed
BB23	1	0	1	0	1	23	run forward normal speed pump
GG17	0	1	0	1	0	17	run reverse normal speed
AD07	1	0	0	0	0	07	run turn left slow speed
AB07	1	0	0	1	0	07	turn on its axis slowly

Fig.4 *Example of state encoding*

One example of chronogram of the interchange of information between the vehicle and the console of control is given in Figure 5. It can be observed the messages transmitted by the console and the corresponding answers by the vehicle. In this case after a series of commands that order the running of the vehicle, all of them answered by the vehicle with its state, two consecutive fails occur due to interference or any other cause, recall that all the radio communication systems are susceptible of loss of information. In this case the vehicle goes to a stop state, restarting the running after receiving another valid command from the console.

 ISBN 13: 978-84-613-5510-5

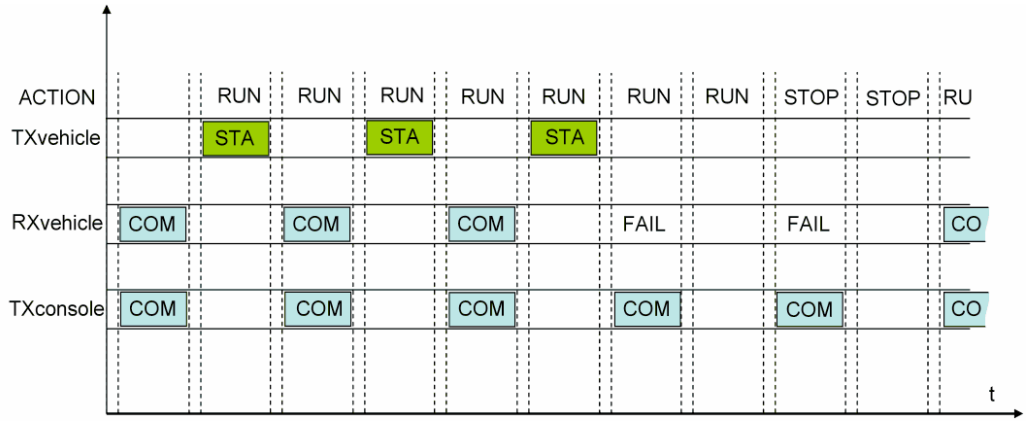


Fig. 5 *sequence of communications packets*

3.1 Security of communications

The fact that communication between the vehicle and the console takes place in real time is of a great importance. There exists a class of algorithms, known as stream cipher algorithms such that all of them treat the information bit-to-bit, usually binary digits and in real time and are very appropriate when buffering is limited some telecommunications applications as could be a radio modem. They are based in Linear Feedback Shift Registers (LFSR), which give an encryption sequence and are very suitable for hardware implementation (cf. [4] for details).
We use good properties of LFSRs to give an efficient selection key method which allows generating sequences of key identifiers. Source an destination share a list of keys and an easy algorithm to generate sequences of identifiers and therefore, these are not sent, which solves questions relative to possible compromised pairs of key-identifier and only some additional information bits about the initial state of the system that generates the sequence of identifiers is required to encrypt/decrypt a big number of blocks of information.

3.1.1 The key selection method

In this case, the source A (console or vehicle) and the destination B (console or vehicle) use a block cipher and share a list of l keys and an algorithm to generate a sequence of positions (identifiers) of the list of keys. The algorithm that we use consist in a binary LFSR (stages store one bit) with k stages, such that $2k \geq l$ and a Boolean function whose input is the state vector of the LFSR at some moment and its output is a number in the range 1-l.
Now if A wants to send a message to B then A proceeds as follows:

 ISBN 13: 978-84-613-5510-5

1. A divides the plain text into packets Pj, j=1,…,n of blocks Bj,i, i=1,…,r of appropriate length .

2. For j=1,…,n, A generates, by means of a Real Time Clock, a random number of k digits, Sj, which is taken as initial state of the LFSR.

3. Using the LFSR and the Boolean function, A generates a sequence of numbers j,i for i=1,…,r in the range 1-l and consider the key whose position in the list is j,i ,kj,i.

4. A encrypts every block Bj,i of the packet Pj, j=1,…,n, using the key kj,i, obtaining packets of blocks Cj, for j=1,…,n.

5. A sends Sj and Cj for j=1,…,n to B.

When B receives Sj and Cj, he uses Sj to generate the sequence of keys necessary to decipher the message.

In case we do not want to send Sj as plain text, we can use some of the characters of the first encrypted block of the packet Cj to get a new seed for the LFSR and so, generate a pseudorandom number which gives an identifier of a new key k to encrypt Sj. In that case, when B receives the message proceeds as follows:

1. B takes selected characters of Bj,1 to get a seed for the LFSR and gets the identifier of the key k.

2. B uses k to get Sj.

3. B uses Sj to get the sequence of keys kj,i and decrypts packet Cjj=1,…,n

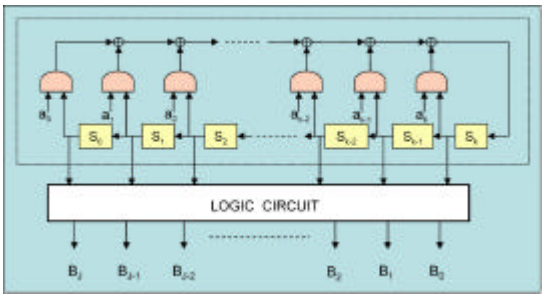


Fig 6. Key selection method

Figure 6 shows a diagram composed by an LFSR whose state vector is taken as input of a logic circuit given by a Boolean function, which produces keys kj,i. This method has been also used in [2] and [3] for telecontrol and telemetry applications.

4. Conclusions

We have introduced a novel teleoperation system for vehicles based on a finite automaton and with a secure communication system which allows to handling vehicles for different purposes in an easy and safe way. This system was

 ISBN 13: 978-84-613-5510-5

implemented on a platform for greenhouses spraying [5]. The functioning given by the designed automaton shows an easily operable vehicle and with an acceptable level of security, in what concerns to possible loss of control and with a high security level of communication in what concerns to possible interferences or outsiders as evidenced after a long testing period [6].

Fig 7. Vehicle and console

4. Acknowledgements

This paper was supported by TEC2009-13763-C02-02. First and second authors are also supported by TIC019. Third author is supported by FQM0211.

5. References

[1] A. MANDOW, J.M. GOMEZ, J. MARTINEZ, V. MUÑOZ, A. OLLERO AND A. GARCIA, *The autonomous mobile robot AURORA for greenhouse operation, vitiation,* IEEE Robotics Automation Mag. **3**(4) (1996) 18-27.

[2] M. DEVY, R. CHATILA, P. FILLATREU, S. LACROIX AND F. NASHASHIBI, WITTEN, *On autonomous navigation in a natural environment*, Robotics and Autonomous Systems **16**(1) (1995) 5-16.

[3] N. NOVAS, J.A. LOPEZ-RAMOS, J.A. GAZQUEZ AND J. PERALTA, *Unidad cifradora/descifradora de mensajes con informacion digital,* Spanish Patent P200402603 (2004).

[4] A. MENEZES, P. OORSCHOT AND S. VANSTONE, *Handbook of Applied Cryptography,* CRC Press (1996).

 ISBN 13: 978-84-613-5510-5

[5] J. SANCHEZ-HERMOSILLA, F. RODRIGUEZ, M. BERENGUER, D. MORALES AND J. GUZMAN, *Mobile Robot for spraying in greenhouses,* RWorkshop on Spray Application Techniques in Fruit Growing, Barcelona (2006).
[6] http://www.ace.ual.es/~jgazquez/icons/televeh1.MPG

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Mutation rate and the maintenance of cooperation: a parsimonious model of somatic evolution and oncogenic transitions

Philip J Gerrish[12], Jorge M Pacheco[2 3], Alan Perelson[4], Claudia P Ferreira[5]

University of New Mexico, Albuquerque NM, USA
Universidade de Lisboa, Lisboa, Portugal
Universidade de Minho, Braga, Portugal
Los Alamos National Labortatory, Los Alamos NM, USA
Universidade de Botucatu, Sao Paolo, Brazil

emails: pgerrish@unm.edu

Background and Objectives: Multicellularity, and biological complexity in general, are ultimately the result of natural selection's paradoxical tendency to foster cooperation through competition. Cooperating communities ranging from complex societies to somatic tissue are constantly under attack, however, by non-cooperating mutants or transformants, called "cheaters".

Methods: We employ simulations and analytical game-theoretic models of interactions on networks with an array of different network topologies.

Results: Structure in these communities promotes the formation of cooperating clusters whose competitive superiority can alone be sufficient to thwart outgrowths of cheaters and thereby maintain cooperation. But we find that when cheaters appear too frequently – exceeding a threshold mutation or transformation rate – their scattered outgrowths infiltrate and break up cooperating clusters, resulting in a cascading loss of community integrity, a switch to net positive selection for cheaters, and ultimately in the loss of cooperation.

Discussion and Conclusions: We discuss the possibility that our model may provide a parsimonious framework for understanding the somatic evolutionary processes leading up to transitions from normal to cancerous tissue.

Grant support: NIH (R01 GM079483 to P.J.G; AI28433, RR06555, P20-RR18754 to A.S.P); European Community (FP7 231807 to P.J.G.); FTC-Portugal (J.M.P.).

 ISBN 13: 978-84-613-5510-5

Model assumptions

Cooperation and multicellularity evolved and are maintained by very complex processes, but a first approximation to these processes may be formulated as a simple evolutionary game:

On average, cooperators "help" others (increasing others' fitness) at their own expense (decreasing their own fitness).

On average, cheaters don't "help" others (don't increase others' fitness or decrease their own fitness).

Interactions are structured on a network (regular, random, or scale free).

Cooperation and cheating are heritable traits, and mutation can toggle between the two.

Main result: critical mutation rate above which cooperation is lost as 2nd order phase transition.

$$\mu_c = (1-q_0)\left(e^{-(1+\beta)}\left(\frac{1+(n-1)q_0\varepsilon}{1+\beta}\right)^{\alpha} - e^{-(1+nq_0\varepsilon)}\left(1+q_0\left[\left(\frac{1+((n-1)q_0+1)\varepsilon}{1+\beta+\varepsilon}\right)^{\alpha}-1\right]\right)^{-1}\right)$$

$$\mu_c \approx ke^{-1}(\alpha+1)(\phi\varepsilon-\beta).$$

α = sampling coefficient
ε = average "help" conferred by cooperators
β = average advantage of being a cheater

 ISBN 13: 978-84-613-5510-5

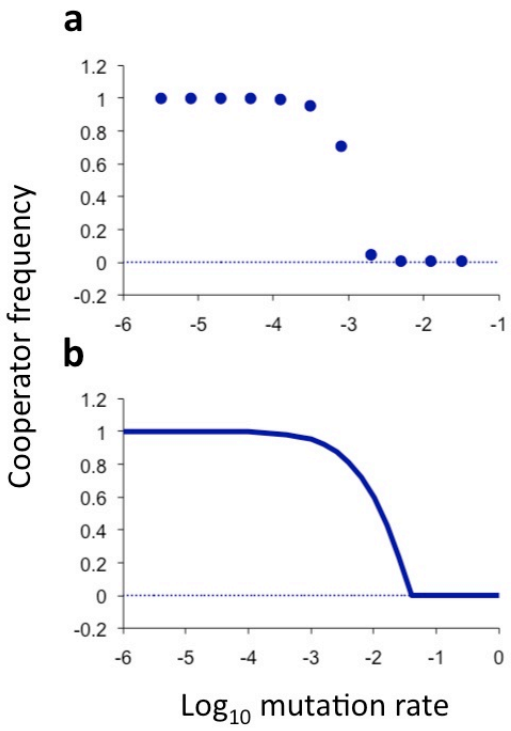


Figure 1. **Equilibrium cooperator frequencies as a function of mutation rate**. (a) As determined from simulated populations that were spatially structured on a three-dimensional grid. The population initially consists of all cooperators, and cheaters arise by spontaneous mutation. The intrinsic advantage of being a cheater (or cost of being a cooperator) was 0.01, the advantage conferred by a cooperating neighbor was 0.05, and total population size was 27,000. (b) As predicted by the analytical equilibrium solution of the "pair approximation" model, using the same parameter values but implicitly assuming an infinite population. The analytical curve reveals a sharp transition between the maintenance of cooperation and the complete loss of cooperation; the sharpness of the inflection resembles that of a second-order phase transition.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Transmission coefficient in monolayer graphene

C. Hdez. Fuentevilla[1], J. D. Lejarreta González[1]
[1] *Department Física Aplicada, University of Salamanca*

emails: chernan@usal.es, leja@usal.es

Abstract

We present a calculation for the transmission coefficient in the monolayer graphene in function of the angle through a barrier and a double barrier.

Key words: transmission, graphene, Klein paradox.

1. Introduction

Graphene is a planar and monoatomic layer of carbon atoms - then is two-dimensional (2D) - arranged on a densily packed honeycomb crystal lattice.

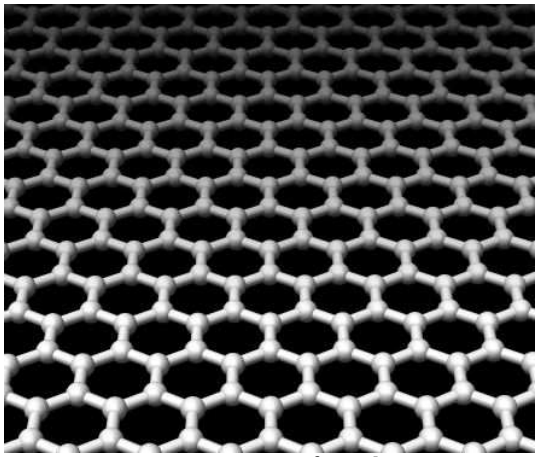


Fig 1.- Structure of graphene

P.R. Wallace wrote the first paper [5] in 1947 to the band structure of graphene and showed the inusual semimetallic behaviour in this material. More time after, in 2004 A. Geim and K. Novoselov has obtained experimentally graphene. They

 ISBN 13: 978-84-613-5510-5

started with graphite three-dimensional and extracted a single sheet (a monolayer of atoms).

The graphene is a zero-gap semiconductor [2],[3], and for low energies the carriers with proximity to the Dirac points can be described by the Dirac-like Hamiltonian

$$\hat{\mathrm{H}} = \hbar \mathrm{v_F} \begin{pmatrix} 0 & \mathrm{k_x} - \mathrm{ik_y} \\ \mathrm{k_x} + \mathrm{ik_y} & 0 \end{pmatrix} = \hbar \mathrm{v_F} \vec{\sigma} \cdot \vec{\mathrm{k}}$$

Where $\vec{k}$ is the momentum, $\vec{\sigma}$ the 2D Pauli matrix and v_F the Fermi velocity that is independent of k, and your value is $v_F \approx 10^6$ m/s. This play the role of speed of light, but is $v_F \approx$ c/300. Givig rise to the conical sections for $|\mathrm{E}|$ <1 eV. We have a linear spectrum $\mathrm{E} = \hbar \mathrm{v_F k}$, leading to the zero effective mass for electrons and holes, and then they are called *Dirac fermions*. They are like particles relativistic described by the Dirac equation for spin ½ particles. The electronic states are composed of states belonging to the different sublattices, and then we use two-component wave-functions (spinors).

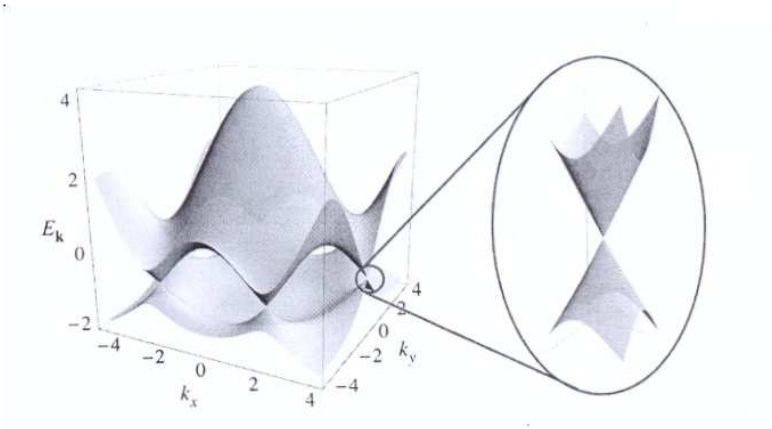


Fig2 .- Energy spectrum in the proximity of the Dirac points.

The graphene is very interesant because it has a exceptional electronic quality. Experimental results from transport measurements show *high electron mobility* μ that can exceed 15,000 cm^2/Vs in the concentrations n as high as 10^{13} cm^{-2} even under ambient conditions [3].

 ISBN 13: 978-84-613-5510-5

2. Theory

The differences in the calculus of the transmission coefficient between traditional semiconductors (AlGaAs, …) and the graphene are :

1. The carriers are governed in the first case by a equation differential of second order and in the second by a equation differential of first order.

$$\left[-\frac{\hbar^2}{2m}\vec{\nabla}^2 + V(\vec{r})\right]\psi(\vec{r}) = E\,\psi(\vec{r}) \qquad \text{Ec. Schrödinger}$$

$$-iv_F\vec{\sigma}\cdot\vec{\nabla}\psi(\vec{r}) = E\,\psi(\vec{r}) \qquad \text{Ec. Dirac}$$

2. In the graphene we will have only carbon atoms, not it´s possible different combinations of elements.

The two components of $\psi(r)$ close to the Dirac point, obeys the 2 D Dirac equation [4].

$$-iv_F\vec{\sigma}\cdot\vec{\nabla}\psi(\vec{r}) = E\,\psi(\vec{r})$$

And we can to write

$$\psi_{\vec{K}}(\vec{k}) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ \pm e^{i\theta_k} \end{pmatrix} \text{ when } \theta_k = \arctan\left(\frac{k_x}{k_y}\right).$$

2.1 Square barrier

We begin with a barrier of width D. Then the potential is

$$V(x) = \begin{cases} 0, & x < 0 \quad \text{region I} \\ V_0, & 0 \le x \le D \quad \text{region II} \\ 0, & D < x \quad \text{region III} \end{cases}$$

The wavefunctions in the three regions can be write in terms of incident and reflected waves, in principle, but we can consider for the total the condition that the incident amplitude is the unity, the reflected amplitude is r and the transmitided amplitude is t, and is not reflected amplitude in the final.

In the region I, II and III :

 ISBN 13: 978-84-613-5510-5

$$\psi_{I}(\vec{r}) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ se^{i\phi} \end{pmatrix} e^{i(k_x x + k_y y)} + \frac{r}{\sqrt{2}}\begin{pmatrix} 1 \\ se^{i(\pi-\phi)} \end{pmatrix} e^{i(-k_x x + k_y y)}$$

$$\psi_{II}(\vec{r}) = \frac{a}{\sqrt{2}}\begin{pmatrix} 1 \\ s'e^{i\theta} \end{pmatrix} e^{i(q_x x + k_y y)} + \frac{b}{\sqrt{2}}\begin{pmatrix} 1 \\ s'e^{i(\pi-\theta)} \end{pmatrix} e^{i(-q_x x + k_y y)}$$

$$\psi_{III}(\vec{r}) = \frac{t}{\sqrt{2}}\begin{pmatrix} 1 \\ se^{i\phi} \end{pmatrix} e^{i(k_x x + k_y y)}$$

When s = sgn(E), s´ = sgn(E-V_0), ϕ = arctan(k_y/k_x), $k_x = k_F \cos\phi$, $k_y = k_F \sin\phi$, k_F the Fermi momentum is $k_F = 2\pi/\lambda$, θ = arctan(k_y/q_x) and $q_x = \sqrt{((V_0 - E)/\hbar v_F)^2 - k_y^2}$.

The coefficients are determined only by the continuity of the wavefunction, because is a equation differential of first order

$$\psi_{I}(x = 0, y) = \psi_{II}(x = 0, y)$$
$$\psi_{II}(x = D, y) = \psi_{III}(x = D, y)$$

Then we obtain t, after we multiply to the conjugate t*, and we will have the transmission coefficient T(ϕ)

$$T_{exac}(\phi) = \frac{\cos^2\theta\cos^2\phi}{[\cos(Dq_x)\cos\phi\cos\theta]^2 + \sin^2(Dq_x)(1 - ss'\sin\phi\sin\theta)^2}$$

We achieve a approximation $|V_0| >> E$ then $\theta \to 0$, and

$$T_{aprx}(\phi) \cong \frac{\cos^2\phi}{1 - \cos^2(Dq_x)\sin^2\phi}$$

2.2 Double barrier

We construct a double barrier because then we will have a well. Then the potential is

 ISBN 13: 978-84-613-5510-5

$$V(x)=\begin{cases} 0, & x<0 & \text{region} & \text{I} \\ V_0, & 0\le x\le D_1 & \text{region} & \text{II} \\ 0, & D_1<x<L & \text{region} & \text{III} \\ V_0, & L\le x\le L+D_2 & \text{region} & \text{IV} \\ 0, & L+D_2<x & \text{region} & \text{V} \end{cases}$$

And the equations are :

$$\psi_{\text{I}}(\vec{r})=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\ se^{i\phi}\end{pmatrix}e^{i(k_x x+k_y y)}+\frac{r}{\sqrt{2}}\begin{pmatrix}1\\ se^{i(\pi-\phi)}\end{pmatrix}e^{i(-k_x x+k_y y)}$$

$$\psi_{\text{II}}(\vec{r})=\frac{a}{\sqrt{2}}\begin{pmatrix}1\\ s'e^{i\theta}\end{pmatrix}e^{i(q_x x+k_y y)}+\frac{b}{\sqrt{2}}\begin{pmatrix}1\\ s'e^{i(\pi-\theta)}\end{pmatrix}e^{i(-q_x x+k_y y)}$$

$$\psi_{\text{III}}(\vec{r})=\frac{c}{\sqrt{2}}\begin{pmatrix}1\\ se^{i\phi}\end{pmatrix}e^{i(k_x x+k_y y)}+\frac{d}{\sqrt{2}}\begin{pmatrix}1\\ se^{i(\pi-\phi)}\end{pmatrix}e^{i(-k_x x+k_y y)}$$

$$\psi_{\text{IV}}(\vec{r})=\frac{f}{\sqrt{2}}\begin{pmatrix}1\\ s'e^{i\theta}\end{pmatrix}e^{i(q_x x+k_y y)}+\frac{g}{\sqrt{2}}\begin{pmatrix}1\\ s'e^{i(\pi-\theta)}\end{pmatrix}e^{i(-q_x x+k_y y)}$$

$$\psi_{\text{V}}(\vec{r})=\frac{t}{\sqrt{2}}\begin{pmatrix}1\\ se^{i\phi}\end{pmatrix}e^{i(k_x x+k_y y)}$$

We will have only two angles, ϕ when the potential is zero and θ when is V_0. We impose another the continuity of the wavefunctios in the limits.

$$\psi_{\text{I}}(x=0,y)=\psi_{\text{II}}(x=0,y)$$
$$\psi_{\text{II}}(x=D_1,y)=\psi_{\text{III}}(x=D_1,y)$$
$$\psi_{\text{III}}(x=L,y)=\psi_{\text{IV}}(x=L,y)$$
$$\psi_{\text{IV}}(x=L+D_2,y)=\psi_{\text{V}}(x=L+D_2,y)$$

3. Results

3.1 Square barrier

$T_{exac}(\phi)$ depends on the width of barrier D, the energy *E*, the high of potential V_0 and the angle of incidence ϕ. The graphic of the transmission coefficient is simetric because $T_{exac}(\phi)=T_{exac}(-\phi)$.
$T_{exac}(\phi)$ is the unity :

 ISBN 13: 978-84-613-5510-5

a/ for the values of Dq_x satisfaying $Dq_x = n\pi$
b/ for normal incidence $\phi = 0$

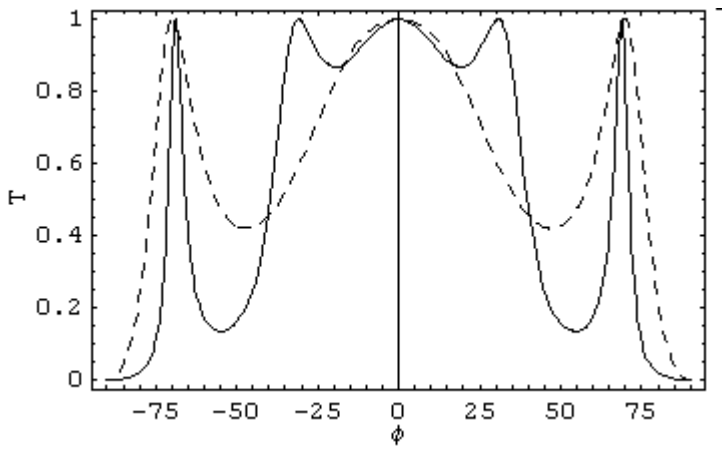


Fig 3.- T_{exac} for E = 80 meV, D = 110 nm, V_0 = 200meV (solid line) and 300 meV (dashed line)

The transmission coefficient unity has the significance the barrier is total transparent, and this is relationated with the *Klein paradox* [1].

$T_{aprx}(\phi)$ depends on D, E, V_0 and ϕ. The difference between T_{exac} and T_{aprox} is more important in the minimum of the function.

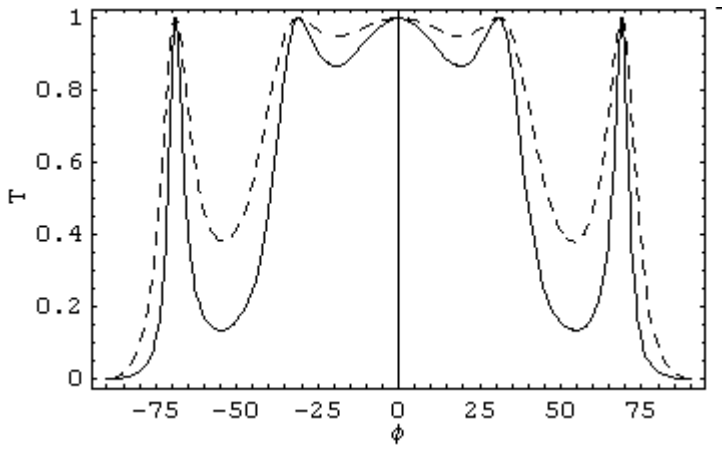


Fig 4.- T_{exac} (solid line) and T_{aprox} (dashed line) for E = 80 meV, V_0 = 200 meV and D = 110 nm

3.1 Double barrier

The function $T(\phi)$ is symmetric in ϕ.

 ISBN 13: 978-84-613-5510-5

When $D_1 = D_2$ we will have a symmetric system, and only the variables are E, V_0, D and L. We will appear three zones with high transmission coefficient even there is some points which arise the unity.

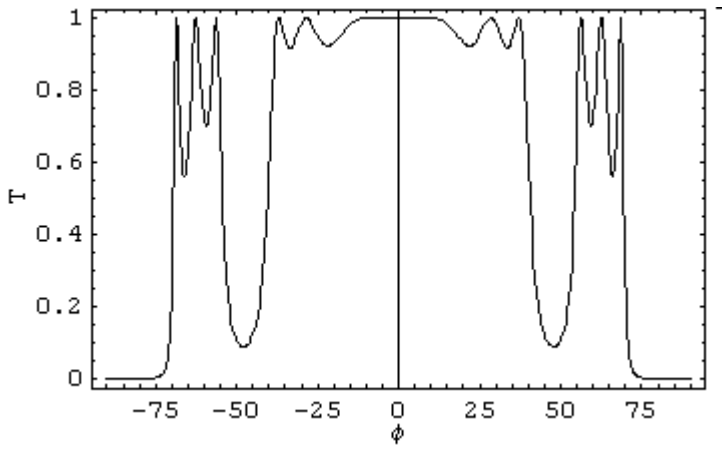


Fig 5.- T for E = 80 meV, V_0 = 200 meV, $D_1 = D_2$ = 110 nm, L = 55 nm

When $D_1 \neq D_2$ it´s different the result of to place the barriers in the position first or second. The behaviour of a system with two barriers when L = D1, is the same that a only barrier with width D1+D2.

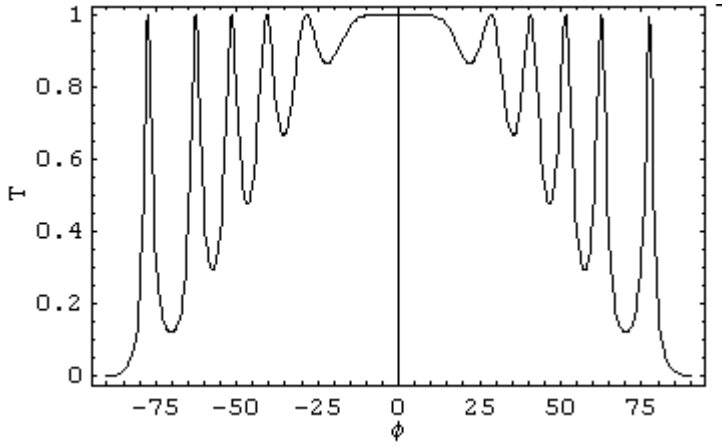


Fig 6.- T for E = 80 meV, V_0 = 200 meV, D_1 = 110nm, D_2 = 220 nm, L = 110 nm

 ISBN 13: 978-84-613-5510-5

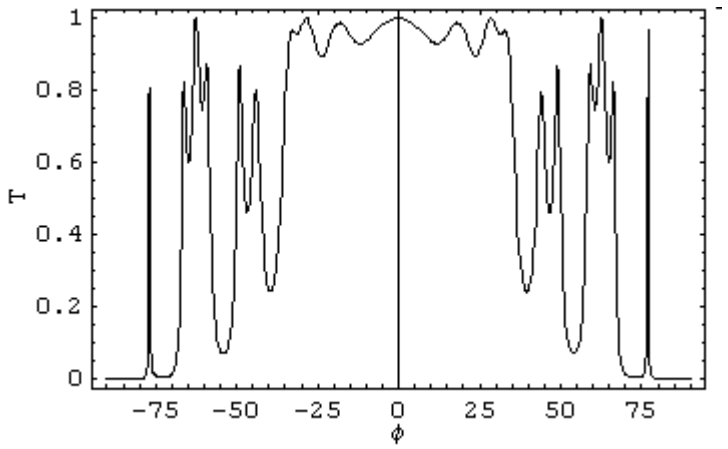


Fig 7.- T for E = 80 meV, V_0 = 200 meV, D_1 = 220nm, D_2 = 110 nm, L = 110 nm

Acknowledgment

This research has been supported in part by the DGICYT under project FIS2009-07880 and also by the project FS/7-2009 of the Fundación Samuel Solórzano Barruso.

4. References

[1] A. CALOGERACOS AND N. DOMBEY, *History and Physics of the Klein Paradox*, Contemp. Phys. **40** (1999) 313-321.

[2] A.H. CASTRO NETO, F. GUINEA, N.M.R. PERES, K.S. NOVOSELOV AND A.K.GEIM , *The electronic properties of graphene*, e-print: arXiv: 0709-1163 (2008).

[3] A.K.GEIM AND K.S. NOVOSELOV, *The rise of graphene*, Nature Mat. **6** (2007) 183-191.

[4] M.I. KATSNELSON, K.S. NOVOSELOV AND A.K. GEIM, *Chiral tunnelling and the Klein paradox in graphene*, Nature Phys. **2** (2006) 620-625.

[5] P.R. WALLACE, *The band theory of graphite*, Phys. Rev. **71** (1947) 622-634.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Improving sample flow in planar preconcentrator

R. Inglés[1], J. Pallarés[2] and J.L. Ramirez[1]

[1] *Dept. of Electronic, Electrical and Automatic Control Engineering, ETSE, Universitat Rovira i Virgili, Tarragona, Spain*
[2] *Department of Mechanical Engineering, ETSEQ, Universitat Rovira i Virgili, Tarragona, Spain*

emails: roser.ingles@urv.cat

Abstract

Preconcentrators are really important to detect gases that are dangerous at very low concentrations. We use Comsol Multiphisics to study their behaviour and improve their design. We are working with a planar device. In such a device most part of the analyte to be detected does not affect the absorbing material. We discover this problem by means of simulation and propose a solution which improves the chamber design forcing a bigger amount of the sample flow to interact with the absorbent material, thus increasing the concentration factor almost twice.

Key words: simulation, fluid flow, preconcentrator

1. Introduction

Several toxic gases such as benzene are dangerous at low concentrations. At present gas sensors are not able to detect so low concentrations. That is why the design and implementation of preconcentrators is an important task. Our aim is to design a microsystem which contains a sensor and a preconcentrator constructed in the same silicon substrate. In this way we obtain a planar device which is technologically and economically more efficient than 3D devices[1]. In such a device most part of the analyte to be detected does not affect the absorbing material. Our proposal improves the chamber design forcing a bigger amount of the sample flow to interact with the absorbent material increasing the concentration factor almost twice.

 ISBN 13: 978-84-613-5510-5

We have simulated the chamber we are using to emulate the microsystem in order to be able to compare our results with experimental ones. It is a cylindrical 6-mm diameter 3.8-mm height chamber. It has input and output 1.27-mm diameter tubes situated at 2 mm above the base (See Fig. 1a). Preconcentrator is located at the base of the chamber (See Fig. 1b).

During the adsorption phase preconcentrator is at ambient temperature and the adsorbent is taking volatiles. During the desorption phase the preconcentrator is heated in order to desorb all the volatiles in a really short time. Adsorption and desorption processes are modelled in the preconcentrator as a simple linear function.

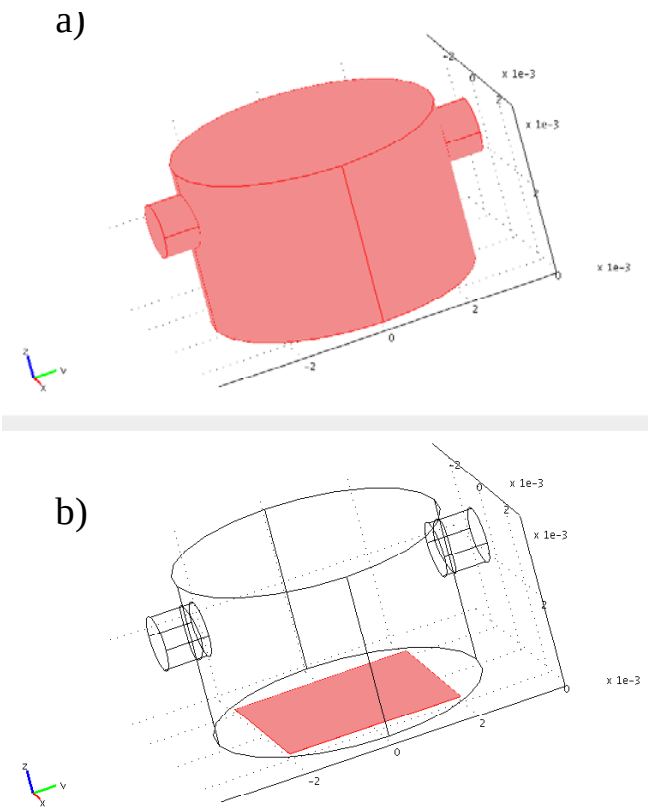


Fig. 1: Preconcentrator Chamber: a) Chamber morphology b) Preconcentrator location

 ISBN 13: 978-84-613-5510-5

2. Fluid flow simulation

We are coupling different models of Comsol Multiphisics in transient mode: Weakly Compressible Navier-Stokes model for fluid flow and Convection and Diffusion model for concentration. Temperature model has not been used because, at working conditions, temperature changes has no effect in flow dynamics as there are no significant density variation, neither convection effects. Thus simulation time is greatly reduced. We use the Weakly Compressible Navier-Stokes model for fluidic. We introduce laminar velocity as input and constant pressure as output boundaries constrains. With the Convection and Diffusion model we simulate the concentration variations. In the preconcentrator there is a two-way reaction which converts analyte free (Af) to analyte in the wall (AW). This is governed by the velocity of reaction.

$$A_W \underset{k\_on}{\overset{k\_off}{\rightleftarrows}} A_f \tag{1}$$

So, next function must be fulfilled in our preconcentrator[2-4]:

$$\frac{\partial C_{A_w}}{\partial t} = -k\_off \cdot C_{A_w} + k\_on \cdot \left(C_{\max} - C_{A_f}\Big|_W \right) \tag{2}$$

In our model, we use this function in the preconcentrator boundary.

First, we simulated the flow in the conventional chamber. We realized that most part of the gas sample crosses the chamber without affecting the preconcentrator because it is at high distance above of it and passes as laminar flow (See Fig. 2). So, we proposed a new design using an “obstacle” which forces the gas sample to go down almost perpendicular to the preconcentrator surface (See Fig. 3). We constructed a “wall” and repeated the measurements in order to see the difference.

 ISBN 13: 978-84-613-5510-5

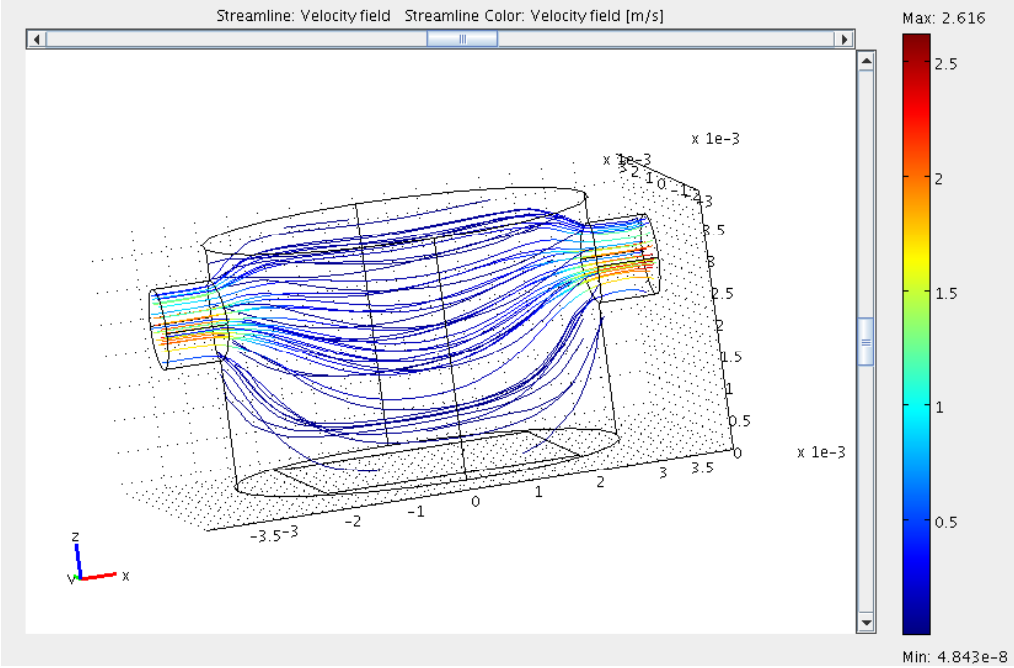


Fig. 2: Streamline of velocity field simulation in a conventional chamber

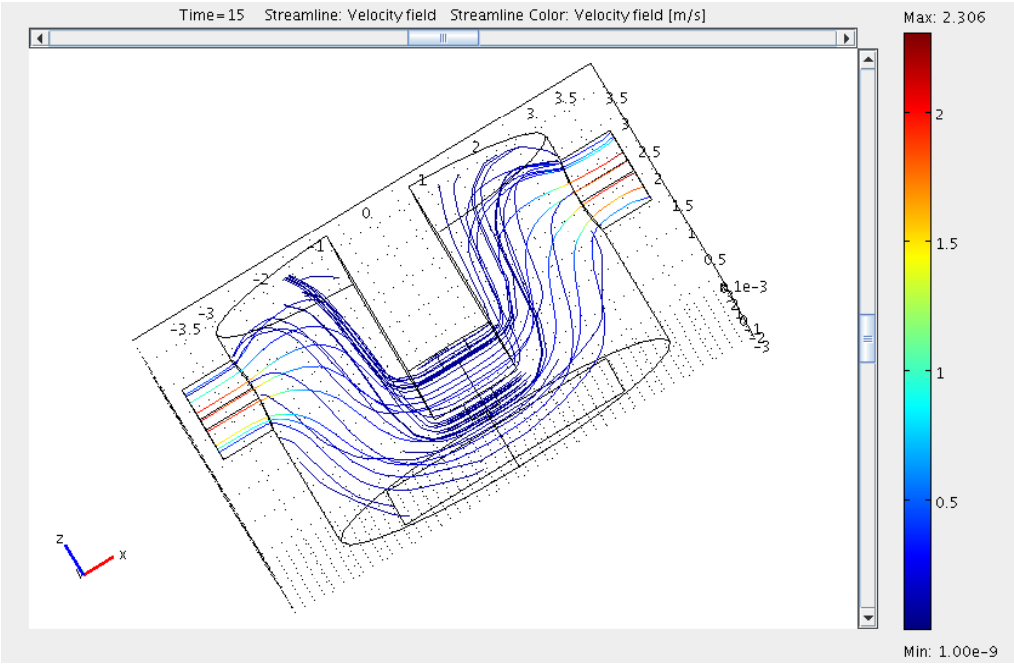


Fig. 3: Streamline of velocity field simulation in a chamber with wall

 ISBN 13: 978-84-613-5510-5

3. Simulation results

In Fig. 4, we compare the amount of benzene adsorbed without wall and with the wall placed in the middle. In Table 1 we can see the concentration of benzene during the adsorption phase. The analyte column represents the amount of analyte going out of the chamber. ΔAnalyte input-output column shows the difference between input and output analyte, with this value we extract the percentage adsorbed and the improvement factor related to this percentage. We observe that the wall enhances the retention capability in a 1.78 factor. So, using this wall, the preconcentrator system adsorbs more analyte in the same period of time.

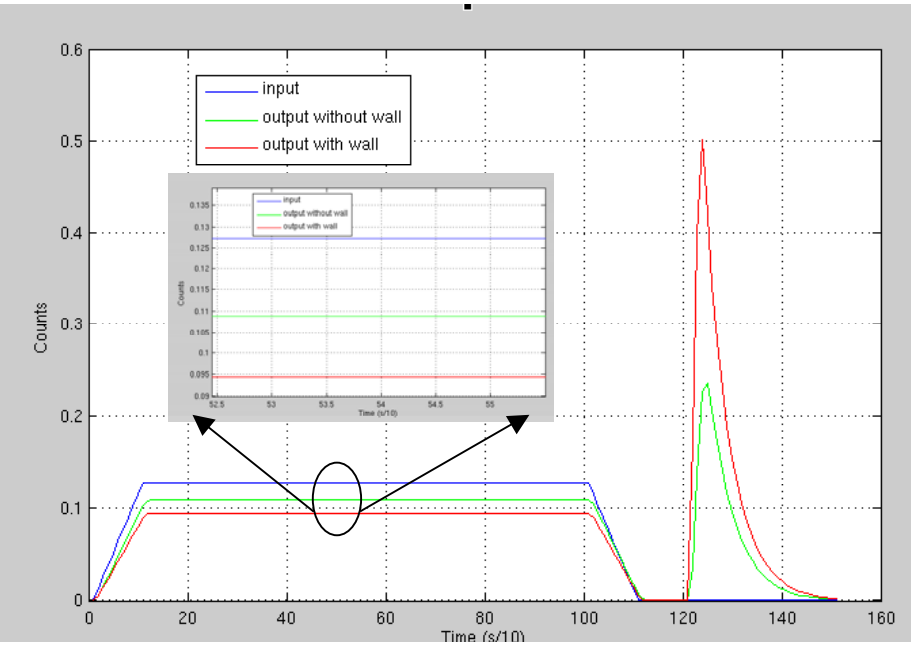


Fig. 4: Comparison results of input and output analyte without and with wall

Table 1: Comparison of simulation results without and with wall

	Analyte (u.a.)	ΔAnalyte input-output (u.a.)	Amount of analyte adsorbed (%)	Retention factor improvement
Input	0.1273			
Output without wall	0.1088	0.0185	14.53 %	1
Output whit wall	0.0944	0.0329	25.84 %	1.78

 ISBN 13: 978-84-613-5510-5

4. Experimental results

Preliminary studies were made on a 4 mm x 4 mm side porous alumina substrate with activated carbon deposited on top as a preconcentrator. However, future implementation will be made on 3 mm x 3 mm preconcentration membranes in silicon technology, already fabricated. We used gas chromatograph mass spectrometry (GCMS) to measure the desorption peak.

A flow of 150 ppb of benzene diluted in CO2 is injected during 10 minutes at ambient temperature, so the preconcentrator is adsorbing the analyte. Then, we introduce helium and heat the preconcentrator to desorb all the benzene.

Experimentally, we have obtained an improvement factor of 1.85, which fits quite well the simulation result.

5. Conclusions

By means of simulations, we detected a ineffienciency which was completely unknown and proposed a solution which was substantiated by the experimental measurements.

6. Acknowledgments

This work is financially supported by the CICYT Project TEC2009-07107

7. References

[1] T. Nakamoto et al, SENSORS AND ACTUATORSA B-CHEMICAL, 69 (2000), pp. 58-62
[2] R. Manginell et al, IEEE SENSORS JOURNAL, (2007) pp. 1032-1041
[3] CL. Chuang, et al.," Kinetics of benzene adsorption onto activated carbon". ENVIRONMENTAL SCIENCE AND POLLUTION RESEARCH, 2003.
[4] CL. Chuang, et al.," Modeling VOCs adsorption onto activated carbon". CHEMOSPHERE, 2003.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

COMPUTATION OF THE RESPONSE TO THE MOVING LOAD OF PERIODICALLY SUPPORTED BEAM

Rachid .Lassoued [1], Guy. Bonnet [2]

[1] *Laboratoire des Matériaux et Durabilité des Constructions Département de Génie Civil Faculté des Sciences de l'Ingénieur Université Mentouri Constantine, Algérie*

[2] *Université Paris Est Marne la Vallée, Laboratoire de Modélisation et Simulation Multi Echelleécanique, CNRS UMR 8208, 5 boulevard Descartes, 77454 Marne la Vallée Cedex, France*

emails: rachid_lassoued@yahoo.fr,guy.bonnet@univ-paris-est.fr

Abstract

A mathematical model is presented to predict the vibration of a rail excited by a moving load. The track is modeled as an infinitely long beam supported over a finite section by discrete support. The supports represent pad/sleeper/ballast system of railway track, they are spaced regularly. The rail idealizes these periodic structures. A periodic system consists on a number of identical elements; coupled together in identical ways to form a whole system. This paper describes the computation of the wave field induced by a moving load on a periodically supported beam. The work starts by calculate the Green function of the free Euler beam without support by using the direct integration. After woods, it introduces the supports into the model established by using the superposition principle which states that the response from all sleepers points and from the external point force add up linearly to give a total response. The periodicity of supports is described by Bloch's theorem. The model developed gives the choice of different support types: mass support, spring support, mass/spring systems…The homogeneous system thus obtained represents a linear differential equation which governs rail response. It is initially solved in the homogeneous case, it admits a no null solution if its determinant is null, this permit the establishment the dispersion equation to Bloch waves and wave bands. The Bloch waves and dispersion curves contain all the physics of the dynamic problem and the wave field induced by a dynamic load applied to the system is finally obtained by decomposition into Bloch waves, similarly to the usual decomposition into dynamic modes on a finite structure. The method is applied to obtain the field induced by a load moving at constant velocity on a thin beam supported by periodic elastic supports.

Key words: Bloch wave**,** Euler beam, Green function, Computation, Moving load

 ISBN 13: 978-84-613-5510-5

1. Introduction

Vibration of infinite periodic structures has been studied extensively in the past 30 years with the focus mainly on free vibration propagation and forced vibration induced by stationary harmonic loads [1]. A mathematical model is presented by Heckl [2] to predict the vibration of a rail excited by a rolling wheel. The track is modeled as an infinitely long beam supported over a finite section by discrete support systems for non moving load. Vibration of an infinite periodic beam subject to a moving harmonic load has been investigated by Belotserkovskiy [3]. In this study, the author considered a single segment only by using boundary conditions derived according to the Euler beam theory. The rail is modeled as an Euler beam with flexible supports in the work of Nordborg [4]. The supports of the beam represent pad/sleeper/ballast system of the railway track; they are spaced regularly. The exact solution, in the frequency domain, of the linear differential equation is presented.

The method used to obtain the response to dynamic load include the computation of transfer matrices ([5][6][7][8], the use of Bloch waves [9] [10] and finite element solutions[11][12].For a Timoshenko beam, Heckl ([13]) produced the dispersion curves and showed that different modes appear, related to the coupling between different degrees of freedom of the beam (flexion, torsion, shear...). The response of a Timoshenko beam to a static (non moving) dynamic load was produced by Hamet [14].

The aim of the paper is to produce the response of a periodically supported thin beam submitted to a moving load, by using the Bloch transform (Allaire [15], Sanchez [16]). Parts of the solution are similar to the one used by (Langley [8]) for 2D structures or Hamet [14] for a Timoshenko beam submitted to a dynamic (non moving) load.

2. Green function for Euler beam

Let us consider a thin beam with an inertia section moment I which is subjected to a moving vertical moving load $F(x,t)$. The vertical beam displacement is the solution of the dynamic beam equation (eq. 1). Euler model is considered:

$$EI\frac{\partial^4 U(x,t)}{\partial x^4} + m\frac{\partial^2 U}{\partial t^2} = F(x,t) \qquad (1)$$

Where m is the mass of the beam by unit of length and E the Young's modulus.

 ISBN 13: 978-84-613-5510-5

If the harmonic concentrated load is applied at the point x_0, in the Fourier space, the movement equation becomes:

$$EI\frac{\partial^4 \tilde{U}(x,\omega)}{\partial x^4} - m\omega^2\tilde{U}(x,\omega) - \delta(x-x_0) = 0 \quad (2)$$

Where $\tilde{U}$ is the Fourier transform of U.

The Green function associated to this equation is the response to an unit load applied at the point t x_0. It is the solution of the equation:

$$\frac{d^4G(x,x_0)}{dx^4} - k_b^4 G(x,x_0) = \delta(x-x_0) \quad (3)$$

With:

$$k_b = \sqrt[4]{\frac{m\omega^2}{EI}} \quad (4)$$

The Green function $G(x,x_0)$ can be now determined by direct integration; it is thus the solution of the equation (3) and represents the vertical vibration at the point x_0, of the beam rail subjected to an harmonic force.

- In any point $x \neq x_0$ (in any point other than the excitation point) the system is free and :

$$\frac{d^4G(x,x_0)}{dx^4} - k_b^4 G(x,x_0) = 0 \qquad x \neq x_0 \quad (5)$$

At the point $x = x_0$, the discontinuity is assumed by the third derivative of $G(x,x_0)$. The function and its two first derived are continuous at $x = x_0$. The integration of the equation (2) on both sides of x_0 and if we take into account the continuity of the function $G(x,x_0)$ at x_0, we obtain :

$$\frac{d^3}{dx^3}G(x,x_0)\Big|_{-}^{+} = \frac{1}{EI} \quad (6)$$

- At $x = \pm\infty$ the function is limited: $|G(x,x_0)| < \infty$

 ISBN 13: 978-84-613-5510-5

The Green function is then expressed by a combination of elementary solutions of the equation (3). Taking into account the condition at infinity, it can be written:

$$G(x,x_0) = a_1 e^{k_b(x-x_0)} + a_2 e^{k_b(x-x_0)} \qquad x < x_0 \tag{7}$$

$$G(x,x_0) = b_1 e^{-k_b(x-x_0)} + b_2 e^{ik_b(x-x_0)} \qquad x > x_0 \tag{8}$$

The coefficients a_1, a_2, b_1, b_2 are evaluated from the Green function and its first derived must satisfy:

$$G(x,x_0)\Big|_{x_{0-}}^{x_{0+}} = 0 \qquad \frac{d}{dx}G(x,x_0)\Big|_{x_{0-}}^{x_{0+}} = 0 \qquad \frac{d^2}{dx^2}G(x,x_0)\Big|_{x_{0-}}^{x_{0+}} = 0 \tag{9}$$

The solution of this problem is exposed by Heckl (Maria **A. Heckl, 2001)** in the more general case of the Timoshenko beam. The result in this case of the Euler beam is a Green function.

$$G(x,x_0) = \frac{1}{4EI\,k_b^3}\left[ie^{ik_b|x-x_0|} - e^{-k_b|x-x_0|}\right] \tag{10}$$

3. The free motion of the periodically supported beam: Bloch waves

We consider a beam periodically supported with N supports placed between $-\infty$ and $+\infty$. The period is given by the regular spacing l. The rigidity of the N^{ieme} discrete elastic support, corresponding to the deformation of the system (elastic support, mass support...), is Z_n , the force transmitted by the discrete support is:

$$F_n = -Z_n U(n,l) \tag{11}$$

It is considered that all the discrete supports are characterized by the same rigidity $Z_n = Z$. The global response of the structure can be described by applying the superposition principle, which consists of the linear summation of all the answers due to the various supports. Then, the displacement with the position x is given by:

$$\eta(x) = \sum_{-\infty}^{+\infty} G(x,x_n) F_n \tag{12}$$

 ISBN 13: 978-84-613-5510-5

Where $G(x,x_n)$ is given by the equation (10) for the Euler beam.

The relation (9) is thus valid for any point x_m.

$$\eta(x_m) + \sum_{-\infty}^{+\infty} Z\eta(x_n)\, G(x_m, x_n) \tag{13}$$

The application of the Floquet theorem gives the relation between two adjacent supports:

$$\eta(x_{m+1}) = \eta(x_n)e^{-\gamma l} \tag{14}$$

With:

$$\gamma = \alpha + ik \tag{15}$$

γ is the Bloch constant propagation. It is generally complex. α is the wave attenuation and k is the Bloch wave number .

By substituting the expression of the Green function given by the equation (9) in the equation (12) and by multiplying this equation by $e^{\gamma ml}$, changing the index of summation $m-n$ by n gives the following equation which defines the relation between γ and the displacement η_0 .

$$\eta_0 + Z\eta_0 \sum_{-\infty}^{+\infty} \left[\frac{1}{4EIk_b^4} \left(ie^{ik_b|n|l} - e^{k_b|n|l} \right) e^{\gamma nl} \right] = 0$$

In the following, only the waves not attenuated will be considered, which means that the sum can be evaluated by usual formulas of the geometrical series.

For the Euler model, we can have:

$$1 + \frac{Z}{4EIk_b^3} \left[\frac{\sin(k_b l)}{\cos(k_b l) - \cos(kl)} - \frac{\sinh(k_b l)}{\cosh(k_b) - \cos(kl)} \right] = 0 \tag{16}$$

$$\cos^2(kl) + \cos(kl) \left[-[\cosh k_b l + \cos k_b l] + \frac{Z}{4EIk_b^3} (-\sin k_b l + \sinh k_b l) \right] + \frac{Z}{4EIk_b^3} [\sin k_b l \cosh k_b l - \sinh k_b l \cos k_b l] + \cos k_b l \cosh k_b l = 0$$

This is a second degree polynomial for $\cos(kl)$. The solution produces the values of the wave number k which is between 0 and. It is a function of k_b which depends on the radial frequency (eq.4). It leads therefore to the dispersion equation.

 ISBN 13: 978-84-613-5510-5

Similar relations were obtained previously by other authors (for example Heckl for a Timoshenko beam).

The shape of the Bloch wave can be obtained by using a similar process. Let us consider the free motion related to a couple k , k_b complying to equation (16).

The displacement at position x between two supports is related to the displacements of all supports by using the Green's function G , leading to:

$$\eta(x) = -Z\eta_0 \left[\sum_{m=-\infty}^{+\infty} G(x, x_m)\eta(x_m) \right] \tag{17}$$

Introducing the Green's function and the propagating constant γ leads to:

$$\eta(x) = -Z\eta_0 \frac{1}{4k_b^4 EI} \left[\sum_{m=-\infty}^{+\infty} \left(ie^{ik_b|x-x_m|} - e^{-k_b|x-x_m|} \right) e^{-\gamma x_m} \right] \tag{18}$$

Computing the sum of the series leads to:

$$\eta(x) = C\left[\frac{N_1}{D_1} + \frac{N_2}{D_2}\right] \tag{19}$$

Where: $C = Z\eta_0 \frac{1}{4k_b^4 EI}$ and where the following notations are introduced:

$$N_1(x,k) = e^{ik} sin(k_b x) - sin\left(k_b(x-l)\right) \tag{20}$$

$$N_2(x,k) = -\left(e^{ikl} sinh(k_b x)\right) - sinh\left(k_b(x-l)\right) \tag{21}$$

$$D_1(k) = cos(k_b l) - cos(kl) \tag{22}$$

$$D_2(k) = cosh(k_b l) - cos(kl) \tag{23}$$

Due to the Floquet's theorem, the Bloch wave may be written:

$$\eta(x) = \eta_1(x) e^{ikx} \tag{24}$$

Where η_1 is a periodic function of x.

The value of η_0 and therefore the value of C must be chosen to build from the Bloch waves an orthonormal basis of $L^2(Y)$ where Y is the periodic cell obtained after transforming the space variable x into the non dimensional variable $X^* = x\frac{2\pi}{l}$. Through this process, the period $[0, l]$ is transformed into $[0, 2\pi]$

 ISBN 13: 978-84-613-5510-5

The constant C is chosen to insure that the L^2 norm of $\eta(X^*)$ $\left(or\ \eta_1(X^*)\right)$ on $[0,2\pi]$ is equal to 1, leading to:

$$C^{-1} = \left[\frac{2\pi}{l}\int_0^l \left|\frac{N_1}{D_1} + \frac{N_2}{D_2}\right|^2 dx\right]^{1/2} \tag{25}$$

4. The response of the structure to moving load

As for the space coordinate, it is convenient, to define the Bloch transform, to use a no-dimensional wave-number K^* define by:

$$K^* = \frac{kl}{2\pi} \tag{26}$$

The Bloch transform of a function $F(x)$ is then defined by:

$$\bar{F}_n(K^*) = \int_0^{2\pi} F(X^*)\overline{\eta_n}(X^*,K^*)dX^* \tag{27}$$

Where $\overline{\eta_n}(X^*,K^*)$ is the complex conjugate of the Bloch wave contained in the n^{th} band obtained by equation (12).
The Bloch decomposition theorem whose proof and conditions of validity can be found in (Sanchez, Allaire) states that.

$$F(X^*) = \sum_{n=1}^{\infty}\int_0^1 \bar{F}_n(K^*)\eta_n(X^*,K^*)dK^* \tag{28}$$

The Bloch transform of equation (1) , considering that the Bloch wave $\eta_n(x,k)$ is the solution of the homogeneous equation for an harmonic equation at radial frequency $\omega_n(k)$, lead to:

$$m\omega_n^2\overline{U_n}(K^*,t) + m\frac{\partial^2}{\partial t^2}\overline{U_n}(K^*,t) = \bar{F}_n(K^*,t) \tag{29}$$

This equation shows that each Bloch component is solution of the dynamic equation for a '1 DOF' system.
Let us use this result to obtain the response of the beam to a load having a constant intensity F_0 which is moving at the velocity V. It means that F is given by:

$$F(x,t) = \delta(x - Vt)F_0 \tag{30}$$

Where δ is the Dirac distribution.
The Bloch transform of $F(t)$ is:

 ISBN 13: 978-84-613-5510-5

$$F_n(K^*,t) = \overline{\eta_n}\left(X^* = Vt\frac{2\pi}{l}, K^*\right) F_0 \frac{2\pi}{l} \tag{31}$$

The Bloch components of the displacement induced by the moving load are therefore solution of:

$$m\omega_n^2 \overline{U_n}(K^*,t) + m\frac{\partial^2}{\partial t^2}\overline{U_n}(K^*,t) = \overline{\eta_n}\left(X^* = Vt\frac{2\pi}{l}, K^*\right) F_0 \frac{2\pi}{l} \tag{32}$$

Its solution is given by:

$$\overline{U_n}(K^*,t) = \frac{2\pi F_0 C^2}{l}\left[\frac{e^{-2i\pi K^*}\sin(k_b Vt) - \sin\left(k_b(Vt-l)\right)}{\gamma_1[\cos(k_b l)-\cos(2\pi K^*)]} - \frac{e^{-2i\pi K^*}\sinh(k_b Vt) - \sinh\left(k_b(Vt-l)\right)}{\gamma_2[\cosh(k_b l)-\cos(2\pi K^*)]}\right] \tag{33}$$

Where γ_1 and γ_2 are given by:

$$\gamma_1 = m(\omega_n^2 - k_b^2 V^2) \tag{34}$$

$$\gamma_2 = m(\omega_n^2 + k_b^2 V^2) \tag{35}$$

The displacement induced by the moving force is finally obtained by composition of all the Bloch components, leading to:

$$U(X^*,t) = \sum_n \int_{K=0}^{1} \overline{U_n}(K^*,t)\eta_n(X^*,K^*)dK^* \tag{36}$$

Coming back to x , k and taking into account that k is involved in η only by $\cos(kl)$, lead to:

$$U(x,t) = \sum_n \int_{k=0}^{\pi/l} V_n(k,x,t)\,dk \tag{37}$$

The function $V_n(k,x,t) = \frac{1}{\pi}\overline{U_n}(k,t)\eta_n(x,k)$ is given by :

$$V_n(k,x,t) = 2F_0 C^2\left[\frac{\overline{N_1}(Vt,k)}{\gamma_1 D_1} + \frac{\overline{N_2}(Vt,k)}{\gamma_2 D_2}\right]\left[\frac{N_1(x,k)}{D_1} + \frac{N_2(x,k)}{D_2}\right] \tag{38}$$

This solution can be transformed by using new dimensionless variables T, X, B, K defined by $x = Xl$, $t = Tl/V$, $k = K/l$, $k_b = B/l$.

 ISBN 13: 978-84-613-5510-5

Using these variables U is given by:

$$U(X,T) = \frac{F_0}{\pi\beta l}\sum_n \int_{K=0}^{\pi} \frac{P}{\alpha_1 \alpha_2 Q} dK \tag{39}$$

Where:

$$\alpha_1 = B^4 - B^2\phi \tag{40}$$

$$\alpha_2 = B^4 + B^2\phi \tag{41}$$

$$\phi = \frac{mV^2l^2}{EI} \tag{42}$$

$$\beta = \frac{EI}{l^4} \tag{43}$$

$$P = \alpha_2 D_2^2 P_{11} + \alpha_1 D_1^2 P_{22} + D_1 D_2(\alpha_1 P_{21} + \alpha_2 P_{12}) \tag{44}$$

$$Q = \int_0^1 \left(D_2^2|N_1|^2 + D_1^2|N_2|^2 + D_1 D_2(\overline{N_1}N_2 + \overline{N_2}N_1)\right) dZ \tag{45}$$

All integrals in Q are obtained explicitly:

$$\int_0^1 |N_1|^2\, dZ = \frac{1}{B}\left[cos(K)\left(sin(B) - Bcos(B) + B - sin(B)cos(B)\right)\right] \tag{46}$$

$$\int_0^1 |N_2|^2 dZ = \frac{1}{B}\left[cos(K)\left(Bcosh(B) - sinh(B)\right) + sinh(B)cosh(B) - B\right] \tag{47}$$

$$\int_0^1 (\overline{N_1}N_2 + \overline{N_2}N_1)\, dZ = \frac{2}{B}\left[cos(K)\left(sin(B) - sinh(B)\right) + cos(B)sinh(B) - sin(B)cosh(B)\right] \tag{48}$$

The terms appearing in P are given by:

$$P_{11} = -cos(K)\left[sin(BT)sin\left(B(Z-1)\right) + sin(BZ)sin\left(B(T-1)\right)\right] + sin\left(B(T-1)\right)sin\left(B(Z-1)\right) + sin(BT)sin(BZ) \tag{49}$$

$$P_{22} = -cos(K)\left[sinh(BT)sinh\left(B(Z-1)\right) + sinh(BZ)sinh\left(B(T-1)\right)\right] + sinh\left(B(T-1)\right)sinh\left(B(Z-1)\right) + sinh(BT)sinh(BZ) \tag{50}$$

$$P_{21} = cos(K)\left[sinh(BT)sin\left(B(Z-1)\right) + sin(BZ)sinh\left(B(T-1)\right)\right] - sinh\left(B(T-1)\right)sin\left(B(Z-1)\right) - sinh(BT)sin(BZ) \tag{51}$$

$$P_{12} = cos(K)\left[sin(BT)sinh\left(B(Z-1)\right) + sinh(BZ)sin\left(B(T-1)\right)\right] - sin\left(B(T-1)\right)sinh\left(B(Z-1)\right) - sin(BT)sinh(BZ)$$

 ISBN 13: 978-84-613-5510-5

(52)

It is convenient to use the dispersion equation with non-dimensional variables

$$1+\frac{\delta}{B^3}\left[\frac{\sin(B)}{\cos(B)-\cos(K)}-\frac{\sin h(B)}{\cosh(B)-\cos(K)}\right]=0 \tag{53}$$

Where the non dimensional parameter δ is given by:

$$\delta=\frac{l^3 Z}{4EI} \tag{54}$$

From the previous results, it can be seen that the displacement depends of the non dimensional position and time and also of the three following parameters:

The parameter δ which is the ratio between the stiffness of the beam and the stiffness of the support and which is involved in the dispersion curve.

The parameter ϕ which is the ratio between the elastic energy and a kinetic energy computed from the moving load velocity.

The parameter $\frac{1}{\beta l}$ related to the flexure stiffness of the beam.

5. Example of application

Let us consider a rail whose properties are given in table 1.

Young's modulus	Section's inertia	Mass per un. length	Support stiffness	Spacing
200	8.10-5	158	30 and 0.02	0.6
GPa	m^4	Kg/m	GN/m	m

Table 1. Physical properties of the beam and support

The dispersion curves are show on figure 1 for the first three passing bands for the stiffer support (Z=30 GN/m). Similar results are obtained for the softer support at lower frequencies.

 ISBN 13: 978-84-613-5510-5

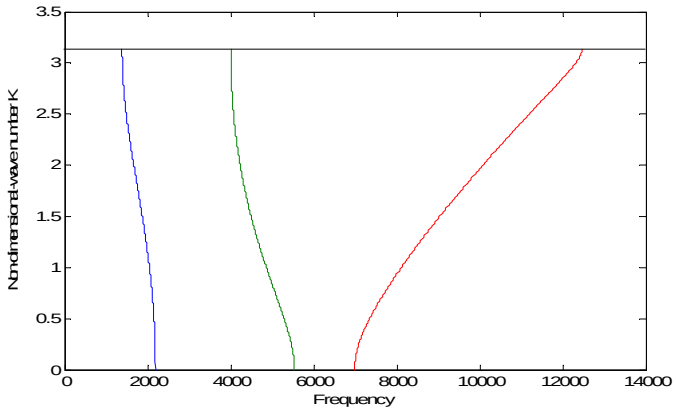


Figure 1 Dispersion curve for the first band tree bands: case for stiffer support

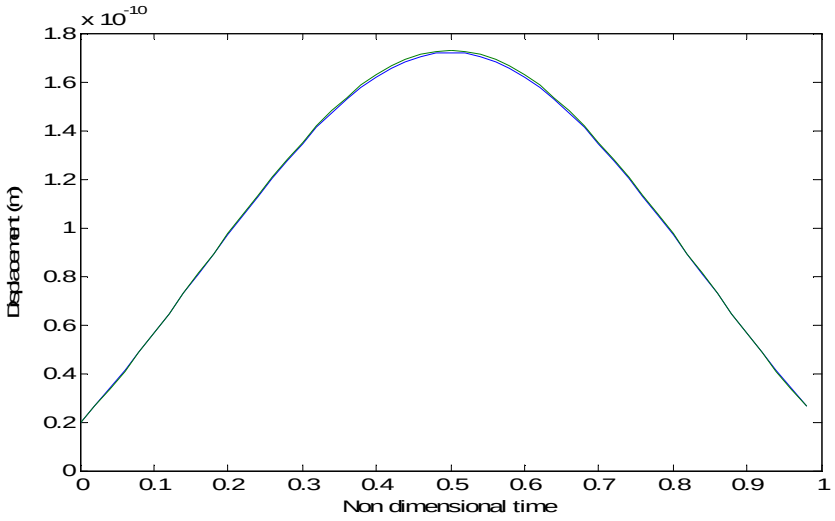


Figure 2 Displacement vs non-dimensional time for the stiffer case (high speed and low speed)

The displacement induced by a unit moving load at the center of a given period is next computed when the load is moving over that period with two speeds: a low speed 10*m/s* and a high speed 140*m/s*. Results are given in the case of stiff support in figure 2 and in the case of soft support in figure 3. It appears that the displacement is the same for low speed and high speed, in the case of the stiffer support, while the displacement depends significantly of the speed for the softer support.

 ISBN 13: 978-84-613-5510-5

This result is due to the fact that the passing bands are at higher frequencies for the stiffer supports. The velocity-dependent term φ in (35) is negligible when compared to B^2 for any velocity.

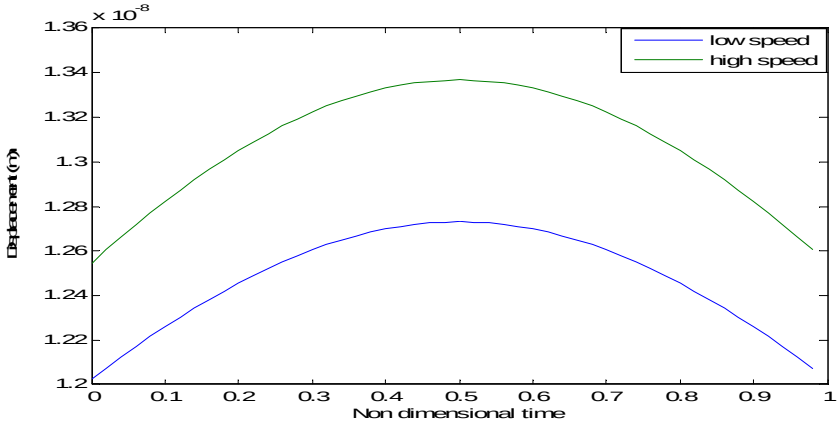


Figure 3. Displacement vs non-dimensional time for the softer case

5. Conclusion

The method of Bloch transform was used to study the dynamics of a beam resting on a periodic support. It allows to compute the response of the beam to any dynamic loading by computing the Bloch transform of the displacement induced by the loading. It involves only to solve a (continous) set of decoupled"1 DOF" dynamic equations. The method is applied to the dynamic response of a thin beam. The authors intend to extent the method to the dynamics of a Timoshenko beam for any loading, implying the coupling between different components of the motion of the beam (torsion, flexion,...).

[1] D.J. Mead, *Wave propagation in continuous periodic structures: research contributions from Southampton*, J.S.Vib. 190 (3) (1996) 495-524.
[2]M. A. Heckl, rail *Noise-can random sleeper spacing's help*, Acoustica. 81 (1995) 559-564.
[3] P.M. Belotserkovskiy, *On the oscillations of infinite periodic beams subject to a moving concentrated force*, J.S.Vib. 193 (3) (1996) 705-712.
[4] A. Nordborg, *vertical rail vibration: point force excitation*, Acoustica, 84 (1998) 280-288.
[5] H.M. Saed, F. Vestroni, *Simulation of combined systems by periodic structures: the wave transfer matrix approach*, J.S.Vib, **213(1)**, (1998) 55-73.
[6] R.S. Langley, *on the modal density and energy flow characteristics of a periodic structure*, J.S.Vib. **172(4)**, (1994) 491-511.
[7] M.L. Munjal, M. Heckl, *Vibrations of a periodic rail-sleeper system excited by an oscillating stationary transverse force*, J.S.Vib, 81(4), (1982 491-500.

 ISBN 13: 978-84-613-5510-5

[8] L. Gry, C. Gontier, *Dynamic modeling of railway track : a periodic model based on a*
generalized beam formulation, J.S.Vib. (1997) **199(4)**, 532-538.
[9] R.S. Langley, The response of two-dimensional periodic structures to point harmonic loading, J.S.Vib. **197 (4)**, (1996) 447-469.
[10] A. Nordborg, *Vertical rail vibrations: point force excitation, Acoustica*, **84**, (1998) 280-288.
[11] V.H. Nguyen, D. Duhamel, *Finite element procedures for nonlinear structures in moving*
coordinates, Part 1: Infinite bar under moving axial loads. Computers and Structures, **84**,
(2006) 1368-1380.
[12] D. Duhamel, B.R. Mace, M.J. Brennan, *Finite element analysis of the vibration of waveguides and periodic structures*, J.S.Vib. **294**, (2006) 205-220.
[13] M.H. Heckl, *Coupled waves on a periodically supported Timoshenko beam*, J.S.Vib.
252(5), (2002) 849-882.
[14] J.F. Hamet, Railway noise: *use of the Timoshenko model in rail vibration studies, Acoustica*, **85**, (1999) 1-12.
[15] G. Allaire, *The Bloch transform and applications, ESAIM Proceedings*, **3**, (1998)65-84.
[16] J. Sanchez Hubert, E. Sanchez Palencia, *Vibration and coupling of continuous systems*,
Springer-Verlag, Berlin, (1989).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Electromagnetic Wave Effect On Semiconductor Device : FDTD Method

S.Latreche[1], S.Labiod[1] and C.Gontrand[2]

[1] Laboratoire Hyperfréquences et Semiconducteurs (LHS), *Dpt. Electronique, Fac. Sciences Ingénieur, Université Mentouri, Constantine. Algerie.*

[2] Institut de Nanotechnologie de Lyon (INL). INSA-Lyon, *Université. Lyon. France*

emails: latreche.saida@gmail.com, samir.labiod@gmail.com, Christian.Gontrand@insa-lyon.fr

Abstract

The electromagnetic wave effects on the behavior of semiconductor devices are investigated by coupling a TM wave (transverse magnetic wave), solution of Maxwell's equations to an active device model. In this paper, 2D (Two-dimensional) simulations verify the expect device-wave interaction. The active semiconductor model is based on the drift –diffusion equations. The coupling between the two models is established by using fields obtained from the solution of Maxwell's equations in a semiconductor device model. Its permit to calculate the current densities witch used to update both the electric and magnetic fields.

The study of an electromagnetic field influence on a semiconductor structure is carried out by MATLAB software. We have developed an efficient and accurate tool to solve the Maxwell's equations by using FDTD method, and finite differences and Euler's method to solve the drift-diffusion equations.

Numerical results relative to the study of two-dimensional device are included to valid the effectiveness of the method.

We observe that the semiconductor device significantly attenuate the input wave as it propagates along the structure.

This is, essentially due to the electromagnetic energy loss by the conducting electrons.

 ISBN 13: 978-84-613-5510-5

Key words: FDTD method, PDE, electromagnetic wave, drift-diffusion model, Gummel's method, Newton Raphson..

1. Introduction

The evolution of electronic market fosters a continuous quest for small-size and low-cost systems.
Propagation and radiative effects become more and more important, most notably, signal reflections due to interconnecting line discontinuities, dispersion, and crosstalk phenomena [1]. So the assessment of potential hazards caused by electromagnetic influence to electronic systems as well as transmission is an essential engineering task. This is essentially due to nearby transmitters and electrically active device.

At high frequencies and reduced circuit sizes, however, such an approach can hardly be pursued, due to the difficulty of properly characterizing (both qualitatively and quantitatively) the coupling phenomena typical of a densely packed circuit.
In these cases, the full-wave solution of Maxwell's equations may provide a more comprehensive investigation tool. To this purpose, different solution techniques can be followed.
In this work, we describe the modeling and the simulation of semiconductor devices were previously achieved by solving various combinations of Poisson and continuity equations conventionally.
The Gummel's method is most commonly used to solve the nonlinear steady-state problem arising from the semiconductor device equations [2], which can be written as sparse nonlinear system of equations. In this approach, the two unknown variables in drift-diffusion model are coupled together through the whole process of computation [3], each equation is solved using Newton's method. It's more robust, and converges in relatively little iteration.
However, interesting physical phenomena arise from the manner in which charge fluctuations and current responses are coupled to the electromagnetic field, a 2D FDTD model has been developed in order to solve the coupled Maxwell's and drift-diffusion equations. This model permits to establish a rigorous simulation of the active device.

2. Semiconductor model

The semiconductor model consists of the Poisson equation and continuity one. These equations represent a coupled nonlinear system of partial differential-integral equations (PDE equations).
In our case, we treat the steady-state condition and further neglect the contribution from holes because the we have an N type substrate where the electrons are predominant versus holes[4].
Then the basic equations become the following

 ISBN 13: 978-84-613-5510-5

1) Poisson's equation

$$\nabla^2 V = \frac{q}{\varepsilon_{SI}}(n - N_D) \tag{1}$$

2) Continuity equation of electron current :
Current equations for electrons and holes must be solved simultaneously.
In the present case, however, a simplified approach is adopted because the type of the substrate is N (n>> p), so we consider only the electrons equation continuity.

$$\frac{\partial n}{\partial t} - \frac{1}{q} Div\left(\vec{j}_n\right) = 0 \tag{2}$$

3) Electron current equation

$$\vec{j}_n = -qn\mu_n \vec{\nabla} V + qD_n \vec{\nabla} n \tag{3}$$

The drift-diffusion approach is known to possibly lack of accuracy if large field changes, either in time and space are encountered.

3. Electromagnetic model

The electromagnetic wave propagation can be completely characterized by solving Maxwell's equations are first-order linear coupled differential equations. These equations coupled differential equations relating the field vectors and current densities at any point in our structure at any time [5].
Maxwell's equations are given by:

$$\vec{\nabla} \wedge \vec{H} = \vec{J} + \varepsilon_{SI} \frac{\partial \vec{E}}{\partial t} \tag{4}$$

$$\vec{\nabla} \wedge \vec{E} = -\mu_{SI} \frac{\partial \vec{H}}{\partial t} \tag{5}$$

4. Electromagnetic wave propagation

The excitation is applied as a TM mode (Transverse Magnetic mode). For the electromagnetic-wave analysis, a sinusoidal excitation is applied in the electrode A of the silicon structure. The electric wave excitation becomes:

$$E = E_0 \sin(\omega t)$$

E_0 represent the maximum amplitude and $\omega = 2\pi f$ the wave pulsation, where $f = 1000\,GH_Z$.
The length wave corresponding is $\lambda = 75\,\mu m$

 ISBN 13: 978-84-613-5510-5

5. Simulated Structure

The TM wave model is then solved for a few picoseconds, to avoid the effects of the electromagnetic wave interference.

The 2D structure used in the simulation is shown in Fig.1. It's constituted by a silicon semiconductor doped $N_D = 10^{18}\,At/cm^3$. The distance between the two electrodes A and B is $50\,\mu m$. The width of the plate is $40\,\mu m$. A constant space step of discretization $\Delta x = \Delta y = h = 1\mu m$ is considered.

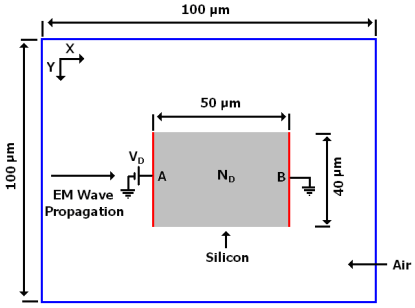


Fig. 1. A view of the simulated device

6. Simulation technique

1) Drift Diffusion model

The discretization uses a first and second order of equation (1) in 2D-finite difference mesh, and the discretization of equation (2) uses Euler's implicit method leads to have equations :

$$H.V^t(i,j+1)+B.V^t(i,j-1)+G.V^t(i-1,j)+D.V^t(i+1,j)-(H+B+G+D).V^t(i,j)=\frac{q}{\varepsilon_{SI}}(n^t(i,j)-N_D) \quad (6)$$

$$a.n^{t+1}(i,j+1)+b.n^{t+1}(i,j-1)+c.n^{t+1}(i-1,j)+d.n^{t+1}(i+1,j)-e.n^{t+1}(i,j)=\frac{h^2}{\mu_n.\Delta t}(n^{t+1}(i,j)-n^t(i,j)) \quad (7)$$

Where $H = B = D = G = \frac{1}{h^2}$

a, b, c, d and e are coefficients depending on the electrostatic potential:

$$a = U_T - \frac{1}{4}(V(i+1,j)-V(i-1,j)$$

 ISBN 13: 978-84-613-5510-5

$$b = U_T + \frac{1}{4}(V(i+1, j) - V(i-1, j)$$

$$c = U_T - \frac{1}{4}(V(i, j+1) - V(i, j-1)$$

$$d = U_T - \frac{1}{4}(V(i, j+1) + V(i, j-1)$$

$$e = 4.U_T + V(i+1, j) + V(i-1, j) + V(i, j+1) - V(i, j-1) - 4.V(i, j)$$

In the case of Ohmic contacts, the electron and hole densities are determined by assuming charge neutrality and thermal equilibrium
At free surfaces, the normal derivatives of current density and electrostatic potential are set to zero.

$V = V_{Contact} + U_T . \ln\left(\frac{n}{n_i}\right)$, where ni is an intrinsic concentration.

Solution of equations (6) and (7) provide the space and time distribution of the unknown functions n and V at each point and each time step in the considered structure which leads to have a two matrix systems which depend in time.

$$A.V_V^t = V_n^t \tag{8}$$

$$B.V_n^{t+1} = V_n^t \tag{9}$$

A and B represent respectively the corresponding matrix coefficient of equations (6) and (7)
V_V and V_n are respectively the electrostatic potential and electron density vectors.
At each time step Gummel's method is most commonly used to solve the coupled system arising from equations (8) and (9). Where these equations can be solved in a decoupled manner using Newton's method [6].
The whole numerical procedure to solve the two system equations can be summarized as:

1. For t from 0 to final time
2. Select time-step which ensures stability convergence.
3. Solve (8) and (9) using Gummel's iterations
4. Set $n^t \leftarrow n^{t+1}$
5. End time iterations

1) Electromagnetic model

The electromagnetic model simulates the evolution of both electric and magnetic fields due to moving free charges [7]

$$\frac{\partial \vec{E}}{\partial t} = \frac{1}{\varepsilon_{SI}}\left(\vec{\nabla} \wedge \vec{H}^{ac} + \vec{J}^{dc} - \vec{J}^{tot}\right) \tag{10}$$

 ISBN 13: 978-84-613-5510-5

$$\frac{\partial \vec{H}}{\partial t} = -\frac{1}{\mu_{SI}} \vec{\nabla} \wedge \vec{E} \tag{11}$$

Where $\vec{J}^{dc}$ is obtained by solving the drift-diffusion model in conjunction with Poisson's equation [8].

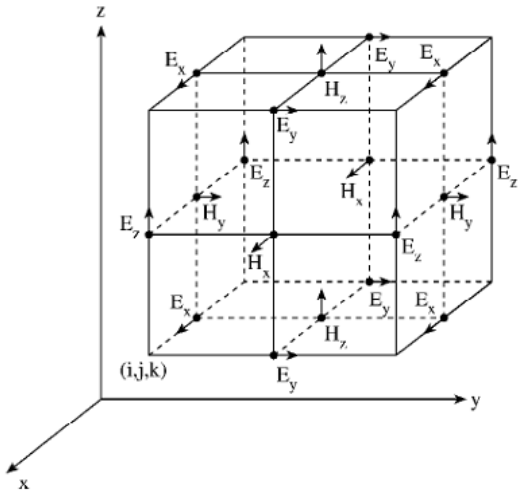


Fig. 2. Positions of the field components in a unit cell of Yee's lattice

Equations (10) and (11) can be replaced with an explicit finite difference approximation in its known values at the previous nth time step. Using the first-order upwind scheme for time and spatial derivatives yields the following equation:

$$E_z^{t+1}(i,j) = E_z^t(i,j) + \frac{\Delta t}{\varepsilon_{SI}}\left(\frac{1}{\Delta y}\left(H_y^t(i,j) - H_y^t(i-1,j)\right) - \frac{1}{\Delta x}\left(H_x^t(i,j) - H_x^t(i,j-1)\right) + J^{dc} - J^{tot}\right) \tag{12}$$

$$H_x^{t+1}(i,j) = H_x^t(i,j) - \frac{\Delta t}{\mu_{SI}} . \frac{1}{\Delta y}\left(E_z^t(i,j) - E_z^t(i,j-1)\right) \tag{13}$$

$$H_y^{t+1}(i,j) = H_y^t(i,j) + \frac{\Delta t}{\mu_{SI}} . \frac{1}{\Delta y}\left(E_z^t(i,j) - E_z^t(i-1,j)\right) \tag{14}$$

Notice from equations (10), (11) and figure (2) that the components E and H are interlaced within the unit cell and evaluated at alternate half-time steps.
Electric and magnetic field components are extracted from two separate sets of equations [9]. At each time step, electromagnetic model and semiconductor equations should be solved sequentially [10], where the equations (12), (13) and (14) gives the electric and magnetic field distributions at each time step, these

 ISBN 13: 978-84-613-5510-5

latter are used by semiconductor model to update the current density at the same time step, which fed back to electromagnetic model again for the following time step [11]. Figure 3 shows flowchart of the sequence FDTD scheme

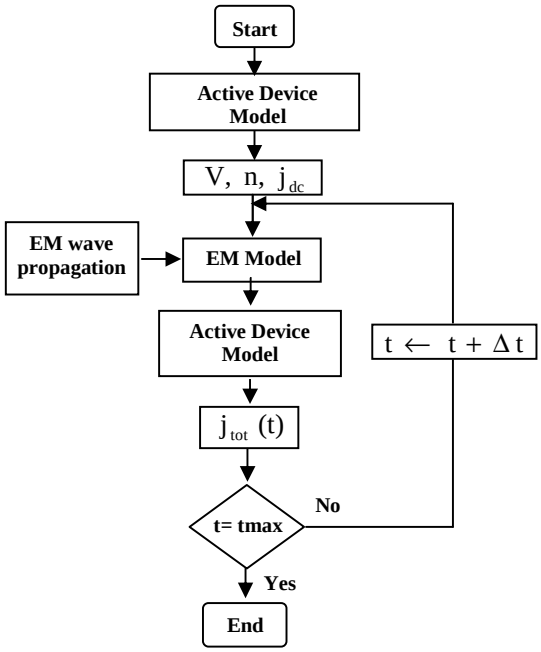


Fig. 3. Flowchart describing the calculate of the total current as a function of electromagnetic wave

In practice, at each time step the iterative process is stopped at the minimum value of iterations such that $\left| \max\left(\frac{\delta n}{n}\right)\right| \langle \varepsilon$, where ε is a fixed tolerance, in our case, we have fixed ε at 10^{-5}. However, since the exact solution is obviously not available, it is necessary to introduce suitable stopping criteria to monitor the convergence of the iteration [12].

7. Results and discussions

For a polarization $V_D = 1v$ the results given by the drift-diffusion model using a finite differences method are in agreement with theoretical concepts.

 ISBN 13: 978-84-613-5510-5

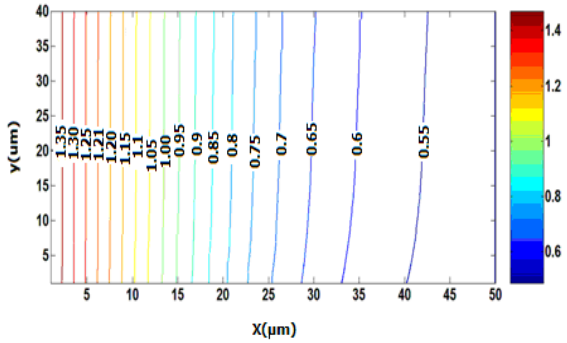


Fig. 4. Contour plot for the equipotential lines

Figure (4) represent the equipotential lines in steps 0.5.
The electrons density is almost constant along the semiconductor. Its value takes approximately the doping value.

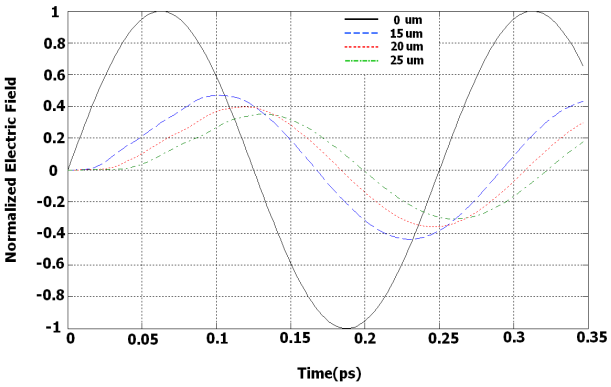


Fig.5. Time variation of the electric field at several points in the x-direction

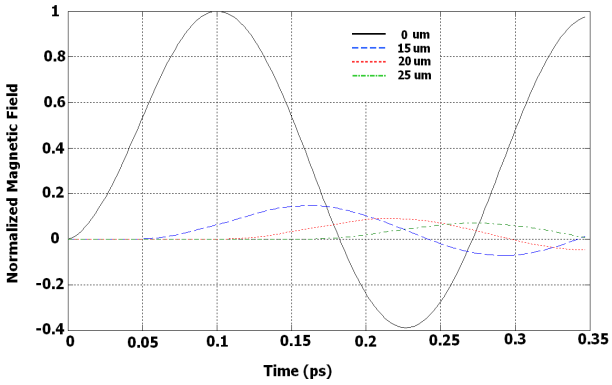


Fig.6. Time variation of the magnetic field at several points in the x-direction

 ISBN 13: 978-84-613-5510-5

Figures (5) and (6) depicts the impute wave evolution at different points along the x-direction.
The input wave decreases in magnitude as it propagates along the x-direction (along the device). This is mainly due to the electromagnetic energy loss to the conducting electrons.
The active device role in attenuating the input wave can be confirmed by the comparison of two waves, one of them propagates on the active device and the other in a semi-insulating one.

8. Conclusion

The effect of electromagnetic wave on electrical characteristics in semiconductor devices is studied, for that we have combined the electromagnetic model with drift-diffusion equations. To this purpose, a FDTD method has been used.
An explicit scheme assumption is formulated, which allows for the fully decoupled of the two models, electromagnetic equations and drift-diffusion ones.
A “leapfrog” scheme is adopted for the FDTD solution of the Maxwell’s equations and “Gummel” scheme for the finite difference solution of the semiconductor equations [8].
It has been shown that a model consisting of a TM wave time domain solution of Maxwell’s equations coupled to the semiconductor model capable of evaluating the effect of the propagating wave on semiconductor behaviour.
The Gummel iterative process is stopped at the exact solution for each time step, where the convergence criterion is fixed for the minimum of $\frac{\delta n}{n}$.
We also concluded that the numerical method used help us to explain some physical phenomena like:
- The increasing in the number of electrons in semiconductor cause the increasing of the conductivity, which leads to an attenuation of the electromagnetic wave in the space.
- The electromagnetic wave leads to the change of the electrons velocity in the space, which cause a discontinuity in the current density.

 ISBN 13: 978-84-613-5510-5

9. Nomenclature

Symbol	*QUANTITY*	*Units*
V	Electrostatic potential	v
n	Electron density	At/Cm^{-3}
N_D	Donor doping density	$At.Cm^{-3}$
q	Electron charge	C
U_T	Thermal voltage	26 mv
μ_n	Electron mobility	$m^2/v.s$
D_n	Electron diffusion coefficient	m^2/s
t	Time	s
Δt	Time increment	s
h	Space increment	m
J_n	Electrons current density	A/m^2
J^{dc}	Direct current density	A/m^2
J^{tot}	total current density	A/m^2
E	Electric field	V/m
H	Magnetic field	A/m
ε_{SI}	Silicon permittivity	F/m
μ_{SI}	Silicon permeability	H/m
H^{DC}	Direct component of magnetic field	A/m
H^{AC}	Alternative component of magnetic field	A/m
ω	frequency	$Rd.s^{-1}$

TABLE 1: UNITS FOR DIFFERENT PARAMETERS

10. References

[1] P. CIAMPOLINI, L, ROSELLI, G, STOPPONI, *Mixed-mode circuit simulation with full-wave analisis of interconnections,* Ieee transaction on electron devices, vol 44, N°11, pp. 2098-2105, Nov 1997.

[2] Patil, M. Ravaioli, U. and Kerhoven, T, *Numerical evaluation of iterative schemes for Drift-Diffution simulation,* Gordon. Breach Science Publisher imprint, India, vol. 8, pp. 337-341, 1998.

[3] W. TSE TAN, *An inexact method for Drift-Diffusion model in Semiconductor Device Simulations,* Dissertation, 2009.

[4] K. HORIO, H. YANAI, T. IKOMA, *Numerical simulation of GaAs MESFET's on the semi-insulating substrate compensated by deep traps,* Ieee transaction on electron devices, vol 35, N°11, pp. 1778-1785, Nov 1988.

[5] M. ALSUNAIDI, S. IMTIAZ, S. EL-GHAZALY. *Electromagnetic wave time-domain model,* IEEE transaction on microwave theory and techniques, vol 44, N°6, pp. 799-808, Jun 1996.

[6] H. HEYDEMAN, *Solution numérique bidimensionnelle des equations générales de transport en régime permanent,* Onde electrique, 1972.

[7] N. MATTHEW, O. SADIKU, *Numerical Techniques in Electromagnetics*, 2nd edition, CRC Press LLC, 2001.

 ISBN 13: 978-84-613-5510-5

[8] A. YASSER, E. JAMES, *Efficient Modeling of PIN Diode Switches Employing Time-Domain Electromagnetic-Physics-Based Simulators*, SLAC-PUB-11281, Jun 2005.

[9] P. CIAMPOLINI, P. MEZZANOTTE, L. ROSELLI, R. SORRENTINO, *Accurate and efficientvcircuitvsimulation with lumped-element FDTD technique*, Ieee transaction on electron devices, vol 44, N°12, pp. 2207-2215, Dec 1996.

[10] R.MIRZAVANDY, A.ABDIPOUR, G.MORADI, *full wave semiconductor devices simulation using ADI-FDTD method*, Progress In Electromagnetics Research M ,Vol.11, pp. 191-202, 2010.

[11] R. GRONDIN, S. EL-GHAZALY, S. GOODNICK, *A review of global modeling of charge transport in semiconductor and full-wave electromagnetics*, IEEE transaction on microwave theory and techniques, vol 47, N°6, pp. 817-827, Jun 1999.

[12] A.QUARTERONI, R.SACCO, AND SALARI, F, *Numerical Mathematics*, 1st ed, vol. 2. Springer-Verlag New York, 2000.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Complexes of Free Helical Gold Nanoclusters and Carbon Monoxide: A Density Functional Study

Xiao-Jing Liu[1] and I.P. Hamilton[1]

[1] *Department of Chemistry, Wilfrid Laurier University, Waterloo, Ontario, Canada, N2L 3C5*

emails: xxliu@wlu.ca, ihamilton@wlu.ca

Abstract

Gold nanoclusters have attracted a surge of interest over the past decade due to their potentially significant applications in heterogeneous catalysis. In particular, when supported on an oxide surface, it has been shown that gold nanoclusters can catalyze the oxidation of carbon monoxide. Free gold nanoclusters often exhibit a variety of stable structural isomers ranging from compact to hollow to helical. The surface gold atoms of these structural isomers have very distinct electronic properties which could result in very different catalytic properties. In this presentation we examine complexes of free helical gold nanoclusters and carbon monoxide.

Key words: Gold nanoclusters, carbon monoxide

1. Introduction

Metal nanoclusters represent a novel organization of matter which is yet to be fully understood. In particular, gold nanoclusters have been shown to exhibit size-related properties that differ significantly from those observed for small clusters or the bulk material [1,2]. It has been established that there exists a close relationship between the properties of nanoclusters and their geometries but it is difficult to elucidate this connection by experimental techniques alone and, in this regard, quantum calculations can be very helpful. Density functional theory (DFT) has become an increasingly important tool in quantum calculations since the effects of electron correlation, which are large for metal nanoclusters, can be included at a moderate computational cost [3]. Relativistic effects, which are

 ISBN 13: 978-84-613-5510-5

large for gold nanoclusters, can be efficiently included via an effective core (or pseudo) potential [4].

Gold nanoclusters have potentially significant applications in heterogeneous catalysis. This area was initiated by Haruta in 1997 when he showed that gold nanoclusters (when supported on an oxide surface) could catalyze the oxidation of carbon monoxide (CO) [5]. The oxide surface was employed to prevent coalescence of the gold nanoclusters and it was initially believed to be inert. However, it subsequently became clear that the oxide surface plays an important role in the catalysis process. Nonetheless, it is of considerable interest to study the complex of a free gold nanocluster and CO and, in particular, to examine the effect of the free gold nanocluster on CO. In this presentation, we focus on helical gold nanoclusters since the surface gold atoms of these structural isomers exhibit very different electronic properties depending on their location. We have shown that Au_{24}, Au_{32} and Au_{40} all have structural isomers which are helical [6]. Although the lowest energy structural isomer is the compact structure, the helical structure is both stable and robust. In this presentation we focus on Au_{40} and examine the strength of the bonding between the surface gold atom and CO, charge transfer to CO, and changes to the CO bond distance and frequency as a result of complex formation.

2. Method

In our DFT calculations, four generalized gradient approximation (GGA) exchange-correlation functionals are employed: the Becke exchange functional with the Perdew correlation functional (BP86), the Becke hybrid three-parameter exchange functional with the Perdew correlation functional (B3P86), the Becke exchange functional with the Perdew-Wang correlation functional (BPW91), and the Becke hybrid three parameter exchange functional with the Perdew-Wang correlation functional (B3PW91). We used the LANL2DZ effective core potential, although other effective core potentials have been successfully used for gold nanoclusters. We believe that these functionals, and BP86 in particular, are appropriately accurate for the neutral gold nanoclusters considered in this presentation. The harmonic-frequency calculations were performed based on the optimized geometries of the nanoclusters. For simplicity, we employ the commonly used Mulliken charge analysis to examine the electronic properties of the nanoclusters. All calculations were performed with the Gaussian 09 program package [7] and, unless otherwise noted, all results are for the BP86 functional.

3. Results

For helical Au_{40} there are 5 gold core atoms, each of which is surrounded by 7 gold surface atoms which spiral in a helical fashion (see Fig. 1, top row). The

 ISBN 13: 978-84-613-5510-5

diameter is 0.553 nm and the length is 1.138 nm. The average Au-Au distance in each surface atom row is 2.84 Å. For bare CO, the C-O bond distance is 1.139 Å while the C-O vibrational frequency is 2120 cm^{-1}. These results compare favorably with the experimental values of 1.128 Å and 2143 cm^{-1} respectively.

For helical Au_{40} a Mulliken charge analysis (using the calculated wave function of the optimized geometry) shows that, although the total charge is zero, the terminal and central surface gold atoms are negatively charged while the intermediate surface gold atoms are positively charged. This indicates that Au_{40} has strong selective reactivity and that nucleophiles will prefer to attack at the terminal and central gold atoms, while electrophiles will prefer to attack at the intermediate gold atoms. In contrast, the surface gold atoms of compact Au_{40} are all negatively charged. For bare CO, the charge on the C atom is -0.041 and the charge on the O atom is 0.041.

We now consider the complex that is formed when CO bonds to a terminal gold atom of helical Au_{40} (which carries a negative charge). As shown in Fig. 1, bottom row, a bond is formed between a single surface gold atom and the carbon atom of CO. Note that the helical structure is largely preserved and Au_{40} does not collapse to the more stable compact structure. In the complex formation there is charge transfer from Au_{40} to CO. The charge on the C atom is now 0.125 and the charge on the O atom is now -0.329 and the total charge of CO is -0.204. Upon complex formation, the C-O bond distance increases to 1.146 Å while the C-O vibrational frequency decreases to 2065 cm^{-1}. This is indicative of a weakening of the C-O bond which means that CO will be more reactive and, in particular, more susceptible to oxidation.

We are currently considering the complex that is formed when CO bonds to an intermediate gold atom of helical Au_{40} (which carries a positive charge). We are also considering the complex that is formed when CO bonds to compact Au_{40}.

4. References

[1] P. SCHWERDTFEGER, *Gold Goes Nano - From Small Clusters to Low-Dimensional Assemblies,* Angew. Chem. Int. Ed. **42**, (2003) 1892-1895. Freeman, San Francisco, 1970.

[2] P. PYYKKÖ, *Theoretical Chemistry of Gold*, Angew. Chem. Int. Ed. **43**, (2004) 4412-4456.

[3] R.G. PARR AND W. YANG, *Density-Functional Theory of Atoms and Molecules*, Oxford University Press, New York, 1989.

[4] P. PYYKKÖ, *Relativistic Effects in Structural Chemistry*, Chem. Rev. **88**, (1988) 563-594.

[5] M. HARUTA, *Size- and Support-dependency in the Catalysis of Gold*, Catalysis Today **36**, (1997) 153-166.

 ISBN 13: 978-84-613-5510-5

[6] X. LIU, I.P. HAMILTON, R. KRAWCZYK AND P. SCHWERDTFEGER, *Free Helical Gold Nanowires: A Density Functional Study,* (Manuscript in Preparation).
[7] M.J. FRISCH ET AL., *Gaussian 09, Revision A.1*, Gaussian, Inc., Wallingford CT, 2009.

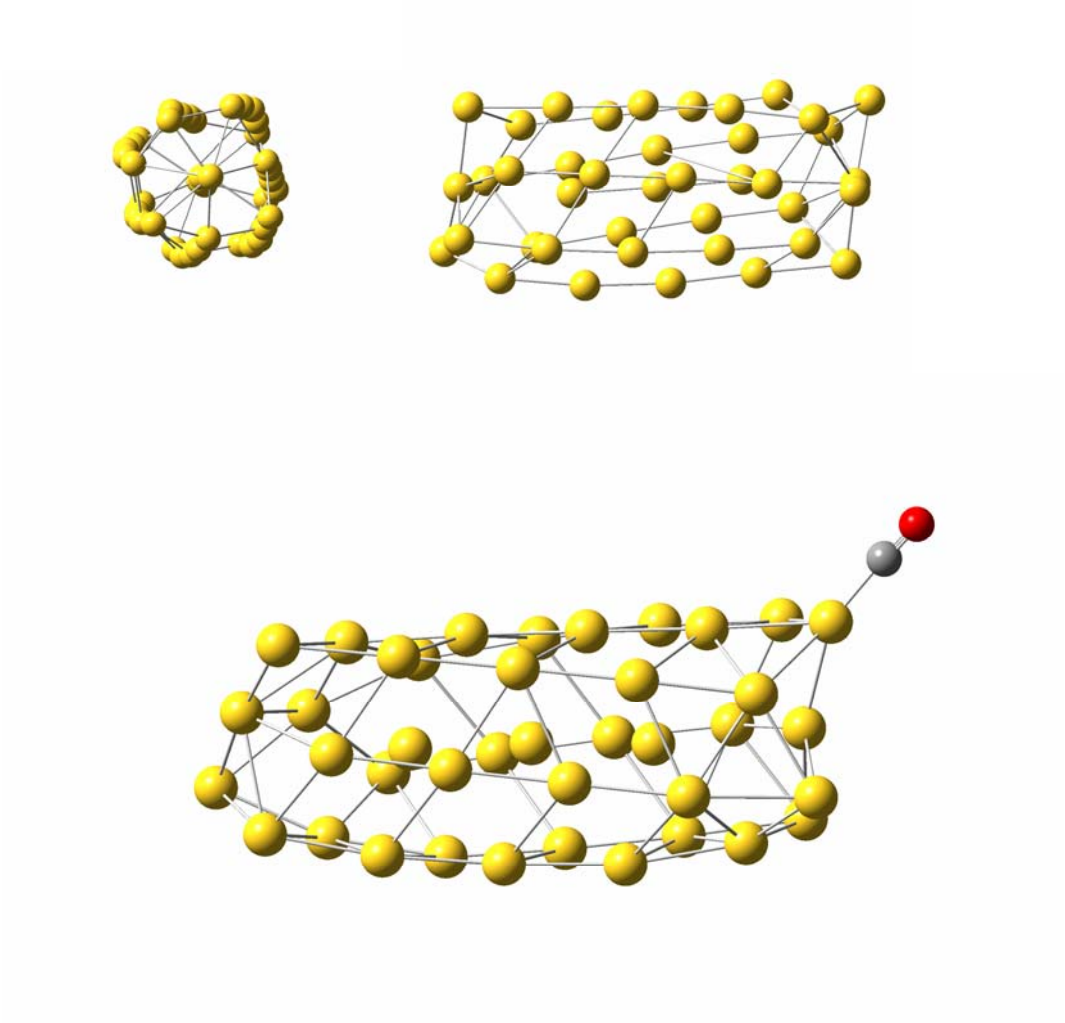


Figure 1. Top row: Two views of helical Au_{40}. Bottom row: Complex between helical Au_{40} and terminally bonded carbon monoxide.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Isotropic Image Analysis for Case Base Creation in a CBR Forecasting System.

Aitor Mata, Mª Dolores Muñoz, Emilio Corchado and Juan M. Corchado

[1] *Department of Computer Science and Automation, University of Salamanca, Spain*

emails: {aitor, mariado, escorchado, corchado}@usal.es

Abstract

This interdisciplinary study presents a novel Case-Based Reasoning (CBR) system that applies an isotropic buffer operator for case-based creation. Commonly used as an image analysis tool by commercial Geographic Information Systems (GIS), the operator in this particular system calculates the areas of different environmental phenomenon for simulation and visualization tasks. The systems presented in this paper use CBR methodology to generate predictions from data sets that were generated after having applied the isotropic buffer operator to the environmental images.

Keywords: Isotropic image analysis, Case-Based Reasoning, Forecasting.

1. Introduction

All customised Geographical Information Systems (GIS) include a *buffer* operator, also known as the *buffering* or influence zone. It is defined as the geometric space of the points that are at a shorter or similar distance to a given object (point, polyline or polygon) [5]. This definition is isotropic or directionally uniform, since the distance of the object to the edge of the buffer is constant in any direction on the plane. Among other fields, this operator is used in the simulated visualization of environmental processes, such as surveys of pesticide and chemical fertilizer contamination in the shallow waters of hydrographic basins; the influence of nitrates and silt levels on the growth of local flora, the

 ISBN 13: 978-84-613-5510-5

environmental impact of installing new industries in close proximity to urban centers, the determination of areas of high seismic risk.

Case-Based Reasoning (CBR) systems make use of past information in order to generate new solutions to new problems. The quality of the information stored within the case base will determine the quality of the solutions offered by these systems. Thus, the isotropic buffer operator is an important element in image analysis, as it provides the CBR system with accurate information that may be used in future situations.

The following section gives a brief explanation of CBR methodology. The third section develops the concept of isotropic image analysis, after which the fourth section describes the CBR system presented in this study, prior to the conclusions, which are advanced at the end of the paper.

2. Case-Based Reasoning

Case-Based Reasoning is a technique that has its origin in knowledge-based systems. CBR systems learn from previous situations [1]. The main element of a CBR system is the *case base*; a structure that stores problems, elements (*cases*), and its solutions. So, a case base can be visualized as a database that stores a collection of problems with some sort of relationship to the solutions to every new problem, which gives the system the ability to generalize in order to solve new problems.

The learning capabilities of CBR systems rely on their own structures, which consist of four main phases [2]: *retrieval*, *reuse*, *revision* and *retention*. The *retrieval* phase consists of finding the cases in the case base that most closely resemble the proposed problem. Once a series of cases have been extracted from the case base, they must then be *reused* by the system. In this second phase, the selected cases are adapted to fit the current problem. After offering a solution to the problem, it is then *revised* to check whether the proposed alternative is in fact a reliable solution to the problem. If the proposal is confirmed, it is *retained* by the system and could eventually serve as a solution to future problems.

CBR is a methodology [16] that has been applied to solve different kind of problems. It is a model that can easily be applied to solve soft-computing problems [15], since the methodology used by CBR is quite easy to assimilate using soft-computing approaches. Further applications are predictive models for the stock market [7], where inputs of different daily values allow a CBR model to assist with stock market investment decisions; construction models, for the generation of functional databases [18] to improve the somewhat chaotic

 ISBN 13: 978-84-613-5510-5

organization of construction projects, and also [6] to select different methods and materials, using expert system-oriented applications.

In most cases, CBR has not been used by itself, but is combined with various artificial intelligence techniques. Growing Cell Structures (GCS) has been used with CBR to automatically create the intern structure of the case base from existing data. It has been combined with multi-agent applications [4] to improve its results. ART-Kohonen neural networks [17], artificial neural networks and fuzzy logic [8] have also been used to complement the capabilities of the CBR methodology.

3. Isotropic image analysis

There are two methods for the generation of influence areas: Voronoi triangulation and the Minkowski Sum [12]. In the latter method, a *secondary polygon or generating polygon* is defined as being located on a point or moving on a polyline or polygon and generating a surface formed by the points that the generating polygon finds on its way. In the isotropic buffer, the generating polygon is a circle, which implies a constant distance between the border of the buffer and the object.

3.1. Von Misses Distribution

We can define the *circular variables* [3] as those that represent directions on the plane, which are quantified by angles that range from 0 to 2π. One of the most important differences with regard to the lineal variables is that, while these can take values of the whole real straight line $(+\infty,-\infty)$, the circular variables take cyclical values and consequently, the sum or difference of observations can surpass 360^{o} and can even result in a negative value, it being possible in such cases to find an equivalent value within the interval $0\text{-}360^{o}$. This characteristic allows the circular variables to be treated differently from the lineal ones, by means of statistical creation, correlation analysis and specific distributions for these types of variables.

Conceptually, a circular distribution can be considered in the same way as a bivariated lineal distribution where the total probability (or total mass) is dispersed within the circle unit. Therefore, in the same way as in the bivariated lineal statistic, a mean vector $\overline{m}$ of module r and mean angle $\overline{\Phi}$ exists in the circular statistic, at the tip of which the mass centre C of the distribution may be found (*Fig. 1*).

 ISBN 13: 978-84-613-5510-5

Let $Z(\Phi)$ be the random variable If we take a monomodal sample of frequencies $n_1, n_2, \ldots, n_j$ in the directions $\Phi_1, \Phi_2, \ldots, \Phi_j$, the mean vector $\overline{m}(r, \overline{\Phi})$ may be defined as

$$r = \sqrt{\bar{x}^2 + \bar{y}^2} \qquad \overline{\Phi} = \begin{cases} \text{Arctan}[\bar{y}/\bar{x}] & \text{si } \bar{x} > 0 \\ 180 + \text{Arctan}[\bar{y}/\bar{x}] & \text{si } \bar{x} < 0 \end{cases} \qquad (1)$$

Where, $\bar{x}$ and $\bar{y}$ are the projections of $\overline{m}$ on the X and Y axes, respectively:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{j} n_i \,\text{Cos}\,\Phi_i \qquad \bar{y} = \frac{1}{N}\sum_{i=1}^{j} n_i \,\text{Sin}\,\Phi_i \qquad N = \sum_{i=1}^{j} n_i \qquad (2)$$

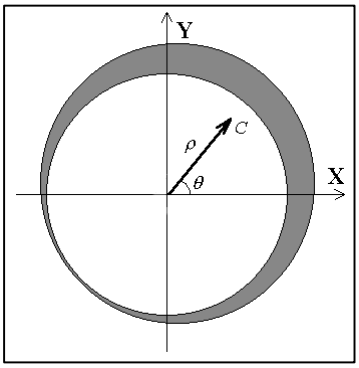


Figure 1.

If the data are contained in intervals of width λ, r should be corrected, the correct module being $r_c = r \cdot c$, where

$$c = \frac{\lambda/2}{\text{Sin}\,(\lambda/2)} \qquad (3)$$

Among the existing circular distributions [14], one of the most widely used for the modelling of circular variables is the Von Misses distribution, whose density function for *v*-modal and symmetric samples is

$$f(\Phi) = \frac{1}{2\pi\, I_0(k)} \text{Exp}[k \,\text{Cos}\, v(\Phi - \theta)] \qquad (4)$$

Where I_0 is the Bessel function of an imaginary pure argument of order 0, v is the number or modes and k it is the concentration parameter [14], that indicates to what measure the distribution around the dominant direction θ is concentrated. The *v*-modal samples should be considered as being extracted from a distribution

 ISBN 13: 978-84-613-5510-5

generated by the overlapping *v* monomodal distributions. When the distances between modes are arbitrary, standard methods do not exist to decompose a *v*-modal sample into *v* monomodal samples; in practice, the multimodal samples are usually shown as bimodal and are diametrically opposed. In this case, it is possible to reduce the bimodal sample to a monomodal sample, duplicating the angles. With the new angles, the average vector is calculated $\overline{m}_2(r_2, \overline{\Phi}_2)$ using Eq.(1)(2)(3). To obtain the symmetrical modal angle $\overline{\Phi}_1$ from the original sample, the angle duplication effect must be cancelled, which gives us $\overline{\Phi}_1 = \overline{\Phi}_2/2$ ó $\overline{\Phi}_1 = \overline{\Phi}_2/2 + 180^o$.

For $k = 0$, $f(\Phi)$ degenerates in an uniform distribution. Mardia demonstrated [11] that the maximum likelihood estimation $\hat{\theta}$ and $\hat{\rho}$ for parameters θ and ρ of a Von Misses distribution are respectively $\overline{\Phi}$ and *r*. Likewise,

$$\frac{I_1(\hat{k})}{I_0(\hat{k})} = r \tag{5}$$

is fulfilled. Hence, the maximum likelihood $\hat{k}$ is the solution of the Eq.(5).

3.2. Minkowski Sum

Given two images, *A* and *B* in R^2, the Minkowski sum is defined as

$$A \oplus B := \bigcup_{b \in B} A + b \tag{6}$$

Where *A* is the generating polygon, and *B* the skeleton or primary element (point, polyline, or polygon. $A \oplus B$ is generated by moving *A* though each element $b \in B$, and then by adding the result of all the translations later on. The translation of the generating polygon *A* through the element $b \in B$ is defined as

$$A + b := \{a + b, a \in A\} \tag{7}$$

If we take a circle as generating polygon *A*, and the group of points $B = \{(2,3),(3,4),(2,5),(1,5)\}$ as the primary element:

$$A \oplus B = [(A + (2,3)) \cup (A + (3,4)) \cup (A + (2,5)) \cup (A + (1,5))] \tag{8}$$

Fig. 2 shows the result, as well as $A \oplus L$ and $A \oplus P$, additions which have respectively taken polyline *L* and polygon *P* as primary elements.

 ISBN 13: 978-84-613-5510-5

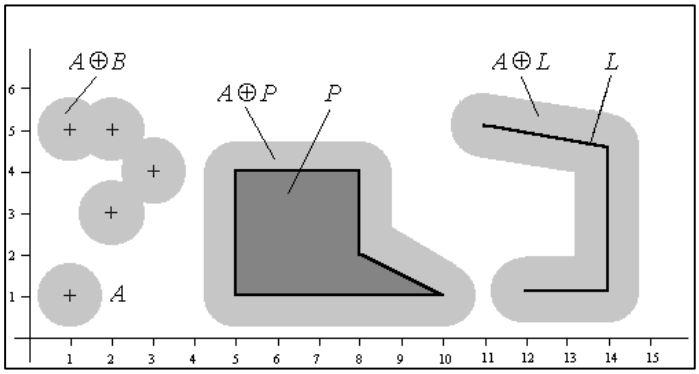


Figure 2.

Conceptually, the Minkowski sum is a dilation or expansion of the primary image B, whose form is determined by the generating polygon A. In the previous example we have chosen a circle as the generating image. The expansion of the primary image is directionally uniform or isotropic, since the generating image is a symmetrical figure with regard to both axes.

4. Forecasting CBR system

In the CBR system presented here, the images to be analyzed are divided into smaller squares. A squared zone determines the area that will be independently analyzed. The values of the different variables in a square area at a certain moment, which define the problem or the situation that has to be solved, is known as *a case*.

4.1. Case base creation

In this study, we have applied isotropic image analysis based on the buffer operator using Von Misses distribution and the Minkowski Sum, both previously introduced in Section 3. Owing to its good adaptation capabilities, this system has been applied to calculate the areas of different environmental phenomenon, which enable them to be modelled.

Once the data is structured, it is stored in the *case base*. Every case has its temporal situation stored, which relates every case with the next situation in the same position. That temporal relationship is what creates the union between *problem* and *solution*. The problem is the past case, and the solution is the future case, the future state of the square under analysis.

Growing Cell Structures (GCS) [9] are used when introducing the data into the case base,. GCS can create a model from a situation organizing the different cases

 ISBN 13: 978-84-613-5510-5

by their similarity. If a 2D representation is chosen to explain this technique, the most similar cells (*cases* in OSCBR) are near one or the other. If there is a relationship between the cells, they are grouped together, and this grouping characteristic helps the CBR system to retrieve the similar cases in the next phase. When a new cell is introduced in the structure, the closest cells move towards the new one, changing the overall structure of the system as shown in (9) and (10). The weights of the winning cell, ω_c, and its neighbours, ω_n, are changed. The new value is represented by $\omega_c(t+1)$, and $\omega_n(t+1)$ respectively. The terms ε_c and ε_n represent the learning rates for the winner and its neighbours, while x represents the value of the input vector.

$$\omega_c(t+1) = \omega_c(t) + \varepsilon_c(x - \omega_c) \quad (9)$$

$$\omega_n(t+1) = \omega_n(t) + \varepsilon_n(x - \omega_n) \quad (10)$$

4.2. Generating predictions

Once the case base has stored the historical data, and the GCS has been structured according to the original distribution of the variables, the system is ready to receive a new problem. When a new problem is introduced into the system, GCS are used once again. The stored GCS behaves as if the new problem were stored in the structure, and finds the most similar cells (*cases* in the CBR system) to the problem introduced into the system. In this case, the GCS does not change its structure, because it is being used to retrieve the most similar cases to the introduced problem. Only in the retain phase does the GCS change, introducing the proposed solution once again if it is correct.

The similarity of the new problem to the stored cases is determined by the GCS calculating the distance between them. Every element in the GCS has a series of values (every value corresponds to one of the principal components created after de FIK-PCA analysis) and the distance between the elements is therefore a multi-dimensional distance, where all the elements are considered to establish the distance between cells. Then, after obtaining the most similar cases from the case base, they are used in the next phase. The selected case bases will be used to generate an accurate prediction according to the previous solutions that relate to the problem that was introduced.

Having retrieved the most similar cases to the problem that has to be solved from the case base, they are used to generate the solution. The prediction of the future probability of finding oil slicks in an area was generated by using an artificial neural network, with a hybrid learning system. It was obtained with an adaptation of *Radial Basis Functions Networks* [10]. The chosen cases are used to train the artificial neural network. Radial Basis Function networks have been chosen

 ISBN 13: 978-84-613-5510-5

because of the reduction of the training time comparing with other artificial neural network systems, such as Multilayer Perceptrons. In this case, in every analysis the network is trained, using only the cases selected from the case base which are the most similar to the proposed problem.

Growing RBF networks [13] are used to obtain the predicted future values that correspond to the proposed problem. This adaptation of the RBF networks allows the system to grow during training gradually increasing the number of elements (prototypes) which play the role of the centres of the radial basis functions. In this case, the creation of the Growing RBF must be made automatically, which implies an adaptation of the original GRBF system. The pseudocode of the growing process and the definition of the error for every pattern is shown below:

$$e_i = {}^{1}/_{p} \cdot \sum_{k=1}^{p} \| t_{ik} - y_{ik} \|, \tag{11}$$

Where t_{ik} is the desired value of the k^{th} output unit of the i^{th} training pattern, and y_{ik} the actual values of the k^{th} output unit of the i^{th} training pattern.

5. Conclusions

We have presented a novel CBR system, by using for the first time a GIS technique based on the use of an isotropic buffer operator.

The areas in our CBR system were calculated by dividing the global images into smaller ones, so that we can apply a different buffer to each one. Changing the size of the buffer will help the system to generate a more accurate analysis, improving the quality of the data in the final case-based solution, resulting in better prediction results.

6. References

[1] Aamodt, A. (1991) A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning, *Knowledge Engineering and Image Processing Group. University of Trondheim*.

[2] Aamodt, A. and Plaza, E. (1994) Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *AI Communications*, 7 (1), 39-59.

[3] Batschelet, E. (1981) Circular statistics in biology, *ACADEMIC PRESS, 111 FIFTH AVE., NEW YORK, NY 10003, 1981, 388*.

[4] Carrascosa, C., Bajo, J., Julian, V., Corchado, J.M., *et al.* (2007) Hybrid multi-agent architecture as a real-time problem-solving model, *Expert Systems With Applications*, 34 (1), 2-17.

 ISBN 13: 978-84-613-5510-5

[5] Chou, Y.H. (1997) *Exploring spatial analysis in geographic information systems*. Onward Press.
[6] Chow, H.K.H., Choy, K.L., Lee, W.B. and Lau, K.C. (2006) Design of a RFID case-based resource management system for warehouse operations, *Expert Systems With Applications*, 30 (4), 561-576.
[7] Chun, S.H. and Park, Y.J. (2005) Dynamic adaptive ensemble case-based reasoning: application to stock market prediction, *Expert Systems With Applications*, 28 (3), 435-443.
[8] Fdez-Riverola, F., Iglesias, E.L., Díaz, F., Méndez, J.R., *et al.* (2007) Applying lazy learning algorithms to tackle concept drift in spam filtering, *Expert Systems With Applications*, 33 (1), 36-48.
[9] Fritzke, B. (1994) Growing cell structures—a self-organizing network for unsupervised and supervised learning, *Neural Networks*, 7 (9), 1441-1460.
[10] Haykin, S. (1999) *Neural networks*. Prentice Hall Upper Saddle River, NJ.
[11] Mardia, K.V. (1975) Statistics of directional data, *Journal of the Royal Statistical Society. Series B (Methodological)*, 37 (3), 349-393.
[12] Okabe, A., Boots, B., Sugihara, K. and Chiu, S.N. (2009) Spatial tessellations: Concepts and applications of Voronoi diagrams (POD), *européen des systèmes automatisés*, 43.
[13] Ros, F., Pintore, M. and Chrétien, J.R. (2007) Automatic design of growing radial basis function neural networks based on neighboorhood concepts, *Chemometrics and Intelligent Laboratory Systems*, 87 (2), 231-240.
[14] Schou, G. (1978) Estimation of the concentration parameter in von Mises-Fisher distributions, *Biometrika*, 65 (2), 369.
[15] Shiu, S.C.K. and Pal, S.K. (2004) Case-Based Reasoning: Concepts, Features and Soft Computing, *Applied Intelligence*, 21 (3), 233-238.
[16] Watson, I. (1999) Case-based reasoning is a methodology not a technology, *Knowledge-Based Systems*, 12 (5-6), 303-308.
[17] Yang, B.S., Han, T. and Kim, Y.S. (2004) Integration of ART-Kohonen neural network and case-based reasoning for intelligent fault diagnosis, *Expert Systems With Applications*, 26 (3), 387-395.
[18] Yu, W. and Liu, Y. (2006) Hybridization of CBR and numeric soft computing techniques for mining of scarce construction databases, *Automation in Construction*, 15 (1), 33-46.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Nonlinear Modelling and Forecasting of Intraday Stock Returns

J.M. Matías [1] and J.C. Reboredo[2]

[1] *Department of Statistics, University of Vigo, Spain*
[2] *Department of Economics, University of Santiago de Compostela, Spain*

emails: jmmatias@uvigo.es, juancarlos.reboredo@usc.es

Abstract

We studied the predictability of intraday stock market returns using both linear and non-linear time series models. For S&P-500 index, we compared simple autoregressive and random walk linear models with a range of nonlinear models that included smooth transition, Markov switching, artificial neural network, nonparametric kernel regression and support vector machine models for horizons of 5, 10, 20, 30 and 60 minutes. The empirical results indicate that nonlinear models outperformed linear models on the basis of both statistical and economic criteria. Specifically, although return serial correlation receded by around ten minutes, return predictability still persisted for up to sixty minutes according to nonlinear models, even though profitability decreases as time elapses. More flexible nonlinear models such as support vector machines and artificial neural network did not clearly outperform other nonlinear models.

Key words Forecasting; Intraday stock returns; Stock price forecasting; High frequency data; smooth transition regression, Markov switching regression, Neural networks, nonparametric kernel regression; support vector machine regressions

 ISBN 13: 978-84-613-5510-5

1. Introduction

The predictability of stock market returns has long attracted much interest of financial researches and practitioners alike as it has profound theoretical and practical implications. The cornerstone of predictability is the idea of financial market efficiency that states that stock returns fully adjust to all relevant information, so they cannot be forecastable.

This paper focuses on return predictability at intraday level. Intraday trading data prediction is particularly important as it offers practitioners the opportunity to gain high annual returns from a trading strategy. We should expect such predictability to weaken as traders' actions to exploit profit opportunities expunge serial return correlation and adjust prices to their new equilibrium levels quickly since the stock market in nowadays is assumed to be highly efficient, so predictability recedes over long horizons. In addition, intraday horizons are important in the management of the risk of a trading desk (see Chew (1994)). Despite its economic and financial importance, the analysis of stock return predictability at very short forecast horizons is scarce in the academic literature. An exception is Clements and Taylor (2003) who analyze interval forecasts of high-frequency data.

We explore intraday nonlinear return predictability for the S&P500 index during the June of 2003 to September of 2003 for different within the day temporal horizons of 5, 10, 20, 30 and 60 minutes. We use a wide set of nonlinear modelling techniques that includes smooth transition autoregressive model, Smooth transition autoregressive model with GARCH errors, Markov switching models, artificial neural networks, non parametric time series regression (NP); and finally, a support vector machines, a technique that comes from the statistical learning theory and whose predictive ability for intraday stock return data was not evaluated as yet to the best of our knowledge. Finally, we compare the forecasting performance of different models in terms of (a) some statistical criteria such as the root mean squared error, proportion of times the signs of returns are correctly forecasted, directional accuracy test of Pesaran and Timmermann (1992), and the popular Dielbold and Mariano (1995) test for the equality of accuracy of competing forecasts; and (b) an economic criteria, using a simple trading strategy guide by forecasts in order to test the relative pay-offs generated by different forecasting models.

2. Methodology

To forecast stock returns we used various models for $E[r_t/I_{t-1}]$, where r_t represents the first difference of the logarithmic stock price, and $I_{t-1}=\{r_{t-1}, r_{t-2},\ldots\}$ is the information set available at time t-1. The information set only included lagged returns, given that we only analyzed the dynamic characteristics of returns and nonsynchronous trading effects may result in autocorrelation in returns. Below we briefly describe the different models considered for the conditional mean.

 ISBN 13: 978-84-613-5510-5

We considered nine forecasting models, the simplest of which were the random walk model (RW) and an autoregressive (AR) specification:

$$r_{t+1} = \beta_0 + \sum_{j=1}^{k} \beta_j r_{t+1-j} + \varepsilon_{t+1}, \quad (1)$$

where k was selected to minimize the Bayes Information Criterion. To account for the possibility that the conditional mean $E[r_t/I_{t-1}]$ could be time-varying in a nonlinear form we employed several nonlinear models in addition to the linear models used as a benchmark. We briefly discuss the set of nonlinear specifications below.

Smooth transition autoregressive models (STAR) models account for the existence of different regimes with different dynamic properties and with smooth transition between regimes (Granger and Teräsvirta, 1993, Teräsvirta et al, 1994). A first-order STAR model with two regimes takes the following form:

$$r_{t+1} = \phi_{10} + \phi_{11} r_t + [\phi_{20} + \phi_{21} r_t] F(z_{t+1}; \gamma, c) + \varepsilon_{t+1}, \quad (2)$$

where $F(z_{t+1}; \gamma, c)$ is the smooth transition function that depends on the transition variable z_{t+1} and the parameters γ, which is the transition rate or smoothness parameter; and where c is the threshold value which represents the change from one regime to another. The transition variable can be defined as a linear combination of the lagged values of r_t, $z_{t+1} = \sum_{h=1}^{H} \alpha_h r_{t+1-h}$. The most widely used smooth transition functions are the logistic function and the exponential function: $F(z_{t+1}; \gamma, c) = 1/1 + \exp(-\gamma(z_t - c))$ and $F(z_{t+1}; \gamma, c) = 1 - \exp(-\gamma(z_t - c)^2)$, with $\gamma > 0$. This model is estimated by quasi-maximum likelihood using the logistic function.

To account for the possible effect of nonlinearities in variance on the mean, we also considered the model in Equation (2) with a GARCH(1,1) errors (STAR-GARCH). Thus, $\varepsilon_{t+1} = \eta_{t+1}\sqrt{h_{t+1}}$, where

$$h_{t+1} = \alpha_0 + \alpha_1 \varepsilon_t^2 + \alpha_2 h_t, \quad (3)$$

$\alpha_0 > 0$, $\alpha_1, \alpha_2 \geq 0$, η_{t+1} is a *i.i.d.* process with zero mean and unit variance.

The main feature of an autoregressive Markov switching (MS) model is the possibility for some of the parameters to switch across different regimes or states according to a Markov process governed by a state variable denoted by s_t (see Hamilton, 1989). A first-order autoregressive MS model has the following specification:

$$r_{t+1} = \alpha_{s_{t+1}} + \beta_{s_{t+1}} r_t + \varepsilon_{t+1}, \quad (4)$$

where ε_{t+1} is *i.i.d.* $N(0, \sigma^2_{s_t})$ and s_t is an unknown state variable that follows a first-order Markov chain, with a transition probability $Pr(s_t = j|s_{t-1}=i)=p_{ij}$ that indicates the probability of switching from state i at time t-1 to state j at time t. For simplicity sake, we assume that there are only two states of the economy, denoted as state one and state two, as in Maheu and McCurdy (2000) and in Perez-Quiros

 ISBN 13: 978-84-613-5510-5

and Timmerman (2000) (e.g., bull and bear markets (Chen and Shen (2007)) or low and high uncertainty in stock markets (Li (2007)).

We can use nonparametric kernel regression (KR) when nonlinearity in the conditional mean cannot be characterized explicitly. In such cases the return conditional mean is specified in a general form as:

$$E(r_{t+1}|I_t) = g(r_t, r_{t-1}, \ldots, r_{t-p+1}), \tag{5}$$

where p is the number of lagged stock returns and where the function $g(\cdot)$ can be approximated locally at each point by a linear function.

As an alternative to nonparametric nonlinear conditional mean, we considered artificial neural networks, which have proven to be useful in capturing nonlinearity-in-mean for forecasting financial time series. Artificial neural networks are a universal approximator in a wide variety of nonlinear patterns (see Hornik et al, 1990) and generate good predictions (Swanson and White, 1995, 1997). The basic structure of neural networks combines many basic nonlinear functions via a multilayer structure, where there is at least one hidden layer between inputs and outputs. The idea is that explanatory variables simultaneously activate the units in the hidden layer through some function, and output is produced subsequently from the units in the hidden layer through another function. The specific type of ANN employed in this study is the multilayer perceptron (MLP) model, the most basic but perhaps most widely used neural network in economic and financial applications. Hence, MLP(p,q):

$$f(\mathbf{x}_t;\boldsymbol{\theta}) = \sum_{j=1}^{q} c_j \psi(\mathbf{w}'_j \mathbf{x}_t + w_{j0}) + c_0$$

where ψ is a sigmoid function (typically, a logistic or hyperbolic tangent function).

The universal approximation properties of MLP (see Leshno, 1993) neural networks permit us to approximate any continuous or integrable function (the universal approximation property). In order to ensure consistency in a stochastic environment, in addition to considering the error associated with the approximation of the regression function via a finite number of parameters, it was necessary to consider the estimation error arising from the use of a limited quantity of data. The consistency of the MLP neural networks for different hypotheses was thus obtained (see Krzyzak, 1996; Fine, 1999), and specifically for dependent observations (Chen, 1999). When the series responds to an autoregressive model, then:

$$y_t = f(\mathbf{x}_t;\boldsymbol{\theta}) + \varepsilon_t$$

where $f(\mathbf{x}_t;\boldsymbol{\theta})$ is a neural network and ε_t is, for example, white noise. Trapletti (2000) demonstrated the stationarity and strongly mixing nature of the series $\{y_t\}$, as also the consistency and asymptotic normality of the least squares estimator for a hypothesis that ensures the identifiability of the network parameters (Hwang and Ding, 1997).

 ISBN 13: 978-84-613-5510-5

MLP training, which usually uses the squared loss, is performed through nonlinear optimization algorithms. In this research we used the Bayesian algorithm proposed by Foresee and Hagan (1997) as it is less dependent on expert criteria.

Another alternative to the previous regression models is the support vector machine (SVM) (Vapnik, 1998, Schölkopf and Smola, 2002). The SVMs for regression are linear models obtained in a new feature space X as a result of a transformation $\boldsymbol{\varphi}: \mathbf{R}^p \to \mathsf{X}$ of the input space, in which an inner product is defined through a positive definite function (kernel), $\langle \boldsymbol{\varphi}(\mathbf{x}_i), \boldsymbol{\varphi}(\mathbf{x}_t) \rangle = k(\mathbf{x}_i, \mathbf{x}_t)$. SVMs for regression have the following general formulation:

$$y_t = f(\mathbf{x}_t) = \langle \mathbf{w}, \boldsymbol{\varphi}(\mathbf{x}_t) \rangle + b .$$

Given a sample, the parameters $\mathbf{w}$, b in the SVM are estimated as the solution to the following regularization problem:

$$\min_{\mathbf{w},b} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{T-p} \ell(y_i, f(\mathbf{x}_i)) \right\} \tag{6}$$

where C is a regularizing constant and ℓ is the ε-insensitive loss:

$$\ell(y, f(\mathbf{x})) = |y - f(\mathbf{x})|_{\varepsilon} = \max\{0, (|y - f(\mathbf{x})| - \varepsilon)\} .$$

The only solution to the problem in Equation (12) is a linear combination of key points of the sample (the support vectors), $\mathbf{w} = \sum_{\text{s.v.}} \alpha_i \boldsymbol{\varphi}(\mathbf{x}_i)$, in such a way that the SVM results as:

$$f(\mathbf{x}_t) = \sum_{i \in \text{s.v.}} \alpha_i \langle \boldsymbol{\varphi}(\mathbf{x}_i), \boldsymbol{\varphi}(\mathbf{x}_t) \rangle + b = \sum_{i \in \text{s.v.}} \alpha_i k(\mathbf{x}_i, \mathbf{x}_t) + b .$$

SVMs share the general form of radial basis function neural networks, which are universal approximators to continuous or integrable functions (e.g., Park, 1993). Consistency results also exist (see Bousquet and Elisseeff, 2002; Steinwart, 2002).

We obtained in-sample and out-of-sample one-step-ahead forecasts that were composed of the estimated parameters of the model and lagged returns. The market timing ability of forecasting models was also compared with a simple buy and hold (B&H) intraday investment strategy for out-of-sample forecasting. This strategy implements a naïve allocation that consist of maintaining a 100% stock index or cash if the quantity predicted exceeds a threshold given by the transaction costs. Hence, the forecast by each predictor determines the position to be taken for the following time period. Thus, if the share price is expected to fall below a threshold on the basis of a particular predictor, then shares are sold if the agent holds assets or they are not bought if the agent holds cash. In contrast, if the share price is expected to rise above a threshold on the basis of a particular predictor, then shares are bought if the agent holds cash or they are not sold if the agent holds assets. The threshold is determined by transaction costs, which are assumed to be low for intraday transactions, as otherwise commissions would

 ISBN 13: 978-84-613-5510-5

erode profits. The value of these cost were determined in our training set as the mean value of rises and falls divided by 1000 (around 3 bps.).

3. Results

Our main empirical findings verify the predictability of stock returns beyond the time serial correlation recedes; i.e., although market is weak form efficiency according to linear models and therefore prices are not linearly predictable, predictable nonlinearity-in-mean is possible up to a 60 minutes time interval. Surprisingly, the forecast performance of nonlinear models decreases very slowly, mainly directionally predictability, which has an important implication for market timing and the resulting active intraday asset allocation management.
we show that simple autoregressive and random walk linear models are surpassed by a wide range of nonlinear models, including smooth transition autoregressive, smooth transition autoregressive with GARCH errors, Markov switching, multilayer perceptron, nonparametric kernel regression and support vector machine models, which potentially capture nonlinearity-in-mean in intraday stock returns. Traditional statistical criteria suggest that nonlinearities-in-mean are relevant to forecasting intraday stock returns both in-sample and out-of-sample and for any intraday time return period. Of the nonlinear models, for short time periods of five minutes, the Markov switching model performed best for in-sample forecasting, whereas kernel regression was the best performer for out-of-sample forecasting. Despite return serial correlation receding and returns behaving as a random walk for more than ten minutes, return predictability still persisted for up to sixty minutes according to nonlinear models. For the longer intraday time periods studied, smooth transition and neural network models appeared to be better performers, even though the Diebold-Mariano test was not conclusive on equal predictability among the models. We also evaluated linear and nonlinear models in terms of economic criteria, using a simple trading rule driven by model predictions and transaction cost. On economic grounds, trading rules based on Markov switching and support vector machine models were the most profitable for short time horizon returns, whereas smooth transition and neural networks behaved better for longer periods, even though profitability decreased as time elapsed.

4. References

[1] BOUSQUET, G., ELISSEEFF, A., *Stability and Generalization,* Journal of Machine Learning Research **2** (2002) 499-526

[2] CHEN, X., WHITE, H., *Improved Rates and Asymptotic Normality for Nonparametric Neural Networks Estimators,* IEEE Transactions on Information Theory **45** (1999) 682-691.

 ISBN 13: 978-84-613-5510-5

[3] CHEN, S., SHEN, C., *Evidence of the Duration-Dependence from the Stock Markets in the Pacific Rim Economies*, Applied Economics **39** (2007) 1461-74.
[4] CHEW, L., *Shock Treatment*, Risk **7** (1994) 63-70.
[5] CLEMENTS, M. P., TAYLOR, N., *Evaluating Interval Forecasts of High-Frequency Financial Data*, Journal of Applied Econometrics **18** (2003) 445-456.
[6] DIELBOLD, F., MARIANO, R., 1995. *Comparing Predictive Accuracy*. Journal of Business and Economic Statistics **13** (1995) 253-263.
[7] FAN J., GIJBELS I., *Local Polynomial Modeling and Its Applications*, Chapman Hall, 1996.
[8] FINE, T., *Feedforward Neural Network Methodology*, Springer, 1999.
[9] FORESEE, F.D., HAGAN, T., *Gauss-Newton Approximation to Bayesian Regularization*. Proceedings of the 1997 International Joint Conference on Neural Networks (1997) 1930-1935
[10] GRANGER, C.W.J., TERÄSVIRTA, T., *Modelling Nonlinear Economic Relationships*, Oxford University Press, Oxford, UK, 1993.
[11] HAMILTON, J.D., *A new approach to the economic analysis of nonstationary time series and the business cycle*. Econometrica **57** (1989) 357-384.
[12] HAMILTON. J.D., *Time Series Analysis*, (1994) Princeton University Press.
[13] HORNIK, K., STINCHCOMBE, M., WHITE, H., *Universal Approximation of an Unknown Mapping and its Derivatives using Multilayer Feedforward Networks*, Neural Networks **3** (1990) 551-560.
[14] HSIEC, D.A., *Testing for Nonlinear Dependence in Daily Foreign Exchange Rates*, Journal of Business, **62** (1989) 339-368.
[15] HSIEC, D.A., X, *Implications of Nonlinear Dynamics for Financial Risk Management*, Journal of Financial and Quantitative Analysis, **28** (1989) 41-64.
[16] HWANG, J.T.G., DING, A.A., *Prediction Intervals for Artificial Neural Networks*, Journal of the American Statistical Association, **92** (1997) 748-757
[17] KRZYZAK, A.; LINDER, T. LUGOSI, G., *Nonparametric Estimation and Classification Using Radial Basis Function Nets and Empirical Risk Minimization*, IEEE Transactions on Neural Networks **7** (1996) 475-487
[18] LEITCH, G., TANNER, J.E., *Economic Forecast Evaluation: Profit vesus the Conventional Error Measures*, American Economic Review **81** (1991): 580-590.
[19] LESHNO, M., LIN, V.Y., PINKUS, A., SCHOCKEN, S., *Multilayer Feedforward Networks with a nonpolynomial activation function can approximate any function*, Neural Networks **6** (1993) 861-867.
[20] LI, M.L., *Volatility States and International Diversification of International Stock Markets*, Applied Economics, **39** (2007) 1867-76.

 ISBN 13: 978-84-613-5510-5

[21]MAHEU, J.M., MCCURDY, T.H., *Identifying bull and bear markets in stock returns*, Journal of Business and Economic Statistics **18** (2000) 100-112.

[22]PARK, J., SANDBERG, I.W., *Approximation and Radial-Basis-Function Networks*, Neural Computation **5** (1993) 305-316

[23]PEREZ-QUIROS, G., TIMMERMAN, A., *Firm Size and Cyclical Variations in Stock Returns*, Journal of Finance **55** (2000) 1229-1262.

[24]PESARAN, M.H., TIMMERMANN, A., *A Simple Nonparametric Test of Predictive Performance*, Journal of Business and Economic Statistics **10** (1992) 461-465.

[25]SATCHELL, S., TIMMERMAN, A., *An Assessment of the Economic Value of Non-linear Foreign Exchange Rate Forecasts*, Journal of Forecasting **14** (1995) 477-497.

[26]SCHÖLKOPFF, B. AND SMOLA, A.J., *Learning with Kernels*. MIT Press, 2002.

[27]STEINWART, I., *Support Vector Machines are Universally Consistent*. Journal of Complexity **18** (2002) 768-791

[28]SWANSON, N.R., WHITE, H., *A Model Selection Approach to Assessing the Information in the Term Structure using Linear Models and Artificial Neural Networks*, Journal of Business and Economic Statistics **13** (1995) 265-275.

[29]SWANSON, N.R., WHITE, H., *Forecasting Economic Time Series using Flexible versus Fixed Specification and linear versus Nonlinear Econometric Models*, International Journal of Forecasting **13** (1997) 439-461.

[30]TERÄSVIRTA, T., TJOSTHEIM, D., GRANGER, C.W.J., *Aspects of Modeling Nonlinear Time Series*. In Engle, R. F., McFadden, D. L. Eds.), Handbook of Econometrics, vol. IV. Elsevier 2919-2957. CHAP. 48, 1994.

[31]TRAPLETTI, A., LEISCH, F., HORNIK, K., *Stationarity and Integrated Autoregressive Neural Network Processes*. Neural Computation **12** (2000) 2427-2450

[32]VAPNIK, V., *Statistical Learning Theory*, John Wiley, 1998.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Screening effect on the convective heat transfer coefficients during vacuum frying of potato cylinders

Mir-Bel J.[1], Oria R.[1] and Salvador M.L.[1]

[1] *Laboratory of Vegetal Food, University of Zaragoza, Miguel Servet 177, 50013 Zaragoza, Spain.*

emails: jmir@cita-larioja.es, mlsalva@unizar.es

Abstract

In this study convective heat transfer coefficients were determined during vacuum frying of potato cylinders (5, 10 mm radius) in sunflower oil at 100, 120 and 140 ºC. These parameters were evaluated with indirect method based on the adjustment of experimental temperatures, obtained with a teflon probe, and a thermal model implemented with the finite elements software COMSOL.

The results obtained shown higher values for the vacuum frying than for atmospheric frying. Geometry and use of vacuum conditions played an important role in the convective heat transfer, being more importants than the oil temperature. A screening effect, originated for bubble collapsing due to high evaporation rate, was also observed in the case of vacuum fried thinner cylinders and was incorporated into the model, with the use of an effective temperature, T_{ef}.

Key words: Vacuum frying, convective heat transfer, bubble screening, finite element method, COMSOL.

1. Introduction

Frying is an extensively process used both in the food industry and domestically. It basically consists of cooking foodstuffs in oil or fat at temperatures well above the boiling point of water. This fast and easy preparation results in products with organoleptic qualities (colour, texture, flavour) much appreciated by consumers. However, these products end up with a high fat content that in some cases reaches a third of their total weight. In recent decades, therefore, numerous complementary processes have been proposed to reduce fat content while retaining sensorial qualities [1].

 ISBN 13: 978-84-613-5510-5

One of these alternative processes is frying at reduced pressure, used for many years in the snacks industry especially in South-East Asia. Working in vacuum conditions reduces the boiling point of water in the foodstuffs which can therefore be eliminated at lower temperatures, allowing frying with oil at a lower temperature. Numerous studies have evaluated the adaptation of this technique with different vegetables, comparing it with frying at atmospheric pressure. It has been found that as well as reducing the final fat content [2,3], vacuum fried products have several other advantages such as a lower acrylamide content [4] and improved organoleptic and nutritional qualities [5,6,7,8]

Coupled heat and mass transfer with bubbling make those processes difficult for modeling and further optimization purposes. This is especially important in the case of vacuum frying as lower pressure increases the evaporation rate and the size of the bubbles. Across the solid surface, there becomes a vigorous movement of evaporation causing a considerable mixing in the oil affecting convective heat transfer coefficient. The knowledge of this coefficient is expected to allow accurate determination of temperature distribution and hence the calculations to lead to development of an optimum frying process.

It was therefore decided to evaluate the convective heat transfer during moderate vacuum frying of potato cylinders.

2. Material and methods

The experimental material was potato (Solanum Tuberosum, cv. Agria) obtained from from local distributors. Each sample consisted of cylinders of different radii (1 and 0.5 cm respectively) and 5 cm in length extracted from the middle of the potato, using a metal punch. Sunflower oil with a high oleic acid content specially prepared for frying was used (Titan, Koipe, Spain).

A pressure cooker (Gastrovac ICC, Spain) with a nominal pressure of up to 20 kPa was used for the reduced pressure frying, sufficient for the pressure levels required. The changes in pressure and temperature were recorded with a piezoresistive pressure transducer (Picovacq PT, Digiterm, France) placed inside the system. In order to obtain suitable working conditions and keep them constant throughout the process, a ratio of 3.5 L of oil for every 100 g of potato was used. Each frying operation consisted of an initial depressurization step with the potatoes and the probe outside the oil, an immersion step with the oil already hot and subsequent removal and

 ISBN 13: 978-84-613-5510-5

draining during 1 min and vacuum breaking during 1 min. Under these conditions the absolute pressures recorded were between 20 and 35 kPa depending on the working temperature and the exact moment of frying, given that the vigorous initial vaporization causes a certain degree of oscillation in the pressure of the system. Before each experiment, the oil was kept at the working temperature and the maximum vacuum level for one hour, and discarded after a maximum of five hours of use. Frying was carried out at temperatures of 100, 120 and 140 ºC.

Similar experiments were carried out at atmospheric pressure and 140 ºC using the same equipment but without applying the vacuum conditions. Three replications were made for each of the conditions.

To determine the heat transfer coefficient during atmospheric and pressure frying conditions an indirect method was used [9, 10]. A cylindrical teflon probe was placed inside the frying oil whit potato cylinders, three big or four small, at different conditions to simulate the bubbling of a normal frying process as can be observer in Fig 1. K-type thermocouples were used to determine the time history of sample temperatures during frying. One thermocouple was placed in the oil to determine the process temperature while the other thermocouple was put across the cylinder radius and fixed near the geometric center of the sample. To determine the convective coefficients the obtained thermal values were adjusted to a mathematical two-dimensional axialsymmetric model.

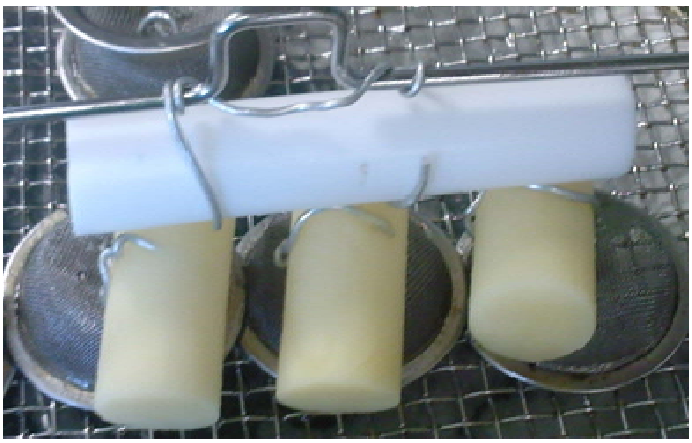


Fig.1. Image of the device used to obtain the experimental data.

Heat transfer from the oil to the teflon probe was simulated using a comercial simulation package COMSOL Multiphysics 3.4 with the heat transfer module (Comsol, Swedish), that makes use of finite element method. General solution form and Direct (UMFPACK) linear system

 ISBN 13: 978-84-613-5510-5

solver were used in simulations. For the probe a heat transfer scheme was adopted as convective at the exterior and conductive in the interior of the cylinder according the following expressions:

$$\rho\ C_p \frac{\partial T}{\partial t} + \nabla(-k\nabla T) = 0 \qquad \text{For the teflon probe}$$

The boundary conditions are:

$$k\ \nabla T = h\ A\ (T_\infty - T_s) \quad \text{For the external boundaries}$$

$$k\nabla T = 0 \qquad \text{For the axes of symmetry}$$

where ρ is the density, C_p is the specific heat, k is thermal conductivity, A is the area, h is the convective heat transfer coefficient, T_∞ is the oil temperature and T_s is the surface temperature.

These equations were implemented using a mesh of 362 nodes with triangular elements. The input data were the thermal and physical properties of the teflon: k, 0.35 W m^{-1} K^{-1}; C_p,1,050 J kg^{-1} K^{-1}; and ρ, 2,200 kg m^{-3} [11]. The convective heat coefficient was adjusted appropriately until the best fit (minimum mean squared error, MMSE) was obtained between the experimental data of temperature at the central point and those predicted from simulation. The parametric optimization was achieved using the Nelder-Mead method through MATLAB 7.0 (Mathworks, USA), specifically the "FMINSEARCH" utility.

3. Results and discussion

Fig. 2 presents changes in the values of the experimental data for the central temperature in the teflon probe for the studied conditions. There are big differences between the different geometries and studied conditions. In the case of the thick cylinders the temperature rises up faster ending close to the set point. For the small ones the temperature remains between 18 and 20 ºC under the oil values and increases slowly. This could be explained attending to the different evaporation rates, just after the vapour escapes from the potatoes the bubbles start collapsing isolating the surface of the probe. This screening effect is more important in the case of vacuum frying, because the bigger bubbles, and for the thinner cylinder, because its ratio surface/volume is higher and their water lose is faster.

 ISBN 13: 978-84-613-5510-5

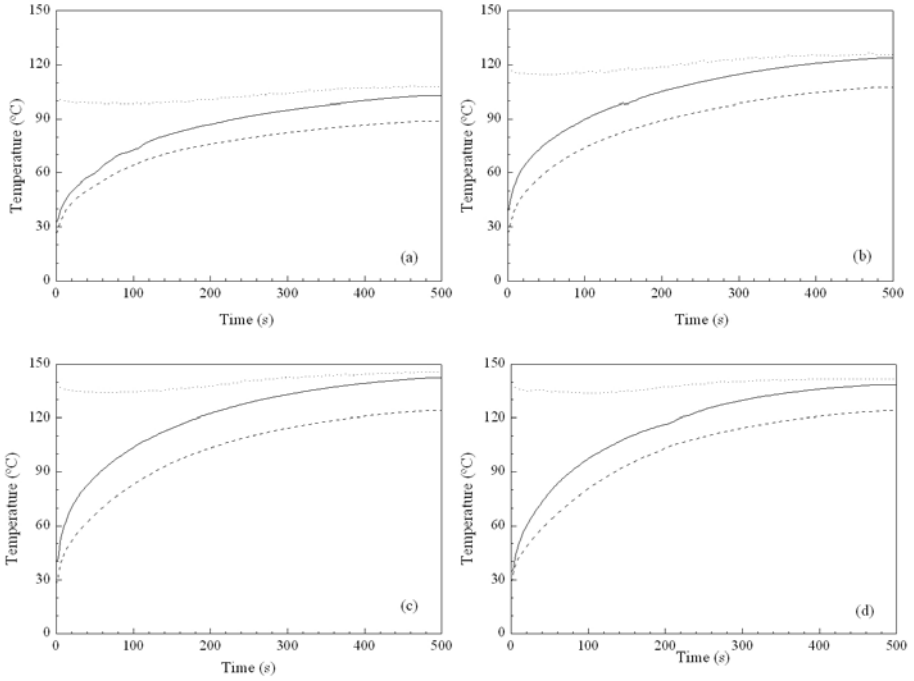


Figure 2. Central probe temperature during frying at different conditions: 100 ºC (a), 120 ºC (b), 140 ºC (c) with vacuum conditions and 140 ºC and atmospheric pressure (d). Oil temperature (dot line), thick potato samples (continuous line), thin potato samples (dash line).

As the convective conditions in the oil depends of the moisture transport the base convective coefficient was supposed to follow an exponential form in parallel to water loss in vacuum frying [2]. This form is also used for other authors in atmospheric frying [12] and is expressed as:

$$h = A + B \cdot e^{(-C \cdot t)}$$

To include the screening effect the effective temperature parameter, T_{ef}, was introduced into the boundary convective condition instead of set temperature, T_∞. With this additional value it is considered that the oil surrounding the surface of the probe is at a lower temperature that the rest due to the bubble screening. The adjusted values for the A,B,C and T_{ef} parameters and the corresponding MMSE are shown in Table 1. This also presents the corrected MMSE, calculated as the results of multiply the MMSE by the numbers of adjust parameters, used in each case to represent the global efficiency of each method.

To include the screening effect, the effective temperature parameter, T_{ef}, was introduced into the boundary convective condition instead of the external temperature, T_∞. This additional value takes into account that the oil surrounding the surface of the probe is at a lower temperature that the

 ISBN 13: 978-84-613-5510-5

rest due to the bubble screening. The adjusted values for the *A,B,C* and T_{ef} parameters and the corresponding MMSE are shown in Table 1, which presents as well the corrected MMSE, calculated by multiplying the MMSE by the fitted parameters, used in each case to represent the global efficiency of each method.

Table 1. Convective heat transfer coefficient *h* values determined with the indirect method.

Heat coefficient	Potato cylinder size	Parameter	Vacuum 100 ºC	Vacuum 120 ºC	Vacuum 140 ºC	Atm. 140 ºC
Exponential $h=A+B.e^{(-C.t)}$	Thick	A (W m^{-2} K^{-1})	155.34	165.00	177.80	130.92
		B (W m^{-2} K^{-1})	1903.64	3684.50	13627.00	1278.57
		C (s^{-1})	0.204	0.160	0.332	0.124
		MMSE	0.91	1.76	1.58	1.18
		MMSE*n	2.73	5.28	4.74	3.54
	Thin	A (W m^{-2} K^{-1})	56.75	57.34	60.78	65.73
		B (W m^{-2} K^{-1})	429.12	406.91	407.54	321.71
		C (s^{-1})	0.066	0.059	0.066	0.075
		MMSE	1.07	0.81	1.29	1.36
		MMSE*n	3.21	2.43	3.87	4.08
Exponential with screen effect $h=A+B.e^{(-C.t)}$	Thick	A (W m^{-2} K^{-1})	128.71	125.00	177.10	120.18
		B (W m^{-2} K^{-1})	2194.46	2684.99	13311.00	1467.97
		C (s^{-1})	0.214	0.131	0.340	0.135
		T_{ef} (ºC)	102.4	123.5	140.1	141.7
		MMSE	0.71	1.63	1.55	1.15
		MMSE*n	2.84	6.52	6.20	4.60
	Thin	A (W m^{-2} K^{-1})	75.80	83.18	87.58	82.14
		B (W m^{-2} K^{-1})	492.11	528.34	500.31	273.64
		C (s^{-1})	0.067	0.078	0.089	0.065
		T_{ef} (ºC)	94.5	112.3	131.5	132.9
		MMSE	0.77	0.56	0.86	1.07
		MMSE*n	3.08	2.24	3.44	4.28

For the first model the MMSE are below 2 so could be considered a good adjust. The parameter '*A*', that represents the heat transfer at high times, increases with temperature in both geometries. The use of vacuum conditions increases this value in the case of thick cylinders while slightly decreases the coefficients for the small ones. There are frying studies that found convective coefficients being proportional [12] or inverse proportional [13] to temperature, depending of the operating conditions. In our case the differences are small comparing with the geometry or the use of vacuum conditions, so we can ignore the effect of temperature for the

 ISBN 13: 978-84-613-5510-5

conditions used. The sum '*A+B*', that represents heat transfer at initial times, show high values for the vacuum frying comparing with atmospheric frying due to the lower agitation observed in this case. The high values that can be registered with this model are similar to the values registered for other authors using similar experimental conditions [9].

The second model shows lower MMSE in all cases. The T_{ef} calculated for the thick cylinders is higher than set temperature, T_∞, being indicative of this modification only affects the adjustment due to the use of an extra parameter. As the corrected MMSE is also high for the thin cylinders fried at atmospheric pressure the same conclusion can be added for these working conditions. In the case of thin cylinders and vacuum conditions T_{ef} is lower than T_∞ in all cases; the difference from the set temperature is between 5.5 and 8.5 ºC showing that the screening is an important effect that has to be considered during frying, specially while using vacuum conditions in products with high evaporation rates. In addition, the introduction of the T_{ef} parameter is mathematically justified by the MMSE*n values, which are lower than the obtained with the first model.

The second model shows lower MMSE in all cases. The T_{ef} calculated for the thick cylinders is higher than the set temperature, T_∞, indicating that this modification only affects the fit due to the use of an extra parameter. As the corrected MMSE is also high for the thin cylinders fried at atmospheric pressure the same conclusion can be added for these working conditions. In the case of thin cylinders and vacuum conditions T_{ef} is lower than T_∞ in all cases; the difference from the set temperature is between 5.5 and 8.5 ºC showing that screening during frying is an important effect that has to be considered, especially while using vacuum conditions in products with high evaporation rates. In addition, the introduction of the T_{ef} parameter is mathematically justified by the MMSE*n values, which are lower than those obtained with the first model.

4. Acknowledgements

The authors express their gratitude to the 'Ministerio de Educación y Ciencia' (Spain) (Project: AGL2007-64252/ALI) and to the 'Diputación General de Aragón' (Project: ALCOTEC 2009/0196) for providing the financial support for the study.

5. References

[1] M. MELLEMA, *Mechanism and reduction of fat uptake in deep-fat fried foods.* Tr. Food Sci. & Tech. **14** (2003) 364–373.

 ISBN 13: 978-84-613-5510-5

[2] J. GARAYO AND R.G. MOREIRA, *Vacuum frying of potato chips*. J. Food Eng. **55** (2002) 181-191.
[3] M. MARISCAL AND P.BOUCHON, *Comparison between atmospheric and vacuum frying of apple slices.* Food Chem. **107** (2008) 1561-1569.
[4] C. GRANDA, R.G. MOREIRA AND S.E. TICHY, *Reduction of acrylamide formation in potato crisps by low-temperature vacuum frying.* J. Food Sci. **69** (2004) 405-411.
[5] S. SHYU AND L.S. HWANG, *Effects of processing conditions on the quality of vacuum fried apple chips.* Food Res. Int. **34** (2001) 133–142.
[6] S. SHYU AND L.S. HWANG, *Effect of vacuum frying on the oxidative stability of oils*. J. Am. Oil Chem. Soc. **75** (1998) 1393-1398.
[7] P.F. DA SILVA AND R.G. MOREIRA, *Vacuum frying of high-quality fruit and vegetable-based snacks.* LWT- Food Sci. Tech. **41** (2008) 1758-1767.
[8] E. TRONCOSO, F. PEDRESCHI AND R.N. ZUÑIGA, *Comparative study of physical and sensory properties of pre-treated potato slices during vacuum and atmospheric frying.* LWT-Food Sci. Tech. **42** (2009) 187–195.
[9] L.J. HUBBARD AND B.E. FARKAS, *A method for determining the convective heat transfer coefficient during immersion frying*. J. Food Pr. Eng. **22** (1999) 201–214.
[10] O. VITRAC, D. DUFOUR, G. TRYSTRAM AND A.L. RAOULT-WACK, *Characterization of heat and mass transfer during deep-fat frying and its effect on cassava chip quality*. J. Food Eng. **53** (2002) 161–176.
[11] J.M. MARÍN AND C. MONNÉ, *Transferencia de calor*, Kronos, Zaragoza (1998).
[12] D.L. BOUCHON AND P.PYLE, *Modelling Oil Absorption During Post-Frying Cooling II: Solution of the Mathematical Model, Model Testing and Simulations.* Trans IChemE, Part C Food Biopr. Proc. **83** (2005) 261-272.
[13] A. YILDIZ, T.K. PALAZOGLU AND F. ERDOGDU, Determination of heat and mass transfer parameters during frying of potato slices. J. Food Eng. **79** (2007) 11–17.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Comparison of GPS observations made in a forestry setting using functional data analysis

C. Ordóñez[1] , J. Martínez[1] , J.F. de Cos Juez[2] , and F. Sánchez Lasheras[3]

[1] *Department of Natural Resources and Environmental Engineering. University of Vigo. Campus Universitario, 36310 Vigo, Spain*
[2] *Department of Mining Exploitation and Prospection. University of Oviedo. E.T.S.I.MINAS, 33007 Oviedo, Spain*
[3] *Research Department, Tecniproject Ltd., 33004 Oviedo, Spain*

emails: javier.martinez@uvigo.es, cgalan@uvigo.es, jfcos@uniovi.es, fsanchez@tecniproject.com

Abstract

GPS receiver observations made in forestry settings are affected by data distortion and signal losses and this negatively affects precision and accuracy measurements. Using a technique for identifying functional outliers, we determine whether there are differences between errors for coordinates obtained at 10 different points of a forest characterized by a set of dasometric data. Our results indicate that the 2 points with highest error correspond to areas with dasometric values that would indicate these areas to have a more dense forest canopy than the remaining areas.

Key words: GPS, forest canopy, functional data, outlier, depth

1.Introduction

Geographic positioning system (GPS) receivers are frequently used in forestry settings for, among other applications [7], the monitoring of harvesting machinery [9] and cadastral forest surveys [19]. In forestry settings, however, measurement precision and accuracy are affected by forest canopy interference in the satellite signals. Tree trunks, branches and leaves distort or break up satellite signals and

 ISBN 13: 978-84-613-5510-5

negatively affect the accuracy and precision of receiver-measured coordinates in these conditions [6], [16]. For a forest of lodgepole pines (*Pinus contorta Douglas,* Gerlach (1989) found that radio signal losses from a satellite could be attributed to tree trunks, branches and foliage 23%, 28% and 36% of the time, respectively.

The typical approach to studying the effect of the forest canopy on GPS positional accuracy is to seek associations between accuracy and dasometric variables characterizing the forest canopy, such as basal area, stand density and the Hart-Becking index [11], [12], [16], [18]. In these cases, tests for comparing statistical distributions (parametric or non-parametric) are generally used to determine whether or not precision values obtained in zones with different forestry variable values can be considered to come from the same population. Analysis of variance (ANOVA) is also frequently used to identify the variables that significantly affect GPS positional accuracy [1]. However, this kind of test is not appropriate when working with a dense set of data collected over time (such as GPS data), as such data is more suitably handled as observations made at discrete points of a smooth stochastic process.

Data mining techniques have advanced to the point where the exploitation of vectorial data has proven to be inadequate; this has led to the emergence of functional data analysis (FDA) [17]. FDA applications are very varied and include environmental research [10], [15], sensors [21], industrial methods [8], [13] and medical research [2], [20].

In the functional approach, comparisons are made overall and take into account the time correlation structure of the data. This is the focus we give to our forest canopy problem. The method used to compare curves is based on the concept of functional depth, which is a measure of the centrality of a given curve within a group of curves [5].

The article is structured on the basis of a description of our methodology, a description of our results and the most relevant conclusions to be drawn from the results.

2.Methodology

We identified outliers using a functional approach, in such a way that the sample of observations was considered to be composed of a series of curves rather than a discrete set of point observations. First the curves were fitted to the discrete data by means of a process called smoothing and then outliers were identified using the concept of functional depth. In this section we explain the basic concepts

 ISBN 13: 978-84-613-5510-5

underlying the methodology for detecting outliers in the initial sample, namely, smoothing, functional depth and outlier identification.

2.1. Smoothing

One of the first studies in the FDA fields [17] considered functional data to be observations at discrete points of continuous random processes. Assume a set of observations $f(t_j)$ in a set of n_p points $t_j \in \mathbb{R}$, where t_j represents each instant of time. These observations can be considered as discrete observations of the function, where is a functional space.

In order to estimate the function $f(t)$, it is considered that $F = span\{\phi_{1,...,}\phi_{n_b}\}$, where $\{\phi_k\}\, k = 1,...,n$ is a set of basis functions. In other words, for each function $f(t) \in \chi \subset F$, we have:

$$f(t) = \sum_{k=1}^{n_b} c_k \phi_k(t) \tag{0.1}$$

The smoothing problem now consists of determining the solution to the following regularization problem [17]:

$$\min_{f \in F} \sum_{j=1}^{n_p} \{z_j - f(t_j)\}^2 + \lambda \Gamma(f) \tag{0.2}$$

where $z_j = f(t_j) + \varepsilon_j$ (ε_j is random noise with zero mean) represents each of the observations of the function f in the instant t_j, Γ is a differential operator that controls the complexity of the function and λ is a regularization parameter.

Bearing in mind (1.1), the problem (1.2) may be written as:

$$\min_{\mathbf{c}} \{(\mathbf{z} - \mathbf{\Phi c})^T (\mathbf{z} - \mathbf{\Phi c}) + \lambda \mathbf{c}^T \mathbf{R c}\}$$

where $\mathbf{z} = (z_1,...,z_{n_p})^T$ is the vector of observations, $\mathbf{c} = (c_1,...,c_{n_b})^T$ is the vector of coefficients expressed in (0.1), $\mathbf{\Phi}$ is the $n_p \times n_b$ matrix with elements $\mathbf{\Phi}_{jk} = \phi_k(t_j)$ and $\mathbf{R}$ is the $n_b \times n_b$ matrix with elements:

$$R_{kl} = \langle D^2\phi_k, D^2\phi_l \rangle_{L_2(\mathbf{T})} = \int_{\mathbf{T}} D^2\phi_k(t) D^2\phi_l(t) dt$$

 ISBN 13: 978-84-613-5510-5

2.2. Functional depth

Depth measurement was originally introduced in the context of multivariate analysis to measure the centrality of a point with respect to a sample. Depth provides a way of ordering a sample from its centre in such a way that the points closest to the centre have greater depth. Like most vectorial concepts, the concept of depth has been generalized to the functional case [5]. Functional depth measures the centrality of a curve within a set of curves.
In this study we focus on 2 of the most widely used depth measurements:

- Fraiman-Muniz depth (FMD): Let the empirical distribution function $F_{n,t}(f_i(t))$ be [5] for the functional sample $\{f_i(t)\}_{i=1}^{n}$, $t \in [a,b]$ be:

$$F_{n,t}(f_i(t)) = \frac{1}{n}\sum_{k=1}^{n} I(f_k(t) \le f_i(t))$$

where $I(\cdot)$ is the indicator function. The FMD for a curve f_i is given by:

$$FMD_n(f_i(t)) = \int_a^b D_n(f_i(t))dt$$

where $D_n(f_i(t))$ is the point depth of $f_i(t), \forall t \in [a,b]$ given by:

$$D_n(f_i(t)) = 1 - \left|\frac{1}{2} - F_{n,t}(f_i(t))\right|$$

- H-modal depth (HMD): The functional mode (based on the mode concept) is defined as the curve most densely surrounded by the other curves in a sample. HMD is expressed as:

$$MD_n(f_i,h) = \sum_{k=1}^{n} K\left(\frac{\|f_i - f_k\|}{h}\right)$$

where $K : \mathrm{R}^+ \to \mathrm{R}^+$ is a kernel function, $\|\cdot\|$ is a norm in a functional space and h is the bandwidth parameter. One of the most widely used norms for a functional space is L^2, expressed as:

$$\left\|f_i(t) - f_j(t)\right\|_2 = \left(\int_a^b (f_i(t) - f_j(t))^2 dt\right)^{1/2}$$

The infinite norm L^∞ is sometimes used:

$$\left\|f_i(t) - f_j(t)\right\|_\infty = \sup_{t\in(a,b)} \left|f_i(t) - f_j(t)\right|$$

 ISBN 13: 978-84-613-5510-5

Different kernel functions $K(\cdot)$ can also be defined, among them the truncated Gaussian kernel:

$$K(t) = \frac{2}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right), t > 0$$

2.3. The functional outlier concept

A functional sample may include elements that, although they do not constitute error in themselves, may feature patterns different from the rest of the sample. Depth measurement, as described above, is used to identify outliers in functional samples.

Depth and outlier are inverse concepts: outliers in functional samples have considerably less depth than non-outliers.

For this study we used the HMD to measure depth. The cutoff C was obtained so that type 1 error—the percentage of correct observations wrongly identified as outliers—was approximately 1%:

$$\Pr(MD_n(x_i(t)) \le C) = 0,01, \quad i = 1,...,n$$

In practice, the distribution of the functional depths for which the value for C needs to be estimated is unknown. Of the different estimation methods available [4], we selected the bootstrapping approach [14].

3. Application. Forest canopy impact on GPS accuracy

3.1. GPS measurements

The sample used in this research $\left\{(t_1, t_2, ..., t_{3600})_j\right\}_{j=1}^{10}$ consists of a set of GPS observations measured, for a set of 10 points, second by second over 1 hour, where t_{ij} represents measurement in instant i (in seconds) over the period of 1 hour at point j.

The data were collected using a double-frequency GPS receiver (HiperPlus, Topcon Positioning Systems, Inc., Livermore, CA, USA) while observing GPS pseudorange and carrier phase.

The GPS experimental data were collected during 2 days over periods of 5-6 hours between 18 and 21 July 2008. Antenna heights ranged from 1.35 to 1.70 m and the logging rate was 1 second. The collection of observations lasted for at

 ISBN 13: 978-84-613-5510-5

least 1.5 hours and the process was repeated 3 times a day. GPS data were revised to ensure continuity and were cut to obtain 10 datasets representing 1 hour.

The observation point was located at latitude 42°41'08.79872"N and longitude 6°38'03.210587" W (WGS84) and at an ellipsoidal height of 933.829 metres (considered the Z_{true} coordinate). These coordinates were calculated by differential correction in static surveying. Post-processing correction was carried out using the PONF base station as the nearest reference station in the regional GNSS network (http://gnss.itacyl.es/). This point was projected and the position set up as the 'true' position for calculating horizontal and vertical accuracy. The UTM coordinates were X_{true}=693814.623 and Y_{true}=4728635.531 (Datum ETRS89; zone 29N).

The planimetric error in each instant of time *i* was calculated using the expression:

$$E_{XY} = \sqrt{\left(X_i - X_{true}\right)^2 + \left(Y_i - Y_{true}\right)^2}$$

The altimetric error in each instant was calculated using the formula:

$$E_Z = \left|Z_i - Z_{true}\right|$$

3.2. Forest environment characterization

With a view to characterizing the forest lots studied, we calculated the parameters associated with the forestry characteristics of each. The parameters were determined by measuring the trees in a radius of 10 metres around the point where the GPS observations were made.
The parameters studied were the following:

- D_m: Normal diameter (measured at a height of 1.3 metres).
- H_m: Mean height.
- H_0: Dominant height (mean height of the 4 thickest trees).
- H_{tt}: Treetop height (total height less the height to the first branch).
- N: Number of feet/hectares.
- G: Basal area (cross-section at normal tree height).
- D_g: Mean squared diameter ($D_g = \sqrt{\frac{4G}{\pi n}}$; n = number of trees).
- HBI: Hart-Becking index (relationship between mean spacing between trees *a* and dominant height H_0 ($IH = \frac{a}{H_0}100$)

 ISBN 13: 978-84-613-5510-5

- V: Wood volume.
- W: Biomass.
- SC: slenderness coefficient (relationship between mean height and mean diameter of the biomass)

Table 1 shows the parameter values for the 10 studied plots and also minimum, maximum and mean values and standard deviation for the entire set.

Table 1. Forestry paramaters for 10 forest lots

Lot	D_m (cm)	H_m (m)	H_0 (m)	H_{tt} (m)	N (ft/ha)	G (m^2/ha)	D_g (cm)	HBI (%)	V (m^3/ha)	W (kg/ha)	SC
1	20.11	17.02	17.85	11.72	732.48	24.47	20.41	20.69	182.89	83696.33	82.42
2	20.28	15.34	16.40	10.36	764.13	25.61	20.31	22.06	191.20	87207.64	74.29
3	28.38	25.32	26.73	17.46	605.82	38.53	29.13	15.20	385.41	179483.25	88.50
4	29.68	27.69	28.01	19.21	668.30	46.84	29.79	13.81	474.05	220912.96	94.46
5	28.66	25.73	26.07	17.30	2509.59	54.38	28.17	7.66	415.45	146441.65	192.57
6	14.92	16.01	15.02	9.85	2037.15	37.20	15.92	14.75	237.06	114335.66	111.71
7	16.41	19.19	19.11	10.56	1751.19	40.50	17.22	12.51	279.05	131151.88	115.13
8	14.42	22.63	22.61	12.43	3056.36	60.18	16.08	8.00	442.99	216048.37	156.75
9	13.46	21.41	22.23	12.48	2960.08	53.96	16.01	8.27	380.39	188170.60	153.83
10	12.73	23.26	22.85	13.06	2992.34	46.35	14.23	8.00	312.88	157547.27	182.95
Min	12.73	15.34	15.02	9.85	605.82	24.47	14.23	7.66	182.89	83696.33	74.29
Max	29.68	27.69	28.01	19.21	3056.36	60.18	29.79	22.06	474.05	220912.96	192.57
Mean	20.12	21.38	21.66	13.62	1811.63	42.72	20.94	13.39	329.86	152467.07	126.62
STD	6.70	4.33	4.48	3.33	1044.93	11.91	6.05	5.28	104.81	49092.53	43.06

4. Results

The first step in identifying possible outliers in the data was to fit curves to the set of values for the planimetric and altimetric errors. The smoothing method described in Section 2.1 was used for this purpose, resulting in a set of 10 curves for each error type. Figure 1 shows the 10 planimetric error curves. The great functional complexity of the sample is evident in the irregularity of the functions.

 ISBN 13: 978-84-613-5510-5

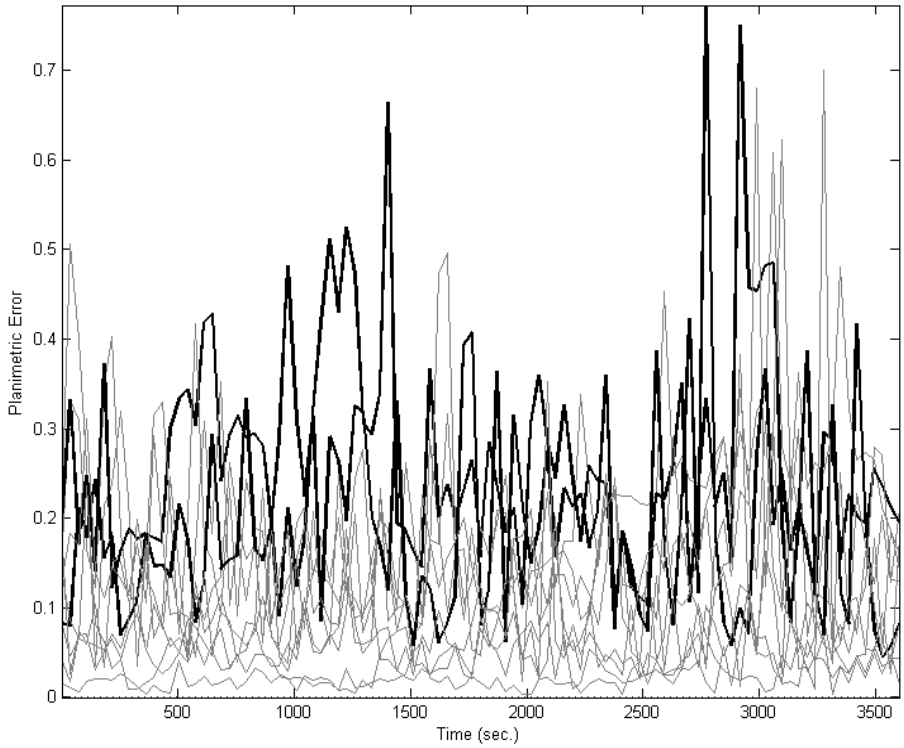


Figure 1. Functional sample for planimetric errors, with 2 outliers depicted in black.

Given the complexity of the sample, it was necessary to select a set of 1000 basis functions in order to obtain the most information possible. The outcome was a fit defined by an R-squared value (RSQ) of 0.99.

In order to study the functional outliers the HMD function was selected considering the norm L^2 of a functional space. The analysis identified 2 functional outliers (depicted in black in Figure 1) in the sample for the specific case of the planimetric errors; these outliers corresponded to Lots 5 and 8 in Table 1. No functional outliers for the error in Z were detected.

This result differs from that obtained using the Kruskal-Wallis test for all the positions except numbers 5 and 8 (the fact that the errors do not follow a normal distribution was first checked and was the reason a non-parametric test was used). This test rejected the null hypothesis, for a 99% significance level, that the 8 error observations (all except numbers 5 and 8) came from the same population. Recall that, since the Kruskal-Wallis test compares the medians of the groups, it is possible to have a tiny p value—clear evidence that the population medians are different— even if the distributions overlap considerably.

 ISBN 13: 978-84-613-5510-5

The 2 points corresponding to the outliers are characterized by high values for tree density (N), volume (V) and biomass (W), low values for the Hart-Becking index and the largest basal areas. Our result corroborates the results obtained by other authors, such as Naesset (1999, 2001)—who found that the basal area is a parameter that has a significant bearing on GPS measurement accuracy—and Rodríguez-Pérez et al. (2007)—who detected an association between the Hart-Becking index and precision.

5. Conclusions

We have analysed the application potential of functional data analysis to the evaluation of the impact of the forest canopy on the accuracy of GPS receiver measurements. Adopting a functional approach means that error measurements over the period of an hour can be considered as a continuous function. In other words, we can work with all the information rather than just with mean values, which is the basis for the traditional statistical approach.

Outliers were detected using functional depth measurement (a generalization of the vectorial case), which provides information on the distance of a function from the centre of a sample. In contrast with conventional vectorial techniques, this methodology does not require a normality hypothesis for the sample (whether this hypothesis would be valid was, nonetheless, checked) and also takes into account the time correlation structure of the data.

The 2 outliers detected in our study can be explained by the fact that they correspond to points located in lots with the highest basal areas and also high tree density, volume and biomass values and a low Hart-Becking index.

References

[1] C.H. DECKERT AND P. BOSLTADR, *Forest Canopy, Terrain and Distance Effects on Global Positioning System Point Accuracy*. Photogrammetric Engineering & Remote Sensing, **62(3)** (1996), 317-321.

[2] D.A. DOMBECK, M.S. GRAZIANO AND D.W. TANK, *Functional clustering of neurons in motor cortex determined by cellular resolution imaging in awake behaving mice*, Journal of Neuroscience, **29(44)** (2009) 13751-13760.

[3] D. EVANS, R. CARRAWAY AND G. SIMMONS, *Use of global positioning system (GPS) for forest plot location*. Southern J. Applied Forestry, **16(2)** (1992) 67–70.

 ISBN 13: 978-84-613-5510-5

[4] M. FEBRERO, P. GALEANO AND W. GONZÁLEZ-MANTEIGA, *Outlier detection in functional data by depth measures, with application to identify abnormal NOx levels*. Environmetrics (2008) 19:331-345. doi: 10.1002/env.878.

[5] R. FRAIMAN AND G. MUNIZ, *Trimmed means for functional data*. Test **10** (2001) 419-440.

[6] H. HASEGAWA AND T. YOSHIMURA, *Application of dual-frequency GPS receivers for static surveying under tree canopies*. J. For. Res. **8** (2003) 103-110.

[7] L.R. KRUCZYNSKI AND A. JASUMBACK, *Forestry Management Applications: Forest Service Experiences with GPS*, Journal of Forestry, **91(8)** (1993) 20-24.

[8] M. LÓPEZ, J.M. MATÍAS, J.A. VILÁN AND J. TABOADA, *Functional Pattern Recognition of 3D Laser Scanned Images of Wood-Pulp Chips*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **4477(1)** (2007) 298-305.

[9] T.P. MCDONALD, E.A. CARTER AND S.E. TAYLOR, *Using the global positioning system to map disturbance patterns of forest harvesting machinery*. Canadian J. Forest Research, **32(2)** (2002) 310–319.

[10] J.M. MATÍAS, C. ORDÓÑEZ, J. TABOADA AND T. RIVAS, *Functional support vector machines and generalized linear models for glacier geomorphology analysis*, International Journal of Computer Mathematics, **86(2)** (2009) 275-285.

[11] E. NÆSSET, *Point accuracy of combined pseudorange and carrier phase differential GPS under forest canopy*. Canadian J. Forest Research, **29(5)** (1999) 547–553.

[12] E. NÆSSET, AND T. JONMEISTER, *Assessing point accuracy of DGPS under forest canopy before data acquisition, in the field and after postprocessing.* Scandinavian J. Forest Research, **17(4)** (2002) 351–358.

[13] J.I. PARK, S.H. BAEK, M.K. JEONG AND S.J. BAE, *Dual features functional support vector machines for fault detection of rechargeable batteries*, IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, **39(4)** (2009) 480-485.

[14] L. PENG AND Y. QI, *Bootstrap approximation of tail dependence function*, Journal of Multivariate Analysis **99 (8)** (2008) :1807-1824. doi: 10.1016/j.jmva.2008.01.018.

[15] J. M. PARUELO AND F. TOMASEL, *Prediction of functional characteristics of ecosystems: a comparison of artificial neural networks and regression models*, Ecological Modelling, **98** (1997) 173-186.

[16] J.R. RODRÍGUEZ-PÉREZ, M.F. ÁLVAREZ AND E. SANZ-ABLANEDO, *Assessment of low-cost GPS receiver accuracy and precision in forest environments*. J. Surv. Eng. **133** (2007) 159-167.

 ISBN 13: 978-84-613-5510-5

[17] J.O. RAMSAY, AND B.W. SILVERMAN, *Functional Data Analysis*. New Yor, Springer, 1997.

[18] I. SAWAGUCHI, K. NISHIDA, M. SHISHIUCHI AND S. TATSUKAWA, *Positioning precision and sampling number of DGPS under forest canopies*. J. Forest Research, **8(2)** (2003) 133–137.

[19] T. SOLER, G. ÁLVAREZ-GARCÍA, A. HERNÁNDEZ-NAVARRO AND R.H. FOOTE, *GPS high accuracy geodetic networks in Mexico*. J. Surv. Eng., **122(2)** (1996), 80–94.

[20]R. VIVIANI, G. GRÖN AND M. SPITZER, *Functional principal component analysis of FMRI data*, Human Brain Mapping, **24(2)** (2005) 109-129.

[21]D. WU, S. HUANG AND J. XIN, *Dynamic compensation for an infrared thermometer sensor using least-squares support vector regression (LSSVR) based functional link artificial neural networks (FLANN)*, Measurement Science and Technology, **19(10)** (2008) nº105202.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Mathematical Modeling of Forest Fire Front Spread

Valeriy Perminov
Belovo Branch of Kemerovo State University

email: valerperminov@gmail.com

Abstract

In this paper the assignment and theoretical investigations of the problems of crown forest fire spread in windy condition were carried out. Mathematical model of forest fire was based on an analysis of known experimental data and using concept and methods from reactive media mechanics. In this context, a study - mathematical modeling - of the conditions of forest fire spreading that would make it possible to obtain a detailed picture of the change in the velocity, temperature and component concentration fields with time, and determine as well as the limiting conditions of forest fire propagation is of interest.

Key words: mathematical, forest fire, ignition, discrete analogue, control volume.

1. Introduction

A great deal of work has been done on the theoretical problem of crown forest fire initiation. Crown fires are initiated by convective and radiative heat transfer from surface fires. However, convection is the main heat transfer mechanism (Van Wagner [1]). The proposed in [1] theory depends on three simple crown properties: crown base height, bulk density of forest combustible materials and moisture content of forest fuel. Also, crown fire initiation and hazard have been studied and modeled in details later (Alexander [2]; Van Wagner [3]; Xanthopoulos [4]; Rothermel [5]; Cruz and others [6]; Albini and others [7]; Scott and Reinhardt [8]).The more complete discussion of the problem of crown forest fires is provided by coworkers at Tomsk University (Grishin [9]; Grishin and Perminov [10]; Perminov [11,12]). In particular, a mathematical model of forest fires was obtained by Grishin [9] based on an analysis of known and original experimental data (Grishin [9]; Konev [13], and using concepts and methods from reactive media mechanics. The physical two-phase models used by Morvan and

 ISBN 13: 978-84-613-5510-5

Dupuy [14,15] may be considered as a continuation and extension of the formulation proposed in [9]. This study gives a two dimensional averaged mathematical setting and method of numerical solution of a problem of a forest fire spread. The boundary-value problem is solved numerically using the method of splitting according to physical processes. It was based on numerical solution of two dimensional Reynolds equations for the description of turbulent flow taking into account for diffusion equations chemical components and equations of energy conservation for gaseous and condensed phases, volume of fraction of condensed phase (dry organic substance, moisture, condensed pyrolysis products, mineral part of forest fuel).

2. Physical and mathematical model

It is assumed that the forest during a forest fire can be modeled as 1) a multi-phase, multistoried, spatially heterogeneous medium; 2) in the fire zone the forest is a porous-dispersed, two-temperature, single-velocity, reactive medium; 3) the forest canopy is supposed to be non - deformed medium (trunks, large branches, small twigs and needles), which affects only the magnitude of the force of resistance in the equation of conservation of momentum in the gas phase, i.e., the medium is assumed to be quasi-solid (almost non-deformable during wind gusts); 4) let there be a so-called "ventilated" forest massif, in which the volume of fractions of condensed forest fuel phases, consisting of dry organic matter, water in liquid state, solid pyrolysis products, and ash, can be neglected compared to the volume fraction of gas phase (components of air and gaseous pyrolysis products); 5) the flow has a developed turbulent nature and molecular transfer is neglected; 6) gaseous phase density doesn't depend on the pressure because of the low velocities of the flow in comparison with the velocity of the sound. Let the point x_1, x_2 , x_3= 0 is situated at the centre of the surface forest fire source at the height of the roughness level, axis $0x_1$ directed parallel to the Earth's surface to the right in the direction of the unperturbed wind speed, axis $0x_2$ directed perpendicular to $0x_1$ and axis $0x_3$ directed upward (Fig. 1).

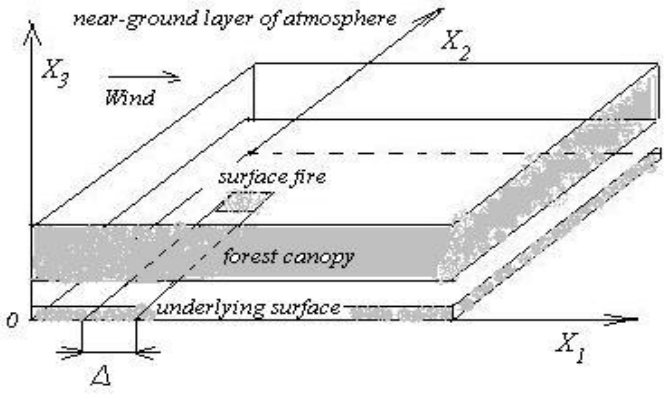


Figure 1.

 ISBN 13: 978-84-613-5510-5

Because of the horizontal sizes of forest massif more than height of forest – *h*, system of equations of general mathematical model of forest fire [9] was integrated between the limits from height of the roughness level - 0 to *h*. Besides, suppose that

$$\int_0^h \phi \ dx_3 = \bar{\phi} h$$

$\dddot{\phi}$ - average value of ϕ. The problem formulated above is reduced to a solution of the following system of equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho v_j) = Q - (\dot{m}^- - \dot{m}^+)/h, \ j = 1,2,3; \tag{1}$$

$$\rho \frac{dv_i}{dt} = -\frac{\partial p}{\partial x_j} + \frac{\partial}{\partial x_j}(-\rho \overline{v_i' v_j'}) - \rho s c_d v_i |\vec{v}| - \rho g_i - Q v_i + (\tau_i^- - \tau_i^+)/h, \ i = 1,2,3; \tag{2}$$

$$\rho c_p \frac{dT}{dt} = \frac{\partial}{\partial x_j}(-\rho c_p \overline{v_j' T'}) + q_5 R_5 - \alpha_v (T - T_s) - (q_T^- - q_T^+)/h; \tag{3}$$

$$\rho \frac{dc_\alpha}{dt} = \frac{\partial}{\partial x_j}(-\rho \overline{v_j' c_\alpha'}) + R_{5\alpha} - Q c_\alpha + (J_\alpha^- - J_\alpha^+)/h, \ \alpha = 1,3; \tag{4}$$

$$\frac{\partial}{\partial x_j}\left(\frac{c}{3k}\frac{\partial U_R}{\partial x_j}\right) - k(cU_R - 4\sigma T_S^4) + (q_R^- - q_R^+)/h = 0; \tag{5}$$

$$\sum_{i=1}^{4} \rho_i c_{pi} \varphi_i \frac{\partial T_S}{\partial t} = q_3 R_3 - q_2 R_2 + k(cU_R - 4\sigma T_S^4) + \alpha_V (T - T_S); \tag{6}$$

$$\rho_1 \frac{\partial \varphi_1}{\partial t} = -R_1, \rho_2 \frac{\partial \varphi_2}{\partial t} = -R_2,$$
$$\rho_3 \frac{\partial \varphi_3}{\partial t} = \alpha_C R_1 - \frac{M_C}{M_1} R_3, \rho_4 \frac{\partial \varphi_4}{\partial t} = 0; \tag{7}$$
$$\sum_{\alpha=1}^{5} c_\alpha = 1, \ p_e = \rho R T \sum_{\alpha=1}^{5} \frac{c_\alpha}{M_\alpha}, \vec{v} = (v_1, v_2, v_3), \vec{g} = (0,0,g).$$

The system of equations (1)–(7) must be solved taking into account the initial and boundary conditions

$$t = 0 : v_1 = 0, v_2 = 0, v_3 = 0, T = T_e, c_\alpha = c_{\alpha e}, T_s = T_e, \varphi_1 = \varphi_{ie};$$

$$t = 0: v_1 = 0, v_2 = 0, v_3 = 0, T = T_e, T_S = T_e, c_\alpha = c_{\alpha e}, \varphi_i = \varphi_{ie}; \tag{8}$$

 ISBN 13: 978-84-613-5510-5

$$x_1 = -x_{1e} : v_1 = V_e, v_2 = 0, \frac{\partial v_3}{\partial x_1} = 0, T = T_e, c_\alpha = c_{\alpha e}, -\frac{c}{3k}\frac{\partial U_R}{\partial x_1} + cU_R/2 = 0; \quad (9)$$

$$x_1 = x_{1e} : \frac{\partial v_1}{\partial x_1} = 0, \ \frac{\partial v_2}{\partial x_1} = 0, \frac{\partial v_3}{\partial x_1} = 0, \ \frac{\partial c_\alpha}{\partial x_1} = 0, \frac{\partial T}{\partial x_1} = 0, \frac{c}{3k}\frac{\partial U_R}{\partial x_1} + \frac{c}{2}U_R = 0; \quad (10)$$

$$x_2 = x_{20} : \frac{\partial v_1}{\partial x_2} = 0, \ \frac{\partial v_2}{\partial x_2} = 0, \ \frac{\partial v_3}{\partial x_2} = 0, \frac{\partial c_\alpha}{\partial x_2} = 0, \frac{\partial T}{\partial x_2} = 0, -\frac{c}{3k}\frac{\partial U_R}{\partial x_2} + \frac{c}{2}U_R = 0; \quad (11)$$

$$x_2 = x_{2e} : \frac{\partial v_1}{\partial x_2} = 0, \ \frac{\partial v_2}{\partial x_2} = 0, \frac{\partial v_3}{\partial x_2} = 0, \frac{\partial c_\alpha}{\partial x_2} = 0, \frac{\partial T}{\partial x_2} = 0, \frac{c}{3k}\frac{\partial U_R}{\partial x_2} + \frac{c}{2}U_R = 0. \quad (12)$$

$$x_3 = 0 : v_1 = 0, \ v_2 = 0, \frac{\partial c_\alpha}{\partial x_3} = 0, -\frac{c}{3k}\frac{\partial U_R}{\partial x_3} + \frac{c}{2}U_R = 0, \quad (13)$$
$$v_3 = v_{30}, T = T_g \text{ при } |x_1| \le \Delta, |x_2| \le \Delta \text{ и } v_3 = 0, T = T_e \text{ при } |x_1| > \Delta, |x_2| > \Delta;$$

$$x_3 = x_{3e} : \frac{\partial v_1}{\partial x_3} = 0, \ \frac{\partial v_2}{\partial x_3} = 0, \frac{\partial v_3}{\partial x_3} = 0, \frac{\partial c_\alpha}{\partial x_3} = 0, \frac{\partial T}{\partial x_3} = 0, \frac{c}{3k}\frac{\partial U_R}{\partial x_3} + \frac{c}{2}U_R = 0. \quad (14)$$

Here and above $\frac{\mathrm{d}}{\mathrm{d}t}$ is the symbol of the total (substantial) derivative; α_v is the coefficient of phase exchange; ρ - density of gas – dispersed phase, t is time; v_i - the velocity components; T, T_S, - temperatures of gas and solid phases, U_R - density of radiation energy, k - coefficient of radiation attenuation, P - pressure; c_p – constant pressure specific heat of the gas phase, c_{pi}, ρ_i, φ_i – specific heat, density and volume of fraction of condensed phase (1 – dry organic substance, 2 – moisture, 3 – condensed pyrolysis products, 4 – mineral part of forest fuel), R_i – the mass rates of chemical reactions, q_i – thermal effects of chemical reactions; k_g, k_S - radiation absorption coefficients for gas and condensed phases; T_e - the ambient temperature; c_α - mass concentrations of α - component of gas - dispersed medium, index α=1,2,3, where 1 corresponds to the density of oxygen, 2 - to carbon monoxide CO, 3 - to carbon dioxide and inert components of air; R – universal gas constant; M_α , M_C, and M molecular mass of α -components of the gas phase, carbon and air mixture; g is the gravity acceleration; c_d is an empirical coefficient of the resistance of the vegetation, s is the specific surface of the forest

 ISBN 13: 978-84-613-5510-5

fuel in the given forest stratum. In system of equations (1)-(7) are introduced the next designations:

$$\dot{m} = \rho v_3, \tau_i = -\rho \overline{v_i' v_3'}, J_\alpha = -\rho \overline{v_3' c_\alpha'}, J_T = -\rho \overline{v_3' T'}.$$

Upper indexes "+" and "-" designate values of functions at x3=h and x3=0 correspondingly. It is assumed that heat and mass exchange of fire front and boundary layer of atmosphere are governed by Newton law and written using the formulas:

$$(q_T^- - q_T^+)/h = -\alpha(T - T_e)/h,$$
$$(J_\alpha^- - J_\alpha^+)/h = -\alpha(c - c_{\alpha e})/hc_p.$$

To define source terms which characterize inflow (outflow of mass) in a volume unit of the gas-dispersed phase, the following formulae were used for the rate of formulation of the gas-dispersed mixture $\dot{m}$, outflow of oxygen R_{51}, changing carbon monoxide R_{52}.

$$Q = (1-\alpha_c)R_1 + R_2 + \frac{M_c}{M_1}R_3, R_{51} = -R_3 - \frac{M_1}{2M_2}R_5,$$
$$R_{52} = \nu_g(1-\alpha_c)R_1 - R_5, R_{53} = 0.$$

Here ν_g – mass fraction of gas combustible products of pyrolysis, α_4 and α_5 – empirical constants. Reaction rates of these various contributions (pyrolysis, evaporation, combustion of coke and volatile combustible products of pyrolysis) are approximated by Arrhenius laws whose parameters (pre-exponential constant k_i and activation energy E_i) are evaluated using data for mathematical models [9,11].

$$R_1 = k_1\rho_1\varphi_1 \exp\left(-\frac{E_1}{RT_s}\right), R_2 = k_2\rho_2\varphi_2 T_s^{-0.5} \exp\left(-\frac{E_2}{RT_s}\right),$$
$$R_3 = k_3\rho\varphi_3 s_\sigma c_1 \exp\left(-\frac{E_3}{RT_s}\right), R_5 = k_5 M_2 \left(\frac{c_1 M}{M_1}\right)^{0.25} \frac{c_2 M}{M_2} T^{-2.25} \exp\left(-\frac{E_5}{RT}\right).$$

The initial values for volume of fractions of condensed phases are determined using the expressions:

$$\varphi_{1e} = \frac{d(1-\nu_z)}{\rho_1}, \varphi_{2e} = \frac{Wd}{\rho_2}, \varphi_{3e} = \frac{\alpha_c\varphi_{1e}\rho_1}{\rho_3}$$

where d -bulk density for surface layer, ν_z – coefficient of ashes of forest fuel, W – forest fuel moisture content.

 ISBN 13: 978-84-613-5510-5

It is supposed that the optical properties of a medium are independent of radiation wavelength (the assumption that the medium is "grey"), and the so-called diffusion approximation for radiation flux density were used for a mathematical description of radiation transport during forest fires.

To close the system (1)–(7), the components of the tensor of turbulent stresses, and the turbulent heat and mass fluxes are determined using the local-equilibrium model of turbulence (Grishin, [9]). The system of equations (1)–(7) contains terms associated with turbulent diffusion, thermal conduction, and convection, and needs to be closed. The components of the tensor of turbulent stresses $\rho\overline{v_i'v_j'}$, as well as the turbulent fluxes of heat and mass $\overline{\rho v_j' c_p T'}$, $\overline{\rho v_j' c_\alpha'}$ are written in terms of the gradients of the average flow properties using the formulas

$$-\rho\overline{v_i'v_j'} = \mu_t\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)-\frac{2}{3}K\delta_{ij},$$

$$-\rho\overline{v_j\,c_p\,T'} = \lambda_t\frac{\partial T}{\partial x_j},\ -\rho\overline{v_j\,c_\alpha'} = \rho D_t\frac{\partial c_\alpha}{\partial x_j},$$

$$\lambda_t = \mu_t c_p/\mathrm{Pr}_t,\ \rho D_t = \mu_t/Sc_t,\ \mu_t = c_\mu\rho K^2/\varepsilon,$$

where μ_t, λ_t, D_t are the coefficients of turbulent viscosity, thermal conductivity, and diffusion, respectively; Pr_t, Sc_t are the turbulent Prandtl and Schmidt numbers, which were assumed to be equal to 1. In dimensional form, the coefficient of dynamic turbulent viscosity is determined using local equilibrium model of turbulence [9]. The thermodynamic, thermophysical and structural characteristics correspond to the forest fuels in the canopy of a different (for example pine [9,11,13]) type of forest. The system of equations (1)–(7) must be solved taking into account the initial and boundary conditions. The thermodynamic, thermophysical and structural characteristics correspond to the forest fuels in the canopy of a different type of forest; for example, pine forest (Grishin, Perminov [11]).

3. Numerical methods and results

The boundary-value problem (1)–(13) is solved numerically using the method of splitting according to physical processes (Perminov [11]). In the first stage, the hydrodynamic pattern of flow and distribution of scalar functions was calculated. The system of ordinary differential equations of chemical kinetics obtained as a result of splitting was then integrated. A discrete analog was obtained by means of the control volume method using the SIMPLE like algorithm (Patankar [16]) The accuracy of the program was checked by the method of inserted analytical solutions. Analytical expressions for the unknown functions were substituted in (1)–(7) and the closure of the equations were calculated. This was then treated as the source in each equation. Next, with the aid of the algorithm described above,

 ISBN 13: 978-84-613-5510-5

the values of the functions used were inferred with an accuracy of not less than 1%. The effect of the dimensions of the control volumes on the solution was studied by diminishing them. The time step was selected automatically. Fields of temperature, velocity, component mass fractions, and volume fractions of phases were obtained numerically. The distribution of basic functions shows that the process of crown forest fire initiation goes through the next stages. The first stage is related to increasing maximum temperature in the ground cover with the result that a surface fire source appears. At this process stage over the fire source a thermal wind is formed a zone of heated forest fire pyrolysis products which are mixed with air, float up and penetrate into the crowns of trees. As a result, forest fuels in the tree crowns are heated, moisture evaporates and gaseous and dispersed pyrolysis products are generated. Ignition of gaseous pyrolysis products of the ground cover occurs at the next stage, and that of gaseous pyrolysis products in the forest canopy occurs at the last stage. As a result of heating of forest fuel elements of crown, moisture evaporates, and pyrolysis occurs accompanied by the release of gaseous products, which then ignite and burn away in the forest canopy. At the moment of ignition the gas combustible products of pyrolysis burns away, and the concentration of oxygen is rapidly reduced. The temperatures of both phases reach a maximum value at the point of ignition. The ignition processes is of a gas - phase nature. Note also that the transfer of energy from the fire source takes place due to radiation; the value of radiation heat flux density is small compared to that of the convective heat flux. In the vicinity of the source of heat and mass release, heated air masses and products of pyrolysis and combustion float up. At $V_e \neq 0$, the wind field in the forest canopy interacts with the gas-jet obstacle that forms from the surface forest fire source and from the ignited forest canopy base and burn away in the forest canopy. In the vicinity of the source of heat and mass release, heated air masses and products of pyrolysis and combustion float up. At $V_e \neq 0$, the wind field in the forest canopy interacts with the gas-jet obstacle that forms from the surface forest fire source and from the ignited forest canopy base. On the windward side the movement of the air flowing past the ignition region accelerates. Figures 2,3 present the distribution of temperature $\overline{T}$ ($\overline{T} = T / T_e, T_e = 300\,K$) (1- 5., 2 – 4.5, 3 – 4, 4 – 3.5) for gas phase. Figures 4-5 present mass concentrations of oxygen $\overline{c}_1$ (1 – 0.5, 2 – 0.7, 3 – 0.8) and volatile combustible products of pyrolysis $\overline{c}_2$ concentrations (1 – 0.05, 2- 0.1, 3 – 0.5) ($\overline{c}_\alpha = c_\alpha / c_{1e}, c_{1e} = 0.23$) for wind velocity V_e= 10 m/s: and a) t=3 sec., b) t=5 sec. We can note that the isotherms is moved in the forest canopy and deformed by the action of wind. Similarly, the fields of component concentrations are deformed. It is concluded that the forest fire begins to spread.

 ISBN 13: 978-84-613-5510-5

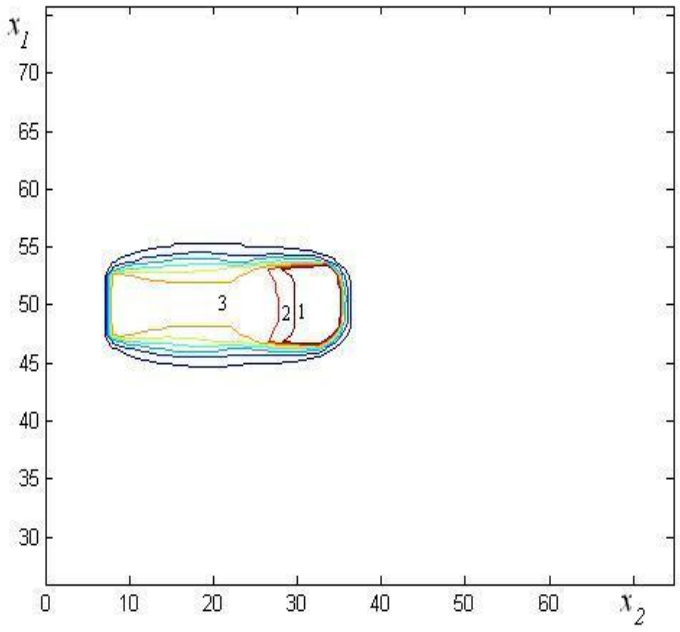


Figure. 2

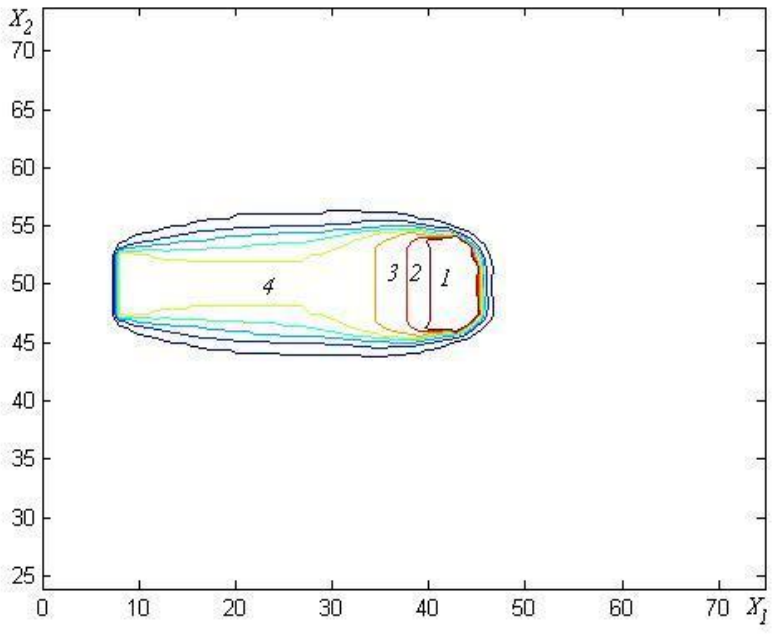


Figure 3

 ISBN 13: 978-84-613-5510-5

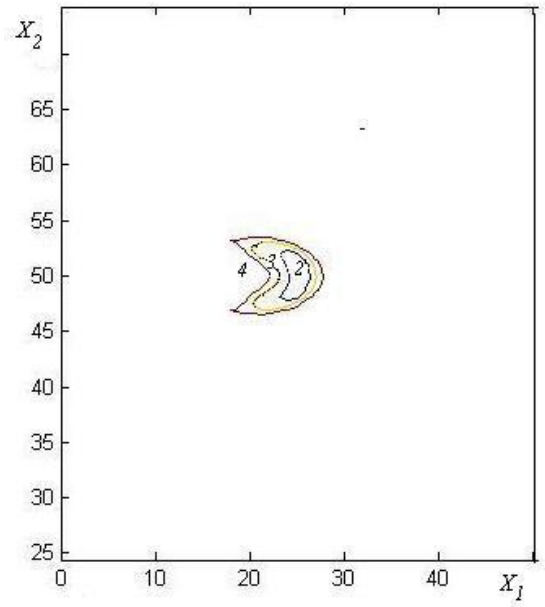


Figure 4

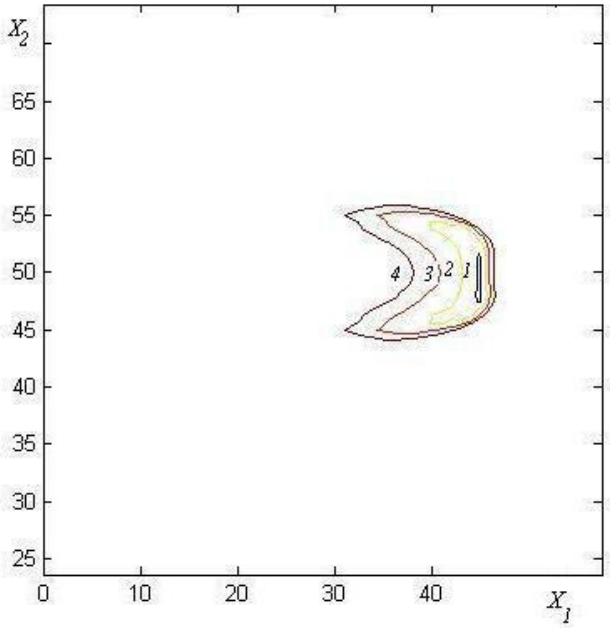


Figure 5

 ISBN 13: 978-84-613-5510-5

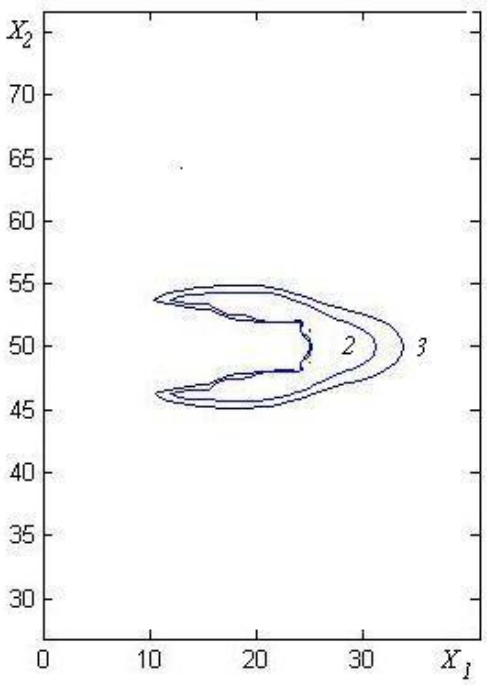


Figure 6.

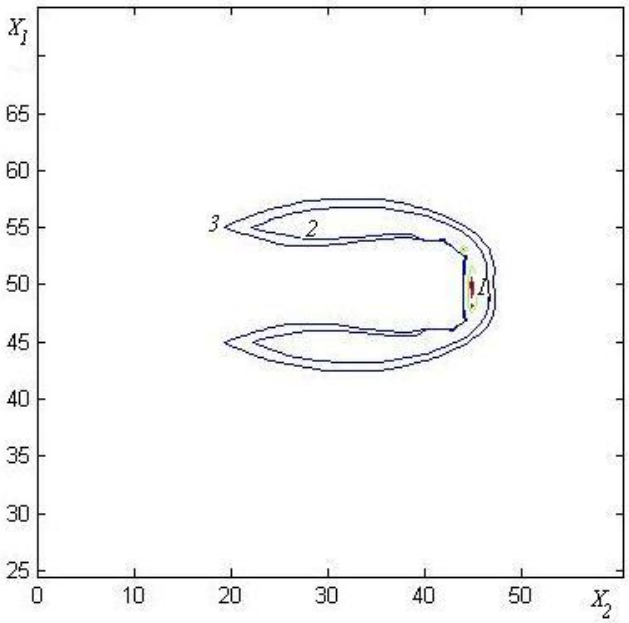


Figure 7

 ISBN 13: 978-84-613-5510-5

4. Conclusion

The results of calculation give an opportunity to evaluate critical condition of the forest fire spread, which allows applying the given model for preventing fires. It overestimates the velocity of crown forest fire spread that depends on crown properties: bulk density, moisture content of forest fuel and etc. The model proposed there give a detailed picture of the change in the velocity, temperature and component concentration fields with time, and determine as well as the influence of different conditions on the crown forest fire initiation. The results obtained agree with the laws of physics and experimental data (Grishin [9]; Konev [13]). From an analysis of calculations and experimental data it was found that for the cases in question the minimum total incendiary heat pulse is 2600 kJ/m^2 (Grishin [9]).· Calculations demonstrated that the value of the radiant heat flux for both problems is considerably less than the convective one , therefore radiation has a weak effect on local and integral characteristics of the problem discoursed above. The results obtained agree with the laws of physics and experimental data.

5. References

[1] C.E. VAN WAGNER, *Conditions for the start and spread of crown fire.* Canadian Journal of Forest Research. **7 (**1977) 23-34.

[2] M.E. ALEXANDER, *Crown fire thresholds in exotic pine plantations of Australasia.* PhD thesis, Department of Forestry, Australian National University, 1998.

[3] C.E. VAN WAGNER, *Prediction of crown fire behavior in conifer stands.* In '10th conference on fire and forest meteorology'. Ottawa, Ontario. (Eds D. C. MacIver, H. Auld and R. Whitewood), (1989) 207-212.

[4] G. XANTHOPOULOS, *Development of a wildland crown fire initiation model.* PhD thesis, University of Montana, 1990.

[5] R.C. ROTHERMEL, *Crown fire analysis and interpretation.* In 11th International conference on fire and forest meteorology. Missoula, Montana, USA, 1991.

[6] M.G. CRUZ, et al, *Predicting crown fire behavior to support forest fire management decision-making.* In *'IV International conference on forest fire research'*. Luso-Coimbra, Portugal. (Ed. D. X. Viegas), 11 [CD-ROM]. (Millpress) 2002.

[7] F.A. ALBINI, et al, *Modeling ignition and burning rate of large woody natural fuels.* Int. Journal of Wildland fire. **5** (1995) 81-91.

[8] J.H. SCOTT et al, *Assessing crown fire potential by linking models of surface and crown fire behavior*. USDA Forest Service, Rocky Mountain Forest and Range Experiment Station. Fort Collins: RMRS-RP-29, (Colorado, USA) 2001.

[9] A.M. GRISHIN, *Mathematical Modeling Forest Fire and New Methods Fighting Them*. Tomsk, Publishing House of Tomsk University, Russia. 1997.

 ISBN 13: 978-84-613-5510-5

[10] A.M. GRISHIN, V.A. PERMINOV, *Mathematical modeling of the ignition of tree crowns. Combustion, Explosion, and Shock Waves,* **34** (1998) 378-386.
[11] V.A. PERMINOV, *Mathematical Modeling of Crown and Mass Forest Fires Initiation With the Allowance for the Radiative - Convective Heat and Mass Transfer and Two Temperatures of Medium*, Ph.D Thesis, Tomsk State University, Russia, 1995.
[12] V.A. PERMINOV, *Mathematical modeling of crown forest fire initiation*. In 'III International conference on forest fire research and 14th conference on fire and forest meteorology'. Luso, Portugal. (Ed. D.X.Viegas), (1998) 419-431.
[13] E.V. KONEV, E.V., *The physical foundation of vegetative materials combustion.*. Novosibirsk, Nauka, Russia. 1977.
[14] D. MORVAN, J.L. DUPUY J.L., *Modeling of fire spread through a forest fuel bed using a multiphase formulation.* Combustion and Flame, **127** (2001) 1981-1994.
[15] D. MORVAN, J.L. DUPUY, *Modeling the propagation of wildfire through a Mediterranean shrub using a multiphase formulation,* Combustion and Flame, **138** (2004) 199-210.
[16] S.V. PATANKAR, *Numerical Heat Transfer and Fluid Flow*, New York, Hemisphere Publishing Corporation, 1981.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Methods for Accurate Motion Tracking and Motion Analysis of the Beating Heart Wall

Bernhard Quatember[1], Wolfgang Recheis, Martin Mayr,[1] Stefanos Demertzis[2], Giampietro Allasia[3], Roberto Cavoretto[3], Alessandra De Rossi[3], and Ezio Venturino[3]

[1] *Innsbruck Medical University (Radiology), Anichstrasse 35, 6020 Innsbruck, Austria*
[2] *Cardiocentro Ticino, Via Tesserete 48, 6900 Lugano, Switzerland*
[3] *Universita degli Studi di Torino (Matematica), Via Carlo Alberto 10, 10123 Torino, Italy*

Key words: Cardiac motion tracking, surface mesh generation, Gordon surface, registration, thin-plate spline transformation, radial basis function

Extended Abstract

1. Introduction

During the past decades, much effort has been devoted to achieving an improved understanding of the contractions of the heart. Numerous methods have been developed and applied to capture specific data from medical images which are then used to describe the motion of the heart. However, the accuracy of these methods is somewhat restricted. Thus, we aimed at an improved method of motion tracking. The pivotal point of our development efforts is the full exploitation of the synergy of the two imaging modalities, viz. cardiac CT and biplane cineangiography, for an accurate quantitative assessment of the regional variations of ventricular motility and to monitor the improvements during therapy. The extraordinarily high spatial and temporal resolution of biplane cineangiography facilitates accurate investigations of the ventricular

 ISBN 13: 978-84-613-5510-5

motility, which, however, are restricted to the regions of the left ventricular surface that are covered by the coronary arteries. Several methods to extend the tracking of the ventricular motion to the entire epicardial surface have been described in the literature. However, the shape of the epicardial surface was only an assumed one and thus not patient-specific. Our multimodal imaging approach to motion analysis is based on a retrospectively ECG-gated cardiac CT data set at the end of the diastole and a biplane cineangiogram of a particular patient. The utilization of an efficient combination of both imaging modalities enables us to exploit their specific favorable characteristics of both of them and to overcome their respective limitations. In particular, we fully take advantage of

- the capacity of cardiac CT to retrospectively reconstruct a highly-accurate 3D image of the epicardial surface at the end of the diastole and
- the excellent motion tracking capability of biplane angiography, in which there is no significant motion blur.

Our techniques comprise several image processing steps, such as segmentation, registration. In these techniques, the realization of specific transformations. moreover, mesh generatiom procedures are carried out. This paper is confined to the generation of a deformable mesh to represent the geometry of the outer surface of the myocardium of the left ventricle and its deformations during the cardiac cycle which are highly complex and difficult to describe quantitatively. Our long-term goal, however, is to exploit our mesh generation approach to develop methods to elucidate the spatial and temporal changes of strain and stress within the myocardium of the rapidly moving heart.

2. Image Processing and Mesh Generation Tasks

In the CT imagery for the end-diastolic position, we carry out a segmentation of the surface of the myocardium (cf. Figure 1). In each transaxial
slice, we define manually an appropriate number of vertices and a spline (NURBS) curve which approximates these vertices.
These splines are subdivided into equal increments. The subdivision points of all transaxial slices are regarded as nodes of a surface mesh. The nodes belonging to each transaxial slice are connected to each other with an interpolating spline (NURBS curve) which is called transversal contour curve. We thus obtain an array of NURBS curves as transversal

 ISBN 13: 978-84-613-5510-5

contour curves of the outer surface of the myocardium. We also compute an array of longitudinal interpolating NURBS curves, each of which passes through corresponding points in all slices. These two arrays of NURBS curves constitute a surface mesh which enables us to achieve a preliminary visual representation of the heart wall (myocardium, left ventricle) for the end-diastolic position in the form of a wireframe model.

3.

Fig. 1. Medical imagery relating to end-diastolic phase: (a)transaxial slice of CT with manually segmented ventricular wall; (b)+(c)biplane angiograms relating to end-diastolic frame of cineangiography

The points corresponding to the entire surface together with well chosen landmarks, constitute the domains of landmark-based thin-plate spline registration. Thin plate spline transformations are not only instrumental for the accomplishment of our image-registration but also for our motion tracking tasks . In the subsequent description of our mathematical concepts and techniques for the analysis of the heart wall motion, we assume the availability of the following for each patient:

- the above-described surface mesh calculated from three-dimensional CT describing the morphology at the end of the diastole, and
- A series of SEATs (skeletonised epicardial artery trees) which have been derived from biplane cineangiograms with the methods which will be described in the full paper. We have developed a method to achieve a largely automatic segmentation of the epicardial arteries in angiograms. In the first stage of our approach:we use a multi-scale Hessian filter to separate the tubular structure of the arteries from the other structures in the image, then we carry out a conversion of these tubular structures into a binary mask, and finally we apply a thinning filter to obtain an initial

 ISBN 13: 978-84-613-5510-5

representation of a skeletonised arterial tree The second stage of our efforts comprises a scan-line-based border detection procedure for an accurate segmentation of the arteries, te calculation of the centre lines in the angiograms, and the three-dimensional reconstruction of these centre lines resulting in space curves which show a tree-like structure. This tree-like arrangement of space curves is regarded as the final version of the skeletonised epicardial artery tree. It will serve as the basis for our registration and motion-tracking tasks. For further details of our segmentation method, please refer to [1].

Registration Tasks. As mentioned above, we carry out a landmark-based image registration with the help of our two imaging modalities, cardiac CT and biplane cineangiography.In particular, our end-diastolic three-dimensional CT data set is correlated with the end-diastolic representation of the heart in biplane cineangiography, our reference image modality. The cineangiographic imagery comprises a relatively large number of frames. One frame refers to the end-diastolic position of the heart. In handling the registration task, we refer to this frame and carry out a lanmark-based thin-plate spline transformation.

Motion Tracking Tasks. Our motion-tracking analyses are also based on a number of thin-plate spline transformations (ca. 60). In the following, we assume that the SEATs are three-dimensionally reconstructed for all frames which are linearly ordered in time). The individual transformations thereby relate to two consecutive frames. The tracking procedure starts with the end-diastolic frame and thus comprises about 60 individual TPS transformations.
Some results of our registration and motion-tracking procedures can be seen in Figures 2 and 3.

 ISBN 13: 978-84-613-5510-5

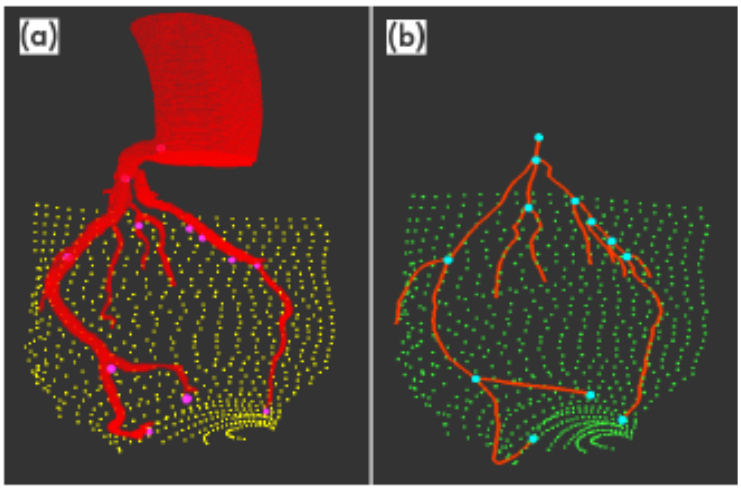


Fig. 2. Image Registration: (a)ventricular surface after segmentation in CT, point landmarks are indicated in magenta; (b)ventricular surface after registration to the SEAT belonging to the end-diastolic phase, point landmarks are indicated in cyan

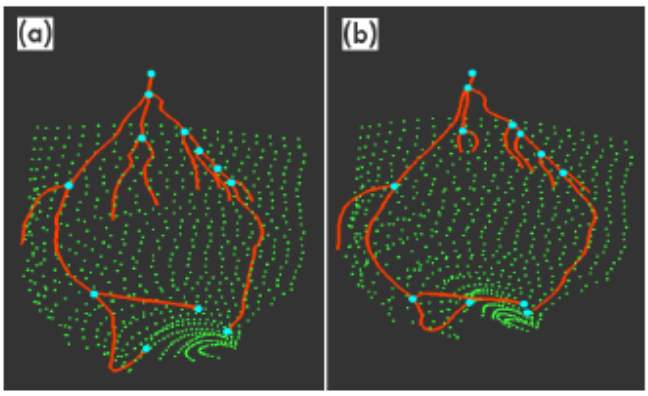


Fig. 3. Comparison between ventricular surface during end-diastolic phase and at the end of the systole: (a)end-diastolic phase, point landmarks are indicated in cyan; (b)end-systolic phase, point landmarks are indicated in cyan

 ISBN 13: 978-84-613-5510-5

3. Analysis and Visualization Aspects

At present, we are elaborating new visualisation and analysis techniques for an accurate quantitative assessment of the regional variations of heart wall motility. From each an every pint of the heart surface, we will be able to draw trajectories which refers to the entire contraction phase of the myocardium. The aforementioned corresponding nodes subdivide the trajectory into equal time intervals . For each and every subdivision point which follows the selected node (material point) in the end-diastolic frame, we plan to compute and to visualize _ the velocity and acceleration vectors and their variations what is even more important, several characteristic quantities of surface deformation kinematics, such as displacement tensors describing surface deformations, surface strain tensors, surface strain rate tensors, surface curvature tensors, and, moreover, tensors of changes of curvature. The underlying concepts of the above tensor quantities are based on differential geometry and thus permit a local analysis of the deformation behaviour. For this reason, these tensor quantities are well suited to reveal the regional variations of the deformation behavior of the heart wall. We plan to derive parameters of diagnosis from these tensors. Moreover, we aim at a visualization of the tensor quantities by using standard methods which we will adapt to our problem area.

4. References

[1] M. MAYR, B. QUATMBER, Development of a special method and a software system for the semi-automatic segmentation of biplane angiograms, in:J. Volkert, T. Fahringer, D. Kranzlmueller, W. Schreiner (eds.), Proceedings of the 2. Austrian Grid Symposium, Innsbruck, Austria, September 21 - 23, 2006, vol. 221 Oesterreichische Computer Gesellschaft, 2007, pp. 220-237.

Acknowledgements

The work described in this paper is partially supported by the ”Austrian GRID” project, funded by the Austrian BMBWK (Federal Ministry for Education, Science and Culture) under contract GZ 4003/2-VI/4c/2004./

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Mathematical Modelling of the Biological Pest Control of the Sugarcane Borer

Marat Rafikov[1] and Elizabeth de Holanda Limeira[1]

[1] *Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas, Universidade Federal do ABC, Santo André, SP, Brazil*

emails: marat9119@yahoo.com.br, bethmatcampinas@yahoo.com.br

Abstract

In this paper, we propose a simple mathematical model of interaction between the sugarcane borer *(Diatraea saccharalis)* and its egg parasitoid *Trichogramma galloi*. In this model the sugarcane borer is represented by the egg e larval stages, and the parasitoid is considered in term of the parasitized eggs. Linear feedback control strategy is proposed to indicate how the natural enemies should be introduced in the environment.

Key words: mathematical modelling, biological control, sugarcane borer, egg parasitoid
MSC2000: AMS 92D25, 34H05, 49N90

1. Introduction

The increase in world demand for ethanol brings an increase of the sugarcane planted in Brazil. The sugarcane borer *Diatraea sacharalis* is reported to be the most important sugarcane pest in south-east region of Brazil [1]. The sugarcane borer builds internal galleries in the sugarcane plants causing direct damages that result in apical bud death, weight loss and atrophy. Indirect damages occur when there is contamination by yeasts that cause red rot in the stalks, either causing contamination or inverting the sugar, increasing yield loss in both sugar and alcohol [2]. It is known that for each 1% of plant infestation by pests the industries lose 0.2% of the ethanol production, that is, in average 25 liters per ha.

One of challenges of the improvements in the farming and harvesting of cane is the biological pest control. A good strategy of biological pest control can increase the ethanol production. Biological control is the use of living organisms

 ISBN 13: 978-84-613-5510-5

to suppress pest populations, making them less abundant and thus less damaging than they would otherwise be [3]. Pests are species that interfere with human activity or cause injury, loss, or irritation to a crop, stored product, animal, or people. One of the main goals of the pest control is to maintain the density of the pest population in an equilibrium level below economic damages. Natural enemies play an important role in limiting potential pest populations. Flooding agroecosystems with parasitoid insects is very effective in lowering the abundance of crop pest insects. Thus, parasitoids are commonly reared in laboratories and periodically liberated in high-density populations as biological control agents of crop pests [4]. *Cotesia flavipes* is important wasp parasitoid of the sugarcane borer larvae in Brazil [1]. In spite of the biological control of *Diatraea saccharalis* by *Cotesia flavipes* is considered successful in Brazil, there are some areas where *Cotesia flavipes* has not the good control. The using of the egg parasitoid *Trichogramma galloi* is considered an interesting option in this case [5].

Thomas and Willis [6] state that introduction of biological agents against weeds and insects is a substantially effective control is less than 40% of the cases. In order for biological control to succeed, the dynamics of the pest and its enemy populations have to be understood. Mathematical modeling is an important tool used in studying agricultural problems. Mathematical modeling applied to the problems of biological pest control allows a qualitative and quantitative evaluation of the impact between the pest and its natural enemy populations. The application of host-parasitoid models for biological control were reviewed in [7].

In this paper, we propose a simple mathematical model of interaction between the sugarcane borer *(Diatraea saccharalis)* and its egg parasitoid *Trichogramma galloi*. In this model the sugarcane borer is represented by the egg e larval stages, and the parasitoid is considered in term of the parasitized eggs. Linear feedback control strategy is proposed to indicate how the natural enemies should be introduced in the environment.

2. Mathematical model of interactions between the sugarcane borer and its parasitoid

Consider two of main stages of development of the sugarcane borer *Diatraea sacharalis* – the egg e larval stages. We assume that there exists only an egg parasitoid (*Trichogramma galloi*) in a common environment. Assuming furthermore logistic growth for the egg population we can propose the following mathematical model that describes interactions between the sugarcane borer and its parasitoid:

 ISBN 13: 978-84-613-5510-5

$$\begin{aligned}
\frac{dx_1}{dt} &= \beta(1-\frac{x_1}{K})x_1 - m_1x_1 - n_1x_1 - \alpha\, x_1x_2 \\
\frac{dx_2}{dt} &= \alpha\, x_1x_2 - m_2x_2 - n_2x_2 \\
\frac{dx_3}{dt} &= n_1x_1 - m_3x_3 - n_3x_3
\end{aligned} \tag{1}$$

were x_1 is the egg density of the sugarcane borer, x_2 is the density of eggs parasitized by *Trichogramma galloi* and x_3 is the larvae density of the sugarcane borer; β is the net reproduction rate; *K* is the carrying capacity the environment; m_1, m_2 and m_3 are mortality rates of the egg, parasitized egg and larvae populations; n_1 is the fraction of the eggs from which the larvae emerge at time *t*; n_2 is the fraction of the parasitized eggs from which the adult parasitoids emerge at time *t*; n_3 is the fraction of the larvae population which moults into pupal stage at time *t*; α is the rate of parasitism.

3. Equilibrium points and stability

The equilibrium points can be obtained by setting to zero the right hand sides of (1):

$$\begin{aligned}
&x_1^*(\beta - \frac{\beta}{K}x_1^* - m_1 - n_1 - \alpha\, x_2^*) = 0 \\
&x_2^*(\alpha\, x_1^* - m_2 - n_2) = 0 \\
&n_1x_1^* - x_3^*(m_3 + n_3) = 0
\end{aligned} \tag{2}$$

We obtain therefore the following points:

$$\begin{aligned}
P_1 &= (0,0,0) \\
P_2 &= \left(\frac{K}{\beta}(\beta - m_1 - n_1),\, 0,\, \frac{n_1K(\beta - m_1 - n_1)}{\beta(m_3+n_3)}\right) \\
P_3 &= \left(\frac{m_2+n_2}{\alpha},\, \frac{\beta}{\alpha} - \frac{\beta}{\alpha^2K}(m_2+n_2) - \frac{m_1+n_1}{\alpha},\, \frac{n_1(m_2+n_2)}{\alpha(m_3+n_3)}\right)
\end{aligned} \tag{3}$$

For P_2 the condition

$$\beta > m_1 + n_1 \tag{4}$$

 ISBN 13: 978-84-613-5510-5

ensures the nonnegativity of the larvae population.
For P_3 the condition of the nonnegativity of the parasitized egg population is

$$\beta > m_1 + n_1 + \frac{\beta}{\alpha K}(m_2 + n_2) \tag{5}$$

Let us consider the Jacobian matrix of the system (1)

$$J = \begin{bmatrix} \beta - \frac{2\beta}{K} x_1^* - m_1 - n_1 - \alpha x_2^* & -\alpha x_1^* & 0 \\ \alpha x_2^* & \alpha x_1^* - m_2 - n_2 & 0 \\ n_1 & 0 & -m_3 - n_3 \end{bmatrix} \tag{6}$$

Stability analysis of the equilibrium point P_1
In this case the Jacobian matrix assumes the form

$$J = \begin{bmatrix} \beta - m_1 - n_1 & 0 & 0 \\ 0 & -m_2 - n_2 & 0 \\ n_1 & 0 & -m_3 - n_3 \end{bmatrix} \tag{7}$$

from which the eigenvalues are easily found to be

$$\lambda_1 = \beta - m_1 - n_1,\ \lambda_2 = -m_2 - n_2,\ \lambda_3 = -m_3 - n_3.$$

It follows then that equilibrium point P_1 is stable if

$$\beta < m_1 + n_1. \tag{8}$$

Stability analysis of the equilibrium point P_2
In this case the Jacobian matrix assumes the form

$$J = \begin{bmatrix} -\frac{\beta}{K} x_1^* & -\alpha x_1^* & 0 \\ 0 & \alpha x_1^* - m_2 - n_2 & 0 \\ n_1 & 0 & -m_3 - n_3 \end{bmatrix}$$

and the eigenvalues are

 ISBN 13: 978-84-613-5510-5

$\lambda_1 = -(\beta - m_1 - n_1)$, $\lambda_2 = \frac{\alpha K}{\beta}(\beta - m_1 - n_1) - m_2 - n_2$, $\lambda_3 = -m_3 - n_3$.

It follows then that equilibrium point P_2 is stable if

$$m_1 + n_1 < \beta < m_1 + n_1 + \frac{\beta}{\alpha K}(m_2 + n_2). \tag{9}$$

Stability analysis of the equilibrium point P_3

In this case the Jacobian matrix has the following form

$$J = \begin{bmatrix} -\frac{\beta}{K}x_1^* & -\alpha x_1^* & 0 \\ \alpha x_2^* & \alpha x_1^* - m_2 - n_2 & 0 \\ n_1 & 0 & -m_3 - n_3 \end{bmatrix} \tag{10}$$

with the characteristic equation

$$(\lambda^2 + a_1\lambda + a_2)(\lambda + a_3) = 0 \tag{11}$$

where the coefficients are given by

$$a_1 = \frac{\beta(m_2 + n_2)}{\alpha K} > 0$$

$$a_2 = \alpha^2 x_1^* x_2^*$$

$$a_3 = m_3 + n_3 > 0$$

According to Routh-Hurwitz condition the eigenvalues of the equation $\lambda^2 + a_1\lambda + a_2 = 0$ have negative real parts if $a_1 > 0$ and $a_2 > 0$. The coefficient $a_2 > 0$ if $x_2^* > 0$, i.e. if the condition (5) is satisfied. It follows then that equilibrium point P_3 is stable if

$$\beta > m_1 + n_1 + \frac{\beta}{\alpha K}(m_2 + n_2) \tag{12}$$

 ISBN 13: 978-84-613-5510-5

4. Numerical simulations of the host-parasitoid interactions without control

For numerical simulations of interactions between the sugarcane borer and its parasitoid were used the following values of model coefficients: $n_1 = 0.1$, $n_2 = 0.1$, $n_3 = 0.02439$, $m_1 = 0.03566$, $m_2 = 0.03566$, $m_3 = 0.00256$, $K = 25000$. These values were obtained based on data published about the use of the egg parasitoid *Trichogramma galloi* against the sugarcane borer *Diatraea saccharalis*these [1], [5], [8].

The value of the parameter β is important for determination the stability of the equilibrium points. When β satisfies the condition $\beta < m_1 + n_1$, the equilibrium P_1 is stable and other points are unstable. Fig.1 shows that for $\beta = 0.13$ and $\alpha = 0.0001723$ all populations go to extinction in this case.

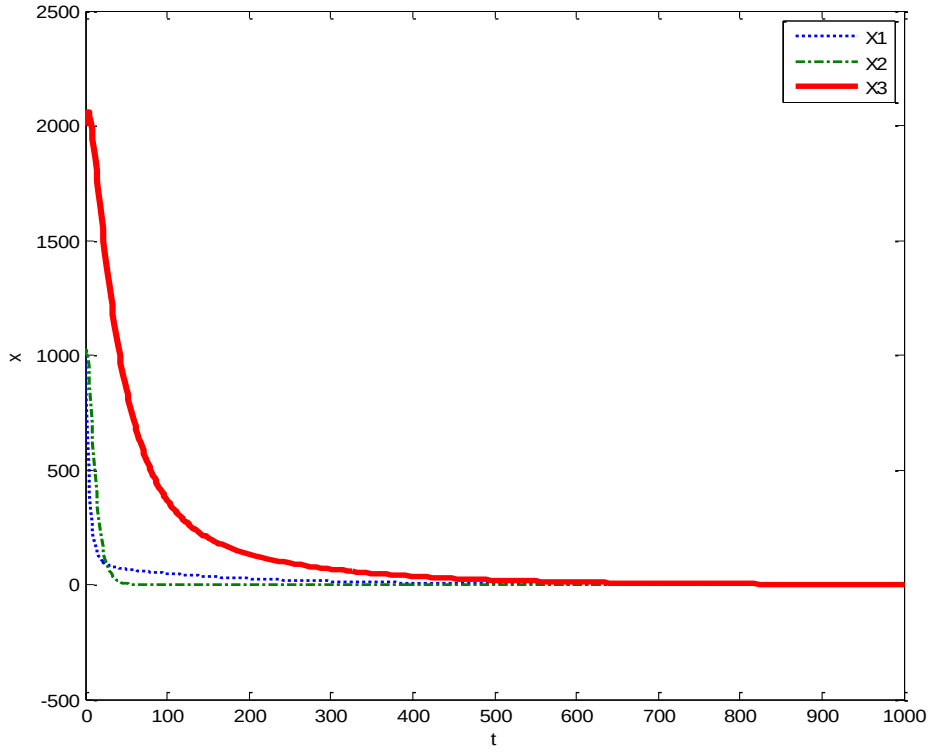


Fig. 1. Evolution of the egg, parasitized egg and larvae populations for $\beta = 0.13$ and $\alpha = 0.0001723$

When β satisfies the condition (9):

 ISBN 13: 978-84-613-5510-5

$$m_1 + n_1 < \beta < m_1 + n_1 + \frac{\beta}{\alpha K}(m_2 + n_2)$$

the equilibrium P_2 is stable and other points are unstable. In this case, the parasitized egg population goes to extinction, and the egg e larvae populations go to positive equilibrium levels. This case is shown in Fig.2 for $\beta = 0.139$ and $\alpha = 0.0001723$

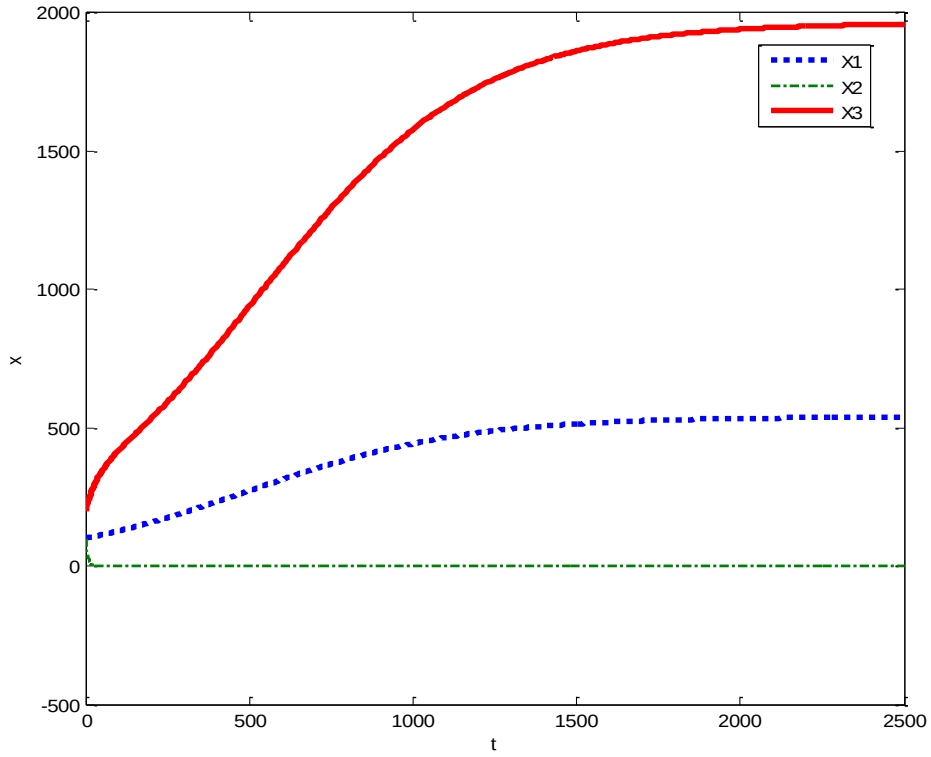


Fig. 2. Evolution of the egg, parasitized egg and larvae populations for $\beta = 0.139$ and $\alpha = 0.0001723$

When β satisfies the condition (12):

$$\beta > m_1 + n_1 + \frac{\beta}{\alpha K}(m_2 + n_2)$$

the positive equilibrium point P_3 is stable and the populations coexist in a common environment. Fig. 3 shows the population oscillations for $\beta = 0.1908$ and $\alpha = 0.0001723$.

 ISBN 13: 978-84-613-5510-5

One can see from Fig. 3 that the sugarcane borer larvae density x_3 takes on values more than the pest density threshold level $x_d = 2500$ numbers/ha [1]. Densities above this level cause economic damages the sugarcane crops. In this case, it is necessary to apply the biological control.

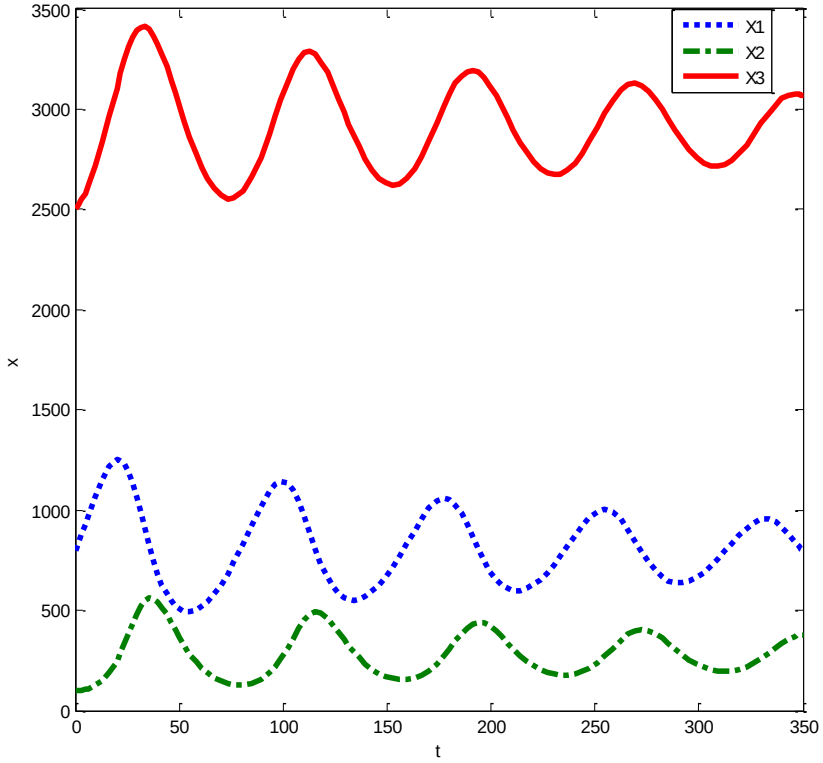


Fig.3. Evolution of the egg, parasitized egg and larvae populations for $\beta = 0.1908$ and $\alpha = 0.0001723$

5. Numerical simulations of the inundative biological control

The main objective of the biological pest control is to maintain the pest population in an equilibrium level below the economic injury level. Thus, parasitoids and predators are commonly reared in laboratories and periodically liberated in high-density populations (inundative biological control) when the pest population reaches a control level [4]. Mathematically, the inundative control can be interpreted by impulsive control function *U* that produces the discontinuous augmentation of the natural enemy population (parasitized egg). The Fig. 4 shows the inundative control applied in initial moment by introduction 20000 parasitoids/ha. From Fig. 4 one can see that the inundative biological control,

 ISBN 13: 978-84-613-5510-5

applied in initial moment by introduction 2000 parasitized egg /ha, maintain the pest population below the value $x_c = 2500$ pests/ha only 88 days. After this period, it is necessary to apply the control again. Another negative factor of the inundative biological control is the high amplitudes of oscillation.

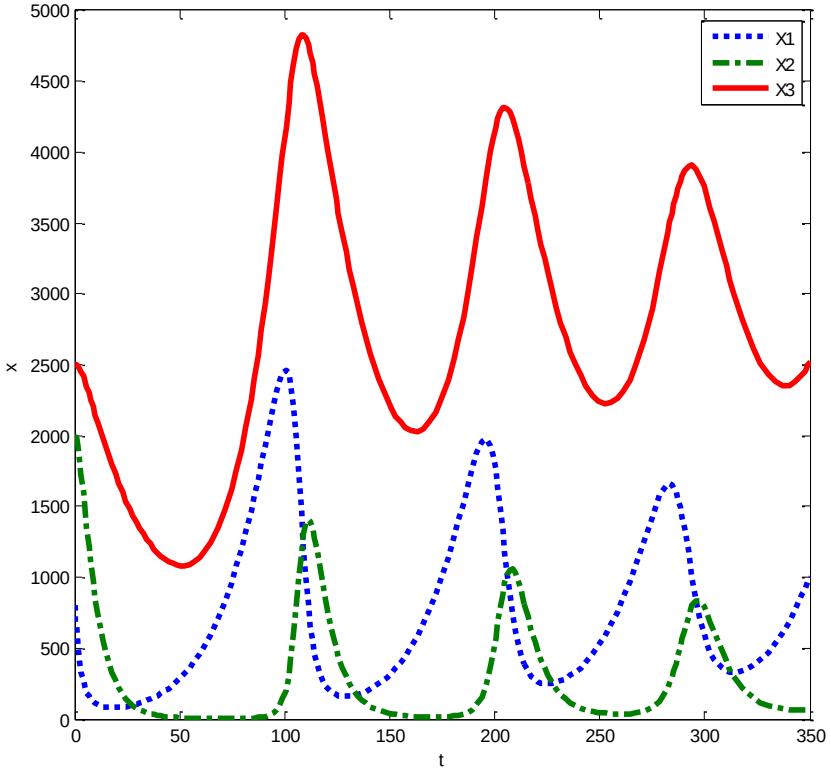


Fig.4. Inundative control application by introduction 2000 parasitoids/ha

6. Optimization of the biological control

We hope to formulate the pest control strategy of the sugarcane borer through the parasitized egg introduction in a common environment. This control must move the controlled system to the steady state in that the larvae density is stabilized without causing economic damages, and that the parasitized egg population is stabilized at the level enough to control the pests.

The dynamic system (1) with control has the following form:

 ISBN 13: 978-84-613-5510-5

$$\begin{aligned}
\frac{dx_1}{dt} &= \beta(1-\frac{x_1}{K})x_1 - m_1x_1 - n_1x_1 - \alpha x_1x_2 \\
\frac{dx_2}{dt} &= \alpha x_1x_2 - m_2x_2 - n_2x_2 + U \\
\frac{dx_3}{dt} &= n_1x_1 - m_3x_3 - n_3x_3
\end{aligned} \tag{13}$$

The goal of the pest control strategy maintains the larvae population at level $x_3^* = x_d < x_c$ by control u^*, where x_d is a designed pest population density below economic injury level. The desired positive steady state with control satisfies the following equations

$$\begin{aligned}
&x_1^*(\beta - \frac{\beta}{K}x_1^* - m_1 - n_1 - \alpha x_2^*) = 0 \\
&x_2^*(\alpha x_1^* - m_2 - n_2) + u^* = 0 \\
&n_1x_1^* - x_3^*(m_3 + n_3) = 0
\end{aligned} \tag{14}$$

From the third equation of the system (14) we obtain the egg density value which is necessary to maintain the larvae population at level $x_3^* = x_d$:

$$x_1^* = \frac{(m_3 + n_3)x_3^*}{n_1} \tag{15}$$

From the first equation of the system (14) we obtain the parasitized egg density value which is necessary to maintain the larvae population at level $x_3^* = x_d$:

$$x_2^* = \frac{\beta(1 - x_1^*/K) - m_1 - n_1}{\alpha} \tag{16}$$

From the second equation of the system (14) we obtain the value of the control u^*:

$$u^* = -x_2^*(\alpha x_1^* - m_2 - n_2) \tag{17}$$

In the general case, the desired steady-state (x_1^*, x_2^*, x_d) of the system (13) controlled by u^* can be unstable. In this case the feedback control *u* can be made so that the desired state becomes asymptotically stable.
Defining the following new variables

 ISBN 13: 978-84-613-5510-5

$$y = \begin{bmatrix} x_1 - x_1^* \\ x_2 - x_2^* \\ x_3 - x_d \end{bmatrix}, \; u = U - u^* \tag{18}$$

and substituting (18) into (13) and admitting (14), we get the following error system:

$$\dot{y} = A\,y + h(y) + Bu \tag{19}$$

where the matrices *A* and *B* are

$$A = \begin{bmatrix} \beta - \dfrac{2\beta x_1^*}{K} - m_1 - n_1 - \alpha\, x_2^* & -\alpha\, x_1^* & 0 \\ \alpha\, x_2^* & \alpha\, x_1^* - m_2 - n_2 & 0 \\ n_1 & 0 & -m_3 - n_3 \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{20}$$

and the vector *h(y)* has a form:

$$h(y) = \begin{bmatrix} -\dfrac{\beta}{K} y_1^2 - \alpha\, y_1 y_2 \\ \alpha\, y_1 y_2 \\ 0 \end{bmatrix} \tag{21}$$

The feedback control *u* can be determinate applying the following theorem.

Theorem [9]**.** If there exist constant matrices *Q*, and *R*, positive definite, being *Q* symmetric, such as that the function

$$l(y) = y^T Q\, y - h^T(y) P\, y - y^T P h(y), \tag{22}$$

is positive definite then the linear feedback control

$$u = -R^{-1} B^T P(t)\, y \tag{23}$$

is optimal, in order to transfer the nonlinear system (19) from an initial to final state

$$y(\infty) = 0 \tag{24}$$

minimizing the functional

 ISBN 13: 978-84-613-5510-5

$$J = \int_0^\infty [l(y) + u^T R u] dt \tag{25}$$

where P the symmetric, positive definite matrix, is the solution of the matrix algebraic Riccati equation

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \tag{26}$$

In addition, with the feedback control (23), there exists a neighborhood $\Gamma_0 \subset \Gamma$, $\Gamma \subset \Re^n$, of the origin such that if $y_0 \in \Gamma_0$, the solution $y(t) = 0,\ t \geq 0,$ of the controlled system (19) is locally asymptotically stable, and $J_{\min} = y_0^T P(0) y_0$. Finally, if $\Gamma = \Re^n$ then the solution $y(t) = 0,\ t \geq 0,$ of the controlled system (19) is globally asymptotically stable.

From theorem one can conclude that if the function (22) is positive definite then the error dynamical system (19) controlled by linear feedback control *u* is asymptotically stable, and hence, the system (13), controlled by

$$U = u^* + u, \tag{27}$$

tends to the desired steady state (x_1^*, x_2^*, x_d).

We illustrate the application of the optimal pest control strategy (27) on the agroecosystem which consisting of sugarcane borer and its parasitoid. We will stabilize the ecosystem (13) at the desired steady state $x_1^* = 549.2$ egg/ha, $x_2^* = 293.67$ parasitized egg/ha, $x_3^* = x_d = 2000$ larvae/ha. The values of x_1^* and x_2^* were calculated from (15) and (16), respectively. In this case, $u^* = 12.15$ parasitized egg/day, and the matrices *A* and *B* have the following form

$$A = \begin{bmatrix} -0.0042 & -0.0946 & 0 \\ 0.0506 & -0.0414 & 0 \\ 0.1 & 0 & -0.0275 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Choosing

$$Q = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}, \ R = [1]$$

we obtain

 ISBN 13: 978-84-613-5510-5

$$P = \begin{bmatrix} 0.3045 & -0.134 & 0.1202 \\ -0.134 & 0.1511 & -0.0517 \\ 0.1202 & -0.0517 & 0.1334 \end{bmatrix}$$

from the solution of the Riccati equation (26).

Finally, we can conclude that the optimal strategy has the following form:

$$U = 12.15 + 0.134\, y_1 - 0.1511\, y_2 + 0.0517\, y_3 \qquad (28)$$

The optimal control (28) is designed to drive the trajectory of the system (13) to desired steady state (x_1^*, x_2^*, x_d), as shown in Fig. 5. Dynamics of the optimal control function (28) is presented in Fig. 6.

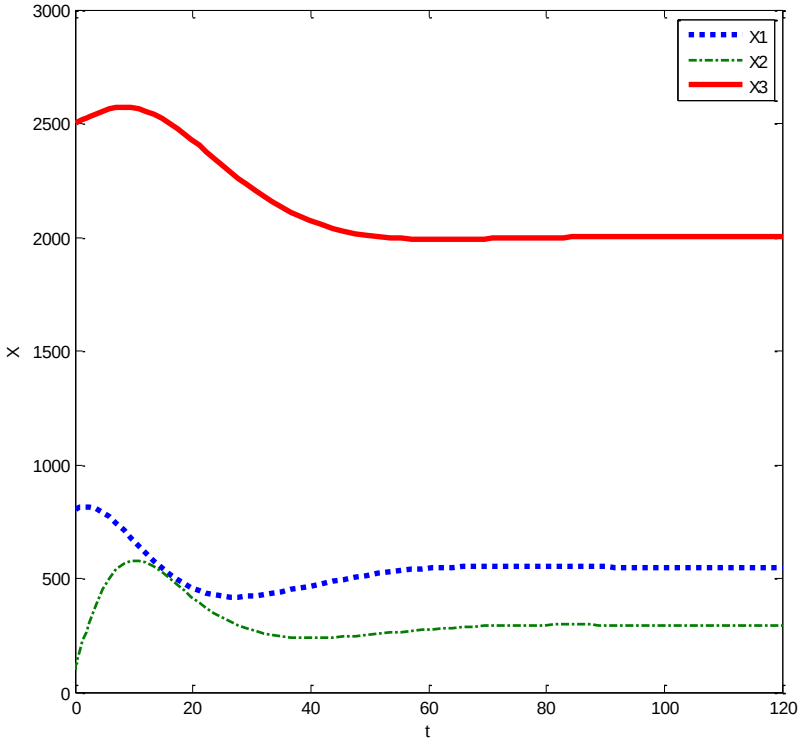


Fig. 5. Evolution of the dynamic system (13) with optimal control

 ISBN 13: 978-84-613-5510-5

Numerical simulations showed that the function *l(y)*, defined by (22), was positive for all considered initial condition values, but it is necessary more investigations to prove if this function is positive definite at a positive space.

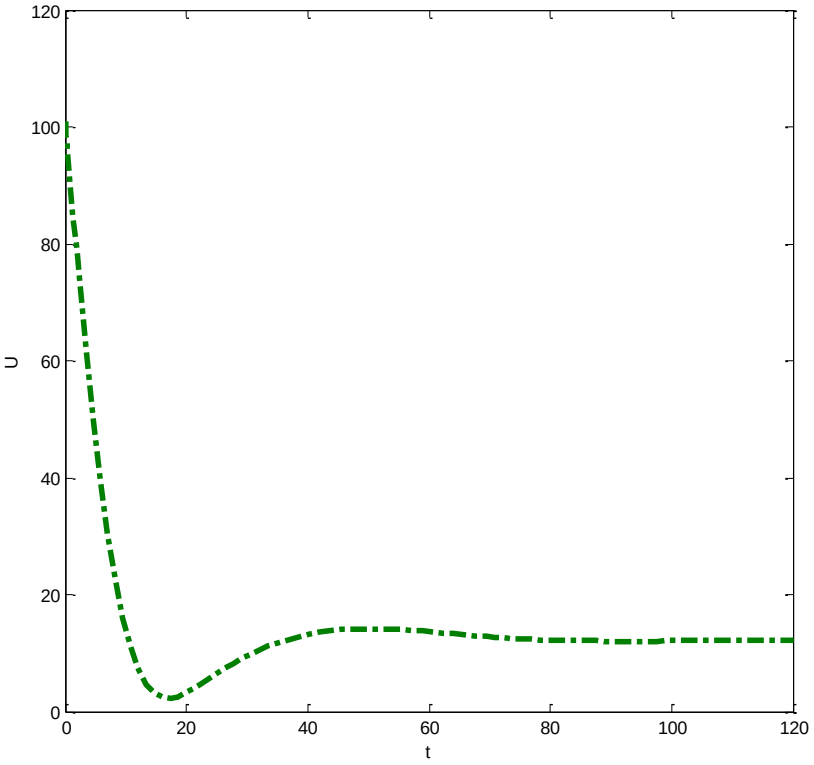


Fig.6. Dynamics of the optimal control strategy

7. Conclusion

Fig. 6 shows that the great amount of parasitoid have to be introduced in initial days. This fact suggests that the proposed feedback control strategy can be integrated into existing biological control technologies, combining the feedback control with the traditional inundative pest control. This control strategy directs the ecosystem to the stable equilibrium point which is reached at 40 days. After this period, according above proposed control strategy, it is necessary to apply the constant control $u^* = 12.15$ parasitized egg/day. It is not economically advantageous to use this constant control. In agricultural practice this control can be substituted by periodic releases of a small population of natural enemies. It is necessary more studies to justify this substitution.

 ISBN 13: 978-84-613-5510-5

Acknowledgments

The authors thank Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) and Conselho Nacional de Pesquisas (CNPq) for the financial supports on this research.

References

[1] J.R.P. PARRA, P.S.M. BOTELHO, B.S.C. FERREIRA, J.M.S. BENTO, *Controle Biológico no Brasil: Parasitóides e Predadores*, São Paulo, Editora Manole, 2002.

[2] N. MACEDO AND P.S.M. BOTELHO, *Controle integrado da broca da cana-de-açúcar, Diatraea saccharalis (Fabr., 1794)(Lepidoptera:Pyralidae)*, Brasil Açucar. **106** (1988) 2-14.

[3] P. DEBACH, *Biological control by natural enemies*, Cambridge, Cambridge University Press, 1974.

[4] H.J. BARCLAY, I. S. OTVOS AND A.J. THOMSON, *Models of periodic inundation of parasitoids for pest control*, Can. Ent. **117**(1970)705-716.

[5] P.S.M. BOTELHO, J.R.P. PARRA, J.F. CHAGAS NETO, C.P.B OLIVEIRA, *Associação do parasitóide de ovos Trichogramma galloi Zucchi (Himenóptera: Trichogrammatidae) e do parasitóide larval Cotesia flavipes (Cam.)(Himenóptera: Brancconidae) no controle de Diatraea saccharalis (Fabr., 1794)(Lepidoptera:Pyralidae) em cana-de-açúcar*, Anais da Sociedade Entomologica do Brasil, 28(3)(1970)491-496.

[6] M.B. THOMAS AND A.J. WILLIS, *Biocontrol - risky but necessary*, Trends in Ecology and Evolution, **13**(1998)325-329.

[7] N.J. MILLS AND W.M. GETZ, *Modeling the biological control of insect pest: review of host–parasitoid models*, Ecological Modelling **93** (1996) 121–143.

[8] J.L. PEREIRA-BARROS, S.M.F. BROGLIO-MICHELETTI, A.J. N. DOS SANTOS, L.W.T. DE CARVALHO, L.H.T. DE CARVALHO AND C.J.T. DE OLIVEIRA, *Ciênc. Agrotec.*, **4**(2005)714-717.

[9] M. RAFIKOV, J.M. BALTHAZAR AND H.F. VON BREMEN, *Mathematical Modeling and Control of Population Systems: Application in Biological Pest Control*, Applied Mathematics and Computation, **200** (2008) 557-573.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Modeling of neutron activation process with Americium Beryllium source. Application to the activation of fluorspar samples

M.A. Rey-Ronco[1], T. Alonso-Sánchez[2], M. P. Castro-García[2]
[1]*Departamento de Energía. Universidad de Oviedo*
[2] *Departamento de Explotación y Prospección de Minas. Universidad de Oviedo*

emails: rey@uniovi.es; tjalonso@uniovi.es; UO21947@uniovi.es

Abstract

This paper shows the mathematical models which represent the phenomena that occur in a neutron activation process. These phenomena are, on the one hand, the neutron flux decreases with the distance between the neutron source and the sample, and on the other hand, the attenuation of the gamma rays originating from the sample activated on its way to the detector position. The development of the mathematical model has been divided into two parts. Firstly, the phenomena are shown separately. Secondly, the phenomena are shown together. Finally, this model is fitted to the neutron activation of a fluorspar sample, and the influence of the two phenomena as defined above can be seen.

Keywords: nuclear activation, deferred gamma rays, mathematical model, fluorspar.

1 Introduction

Neutron activation is a process in which an atom emits a characteristic radiation when it is excited by a neutron. This system can be used to determine the presence of certain elements in a sample.

Neutron activation analysis (NAA) was discovered in 1936 by Hevesy and Levi, who found that samples containing certain rare earth elements became highly radioactive after exposure to a source of neutrons. This observation led to the use of induced radioactivity for the identification of elements.

In the last several decades, this technique has been applied to determine a great variety of elements in many disciplines. These include environmental science as well as, biological, geological, and material science.

The basic elements used in neutron activation are:

 ISBN 13: 978-84-613-5510-5

- an radioactive source that allows the irradiation of a sample by neutrons and,

- a radiation detector that reads gamma radiation emitted by a sample during the decaying of the radioactive products. This radiation is produced in a given time and is characterized by the energy and time during which it occurs, and is characteristic of each element.

A neutron activation process is characterized by two phenomena that occur:

- the one during radiation which supposes that the neutron flux decreases with the distance between the neutron source and the sample [1].

- with the radiation reading, that implies the attenuation of the gamma rays originated in the decay of ^{16}N traversing the sample to the detector position. This attenuation is exponentially dependent on a characteristic attenuation coefficient of the sample [2].

2 Definition mathematical models

In the procedure used, the base of the container sample is located at a distance “a” of the irradiation position (source), and a distance “b” of the reading position (detector), Figure 1.

On the one hand, neutrons emerging from the source after they traverse the space “a” (air) arrive at the sample where they do or do not interact with the sample, resulting in the activation of the fluorine atom. Neutron flux is reduced with the distance from the source.

On the other hand, the produced gamma rays traverse the sample to the detector position, and are attenuated as they travel.

 ISBN 13: 978-84-613-5510-5

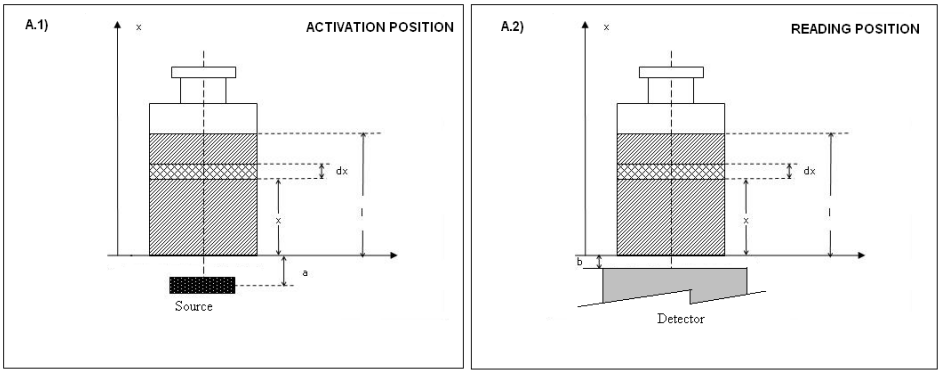


Figure 1. A.1) Geometry considered in the activation position. A.2) Geometry considered in the reading position.

2.1 First model

In the first model, we refer specifically to the phenomenon of the reduction of the neutron flux with distance from the source. It is supposed that the activation of a differential element *dx* of the sample only depends on the distance *(x+a)* in the direction *x* to the source, and that the dependency ratio is inversely proportional to the square of the distance of the differential element at the center of the source. Consequently, it is assumed that gamma rays are not attenuated by distance.

In this process, we have observed the following proportionalities (Figure 2):

- Counts from the sample and reading at the detector C_m are proportional to the counts C_p produced by the sample. The proportionality constant depends on the efficiency of the detector.

- Counts produced in the sample C_p are the integral from $x=0$ to $x=l$ of the differential counts dC_{px} in an element of base S and height dx.

- The differential of counts produced dC_{px} at the differential element of height dx is proportional to the number of excited atoms in the differential element (dN_{ax}).

- The number of excited atoms dN_{ax} depends on neutron flux and the number of fluorine atoms dN_{fx} at that point.

- The neutron flux Φ_{nx} is inversely proportional to the square of the distance between the point and the source $(x+a)$.

 ISBN 13: 978-84-613-5510-5

- The number of fluorine atoms in a sample point dN_{fx} is directly proportional to the product of the grade y and the mass dm of differential element.

- The mass dm depends on the volume of differential element dV and the density ρ_m.

- The volume dV is defined by the section of the sample S and by the height of differential element dx.

- The density of the sample ρ_m is in relation to the density of the fluorspar ρ_1, the sterile ρ_2, and the sample grade in per unit.

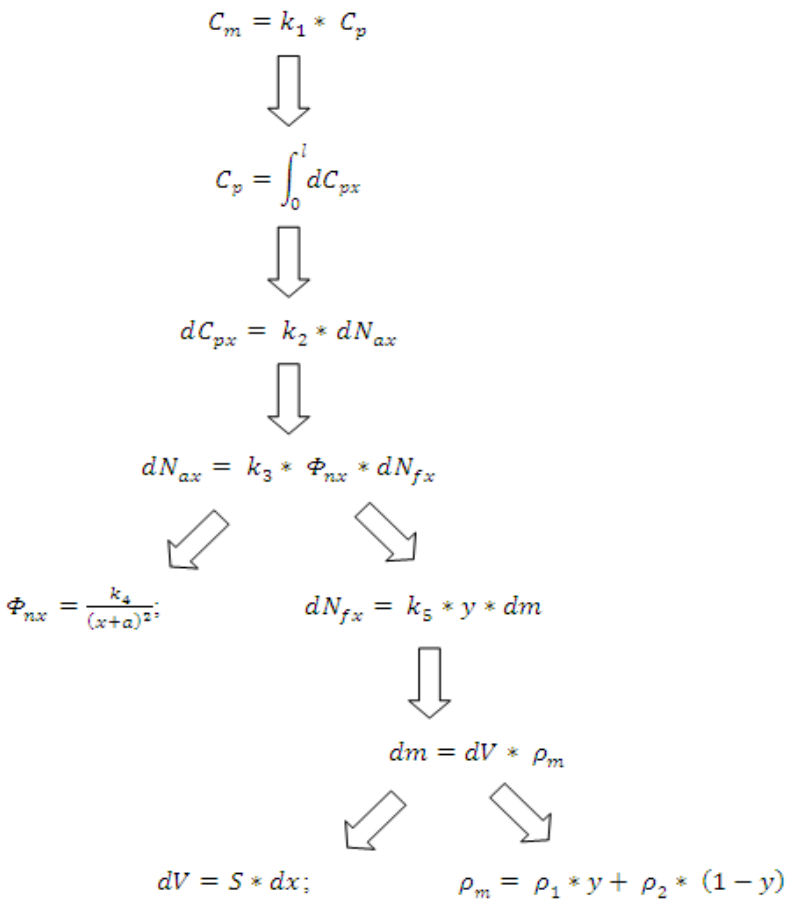


Figure 2. Expressions used in the proportionalities

 ISBN 13: 978-84-613-5510-5

Substituting these terms:

$$C_m = k_1 * k_2 * k_3 * k_4 * k_5 * (P * y + Q * y^2) * \int_0^l \frac{dx}{(x+a)^2}$$

Where,

$$Q = S * (\rho_1 - \rho_2)$$

And,

$$P = S * \rho_2$$

The parameters *Q* and *P* are constants, greater than 0 and have dimensions of mass per unit length.

Making,

$$K = k_1 * k_2 * k_3 * k_4 * k_5$$

and integrating leads to the equation,

$$C_m = K * (Q * y^2 + P * y) * \frac{l}{l+a}$$

We can express the value of the height of the sample *l* (in cm) in function of the mass *m* (in grams), of the cross section of a container sample *S* (in cm^2), of the grade, of the fluorspar density ρ_1 and of the sterile density ρ_2,

$$l = \frac{m}{(\rho_1 * y + \rho_2 * (1-y)) * S}$$

Then, in this model the relationship between counts, mass and grade is the following:

$$C_m = \frac{K * m * (Q * y^2 + P * y)}{a * (m + Q * a + y + P * a)} + F$$

Where,

Cm is the integral of the counts from the sample in a certain range of channels for this first model,

- *P* y *Q* are constants and their values are greater than 0,
- *a* is the distance between the source and the base of the sample,
- *y* is fluorspar grade constant in the sample expressed per unit,

 ISBN 13: 978-84-613-5510-5

- l is the height of the sample .

- F counts from detector background without sample.

This equation is a surface in three dimensions whose X and Y axes are mass and grade and the Z axis is the counts (Figure 3).

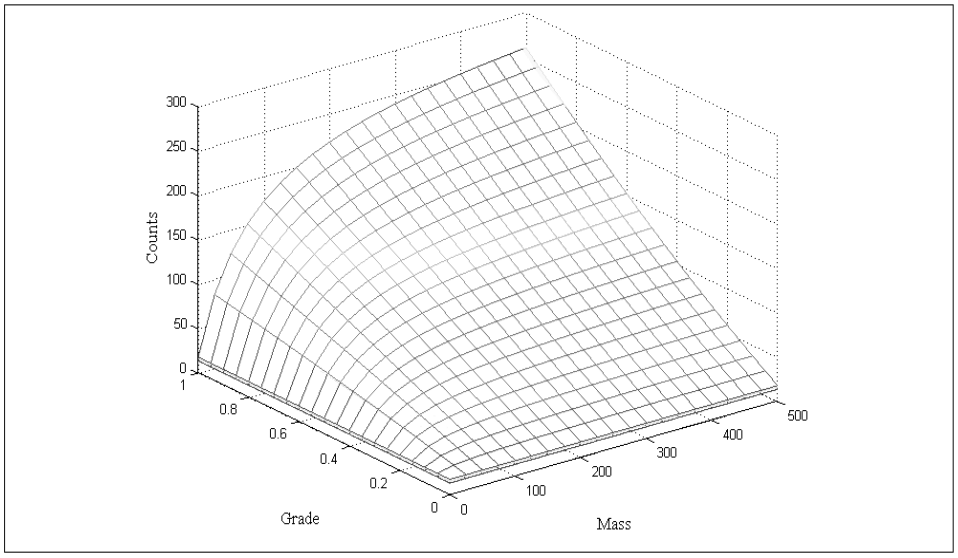


Figure 3. Representation of the equation in the study carried out with MATLAB

2.2 Second model

In the first model, we refer specifically to the phenomenon suffered by gamma rays produced from an irradiated sample before being detected. In the radiation reading position, the detector is at a distance b from the sample, as shown in Figure 1. The value b is very small, negligible compared with the height l of the sample. This fact together with the attenuation of gamma rays in the air being lower than in the sample, leads to the observation that attenuation takes place only in the mass of the sample.

In this process, we have observed the following proportionality, some of them identical to previous case:

- The counts from the sample and reading in the detector C_m have an exponential relation to the counts dC_{px} produced by the sample. The proportionality constant depends on the efficiency and the coefficient of radiation attenuation:

 ISBN 13: 978-84-613-5510-5

$$C_m = k_1 * \int_0^l e^{-\mu * x} * d\, C_{px}$$

- As in the previous case, the count differential dC_{px} produced in the differential element of height dx is proportional to the number of excited atoms in the differential element dN_{ax}. The number of excited atoms dN_{ax} depends on neutron flux Φ_n (constant in this hypothesis) and of the number of fluorine atoms dN_{fx} at that point. The number of fluorine atoms at a sample point dN_{fx} is directly proportional to the product of the grade y and of the mass dm of the differential element, and as in the previous case, the mass dm depends on the volume of the differential element dV and on the density ρ_m.

Proceeding as previously, we obtain the equation for this second model, which includes the background, as follows:

$$C_m = (K' * Q * y^2 + K' * P * y) * \frac{1}{\mu}\left(1 - e^{-\mu * \frac{m}{Q * y + P}}\right) + F$$

The parameters Q and P are the same as in the previous case, while the factor that includes all proportionality constants K' is different.

2.3 Third model

In this third model, it is supposed that the activation of a differential element sample depends on the distance $(x+a)$ in the direction x to the source, and that gamma rays are attenuated on their way to the detector following an exponential $y = ae^{-bx}$.

In this case, the number of counts C in the energy range considered that come to the detector by the neutron activation effect can be expressed as:

$$Solution\ of\ the\ model\ 3\ = \alpha * solution\ of\ the\ model\ 1 + \beta * solution\ of\ the\ model\ 2$$

α y β being the weight coefficients of each phenomenon individually in the final model, which fulfill the condition of being positive and their sum is equal to 1.

3 Checking the models

In order to determine which phenomena have greater impact on neutron activation, and therefore, the best model that fits to reality, each model is applied to neutron activation of a fluorspar sample from a concentration plant. Equipment used

 ISBN 13: 978-84-613-5510-5

consists of an Americium Beryllium source of 1 Ci of activity and a gamma ray detector of the type NaI. A prototype has been designed [3].

The reason for using fluorspar samples for testing the models is that the radiation of fluorspar with neutrons from Americium Beryllium source emits a characteristic high energy (6.13 MeV) which comes, according to previous studies [4], from the fluorine present in CaF_2. This radiation comes from ^{16}N originated, only and exclusively, from the nuclear reaction $^{19}F(n,\alpha)^{16}N$. Due to the characteristics of the detector, the energy spectrum of the sample, does not give a single peak at 6.13 MeV, but has a certain width, and has some 'echoes' called 'escape peaks' at 5.11 and 5.62 [5]. For this reason, the counts used in the study are in a range and are not in this exact value (6.13MeV). The authors know the reactions produced in this mineral using the activation for analyzing fluorine in a fluorspar samples. An activation procedure and a mathematical method that increases the sensitivity were designed [6].

The fluorspar samples with variable fluorite grades *[y]* from 4 to 97% and variable masses *[m]* from 50 to 450 g (taken at intervals of 50 to 50) have been used. Samples are found in the same state of humidity and particle size.

3.1 Adjusting the first approach to the experimental data

Equipment was designed specifically to irradiate with neutrons and read the gamma rays emitted from fluorspar samples with the grades and masses specified above. Irradiation and reading times were adjusted according to the reaction sought and the coefficients K, a, Q, P, and F were determined. Nonlinear regression was used to determine the coefficients. The statistical program SPSS [7] was used. In this work only the coefficients obtained from taking the average values of the counts detected in the different tests, with an energy range between 4.5 and 6 are shown.

Algorithms used by statistical programs for nonlinear regressions are iterative processes which require the assignment of initial values for the coefficients. Table 1 shows the initial values for the parameters of the above equation.

Table 1. Initial parameters used for nonlinear regression

Density ranges	Parameters	Initial values
		$K=1$
$\rho_1=1.5\text{-}1.9 g/cm^3$	$P=S*\rho_2$ between 30-50	$P_0=45$
	$Q=S*(\rho_1-\rho_2)$ between 3 and 16	$Q_0=9.6$
$\rho_2=1.1\text{-}1.4 g/cm^3$	a between 0.5 and 8 cm	$a_0=2cm$
	F between 10 and 25 counts	$F_0=13$ counts$=1$

 ISBN 13: 978-84-613-5510-5

Under these initial conditions the program was put into action and the parameters obtained are shown in Table 2.

Table 2. Summary of the model parameters (Model 1)

PARAMETERS. MODEL 1	
Parameters	Energy 4.5-6 MeV
K	9.798
a	1.832
Q	7.515
P	38.978
F	13.598
R2	0.994

3.2 Adjustment of the second approach to the experimental data

Regression was carried out using the experimental results with a sample group. The procedure was repeated with the energy and value types. The parameters of the resulting equation are reflected in Table 3.

Table 3. Summary of the model parameters (Model 2)

PARAMETERS. MODEL 2	
Parameters	Energy 4.5-6 MeV
K'	1.992
μ	0.505
Q	10.122
P	43.464
F	13.6
R2	0.991

3.3 Adjustment of the third approach to the experimental data

Establishing restrictions of attenuation coefficient μ in the processing of the third model with the program SPSS, we are found that the correlation coefficient in this third model is lower than in previous models, so it follows that the phenomenon of gamma rays attenuation (expressed by β) is negligible.

 ISBN 13: 978-84-613-5510-5

4 Comparison between models

In the first model a very high attenuation coefficient μ is obtained and the parameters *a, K, P, Q* y *F* are consistent with the real values. However, the attenuation coefficient μ obtained in the second model is about 0.5, which is 10 times higher than expected given the composition of the sample. It therefore follows that the first model is the closest to reality.

5 Validity of the first model

Fluorite grades are compared with grades obtained from the first model for activated fluorspar samples with the parameters identified above. These parameters are K=9.798; a=1.832; Q=7.515; P=38.978; F=13.598.

The sample grade of mass known m after activation with a count number C emitted is obtained by:

$$y=\frac{[(C-F)*Q*a^2-K*m*P]\pm\sqrt{[K*m*P-(C-F)*Q*a^2]^2+4*(K*m*Q)*[(C-F)*(P*a^2+m*a)]}}{2*K*m*Q}$$

This equation is illustrated in Figure 3, and the comparison is shown in Figure 4. Note the linearity of the response and the validity of the model for all range of grades.

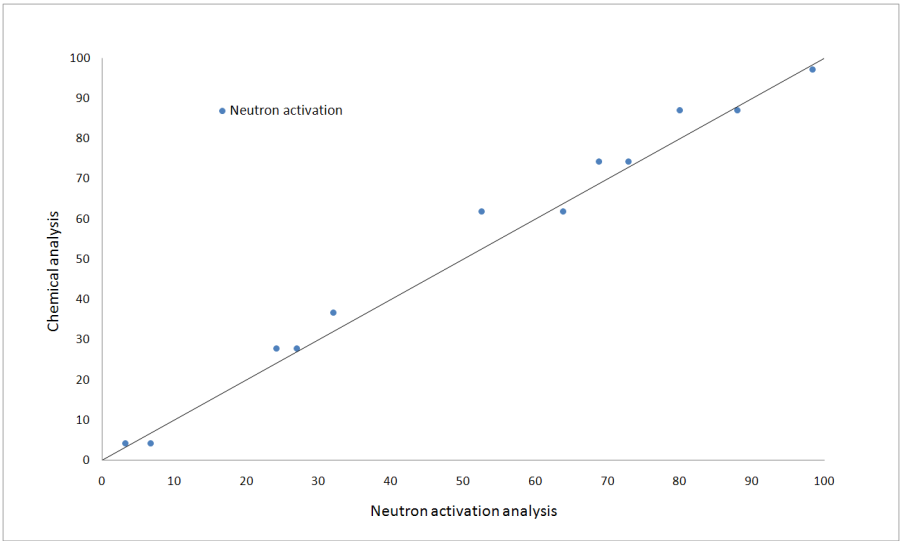


Figure 4. Comparison between chemical analysis and neutron activation with samples used in the deduction of the model

 ISBN 13: 978-84-613-5510-5

6 Conclusions

A neutron activation process from two basic and simultaneous phenomena has been modeled. The phenomena are the reduction of the neutron flux that activates the sample and the attenuation of gamma rays produced during the activation that reach the detector.

The phenomenon of the reduction of the neutron flux is controlled by the inverse of the square of the distance between the sample and the source.
The phenomenon of the attenuation of gamma rays is controlled by an exponential law that depends on the attenuation coefficient μ, characteristic of sample.

The model was assayed by way of irradiation from a fluorspar sample whose fluorine content is reflected by the emission of high- energy gamma radiation (6.13 MeV).

From the correlation of the results with the models and the experiments, it follows that only the effect of reduction of the neutron flux with the distance can be considered as the effect of the other effect is low.

A high correlation coefficient (~1) has been obtained from the model. In addition the parameter values *a, K, P, Q* and *F* are consistent with the real values.

After comparison between the values obtained from chemical analysis and those from neutron activation, the model was considered suitable.

7 Acknowledgements

This work has been supported by Mineral Products and Derivatives Company, SA (Minersa), Government of the Principality of Asturias, Research grant scheme of the University of Oviedo. The authors would like to thank the above-mentioned bodies for their collaboration and financial support during this study.

8 References

[1] Parry S. J., 1991. Activation spectrometry in chemical analysis chemical analysis. A Wiley- Interscience publication. United Stated, America, NY.

[2] Robu E., Gioani C., 2009 Gamma- Ray self- attenuation corrections in environmental samples. Romanian reports in physics, 61, 295-300.

 ISBN 13: 978-84-613-5510-5

[3] Alonso-Sánchez, T., Rey-Ronco M.A, Castro-García M.P. A neutron activation technique for the analysis for fluorine in fluorspar samples. International Journal of Mineral Processing. 94, (2010), 1-13.

[4] Rey Ronco, M. A. Desarrollo de un método rápido basado en técnicas de activación neutrónica para la determinación del contenido en flúor de muestras de mineral de fluorita. Doctoral Thesis. Universidad de Oviedo, 2007. <http://www.tesisenred.net/TDR-0802107-133201/index_cs.html>

[5] Maki Y., Nojiri T., Masilungan B.A. The determination of fluorine by cyclic activation analysis method using ^{241}Am-Be neutron source. Radioisotopes 23, (1974), 149-154.

[6] Rey-Ronco M. A., Alonso-Sánchez T., Castro-García M. P. Mathematical study to improve the sensitivity in the neutron activation analysis of fluorspar. Journal Mathematical Chemistry (2010), doi: 10.1007/s10910-009-9652-z.

[7] Norman H., C. Hadlai Hull, Dale H. Bent., 1968. Statistical Package for the Social Sciences (SPSS).

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Ultrasonic Sensors with Mechanical Couplers: Simulation and Validation

Mónica Fernández[1], Cristina Rodríguez[1], Juan M. Perez Oria[1], M. Ibarra and Luciano Alonso[1]

[1] *Electronic Technology and Automatic Control Systems Department, University of Cantabria*

emails: monica@teisa.unican.es, cristina@teisa.unican.es, oria@teisa.unican.es, ibarra@teisa.unican.es,alonso@teisa.unican.es

Abstract

This paper presents the results obtained in a study of the behaviour of mechanical elements coupled to an ultrasonic sensor using finite element techniques, which can modify the radiation pattern of the original sensor. These results have been obtained using Comsol multiphysics modelling. The effect caused by the sensor size on the radiation acoustic pressure has also been evaluated. In this paper we also present the experimental validation of these simulations.

Key words: ultrasonic, sensors, horns, pattern

1. Introduction

The use of ultrasonic sensors has varied applications including basic ones such as object detection or measurement of distances. For some production process tasks we find that, although the applications are not very difficulty, they involve a great time cost in processes, such as the detection of defects. Some of the more complex applications include recognition and identification of objects.

A common factor in all these applications is that the operating frequency of ultrasounds limit the operating distance of the ultrasonic sensor, due to the already known fact that increasing the operating frequency also increases the attenuation of ultrasonic waves. In contrast, an increase in the frequency of the ultrasound provokes a narrowing in the radiation lobe. Given these facts, it follows that by increasing the operating frequency the directional behaviour of the sensor can be improved, but at the expense of reducing the working distance. In applications

 ISBN 13: 978-84-613-5510-5

needing a broader or a narrower lobe than that obtained by the sensor itself, there are several options, ranging from the use of sensors of different frequency, appropriate for the application, or increasing the sensor diameter to obtain more power, using arrays of sensors to obtain the radiation pattern. This last option intrinsically leads to an increase in hardware and software complexity. An alternative solution is to couple a mechanical element, commonly called horn. As would be expected, physical and geometrical characteristics of the coupled element influence the new ultrasonic radiation pattern. This is where the need arises to find a simulation model, to find the relation between the coupler parameters and the desired characteristics of the radiation lobe.

To characterize this technique, it is necessary to use software tools to be able to analyse the simulated models. Among the different techniques for simulation, such as boundary element, finite difference, finite element, we choose the latter because of its availability and versatility. The use of finite elements and Comsol software has already provided goods results in previous works.

In this paper, we present the results obtained in the radiation pattern using an ultrasonic sensor in two different cases. The first one is the ultrasonic sensor in free radiation and the second one is when a straight horn is linked to the sensor. We also present the variation in sound pressure on the radiation axes when the size of the sensor changes.

2. Simulations descriptions

Taking into account the results already obtained in previous work [1] [2] in the Comsol simulations to study the modifications produced in the ultrasonic radiation patterns using couplers, we have tried to obtain a simulation that describes the real problem more faithfully. This was the reason for simulating the system using the multiphysics modelling that Comsol makes possible.

Simulations have been divided into different parts. The first one is to obtain the ultrasonic radiation lobe in free radiation. The second explores the effect produced when the sensor is provided with a horn with zero opening angle, that is, a tube of a certain length but it with no opening (straight horn). Finally, the effects produced on the sound pressure at on the axis of radiation has been simulated as function of the sensor size.

For a description of the model using Comsol multiphysics, the division of the problem into three distinct domains as shown in Figure 1 has to be taken into account. The first one refers to the sensor itself, while the second refers to acoustic wave propagation in air. In the last of the domains we have defined a far-field zone, which provides the attenuation of the wave for longer distances. In this work, the *y axis* (radiation axis) has been fixed as the axis of symmetry, in order to reduce the large computational cost involved in simulation using the finite

 ISBN 13: 978-84-613-5510-5

element technique. In this way, the decrease in the numbers of equations is used to increase the extent of the simulations, optimizing the computational resources.

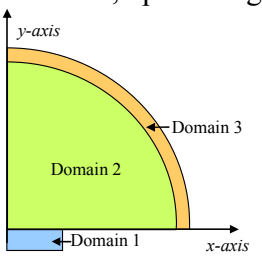


Figure 1: Representation of the domains used in the simulation

For the first of the domains, which refers to the sensor itself, and taking into account the Comsol enhancement, which enables the different kinds of physics to be combined, the piezoelectric crystal part has been used. In our case, it was established that the sensor was a piezoelectric crystal of PZT5-H (Lead Zirconite Titanate), which is a material commonly used in transducers. Then a voltage difference was applied between its upper and lower surfaces.

The second and third domains refer to ultrasonic propagation in air and therefore must fulfill the Helmholtz equation, which is shown in equation (1).

$$\Delta p(\vec{r}) + k^2 p(\vec{r}) = 0 \qquad (1)$$

where Δ is the Laplace operator, p is the acoustic pressure, r is the position, k is the wave number, which is valid in the working domain established.
Having established the domain, you must define its boundary conditions. For the piezoelectric crystal the condition of symmetry in the *y-axis* is established, and a voltage difference between the upper and lower surfaces of the sensor is applied. For the second and third domains, the axis of symmetry is defined as the *y-axis* and in the other contours, except the interface between domains 1 and 2, as the condition of wave propagation in the air, equation (2).

$$\frac{\partial p}{\partial n} = ikp \qquad (2)$$

where i is the imaginary unit.

For the interface between domains 1 and 2, that is, between the upper surface of the sensor and air, the boundary condition is given by equation (3).

$$n(\nabla p) = a_n \qquad (3)$$

where n is the outward normal and a_n is the normal acceleration. Thereby getting the sound pressure produced by the voltage difference applied to two surfaces of the piezoelectric crystal to pass the domain in which the acoustic wave propagates. Note that between the second and third domains, this condition is not produced because the domain characteristics are the same.

 ISBN 13: 978-84-613-5510-5

Under these conditions, the radiation pattern obtained for the case of an ultrasonic sensor operating at a frequency of 25KHz. and a 7mm. sensor radius is shown in Figure 2.

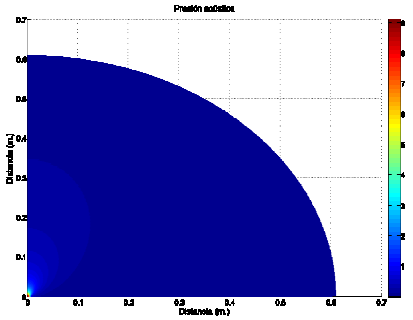


Figure 2: Radiation pattern for multi-physical modelling using a sensor in free radiation

It is important to note that the workspace of the simulation is small, only reaching up to 0.6m. This is because the computational cost required by the software is very high, this being one of the main drawbacks and problems encountered in the work.

The second objective of this study was to obtain an ultrasonic radiation lobe when the sensor is attached to a mechanical horn-type element. In addition, this paper presents the case when the horn used was a straight tube, that is, without opening angle and a given length, such as occurs in case in pieces of couplers types. In the case of multi-physical modelling, the introduction of this new element does not increase the complexity excessively, it is only to add a fourth domain, with the same characteristics as domains 2 and 3 defined above, that is, a domain in which the acoustic wave propagates. Figure 3 shows the new situation

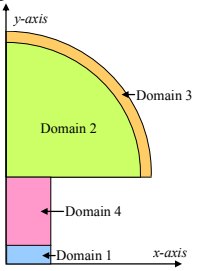


Figure 3: Representation of the domains in the case of couplers

It should be noted that right now the boundary condition between the upper surface of the sensor and the air is between domains 1 and 4, while for the surface binding domains 4 and 2 we do not have to establish any special status as both possess the same characteristics. In addition, the boundary condition established at the wall of the horn is total reflection as shown in equation (4).

 ISBN 13: 978-84-613-5510-5

$$\frac{\partial p}{\partial n} = 0 \qquad (4)$$

Figure 4 shows the radiation pattern obtained for a sensor operating at a frequency of 25Khz. and with a straight coupled horn of length 3cm. and a sensor radius of a 7mm.

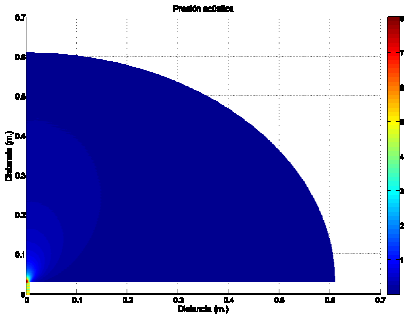


Figure 4: Radiation pattern of a 25KHz. sensor with a 3cm straight horn attached

In the final part of the work, the influence of the size of the ultrasonic sensor on the acoustic pressure on the radiation axis is studied for the case of free radiation. Figure 5 shows that increasing the size of the sensor, the acoustic pressure on the shaft also increases.

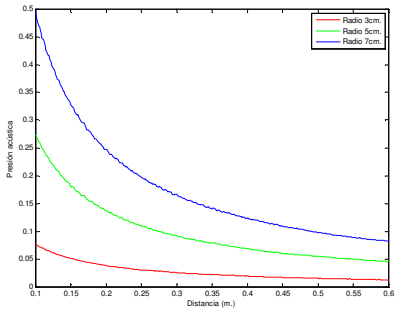


Figure 5: Acoustic pressure on the radiation axis when the sensor size is changed

3. Measurement system description

To perform the experimental evaluation of the simulations we used a robotic positioning system inside an environmental chamber.

For the ultrasonic system, two Hexamite long-range sensors (up to 30m.) were chosen. They work with an excitation frequency of 25 KHz. One of the sensors will be used as a transmitter, it being necessary to use a conditioner for sensor excitation, provided by the same manufacturer. For the reception stage, the corresponding conditioner of the ultrasonic sensor has not been used because for the working distances of the measurements, i.e. up to 3m, it is not necessary.

 ISBN 13: 978-84-613-5510-5

For the sound pressure at all points in an XY plane, we chose to use a Yamaha robot, model BX SBX RCX40 with a RCX40 controller that allows us to vary both the distance and the angle between the transmitter and receiver. The distance between transmitter and receiver varies from 0 to 3m. with a 1cm. step, while the scan angle will be between 0º and 90º with a step of 1º, due to the symmetry. However, it is important to project the large number of points obtained for each register, more than 27,000 echoes.

Both systems are situated inside an environmental chamber, which can vary both the temperature and humidity. In Figure 6, a view from inside the climatic chamber of the measuring system is shown

Figure 6: Transmitter positioned on the robot inside the climatic chamber

A closed-circuit television/video camera was also placed inside the climatic chamber in order to continue to capture and process of ultrasonic echoes from the outside the chamber. All this is handled by a data acquisition card, supplied by National Instruments and both data acquisition and the further processing is done with Matlab. It has been taken into account that the simulation process which is performed with Comsol can also work in Matlab, which is a very important point for a later comparison of simulations and measurements.

Figure 7 shows the ultrasonic radiation lobe in free radiation obtained in the measurements and represented in Cartesian coordinates.

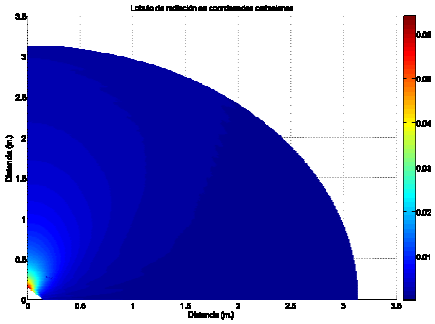


Figure 7: Ultrasonic radiation lobe in free radiation

In the case of coupling to the sensor of a straight horn of 3cm. length, we used the piece shown in Figure 8. Notice that the inside of the coupler is a straight pipe

 ISBN 13: 978-84-613-5510-5

with the same diameter as the sensor, while on the outside the coupler is threaded to be able to connect horns with different geometry.

Figure 8: Straight horn in the measurements used

Figure 9 shows the radiation lobe of the ultrasonic sensor with the coupler in the previous figure, also in Cartesian coordinates.

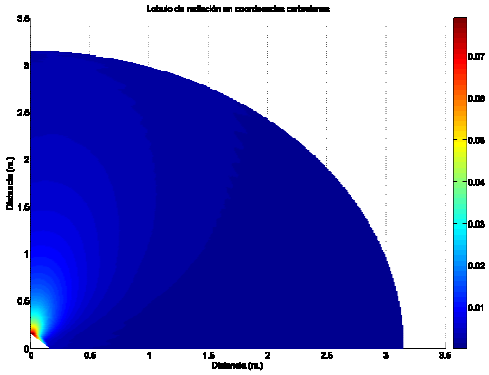


Figure 9: Ultrasonic radiation lobe of the ultrasonic sensor with the coupler

Figure 10 shows the couplers used to obtain the sound pressures in the radiation axis for the same ultrasonic sensor but with a different internal diameter.

Figure 10: Couplers used

4. Experimental validation

Figure 11 shows the simulated sound pressure on the radiation axis in the case of the free radiation sensor and in the case of the horn attached to the sensor.

If in Figure11 the graph corresponding to the acoustic pressure of the sensor in free radiation is displaced an identical distance from the straight horn, that is, 3cm. then Figure 12 is obtained, in which the two acoustic pressures in the radiation axis match.

 ISBN 13: 978-84-613-5510-5

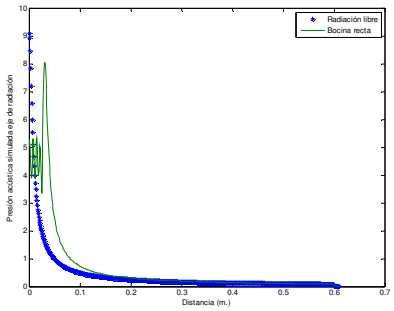


Figure 11: Simulated acoustic pressure on the radiation axis for sensor in free radiation and with coupler

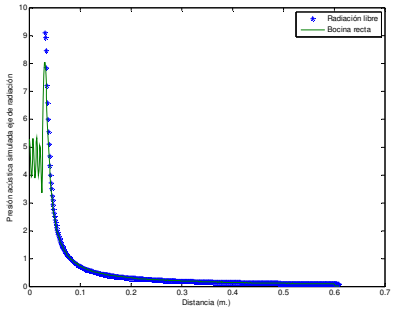


Figure 12: Simulated acoustic pressure on the radiation axis for the sensor both with and without being displaced the length of the horn.

If a cut is made in the *y-axis* to observe how the sound pressure on the x-axis varies, taking into account the shift that occurs when you the horn coupling is done, Figure 13 can be obtained.

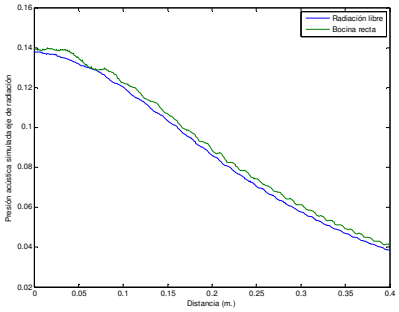


Figure 13: Acoustic pressure simulated on the x-axis for free sensor and with the horn coupled, taking into account the shift of the length of the horn.

From the above, and the radiation patterns obtained in Figures 2 and 4, we can conclude that the radiation pattern has not been modified by the coupling of the straight horn, that is, the effect of the straight coupler is the same as situating the ultrasonic sensor at the top of the horn.

 ISBN 13: 978-84-613-5510-5

Viewing the simulations made varying the size of the sensor at the acoustic pressure on the axis of radiation; it appears that the sound pressure increases with the diameter. The known relation expressed in equation (5) is verified, which provides the sound pressure at a distance *d* from the oscillator.

$$P = P_o \cdot 2 \cdot sen\left[\frac{\pi}{\lambda}\left(\sqrt{\frac{D^2}{4}+d^2}-d\right)\right] \qquad (5)$$

where *D* is the diameter of the oscillator and λ the wavelength.

For large distances, $d >> D^2/4\lambda$, the above equation can be approximated by equation (6), that is, the pressure is proportional to the square of the diameter of the sensor. Table 1 shows the proportional constants obtained in each case.

$$P = P_o \cdot \frac{\pi \cdot D^2}{4\lambda} \qquad (6)$$

Sensor Size (mm.)	Proportionality constant
14	4.2
10	4.5
6	3.6

Table 1: Simulated proportionality constant depending on the size of the sensor

If we analyze the acoustic pressure along the axis of radiation to the measures, both for the case of free radiation is coupled as when you get Figure 14, which shows that there is little difference between the case of free radiation horn coupled radiation increases with distance.

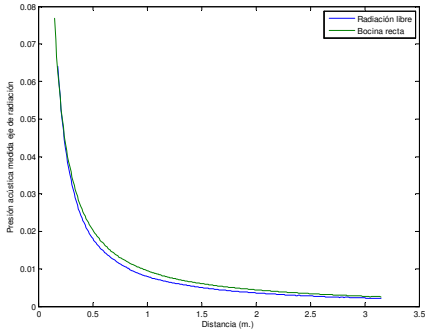


Figure 14: Acoustic pressure measured on the axis of radiation, radiation free and coupled horn

Based on this result, the main conclusion is that by using a coupler in the form of a straight horn with different diameter, change the size of the sensor while maintaining the same intrinsic features of the sensor.

 ISBN 13: 978-84-613-5510-5

Figure 15 shows the sound pressure measured on the radiation axis when the couplers shown in Figure 10 are used. In this figure, the inner diameter of the coupler is varied from 3mm. to 7mm. in radius with a step of 2mm. These results agree with those predicted in the simulations and expressed in the equation (5), that is, increasing the sensor size increases the pressure on the axis of radiation.

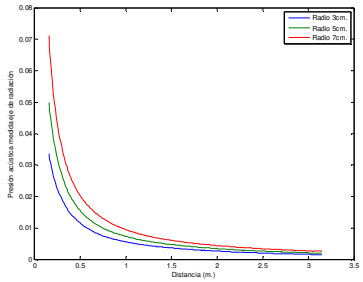


Figure 15: Acoustic pressure on the radiation axis for different sizes of sensors.

Table 2 shows the proportionality constants of the measurements obtained in each case.

Sensor Size (mm.)	Proportionality constant
14	0.9
10	1.3
6	2.6

Table 2: Constant of proportionality of measures according to the size of the sensor

If a comparison is made between the acoustic pressure on the radiation axis for simulations and measurements, both with coupler, Figure 16 is obtained, in which you can see that the sound pressures correspond quite faithfully, after a scale adjustment.

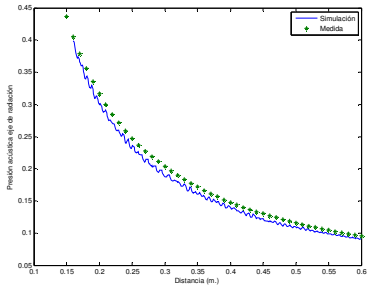


Figure 16: Acoustic pressure on the axis of radiation, measured and simulated, with a straight horn

In addition comparisons can be made between the simulations and measurements, for different sensor sizes, producing the graph shown in Figure 17, in which the scales have been adjusted.

 ISBN 13: 978-84-613-5510-5

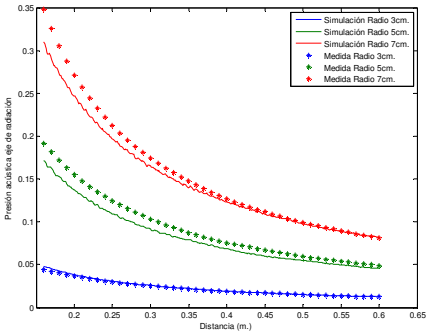


Figure 17: Acoustic pressure on the radiation axis, simulated and measured, for different sensor sizes

5. Conclusions

Analyzing the simulations, it can be concluded that the use of a coupling element with the shape of a straight horn does not affect the radiation pattern of the sensor. This has been corroborated by the laboratory measurement of the radiation pattern in the case of a free radiation sensor and for a sensor coupled to a horn. This implies that when using a single sensor, different sizes can be used and different horns with different lengths and apertures can be attached to find the most suitable radiation lobe for a given application.

Furthermore, it has also been shown that sensor size affects acoustic pressure, and the mathematical dependence on sensor diameter has been demonstrated.

As noted before, the main problem is the large computational cost entailed when performing the simulations. This is so high that it is not possible to simulate long distances, comparable to the measurement distances.

Acknowledgments

This work has been carried out under the sponsorship of the Spanish Ministry of Science and Innovation (MICINN) in the CICYT with reference DPI2007-640 295.

6. References

[1] M. FERNÁNDEZ, C. RODRÍGUEZ, L. ALONSO, J. PÉREZ ORIA, *Validación experimental de modelos simulados para la conformación de lóbulos de radiación ultrasónica,* XXIX Jornadas de Automática. Tarragona, España 2008.

[2] M. FERNÁNDEZ, C. RODRÍGUEZ, L. ALONSO, J. PÉREZ ORIA, *Simulación del diagrama de radiación ultrasónico modificado por bocinas y*

 ISBN 13: 978-84-613-5510-5

validación experimental del modelo de elementos finitos, XXVII Jornadas de Automática. Almeria, España 2006.

[3] L. ALONSO, J.M. PEREZ ORIA, *A Modified Finite Differences Method for Analysis of Ultrasonic Propagation in Non-Homogeneous Media,* Journal of Computational Acoustics (JCA). Vol. 18, No. 1 (2010) 31–45

[4] COMSOL Web Page: http://www.comsol.com/products/tutorials/

[5] ALONSO L., RODRÍGUEZ C., FERNÁNDEZ M., ROBLA S., SARABIA E. G., PÉREZ ORIA J.,*Conformación mediante bocinas de lóbulos de radiación de sensores ultrasónicos*, XXV Jornadas de Automática. Ciudad Real, España 2004.

[6] ALONSO L., ORIA J.P., FERNÁNDEZ M., RODRÍGUEZ C., ROBLA S. *Qualitative Analysis of the Influence of Horns on Ultrasonic Lobes*, Control and Applications, Cancún 2005.

[7] DAVID S. BURNETT, *Finite Element Analysis From Concepts to Applications*, Addison Wesley, 1988.

[8] YOUNG W. KWON, HYOCHOONG BANG, *The Finite Element Method Using Matlab,* CRC Press, 1996.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Detection of Imperfections within Historic Walls Using Ground-Penetrating Radar

Gonzalo Safont[1], Addisson Salazar[1], Jorge Gosalbez[1], Luis Vergara[1]

[1] *Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universidad Politécnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain*

emails: gonsaar@upvnet.upv.es, asalazar@dcom.upv.es, jorgocas@dcom.upv.es, lvergara@dcom.upv.es

Abstract

This paper presents a novel application of Ground Penetrating Radar (GPR) to the evaluation of ashlar masonry walls. Several experiments were made on a scale replica of a historic ashlar masonry wall. These models were loaded with different weights, and the corresponding B-Scans (radargrams) were obtained. Several kinds of flaws and inhomogeneities were detected from the analysis of the radargrams.

Key words: GPR, NDT, flaw detection, historic walls

1. Introduction

We here present the results obtained applying auscultation of scaled ashlar masonry walls using Ground Penetrating Radar [1]-[4]. We analyzed two masonry walls, one being homogeneous in structure and the other with many imperfections drilled in it. We intend to detect inhomogeneities inside the wall and characterize the propagation of electromagnetic waves inside the masonry when under different loads.

Equipment employed consisted of a GPR, a SIR 3000 from Geophysical Survey Systems, Inc. We used a 1.6 GHz (Figure 1.b) mounted on an encoder. The receiving antenna has a size of 3.8 x 10 x 16.5 cm and was adequate for both

 ISBN 13: 978-84-613-5510-5

vertical and horizontal measures.

The configuration parameters used for data capture were: distance mode, 156 scans per meter, range of 10 ns (20 ns in round-trip-time) and 1024 samples per scan.

Measures were taken along the surface of the wall, following 7 columns (2.2 m high each) and four rows (2.87 m long each). The resulting sampling grid is shown in Figure 2. It must be noted that measures were taken from side A, while on the opposite side of the wall there were different sensors and devices (from other NDT) that served as reflectors.

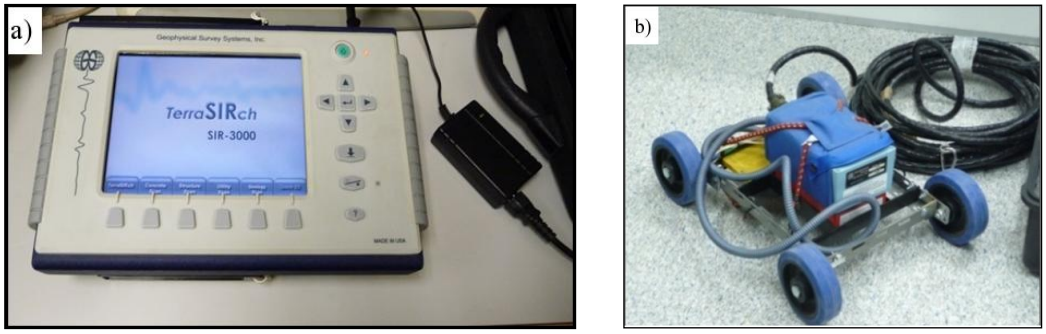


Figure 1. Photograph of a GPR system: a) conditioning, recording and processing system SIR 3000; b) 1.6 GHz antenna (model 5100) with encoder.

We will try to obtain the dielectric constant to use in our work. We know each A-scan has 1024 time samples and so the time sampling is:

$$dt = \frac{10 \text{ ns}}{1024 \text{ samples}} = 9.76 \text{ ps/sample}$$

Using the first trace of horizontal line 2, we find there is a distance of 426 samples between the start and the end of the wall, see Figure 4. Given the distance the signal travelled during that time (20.4 cm, as measured on site) we get a relative dielectric constant:

$$\varepsilon_r = \left(\frac{c \cdot dt \cdot \#\text{samples}}{\Delta z} \right)^2$$

From here on we will use this value in our calculations.

 ISBN 13: 978-84-613-5510-5

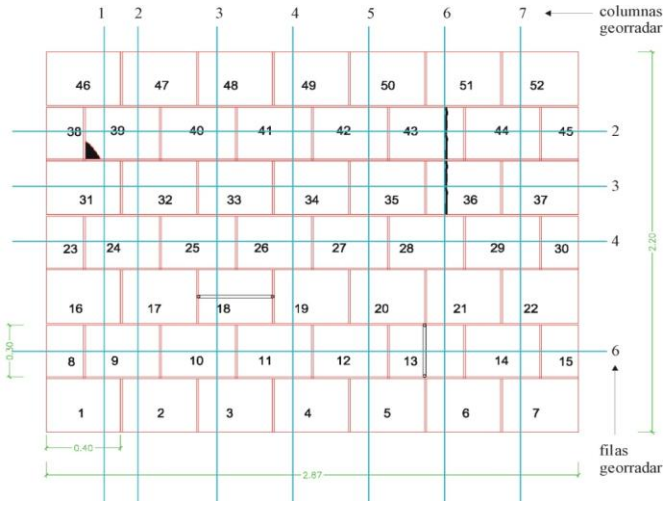


Figure 2. Geometry of the wall under test. Ashlars are lined in red while auscultation trajectories are colored in blue.

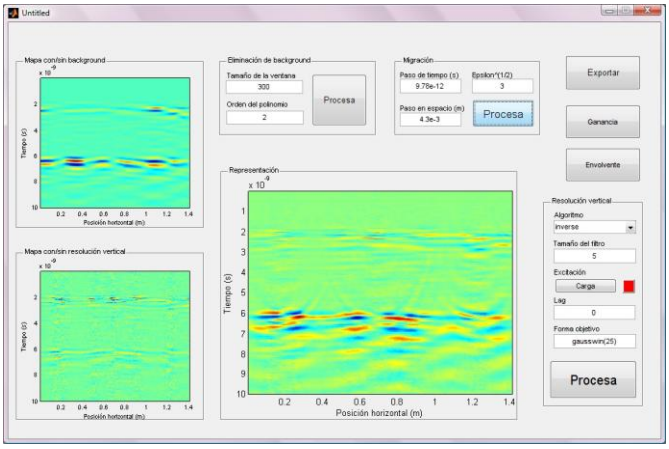


Figure 3. Application used for processing.

2. GPR System Application Interface

A GUI (Graphic User Interface) was made for simple and visual processing of GPR-captured data. It has two parts: a main program (see Figure 3) and a secondary window used for the selection of the excitation signal.

The main window links to three common processing steps of B-scan data: background signal removal, depth resolution enhancing and Kirchoff migration. We used different ICA algorithms for background signal removal [5]-[8]. The results of each of these steps are shown on screen, with the most recent result being shown of the central, biggest figure. There are also three buttons not related to this processing: an export data button (including parameters used and the most

 ISBN 13: 978-84-613-5510-5

recent result), an AGC (Automatic Gain Control) button to enhance contrast and an envelope button, which shows the time envelope of the most recent result (sometimes used for data interpretation). The last two apply to the most recent result (the one shown on the central figure) and are discarded if any further processing is attempted. Exported data values, on the other hand, do include AGC and envelope calculation if they have been applied. In addition we used methods of depth resolution enhancement and cepstral deconvolution [9].

2. General Analysis of the Wall

Figure 4 shows a detail of the background corresponding to the opposite side of the background signal for the wall. The left part of the figure corresponds to row 6, while the right part belongs to column 6.

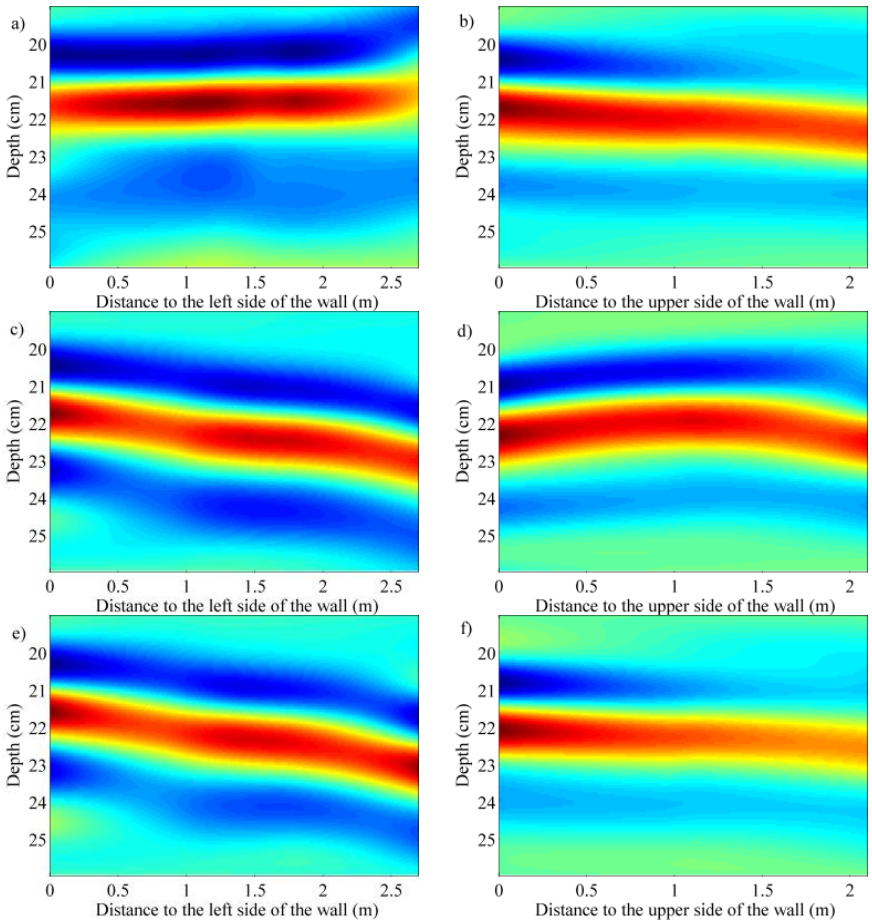


Figure 4. Background at the opposite side of the wall for different values of the load. a) row 6, no load; b) column 6, no load; c) row 6, 50 mt load; d) column 6, 50 mt load; e) row 6, 80 mt load; f) column 6, 80 mt load.

 ISBN 13: 978-84-613-5510-5

Row number 6 has a greater variation of its propagation conditions. This is noticeable from the way the opposite side of the wall seems to move away (see Figure 4), indicating a loss in velocity of propagation inside the material. This is due to a worsening of transmission properties due to the load. We noticed that the difference between 0 and 50 mt (metric tons) is greater than the one between 50 and 80 mt.

The column shown (number 6) also shows a difference in depth between 0 and 50 mt loads. The difference is stronger on both ends of the wall and matches with what shown of row 6 (the right parts of the representations for row 6 and for column 6 correspond to the same area of the wall).

Generally speaking, other representations of the background of other rows and columns follow these same tendencies (that is, they show a stronger effect at the start and end of the B-scan). Nevertheless, they keep the same values in the central area of the wall. This behavior seems consistent with in situ measurements of the wall's distortions.

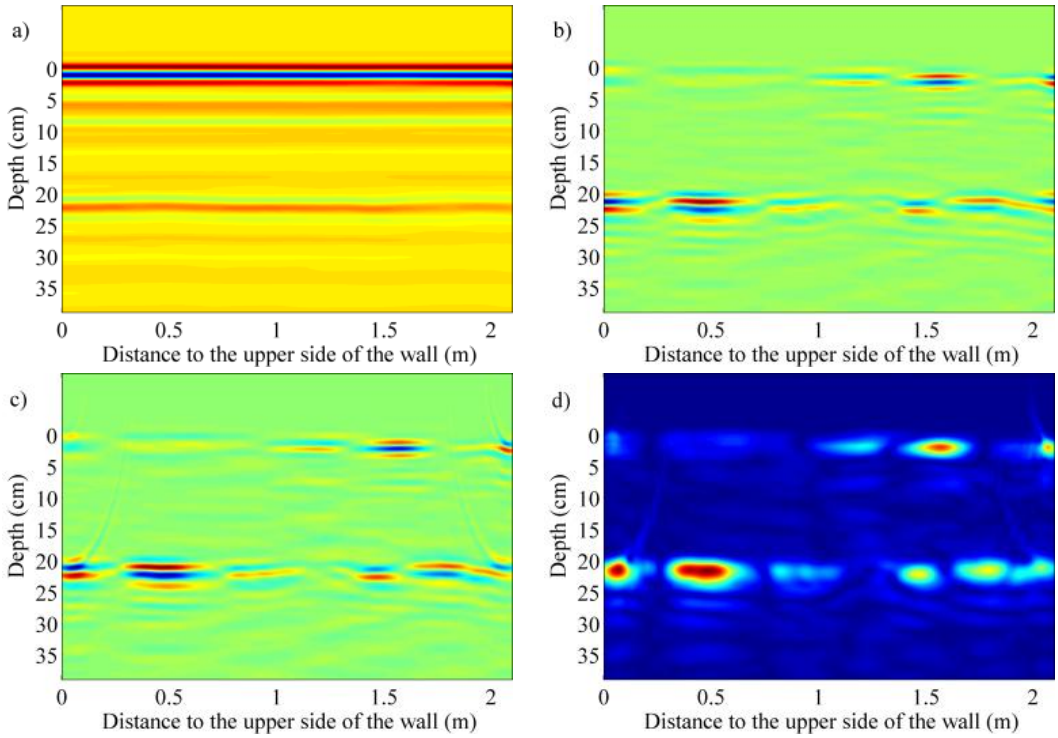


Figure 5. Steps in the processing of radargrams for flaw detection: a) original map; b) background removal; c) migration; d) contrast enhancing. Wall, column 1, 80 mt load.

 ISBN 13: 978-84-613-5510-5

3. Analysis of Flaws Within the Wall

For the purpose of flaw detection, some algorithms were implemented to emphasize the discontinuities (typically due to changes in the material) in the radargrams. These methods, as seen on previous sections, were: background removal, depth resolution enhancing, Kirchoff migration and improvement of the contrast in the B-scan.

Because it is a homogeneous wall, there are no important flaws within the ashlars that compose the wall. This means discontinuities found in the radargrams are due only to mortar interfaces with the ashlars. These interfaces are the less dense material in the structure and thus are the most susceptible to strains because of a compression load. We can check the compression suffered under a load of 80 mt in Figure 6. This Figure also includes the unloaded radargram for comparison.

As a matter of fact, we can notice the greatest effect takes place over the central area of the wall, where the compression forces are bigger. This means that the mortar interfaces between ashlars in the wall under study are no longer visible.

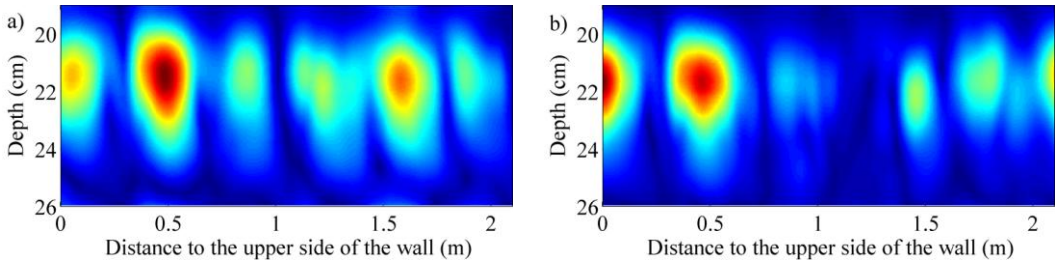


Figure 6. Effect of the load on the detected discontinuities. First column; a) no load; b) 80 mt load.

4. Conclusion

The proposed approach for the auscultation of historical masonry walls with ground-penetrating radar (GPR) has proved to be effective for the detection of flaws and the characterization of walls under load. It was possible to detect flaws with a size of millimeters and variations in the interfaces between ashlars and mortar caused by effect of the compression suffered under load.

Radargrams are techniques that allow representing the internal structure of walls. Nevertheless, one must take into account that inhomogeneities inside the walls will be more or less visible due to their geometry and their physical properties (especially their contrast with the surrounding medium). Because of this, adaptive processing techniques are required to successfully show the different flaws.

 ISBN 13: 978-84-613-5510-5

In addition, we would like to point out that the proposed approach is a low-cost operation with no required previous preparation of the material or a complex and extensive setup. Even more, one can capture data continuously along the surface of the wall (and its interior) in a short amount of time.

5. Acknowledgments

This work has been supported by the *Generalitat Valenciana* under grant PROMETEO/2010/040; the Spanish Administration and the FEDER Programme of the European Union under grant TEC 2008-02975/TEC; and the *Generalitat Valenciana* under grant GV/2009/003 within the research and development programme for Emergent Research Groups.

6. References

[1] A. ZHAO, Y. JIANG AND W. WANG, *Exploring Independent Component Analysis for GPR Signal Processing*, Progress In Electromagnetics Research Symposium (2005) 750-753.

[2] F. ABUJARAD AND A. OMAR, *Comparison of Independent-Component Analysis (ICA) Algorithms for GPR Detection of Non-Metallic Land Mines*, Proceedings of SPIE. **6365** (2006) 6365.1-6365.12.

[3] J.X. LIU, B. ZHANG AND R.B.WU, *GPR Bounce Removal Methods Based on Blind Source Separation*, Progress In Electromagnetics Research Symposium (2006) 256-259.

[4] P.K. VERMA, A.N. GAIKWAD AND M.J. NIGAM, *Analysis of Clutter Reduction Techniques for Through Wall Imaging in UWB Range,* Progress in Electromagnetics Research B **17** (2009) 29-48.

[5] J.F. CARDOSO AND A. SOULOUMIAC, *Blind Beamforming for Non Gaussian Signals*, IEEE Proceedings-F **140** (1993) 362-370.

[6] A. BELOUCHRANI, K.A. MERAIM, J.F. CARDOSO AND E. MOULINES, *Second-Order Blind Separation of Temporally Correlated Sources,* Proceedings of the International Conference on Digital Signal Processing (1993) 346-351.

[7] A. ZIEHE AND K.R. MULLER, *TDSEP – An Efficient Algorithm for Blind Separation Using Time Structure*, Proceedings of the Eight International Conference on Artificial Neural Networks (1998) 675-680.

[8] A. SALAZAR, L. VERGARA, A. SERRANO, J. IGUAL, *A general procedure for learning mixtures of independent component analyzers,* Pattern Recognition, **43** (2010) 69-85.

[9] J.M. REYNOLDS, *An Introduction to Applied and Environmental Geophysics*, Wiley, 1997.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Estimation of Missing Seismic Data based on Non-linear Systems

Gonzalo Safont[1], Addisson Salazar[1], Jorge Gosalbez[1] and Luis Vergara[1]

[1] *Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universidad Politécnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain*

emails: gonsaar@upvnet.upv.es, asalazar@dcom.upv.es, jorgocas@dcom.upv.es, lvergara@dcom.upv.es

Abstract

This paper presents a new method for the reconstruction of missing data in seismic signals. The method is based on non-linear (NL) systems considering non-Gaussian statistics in the probability density function of the seismic data. We propose different NL structures by combining different techniques for the linear and non-linear stages. The linearity in the data is recovered using kriging and cross correlation, and the data non-linearity is reconstructed using direct sample estimation and a third order polynomial approximation. The results by linear and NL structures are compared with the results of Multi-Layer Perceptron and Radial Basis Function neural networks.

Key words: kriging, non-linear systems, seismic signals, missing data

Introduction

In seismic analysis, the wavefield generated by seismic sources is captured by a number of sensors located over a large area. Reconstruction of missing data and data interpolation are important issues in the processing of seismic signals. Incomplete data happen due to problems, such as disconnection of sensors during signal acquisition or need for resampling of data at particular non-measured locations. Several techniques have been proposed to deal with these issues, for instance: matching pursuit [1], wave equation-based interpolation [2],

 ISBN 13: 978-84-613-5510-5

autoregressive spectral extrapolation [3], prediction error filtering interpolation [4], and Fourier reconstruction [5]-[7]. The problem of seismic signal reconstruction can be posed as an inverse problem, where from incomplete data a recovering of the complete seismic wavefield is attempted.

In this paper, we propose a new method for seismic trace reconstruction based on Wiener structures that are composed of a linear processor followed by a non-linear processor. Wiener, Hammerstein, and Wiener-Hammerstein structures represent simple methods to build non-linear predictors capable of prediction for non-Gaussian data [8]-[10].

Several examples of seismic data reconstruction are included using real seismic data from a public dataset of BP Amoco [12]. In order to evaluate the proposed method, maps with simulated missing data were generated by subtracting zones of 4 contiguous A-scans from the complete data set. The quality of the data reconstruction was assessed using mean squared error (MSE), Kullback-Leibler distance (KLD), and absolute values of the differences in variance and kurtosis between the original and the reconstructed data [13].

Prediction Based on Wiener Systems

Wiener structures have been used to model non-linear systems in quite different applications, such as blind deconvolution in digital communications [14] and prediction applied to infrared signals [15]. The structure consists of a linear stage followed by a zero-memory non-linear stage (see Figure 1).

From Figure 1, $x_p(n+l)$ is the output of the linear predictor (i.e. $x_p(n+l)$ is the linear prediction of $x(n+l)$ from the past samples $x(n), x(n-1), \ldots, x(n-N+1)$). N is the order of the linear predictor and l the prediction time lag. Thus, the output of the Wiener system is the conditional mean defined as

$$G\left(x_p(n+l)\right) = E\left[x(n+l)/x_p(n+l)\right] \tag{1}$$

Assuming stationarity, a sample estimate of $G(\cdot)$ can be made. In this paper we implement Wiener structures with kriging or cross correlation for the linear stage and direct sample estimation or a third order polynomial approximation for the non-linear stage. Let us review the techniques applied in the two stages of the Wiener structure.

 ISBN 13: 978-84-613-5510-5

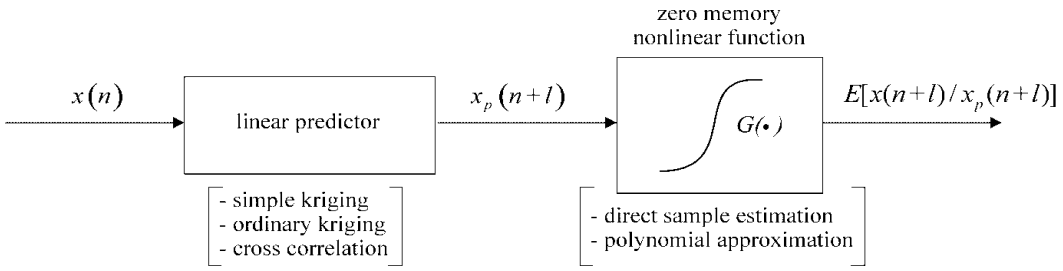


Figure 1. Scheme for nonlinear prediction.

Linear Stage

Cross correlation

This method consists of assigning a weight to each point involved in the prediction. The weights are obtained from the linear transform that minimizes the mean squared error $E\left[\left\|\mathbf{w}-\mathbf{w}_{pD}\right\|^2\right]$, where $\mathbf{w}$ are the original signal values $\mathbf{w}=[w(n+1),\ldots,w(n+D)]^T$ and $\mathbf{w}_{pD}=\mathbf{H}*[w(n+1),\ldots,w(n+D)]^T$ are the predicted values. This problem is a particular case of the Wiener-Hopf equation, and the optimum weight matrix $\mathbf{H}$ is obtained by means of

$$\mathbf{H}=\mathbf{R}_{xw}*\mathbf{R}_{ww}^{-1} \tag{2}$$

The generic elements of these matrixes are: $R_{ww}(i,j)=R_w(i-j), i=1,\ldots,N,$ $j=1,\ldots,N;$ and $R_{xw}(i,j)=R_w(i+j-1), i=1,\ldots,D, j=1,\ldots,N,$ being $R_w(m)$ the autocorrelation function of the signal, N the number of samples used and D the number of values to be predicted.

Kriging

This algorithm estimates the weights $\lambda_i(\mathbf{r})$ such that the estimator for the interpolated value $\hat{Z}$ has an optimal relationship with a given set of data values $Z(\mathbf{r}_i), i=1,\ldots,N$. Thus,

$$\hat{Z}(\mathbf{r})=\sum_{i=1}^{N}\lambda_i(\mathbf{r})\,Z(\mathbf{r}_i) \tag{3}$$

where $\mathbf{r}$ is the position vector.

 ISBN 13: 978-84-613-5510-5

The restrictions, minimum residual variance $\sigma_e^2 = E\left[\left(\hat{Z}(\mathbf{r}) - Z(\mathbf{r})\right)^2\right]$ and $E\left[\hat{Z}(\mathbf{r}) - Z(\mathbf{r})\right] = 0$, impose the condition that the estimator is both unbiased and gives the least dispersion.

Kriging established the concept of structural analysis. The variogram indicates the degree of correlation between values of the variable as a function of distance. The definitions that are relevant to kriging are the covariance (C) and the semivariogram (γ). They are defined by

$$\begin{gathered}\gamma(\mathbf{r}_1, \mathbf{r}_2) = var\{Z(\mathbf{r}_1) - Z(\mathbf{r}_2)\}/2 \\ C\big(Z(\mathbf{r}_1), Z(\mathbf{r}_2)\big) = E[(Z(\mathbf{r}_1) - E[Z(\mathbf{r}_1)])(Z(\mathbf{r}_2) - E[Z(\mathbf{r}_2)])]\end{gathered} \tag{4}$$

with $var\{\cdot\}$ being the variance. We used two different methods. Simple kriging obtains the weights by solving the system of equations (5):

$$\begin{bmatrix} C(Z(\mathbf{r}_1), Z(\mathbf{r}_1)) & \cdots & C(Z(\mathbf{r}_1), Z(\mathbf{r}_N)) \\ \vdots & \ddots & \vdots \\ C(Z(\mathbf{r}_N), Z(\mathbf{r}_1)) & \cdots & C(Z(\mathbf{r}_N), Z(\mathbf{r}_N)) \end{bmatrix} \begin{bmatrix} \lambda_1(\mathbf{r}) \\ \vdots \\ \lambda_N(\mathbf{r}) \end{bmatrix} = \begin{bmatrix} C(Z(\mathbf{r}_1), Z(\mathbf{r})) \\ \vdots \\ C(Z(\mathbf{r}_1), Z(\mathbf{r})) \end{bmatrix} \tag{5}$$

On the other hand, ordinary kriging assumes an unknown, constant trend μ. The system of equations to be solved changes to (6):

$$\begin{bmatrix} \gamma(\mathbf{r}_1, \mathbf{r}_1) & \cdots & \gamma(\mathbf{r}_1, \mathbf{r}_N) & 1 \\ \vdots & \ddots & \vdots & 1 \\ \gamma(\mathbf{r}_N, \mathbf{r}_1) & \cdots & \gamma(\mathbf{r}_N, \mathbf{r}_N) & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1(\mathbf{r}) \\ \vdots \\ \lambda_N(\mathbf{r}) \\ \mu \end{bmatrix} = \begin{bmatrix} \gamma(\mathbf{r}_1, \mathbf{r}) \\ \vdots \\ \gamma(\mathbf{r}_N, \mathbf{r}) \\ 1 \end{bmatrix} \tag{6}$$

Non-linear Stage

Direct sample estimation

The non-linear function that relates known to predicted data is estimated by a sliding window,

$$G\big(x_{pl}(i)\big) = \frac{1}{\Delta} \sum_i^{i+\Delta-1} x_l(i) \tag{7}$$

where $\mathbf{x}_{pl}$ is a vector with the predicted data sorted from highest to lowest, and $\mathbf{x}_l$ is a vector with the known values corresponding to each of the predicted values. The window defined in (7) is rectangular, but any kind of window can be used.

 ISBN 13: 978-84-613-5510-5

Polynomial approximation

In [15] we developed the following one-dimensional polynomial approximation for the non-linearity of the Wiener system. Assuming that $x \equiv x_l(n+l)$ and $x_p \equiv x_{pl}(n+l)$,

$$G(x_p) = \sum_{m=1}^{\infty} \frac{1}{m!} C_m(x, x_p) H_m(x_p) \tag{8}$$

where $C_m(x, x_p)$ is the cross-cumulant defined as $C_m(x, x_p) = cum\left(x, \overbrace{x_p, \dots, x_p}^{m\ times}\right)$ and H_m is the Hermite polynomial of order m. For the previous application ([15]) $m \geq 3$ was found to be enough to obtain approximate Gaussian predictions.

Results and Discussion

We used a dataset created by BP Amoco corresponding to Carpatheans thrusting over the North Sea [12]. From this dataset, a single 2D cut of the whole data model was used (see Figure 2.a). This 2D model has 252 scans, located at intervals of 25 m, and 314 time samples per scan. Time sample rate is 9.9 ms starting at time t = 0. For data reconstruction purposes, we considered that four consecutive scans had failed and no data could be recovered from them. These four scans were located for offsets between 1525 and 1600 m (see Figure 2.b) for a total of 1256 missing values, 1.6% of the available data. Some acronyms were used for the different methods applied for the sake of simplicity. They are shown in Table 1.

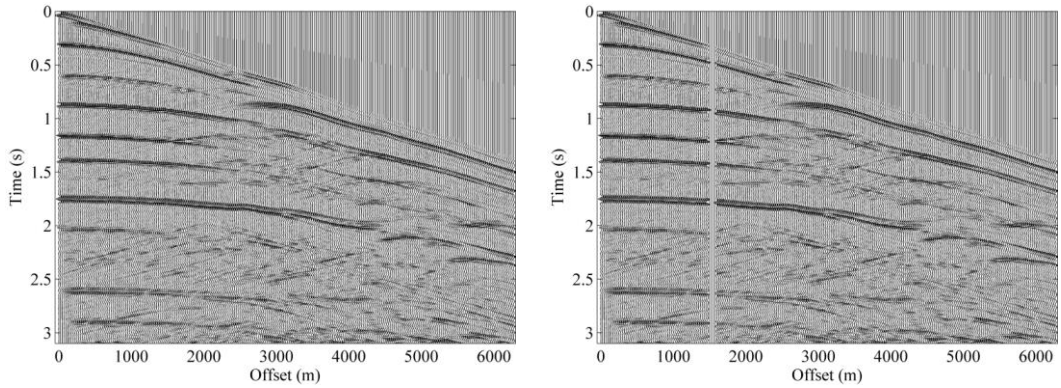


Figure 2. Complete dataset (a) and dataset after for scans were removed (b).

 ISBN 13: 978-84-613-5510-5

For a better estimation, we divided the missing traces to be reconstructed in small zones. The size of these zones was adjusted depending on the applied method, see Table 1. The zones were estimated using training data blocks from zones directly adjacent to them.

Figure 3 shows the nonlinearity present in one of the training data blocks and the estimated curves for the different methods. As we can see, the RBF neural network yields the better result in this particular case, with direct sample estimation being a close second. The polynomial fitting was not able to approximate the nonlinearity. MLP was able to model some of it but negative values were incorrectly modeled. That points to an over-fitting of the positive values in the training set.

Table 1. Acronyms used for the different methods and sizes of zones to be reconstructed. Training data blocks and zones are the same size. These sizes are in number of samples.

Acronym	Linear predictor	Nonlinear estimator	Zone size
CC	Cross-correlation		1 x 5
CC+SE	Cross-correlation	Direct sample estimation	1 x 5
CC+Poly	Cross-correlation	Polynomial fit	1 x 40
SK	Simple kriging		1 x 5
SK+SE	Simple kriging	Direct sample estimation	20 x 10
SK+Poly	Simple kriging	Polynomial fit	30 x 10
OK	Ordinary kriging		10 x 5
OK+SE	Ordinary kriging	Direct sample estimation	10 x 5
OK+Poly	Ordinary kriging	Polynomial fit	20 x 5
MLP	Multi-Layer Perceptron neural network		65 x 12
RBF	Radial Basis Function neural network		15 x 5

Figure 4.a and Figure 4.b show four different error measurements for the whole reconstructed data: Mean Squared Error (MSE), Kullback-Leibler distance (KLD), absolute difference in variance (ΔVAR) and absolute difference in kurtosis (ΔKUR). There is an abnormality in Figure 4.a for OK+Poly. This was caused by a problem in the polynomial fitting: a few of the predicted data were outside the values given in their training set, and so the resulting polynomial fit was bad and yielded values out of range. This can be further confirmed in Figure 4.b.

 ISBN 13: 978-84-613-5510-5

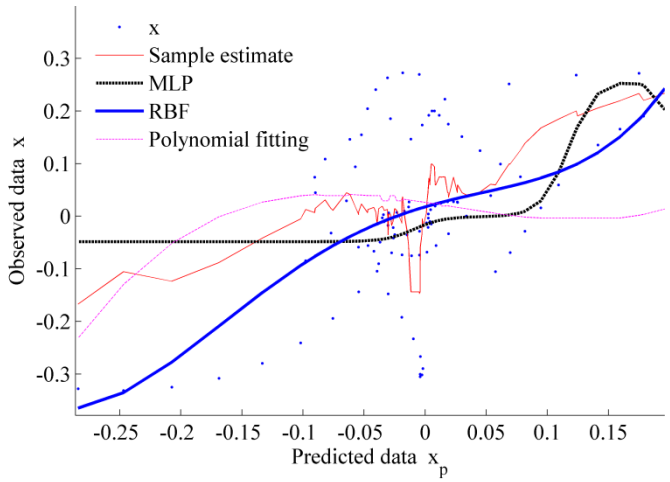


Figure 3. Estimate of the nonlinearity of a given training data block for all considered methods. x are the data to be fitted.

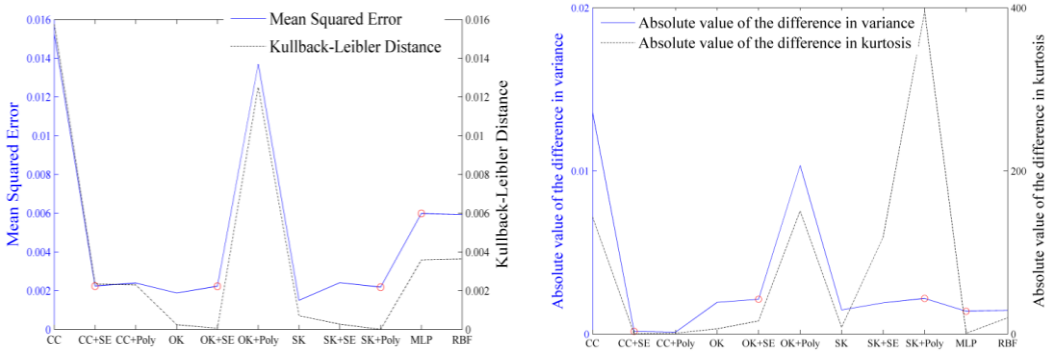


Figure 4. Error estimators for the different methods: a) Mean Squared Error (left) and Kullback-Leibler distance (right) for all considered methods; b) absolute difference in variance (left) and kurtosis (right) between predicted data and real values. Best predictions are marked with circles.

In Figure 4.b, variance and kurtosis for methods using polynomial approximations of the nonlinearities are further from the desired values. This is especially true in the case of kurtosis, which is a measure of the amount of outliers for a given data distribution. Real data had a kurtosis of 31.90 and a variance of 2.2·10-3.

Figure 5 shows zoom-ins of two different reconstructed maps of missing data. The map given in Figure 5.a was obtained with CC+SE, while the second map (Figure 5.b) was obtained using a MLP neural network. The CC+SE results are better that those obtained with the MLP neural network Comparing them we appreciate CC+SE obtains results similar to the desired result; as a matter of fact, reconstructed values are hardly distinguishable from neighboring values.

 ISBN 13: 978-84-613-5510-5

It is worth noting that while CC+SE does not achieve the best possible MSE, it has a low Kullback-Leibler distance and higher-order statistics very close to those of the real data. Part of the reason for its comparatively high mean squared error is that its values are more attenuated than those obtained with other methods.

Conclusion

We demonstrated the feasibility of a procedure based on Wiener systems for recovering realistic estimates of data structures of missing seismic traces. The versatility of the procedure allows linear and non-linear dependencies of the data to be modelled using different techniques. The accuracy of the recovered data was evaluated by the mean squared error, density distance, kurtosis, and variance between the estimate and the real data. The best results were obtained using ordinary kriging in the linear part and direct sample estimation in the non-linear part of the Wiener structure.

There are several research lines open from this work, such as including priors of the data distributions, attempting other linear and non-linear techniques, and processing in the frequency domain.

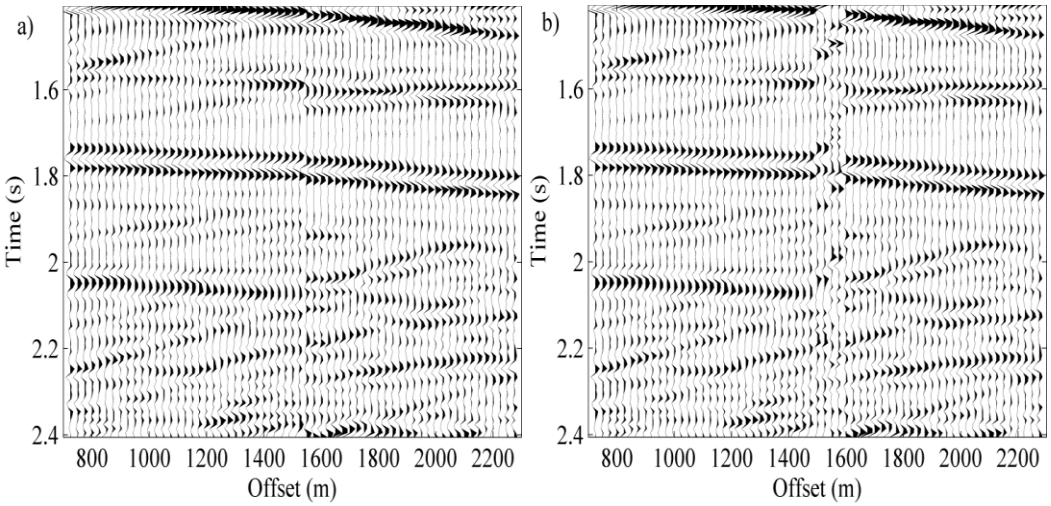


Figure 5. Wiggle plot showing data predicted using: a) CC+SE; b) MLP.

Acknowledgments

This work has been supported by the *Generalitat Valenciana* under grant PROMETEO/2010/040; the Spanish Administration and the FEDER Programme of the European Union under grant TEC 2008-02975/TEC; and the *Generalitat Valenciana* under grant GV/2009/003 within the research and development programme for Emergent Research Groups.

 ISBN 13: 978-84-613-5510-5

References

[1] F.J. HERRMANN AND G. HENNENFENT, *Non-parametric seismic data recovery with curvelet frames*, Geophysical Journal International **173** (2009) 233-248.

[2] J. RONEN, *Wave-equation trace interpolation*, Geophysics **52** (1987) 973–984.

[3] H. KARSH, *Further improvement of temporal resolution of seismic data by autoregressive (AR) spectral extrapolation*, Journal of Applied Geophysics **59** (2006) 324-336.

[4] S. SPITZ, *Seismic trace interpolation in the F-X domain*, Geophysics **56** (1991) 785–794.

[5] A.J.W. DUIJNDAM, M. SCHONEWILLE AND K. HINDRIKS, *Reconstruction of band-limited seismic signals, irregularly sampled along one spatial direction*, Geophysics **64** (1999) 524–538.

[6] B. LIU AND M.D. SACCHI, *Minimum weighted norm interpolation of seismic records*, Geophysics **69** (2004) 1560-1568.

[7] P. ZWARTJES AND A. GISOLF, *Fourier reconstruction of marine-streamer data in four spatial coordinates*, Geophysics **71** (2006) V171-V186.

[8] N. WIENER, *Nonlinear Problems in Random Theory*, M.I.T. Press, Cambridge, 1958.

[9] E.W. BAI, *An optimal two-stage identification algorithm for Hammerstein–Wiener nonlinear systems*, Automatica **34** (1998) 333–338.

[10] K. ZHANG AND L.W. CHAN, *Practical Method for Blind Inversion of Wiener Systems*, Proceedings of the IEEE International Joint Conference on Neural Networks (2004) 2163-2168.

[11] J.S. WANG AND Y.L. HSU, *Dynamic nonlinear system identification using a wiener-type recurrent network with OKID algorithm*, Journal of Information Science and Engineering **24** (2008) 891-905.

[12] J. ETGEN AND C. REGONE, *Strike shooting, dip shooting, widepatch shooting - Does prestack migration care? A model study*, 68th Annual International Meeting, SEG, Expanded Abstracts (1998) 66-69.

[13] D.J. MACKAY, *Information Theory, Inference, and Learning Algorithms*, 4th edition, Cambridge University Press, Cambridge, 2005.

[14] S. HAYKIN, *Unsupervised Adaptive Filtering Volume 2: Blind Deconvolution*, 1st edition, Wiley-Interscience, 2000.

[15] L. VERGARA AND P. BERNABEU, *Simple approach to nonlinear prediction*, Electronic Letters **37** (2001) 926-928.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Implementing a GPU fuzzy filter for Impulsive Image Noise Correction

M. Guadalupe Sánchez[1], Vicente Vidal[2], Jordi Bataller[2] and Josep Arnal[3]

[1] *Departamento de Sistemas y Computación, Instituto Tecnológico de Cd. Guzmán, 49100 Cd. Guzmán, Jal. México*
[2] *Departamento de Sistemas Informáticos y Computación, E.P.S. Gandia, Universidad Politécnica de Valencia, 46730, Grao de Gandia Valencia, Spain*
[3] *Departamento de Ciencia de la Computación e Inteligencia Artificial, Universidad de Alicante, 03071, Alicante, Spain*

emails: msanchez@dsic.upv.es, vvidal@dsic.upv.es, bataller@dsic.upv.es, arnal@dccia.ua.es

Abstract

The problem of correcting images with noise has been widely studied in the digital image processing literature, and many techniques and algorithms have been suggested for this purpose. However, due to their high computational cost, and in general their application for large sized images, the need exists to develop the same executable files in parallel. The implementation of image correction algorithms on the CUDA platform is a relatively new field. Although the platform is easy to program, it is not easy to optimize the applications due to the number of decisions that have to be made. Thus, it is necessary to perform an analysis in order to identify the best configuration for each problem. This paper reports an optimization study on the use of the CUDA platform with fuzzy metric and the concept of peer group for noise correction in images.

Key words: noise correction in images, GPU, CUDA

 ISBN 13: 978-84-613-5510-5

1. Introduction

In recent years, the incorporation of GPUs (Graphics Processor Units) in graphic cards has achieved significant improvements in computational speed, offering a high parallel processing level combined with a very competitive price. For this reason, developments based on this hardware have become increasingly widespread, not only for graphic implementations but also for general purpose applications. The most commonly used programming platform for these graphic cards is CUDA (Compute Unified Device Architecture) [10]. Its multiple fields of application include medicine, astrophysics, biology, computational chemistry and signal processing, among many others. For example, [1] provides an illustration of biomedical image processing for color and phenotype analysis. In [13], the performance obtained by processing various algorithms in several classic images was analyzed, whilst in [2], two proposals for accelerating bioinformatics applications used to analyze DNA sequences are presented. [5] reports the implementation of algorithms for hyperspectral image analysis, and another spectral imaging application is described in [3]. Finally, [7] depicts the scope of FFT (Fast Fourier Transform) for filtering images.

Whilst it is relatively easy to use the CUDA platform to program the GPU, and the process is well documented, the problem lies in the difficult task of optimizing application performance, due to several hardware restrictions and the multiple types of memories included, which are organized on several levels and display different storage capacities, different access patterns and other limitations. Thus, it is necessary to carry out a specific study in order to identify the best approach to using the resources offered by CUDA in each case. Such a study is the subject of the present paper, applied in this case to noise correction in images.

The images used in this study were all in RGB format (three color channels: red, green and blue, with values in a range of 0 to 255), with impulsive noise (where several pixels have changed the value of one of their channels to maximum or minimum, white or black respectively). Many algorithms have been proposed for correcting impulsive noise, for instance those mentioned in [4]. In the present study, the process of noise correction was divided into two steps: (1) erroneous pixel detection, also divided into two phases, and (2) the elimination of these pixels. The fuzzy metric is used [11] together with the concept of peer group $\mathcal{P}$ previously mentioned in [12]. In this concept, a set is of pixels created similar to one already given and then a decision is made, according to the cardinality of the set, as to whether the pixel should be treated as corrupted or uncorrupted by noise. In this research, we implemented an adaptation of the algorithms 1 and 3 previously proposed in [6], and algorithm 2 is modified so that each thread labels the pixel as corrupted or not.

This paper is organized as follows: In section 2, the GPU is described together with CUDA programming models. Section 3 illustrates the noise correction method employed and its implementation on CUDA. Experimental results are shown in section 4, and lastly, the conclusions are presented in section 5.

 ISBN 13: 978-84-613-5510-5

2. GPU and CUDA programming models

Recent years have witnessed a spectacular growth in the parallel calculation capacity of Graphic Processing Units (GPU), due to the high demand for video game applications. Although initially these constituted the principle application of this hardware, in November 2006 NVIDIA introduced CUDA, a technology which enables these units to be used to develop programs for other calculation purposes. Such applications have become increasingly popular in the scientific community due to their combination of reasonable costs and good calculation power.

Physically, the GPU, called a *device* in CUDA terminology, contains a set of multiprocessors. These execute programs following the SIMD (Single Instruction, Multiple Data) model, whereby each of the multiprocessor's processor clock cycles executes the same instruction applied to different data, taking into account each application if this instruction is performed by a different thread.

A device has a physical memory (the size of which may vary from 384MB to 1GB, in NVIDIA 9 series) that can be used in different ways. The main use is as global shared memory among the GPU multiprocessors. However, this memory also permits its several areas to be used in other modes:

- As local memory. Each thread may make individual use of 16KB of global memory.
- As texture. In this case, an area of the global memory is blocked in order to be used as read-only, in shared mode and optimized to store structures (arrays) of 1, 2 or 3 dimensions.
- As a constant reading area, shared between all the threads.

Internally, each multiprocessor has four kinds of memories [9]:

- A set of 32 bit registers per processor with read or write access.
- A read/write 16KB cache memory for optimizing access to the global memory, shared by all the multiprocessor processors.
- A constant read-only 64KB cache memory, shared by all the processors, which speeds up reading of the constant memory.
- A read-only cache memory called texture cache, which is shared by all the processors and accelerates reading of the texture memory.

The large variety of memories and their different features complicate the task of achieving optimum performance in programs using CUDA. One of the main issues that must be considered in order to obtain efficient programs is the coalescence of accesses to global memory.

Global memory is addressed in 16 or 32 byte displacements. Furthermore, it is possible to read 4, 8 or 16 Bytes of the global memory in a simple instruction. However, if a variable is not stored just after a memory address multiple of 16, or if its size is not a multiple of 4, more than one access must be performed, penalizing performance. In addition, when the GPU contains a large number of threads, this problem intensifies since even two threads may compete for access to the same memory area. When this happens, it is said that the accesses do not have coalescence [8], that is, they are not well aligned, a situation to be avoided at all costs. The most elemental method (although there are others) is for a thread with

 ISBN 13: 978-84-613-5510-5

value *i* access to a value *i* variable (or array index), and the access address to that point to become multiples of 16.
According to the CUDA programming model, it is necessary to distinguish between the code that is executed on the CPU ("host code" following CUDA terminology) and the code that is executed on each GPU core ("device code"). In particular, the function which we propose here for execution on the GPU takes the name of "kernel". A kernel is processed in parallel by a set of threads which will apply the instructions of that function to a different part of the data in the memory. When the GPU has several multiprocessors, the threads must be grouped in "blocks" to clarify assignment of the execution of a block to a specific GPU multiprocessor (nowadays, a block can contain a maximum of 512 threads). This particularity implies that threads can only be synchronized if they belong to the same block. Once a block of threads has finished, new blocks are launched in the empty multiprocessors.
Meanwhile, the CPU code must analyze the following steps in order to launch execution of the GPU kernels:

- Copy data from the host memory to the device memory.
- Run the kernel, deciding on the number of threads (and their organization in blocks) needed for the processing.
- Wait for kernel execution to finish before moving data from the GPU memory to CPU memory.

A kernel has a predefined variable enabling each thread to recognize its ID and, from this value, identify data to be processed. Initially, data are available in the GPU global memory but if coalescence problems arise, it becomes necessary to consider whether it would be advisable to copy data from the global memory to the cache memories of each multiprocessor in order to process them (returning them later to the global memory). Another option is to use textures, if part of the information is read-only and the information contained is organized dimensionally (in arrays).
As can be seen, the design problems of a CUDA program are:

- Deciding the number of threads and their organization in blocks.
- Deciding at each moment the best location (among the different available memories) for the input and output data, and how to access to them, performing the necessary copies at the appropriate time.

3. Image noise correction and implementation strategies with CUDA

In the present study, the process of image noise correction was divided into two steps. The first step was to detect erroneous pixels and the second, to eliminate them. The process was divided into two steps so that in the elimination step, the condition of neighboring pixels could be taken into account (whether previously evaluated as corrupted or not) when defining the new values for corrupted pixels.

For the detection stage, the fuzzy distance between the vectors of the color image x_i and x_j was used, which is given by the following function:

 ISBN 13: 978-84-613-5510-5

$$M(x_i, x_j) = \prod_{l=1}^{3} \frac{mín\{x_i(l), x_j(l)\} + k}{máx\{x_i(l), x_j(l)\} + k}.$$

where (x_i (1), x_i (2), x_i (3)) is the color vector for the pixel x_i in space color RGB. In [6], k =1024 was shown to be an appropriate setting, and this was therefore the value that was used in the present study. Fuzzy distance measure is employed in peer group $\mathcal{P}(x_i, d)$, giving the central pixel x_i in a window W with size n x n and $d \in [0,1]$; $\mathcal{P}(x_i, d)$ represents the set:

$$\{x_j \in W \quad : M(x_i, x_j) \geq d\}.$$

The peer group associated with the central pixel of W is a set consisting of the central pixel x_i and its neighbors belonging to W, whose distance from x_i exceeds d. After several tests, d =0.95 proved to be a good value for d.
Detection was also divided into two phases, with two kernels for execution. In the first step (described in algorithm 1), the image was divided into (N_1 x N_2)/n windows W disjointed with dimension n x n, where $n \in \{3,5,7, \ldots\}$; in the present case, n =3 was considered and the kernel was configured to set (N_1 x N_2)/n threads. Each thread analyzes their n x n pixels of W, x_i as the central pixel in W, and (f, c) for the thread. Given the parameter d, each thread calculates its peer group.

Require: m, d, image noise
1: Each thread defines the row and column corresponding to the central pixel x_i of the windows W disjoint.
2: Each thread builds its Windows W of pixels.
3: Calculate $\mathcal{P}(x_i, d)$ in W.
4: **If** ($\sharp\mathcal{P}(x_i, d) \geq m+1$)*
5: thread mark:
6: $\forall x_j \in \mathcal{P}(x_i, d)$, x_j as uncorrupted.
7: $\forall x_k \in W, x_k \notin \mathcal{P}(x_i, d)$, x_k as undiagnosed.
8: **else**
9: thread mark:
10: pixel x_i as provisionally corrupted.
11: $\forall x_j \in W, j \neq i, x_j$ as undiagnosed.
12: **end if**

Algorithm 1: S1P1 - Step 1 Phase 1 Detection of erroneous pixels.

Once the value of m has been established (the optimal value according to [6], if the window is n x n, is m =n - 1; in our case n =3 and m =2), if the peer group contains

 ISBN 13: 978-84-613-5510-5

* $\sharp$ representa la cardinalidad del conjunto

at least m+1 elements, then the thread labels the central pixel x_i as uncorrupted and peer group members as uncorrupted. On the other hand, if the central pixel is declared as corrupted, members of the peer group are left as undiagnosed, and also as members of the window outside the peer group.
In the second step (described in algorithm 2), the kernel was reconfigured to set more threads in each block and many threads were set so that each thread processed one item of pixel data. The thread corresponding to the pixel x_i (central pixel of a n x n window) labeled as undiagnosed calculates the peer group and if this satisfies the cardinality m+1, the central pixel is diagnosed as uncorrupted; if not, it is diagnosed as corrupted.

Require: m,d, threads corresponding to the pixels labeled as not diagnosed
Each thread defines the row and column corresponding to the central pixel x_i.
Each thread builds its Windows W of pixels.
1: Calculate $\mathcal{P}(x_i, d)$ in W.
2: **If** $(\sharp\mathcal{P}(x_i, d) \geq m+1)$
3: thread mark:
4: pixel x_i as uncorrupted.
5: **else**
6: thread mark:
7: pixel x_i as corrupted.
10: **end if**

Algorithm 2: S1P2 - Step 1 Phase 2 Detection of erroneous pixels.

In the correction step (described in algorithm 3), we used an equal number of threads for kernel 2. The threads corresponding to pixel x_i apply filtering by substitution where this has been labeled as corrupted in step 1. The replacement value is determined by creating a window W for the pixel x_i and calculating the AMF (Arithmetic Mean Filter) on corrupted pixels W. Threads with uncorrupted pixel values continue as before.

Require: n, threads corresponding to the pixels labeled as corrupted
Each thread defines the row and column corresponding to the central pixel x_i.
Each thread builds its Windows W of pixels.
1: Calculate the AMF of uncorrupted pixels W.
2: Thread corresponding to the pixel x_i replaces value obtained by the AMF.

Algorithm 3: S2 - Step 2 Elimination of erroneous pixels.

To determine the number of threads per block that best fits the application, a heuristic study concluded that 64 x 64 threads per block gave lowest computational costs. The function *cudaMallocPitch* was used to ensure optimal global memory alignment of the pixels through textures.

 ISBN 13: 978-84-613-5510-5

As can be observed in section 2, a series of choices must be considered in order to implement an algorithm on CUDA. In the present case, two strategies were employed:

a. Storing the image with 3 channels per pixel or 4 channels per pixel.
b. Accessing data through the texture memory or not

Deciding whether to store the image in RGB (three channels) or RGBA (3 channels + padding) is necessary since the RGB format uses 3 bytes and thus does not achieve access coalescence. However, if a padding byte is added, even using a time for it, the accesses will fit in blocks of 4 bytes and performance may improve. Furthermore, the addition of a fourth byte can be used to indicate pixel status: corrupted, uncorrupted or undiagnosed.
In addition, we evaluated the improvements deriving from the use or not of texture memory. For this research, two textures were used: one in the first phase of detection, another in phase 2 and in the elimination phase. In all cases, these were used with the purpose of reading x_i neighboring pixels in the fuzzy and peer group calculations. In the detection phase, and once the pixels are in the device memory, each block thread reads its corresponding pixel for analysis, together with its neighbors, through two texture. At the end of this stage, the RGB fill channel contains the state of the pixel: corrupted, uncorrupted or undiagnosed.
In the elimination phase, each thread reads its corresponding pixel, together with its neighbors, through the two texture. Once this is completed, the new values are written onto the pixel which has been analyzed.

4. Experimental results

This section presents the results obtained for the implementations discussed in section 3. The CPU used was a Mac OS X Intel Xeon Quad-Core processor at 2 x 2.26 GHz with 8GB memory. The GPU was an NVIDIA GeForce GT 120 with 512MB of memory. Our implementation used C language. Many images are used in the area of image processing; for the present research, the *lenna* image [6] was employed, with RGB format square dimensions 256, 512, 1024 and 2048 pixels and 5 and 10% noise impulse.
For each algorithm, we designed both the CPU serial code and the GPU parallel code and then compared execution time. When calculating execution time, data transfer time from host memory to device memory was not considered.
The first experiment on GPU was to run the three algorithms for correction of erroneous pixels in RGB and RGBA format (strategy *a* Section 3), whilst the second experiment used textures (strategy *b* of section 3). Table 1 and Figure 1 show a comparison of computational costs obtained by the process described.

 ISBN 13: 978-84-613-5510-5

Size	5% noise			10% noise		
	RGB	RGBA	Texture	RGB	RGBA	Texture
246	4.25	3.09	1.79	4.77	3.47	1.95
512	20.56	12.59	7.11	24.38	14.41	7.88
1024	87.14	51.01	29.33	105.80	59.21	32.50
2048	408.67	250.96	136.91	503.79	278.29	141.24

Table 1.Processing time (msec) for the CPU and GPU-based implementations.

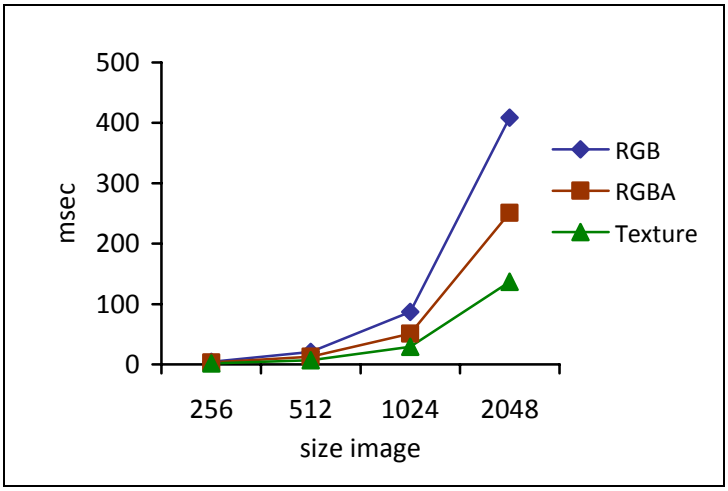


Figure 1.Comparation GPU for RGB, RGBA and Texture.5% impulse noise.

It can be seen that in the case of using the RGBA format on GPU with 5% noise, the improvement is at worst 27% for the 256 image size and at best, 41% for the 1024 image size, compared to RGB implementation. On the other hand, with 10% noise, the performance is at worst 27% for the 256 image size and at best, 44% for 2048 image size. The best performance was obtained when using GPU with texture, since the improvement is approximately 45% with 5% noise and 43% to 49% with 10% noise compared to RGBA (without texture) implementation.
The results regarding relative GPU time spent by each kernel are shown in Table 2.
As can be seen, the computational cost of S1P1 is less than that for S1P2 in all cases. For RGB strategy S2, computational cost is less than that for S1P2 with image sizes 2048.

 ISBN 13: 978-84-613-5510-5

strategy	kernel / size	Step 1. Phase 1 (S1P1)	Step 1. Phase 2 (S1P2)	Step 2 (S2)
GPU time. RGB	256	0.59	1.77	1.90
	512	3.53	8.48	8.55
	1024	17.41	34.61	35.12
	2048	77.56	175.03	156.08
GPU time. RGBA	256	0.44	1.29	1.36
	512	2.33	4.61	5.64
	1024	11.03	17.40	22.58
	2048	55.08	93.97	101.91
GPU time. Texture	256	0.43	0.63	0.73
	512	1.69	2.56	2.88
	1024	7.34	10.22	11.76
	2048	40.36	44.05	52.51

Table 2. Computational times for kernels.

To conclude the comparison, CPU times are compared with GPU times (image with textures) in Table 3 and Figure 2.

size	Textures GPU	CPU	Speedup
246	1.79	111.88	62.30
512	7.11	467.63	65.69
1024	29.33	1932.72	65.89
2048	136.92	7907.12	57.75

Table 3. Speedup for different image sizes.

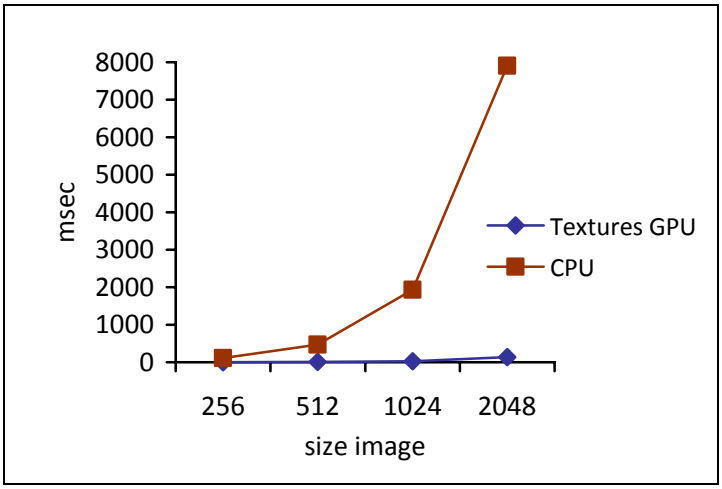


Figure 2.Comparison GPU (texture strategy) and CPU.

 ISBN 13: 978-84-613-5510-5

As can be seen, excellent results were obtained by using optimization with textures on GPU compared with CPU for this application.
Figure 3 show the speedup (speed performance of one implementation with respect to another) when comparing the sequential version running on CPU and the parallel version on GPU using textures. As can be seen, even the worst result for the GPU version is 57 times faster than sequential, which is an excellent outcome.

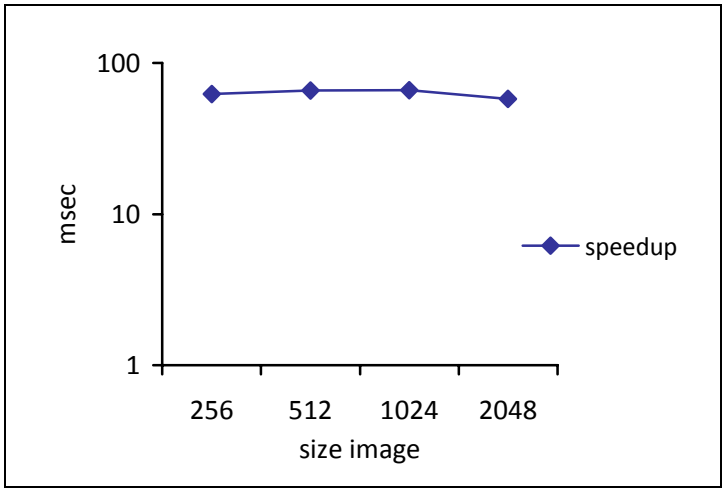


Figure 3. GPU vs CPU speed-up for different image sizes.

Finally, although not the objective of this research, we would highlight the quality obtained for impulse noise correction using the method described in Section 3, of 5 and 10% in different image sizes compared with the result in [6]. Table 4 shows the results using the measures PSNR (Peak Signal-to-Noise Ratio) [12], MAE (Mean Absolute Error) and NCD (Normalized Color Difference).

	5%			10%		
Filter	MAE	PSNR	NCD (10 -2)	MAE	PSNR	NCD (10 -2)
AMF	16.415	22.422	7.358	16.825	21.947	8.798
VMF	2.775	32.051	1.781	2.874	31.767	1.847
DDF	2.764	31.808	1.663	2.982	31.325	1.788
BVDF	2.985	31.382	1.720	3.248	30.773	1.869
FIVF	0.414	35.685	0.482	0.743	34.158	0.669
PGF m=2	0.404	37.996	0.297	0.696	35.257	0.553
FMPGF m=2	0.521	36.196	0.370	0.900	33.390	0.687
GPU FMPGF m=2	0.500	36.220	0.420	1.187	33.989	0.790

Table 4.Quality Comparison.

As can be seen, the quality achieved by implementing this algorithm in parallel is competitive.

 ISBN 13: 978-84-613-5510-5

5. Conclusions

In this paper we have described a study conducted in order to determine the best method for implementing image correction processing in RGB format with impulsive noise on a GPU using a CUDA platform. This processing was divided into two steps: noise detection and noise elimination. For detection, the fuzzy measure and the concept of peer group were used, obtaining a set with pixels similar to a given and then deciding, according to the cardinality of the set, whether the pixel was noisy or not. In the correction stage, corrupted pixel values were replaced by calculating the mean of those neighbors not labeled as corrupted in the first stage.

The experimental results show that if the accesses are coalescing, the results improve significantly. This is demonstrated by the versions in which we added one padding byte to the three-byte RGB format dedicated to each pixel. Additionally, this byte is useful for storing the assessment of whether the associated pixel is corrupted or not, making it unnecessary to access another area of global memory for this purpose. Furthermore, we have also shown that the use of textures for accessing data in global memory is less complex compared, for example, with versions used in multiprocessor caches to avoid accesses to the global memory (since this would have to include codes performing the copy.) Likewise, through the use of textures we have obtained outstanding results in speed, compared to sequential versions of the implementation, in the order of approximately 65%.

Based on the results of this study, we would suggest two future lines of research. Firstly, optimal utilization of CUDA on GPUs would be interesting in order to obtain implementations for large images, spread the processing load between multiple GPUs available in the system and evaluate performance. The second line of research would be GPU implementation of the improvements in computational times and quality achieved using the implementation developed in this work.

6. Acknowledgements

This work was funded by the Spanish Ministry of Science and Innovation (Project TIN2008-06570-C04-04) and Ma. Gpe. would also like to acknowledge DGEST-ITCG for the scholarship awarded through the PROMEP program (Mexico).

References

[1] A. Ruiz, M. Ujaldón, J. A. Andrades, J. Becerra, K. Huang, T. Pan, J. Saltz, "*The GPU on biomedical image processing for color and phenotype analysis*" (2007)

[2] A. J. Peña, J. M. Claver, A. Sanjuan, V. Arnau, "*Análisis paralelo de secuencias de ADN mediante el uso de GPU y CUDA*". ANACAP 2008, 20-21 Noviembre 2008

[3] A. J. Plaza, J. Plaza, S. Sánchez, A. Paz, "*Optimization of a Hyperspectral Image Processing Chain Using Heterogeneous and GPU-Based Parallel Computing Architectures*" . Proceedings of the International Conference on

 ISBN 13: 978-84-613-5510-5

Computational and Mathematical Methods in Science and Engineering CMMSE 2009, 30 Junio, 1–3 Julio 2009.

[4] R. C. Gonzalez, R. E. Woods. "*Digital Image Processing*". Person Education. 3rd. ed.,(2008). ISBN 0-13-505267-X 978-0-13-505267-9

[5] J. Setoain, C. Tenllado, M. Prieto, D. Valencia, A. Plaza, J. Plaza, "*Procesamiento paralelo de imágenes hiperespectrales utilizando unidades de procesamiento gráfico domesticas*". XVII Jornadas de Paralelismo 2006, Albacete, España, Septiembre, 2006

[6] J. G. Camarena, V. Gregori, S. Morillas, A.Sapena, "*Fast detection and removal of impulsive noise using peer group and fuzzy metrics*", Journal of Visual Communication and Image Representation, 19 (2008) 20-29

[7] M. Kenneth, A. Edward, "*The FFT on a GPU*". SIGGRAPH/EUROGRAPHICS Conference On Graphics Hardware.San Diego, California, 2003, pp.112-119

[8] NVIDIA Corporation, NVIDIA CUDA C Programming Best Practices - Guide CUDA Toolkt 2.3, http://www.nvidia.com/page/home.html , 2009

[9] NVIDIA Corporation, NVIDIA Programming Guide Version 2.3.1, http://www.nvidia.com/page/home.html , 2009

[10] NVIDIA, http://www.nvidia.es/page/home.html

[11] S. Morillas, V. Gregori, G. Peris, A. Sapena, "*Local self-adaptive fuzzy filter for impulsive noise removel in color images*", Science Direct Signal Processing 88 (2008) 390-398.

[12] B. Smolka, A. Chydzinski, "*Fast detection and impulsive noise removal in color images*", Real-Time Imaging 11 (2005) 389-402.

[13] Z. Yang, Y. Zhu, Y. Pu, "*Parallel Image Processing Based on CUDA*", International Conference on Computer Science and Software Engineering 2008, Dec 12-14.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Gas Transport in the Near-Surface Porous Layers of Cosmic Bodies

Yu. Skorov[1,2], R. van Lieshout[3], J. Blum[1], H. U. Keller[1]

[1] *Institute for Geophysics and Extraterrestrial Physics, Technical University of Braunschweig, Germany*

[2] *Max Planck Institute for Solar System Research, Germany*

[3] *Astronomical Institute Anton Pannekoek, University of Amsterdam, The Netherlands*

e-mail: skorov@mps.mpg.de

Abstract

The gas transport through non-volatile random porous media is investigated numerically. We extend our previous research of the transport of molecules inside the uppermost layers of space bodies, assess the validity of the simplified capillary model and its assumptions to simulate the gas flux trough the porous non-volatile layer as it has been applied in planetary physics. A microphysical computational model for molecular transport in random porous media formed by packed spheres is presented. The main transport characteristics such as the mean free path distribution and the permeability are calculated for a wide range of model parameters and compared with those obtained by more idealized models. Finally a practical way is suggested to adjust the algebraic Clausing formula taking into consideration the nonlinear dependence of permeability on layer porosity. The retrieved dependence allows us to accurately calculate the permeability of layers whose thickness and porosity vary in the range of values expected for the near-surface regions of a cometary nucleus.

Key words: gas, porous media, planetary physics

As the physical models of the surface layer structure become more sophisticated and possibly more realistic (e.g. material may be porous, its composition may include a variety of volatile and non-volatile additives, etc.) determination of the

 ISBN 13: 978-84-613-5510-5

effective gas production is becoming a more and more complicated problem. Various aspects of this fundamental problem of planetary physics were considered by us in numerous articles published over the past few years (e.g., [1], [2]). Here we focus on the release of gas through a porous non-volatile surface layer of a cosmic body.

1. Capillary models of gas transfer through a porous dust layer

The vast majority of publications containing theoretical modeling of gas transfer in the uppermost porous layers of a space body uses one and the same basic algebraic formula to calculate effective gas activity - namely, the formula of Knudsen, that describes the mass flow rate per unit capillary area

$$\Psi_K = \left(\frac{32m}{9\pi k}\right)^{1/2} \frac{r}{L}\left(\frac{P_t(T_t)}{\sqrt{T_t}} - \frac{P_b(T_b)}{\sqrt{T_b}}\right)$$

where r is the channel radius, L is the length, (P_b, T_b) and (P_t, T_t) are pressures and temperatures at the bottom and top of the channel, respectively. This formula is very popular in planetary physics and has been used without change for almost thirty years [3]. The Knudsen formula refers to a very simple model of the porous medium as a bundle of disjoint, straight cylindrical capillary channels of radius r, and length L with diffusively scattering walls. The gas in the channel is in the free-molecular regime, i.e. the intermolecular collisions are negligible, whereas scattering by the walls plays a major role. A simple generalization of the Knudsen approach exists which allows us to calculate the rarefied gas flow in a Knudsen regime through a cylindrical tube of arbitrary length with diffusion scattering walls with high accuracy. This is the Clausing formula which apparently was for the first time considered in cometary physics by Steiner [4]:

$$\Psi_C = \left(\frac{m}{2\pi k}\right)^{1/2} \frac{20 + 8(L/r)}{20 + 19(L/r) + 3(L/r)^2}\left(\frac{P_t}{\sqrt{T_t}} - \frac{P_b}{\sqrt{T_b}}\right)$$

He also performed a quantitative comparison of the Knudsen and the Clausing formulas and showed that even for a sufficiently long channel in which the ratio of length to radius equals 10, the gas flow calculated using the former formula is overestimated by 50%, while for short tubes ($L \approx r$) the relative error is about eight times higher. Later Skorov et al. [1] applied this approach for modeling a gas flow through a cylindrical tube with icy walls of varying temperature.

Both formulae considered above were obtained for the molecular gas flow in a tube, while our ultimate goal is the calculation of gas flow in natural stochastic porous media, where statistically the length of a void is the same in any arbitrary

 ISBN 13: 978-84-613-5510-5

direction. The transition from the model of a single pipe to the model of porous medium can not be realized in a simple way when the capillary approach is used. The other serious obstacle for use of the capillary models in planetary physics applications is the anisotropic character of the model generated medium. In order to compensate for this problem an additional model parameter the above-mentioned tortuosity, τ, is usually added. The formal purpose is to replace a straight cylindrical channel by a broken one of greater length and thereby to make the model environment more isotropic. Unfortunately the tortuosity can be introduced in a non-unique way into the simplest capillary model. For a natural porous media the tortuosity is an empirical quantity, that should be determined from the experiments. Note that, in contrast to porosity, tortuosity can not be easily measured directly. Often it is derived from independent measurements of the Knudsen diffusion coefficient and porosity.

2. Models of granular packed bed

There is an alternative way to describe the Knudsen diffusion in a porous medium - a way where the molecular gas flow is regarded to be external to the nonvolatile matrix: “flow around obstacles”. The matrix itself is constructed (composed) of elementary scattering or absorbing objects, for example, spheres. Interaction of gas molecules with the surfaces of matrix elements is similar to interaction with the walls of a capillary in the models of the first type. Thus, the weakening of gas flow and reduction of effective diffusion rate can be well modeled as before.

We generate porous media consisting of mono-disperse spheres using two different methods: ballistic deposition (RBD) and random sequential packing (RSP). For RBD spheres are dropped one by one vertically into a control volume where they touch either the bottom or another sphere. The porosity of media generated by RBD is about 0.85, with a more compact base and a fluffier top. For RSP spheres are placed one by one at random locations within a control volume. Locations that would result in overlapping spheres are rejected. This results in a homogeneous, isotropic medium with a porosity that can be determined by specifying the total number of spheres used. We generated media with porosities of 0.65, 0.70, 0.75, 0.80 and 0.85.

In order to estimate the transport properties of the model media we apply the random walk algorithm used extensively in studies of disordered granular media. The major characteristics of a porous layer that are important for planetary applications are the permeability and the resulting return flux. These characteristics are examined by tracing the geometric paths of a large number of test particles through the medium using the Monte Carlo method. We use 100.000 test particles for each simulation. Since the outflow is assumed to be rarefied, intermolecular collisions and particle velocities are not considered. Therefore, the geometric paths of test particles only are a result of their starting direction and

 ISBN 13: 978-84-613-5510-5

subsequent interactions with the spheres that represent the nonvolatile matrix. These interactions are modeled as either specular reflections or diffuse scattering.

3. Results and conclusions

The present work aims at three primary goals:

- Revise or adjust the capillary models used in planetary physics to describe the transport of sublimation products through porous nonvolatile layer accurately.
- Present alternative description of porous media based on ballistic deposition and random sequential packing methods. Use direct statistical simulation to retrieve major geometrical and transport properties of model media as a function of porosity and layer thickness.
- Suggest a way to adjust the Clausing formula taking into consideration the nonlinear dependence of permeability on layer porosity.

In view of these goals we summarize below the main results:

Knudsen's formula is not applicable for modeling the gas transport through short channels. Satisfactory agreement with experimental data is achieved only when the channel radius is much smaller than its length. As an alternative, Clausing's formula can be used. This formula gives an exact agreement with the experimental data for straight cylindrical channels with an arbitrary ratio of radius to length. However, the spatial anisotropy of the capillary model leads to the fact that its transfer characteristics are highly different for different directions. The transition from the permeability of one channel to the permeability of the medium can not be accurately and correctly generalized. Adding an additional linear factor - tortuosity to the formula does not solve the problem, but on the contrary, only confuses the situation.

To avoid these inconsistencies accurate statistical calculations are performed for media formed either by ballistic deposition (RDB) of test particles or by random filling of a control volume (RSP). Two types of interaction of molecules with scattering spheres are tested: diffuse and specular scattering. It turns out that for the random model of the porous medium the permeability is virtually independent of the type of interaction. We show that a relatively small variation of porosity (not more than 30%) leads to a strong change of permeability. The permeability depends on the medium porosity in a nonlinear way.

In order to overcome the resource consuming calculations for direct use of the statistical models, we present a practical way to calculate the effective permeability. We preserve the overall structure of Clausing's formula, accurately describing the kinetics of transport through a single cylindrical capillary. To take into consideration the porosity of the medium in an appropriate manner we

 ISBN 13: 978-84-613-5510-5

assume that the effective radius of the capillary is an unknown function of porosity. The explicit form of this functional dependence is derived from a nonlinear approximation based on statistical modeling results. Thus, the effective permeability, as before, depends on the thickness of the layer and its effective pore size, which in turn is a function of porosity. The retrieved algebraic expression allows us to accurately calculate the permeability of layers whose thickness and porosity vary in the range of values expected for the near-surface regions of a cometary nucleus. The simplicity of this approach makes it practical to include the computational block that accurately describes the transport of gas in the overall thermal model of a cometary nucleus.

This work was supported by the German Research Foundation (DFG grant BI 298/9-1).

4. References

[1] Skorov, Yu. V.; Kömle, N. I.; Keller, H. U.; Kargl, G.; Markiewicz, W. J. Icarus, 153, 180-196, 2001.
[2] Davidsson, B. J. R.; Skorov, Yu. V. Icarus, 168, 163-185, 2004.
[3] Fanale, F. P.; Salvail, J. R. Icarus, 60, 476-511, 1984.
[4] Steiner, G. Astronomy and Astrophysics, 240, 533-536, 1990.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Fuzzy Model for Improving Accuracy in Real-Time Location Systems

Dante I. Tapia[1], Ricardo S. Alonso[1], Juan F. De Paz[1], Cristian I. Pinzón[2], Javier Bajo[1] and Juan. M. Corchado[1]

[1] *Departamento Informática y Automática Universidad de Salamanca*
[2] *Universidad Tecnológica de Panamá*

{dantetapia, ralorin, fcofds, cristian_ivan p, jbajope, corchado}@usal.es

Abstract

Wireless Sensor Networks (WSN) have become much more relevant in recent years, mainly because they can be used in a wide diversity of applications. Real-Time Location Systems (RTLS) are one of these applications and represent a currently growing market. However, accuracy in RTLS is still a problem requiring novel solutions. This paper presents an innovative mathematical fuzzy model for improving the accuracy of RTLS. The proposed approach and the preliminary results obtained are presented in this paper.

Key words: Wireless Sensor Networks, Real-Time Location Systems, Location Algorithms, Fuzzy Logic, Artificial Neural Networks.

1. Introduction

Wireless Sensor Networks allow us to obtain information about the environment and act on this, expanding users' capabilities and automating daily actions. One of the most interesting applications for WSN is the Real Time Localization System (RTLS). Although outdoor localization is covered to a large degree by systems such as GPS, indoor localization is still an area in need of development, especially with respect to locating people or objects within an enclosure [13] [14].

 ISBN 13: 978-84-613-5510-5

Therefore, it is in indoor spaces that localization presents the most difficulty. For this, it has become necessary to develop systems that allow the performance of efficient localization in terms of precision and optimization of resources (for example, sensor infrastructure and calculation capacity). The process necessary for carrying out localization must take into account the type of sensors used and the algorithm applied for the calculation of the final position based on the information recovered by these sensors.

Amongst the technologies that are currently used most in the development of RTLS are, RFID (*Radio Frequency IDentification*), Wi-Fi y ZigBee [3] [4] [5]. However, in addition to the technology used, it is necessary to establish mathematic models that allow us to determine the position from the signals recovered. For this, various algorithms exist, such as Triangulation, Fingerprinting and Multilateration [10]. However, these models present important disadvantages when developing a precise localization system, especially indoors. Therefore, it is necessary to define new models that allow the improvement of precision in this type of system.

In this article, a new model based on fuzzy logic is presented in order to improve the precision of localization systems based on wireless sensor networks, in real time. The basic functioning of these systems is as follows: Firstly, it is necessary to place a fixed node network within the space where localization will be carried out. In turn, a series of mobile nodes exist, generally called "tags", which periodically transmit a signal that contains their identifier in the network. That signal is detected by the fixed nodes within their coverage area, containing power measurements (*RSSI: Received Signal Strength Indication*) and quality (*LQI: Link Quality Indicator*) of the signal received. A central node compiles all the reference measurements from all of the fixed nodes in the network and sends them to a computer to be processed. The model proposed in this article takes RSSI as inputs and based on this executes an estimation of the position of each of the mobile nodes in the system. In a first stage, the model establishes the most probable position of each mobile node based on the RSSI levels. In the second stage, the data generated is used by the diffused model to train an MLP neuronal network [2]which will be what finally estimates the positions when the system has already been trained.

The paper is structured as follows: Section 2 presents different localization techniques. Section 3 describes the planning model. Section 4 describes a set of tests evaluating our proposal.

2. Localization Systems

Localization Systems allow the identification and localization of different elements in an environment. Localization Systems are composed of two elements: sensors and tags. The tags are placed on the elements while the sensors are normally placed in fixed points, that way generating a sensor network which allows us to locate different devices.

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments Currently, different systems of localization exist based on the technology used, and the different alternatives are:

- GPS: The operation of a real time localization system based on GPS (Global Positioning System) basically consists of a set of satellites (fixed transmitters) that constantly send information, which is collected by mobile devices (receivers). The receivers calculate their position based on the coordinates of the satellites, so the more satellite references had, the better the precision. It is necessary to have at least 3 satellite references in order to be able to calculate the position.
- GSM/GPRS: Mobile phone operators also offer localization services. Their operation is based on using the same network of antennas that the telephone service provides. In this case, localization can be carried out as much by the mobile device as by the service provider, due to the fact that antennas and devices both act as transmitters and receivers. To calculate localization, they use parameters such as the time of arrival of the signal, incidence angles, triangulation of signals or belonging cells.
- RFID: Radio Frequency IDentification (RFID) [3] is another of the alternatives used for the development of real time localization systems. Its operation is based on a network of RFID readers and tags. The readers transmit a constant RF signal, which is collected by the tags, which in turn respond to the readers by sending a number of identification. In this type of localization, each reader covers a determined zone through its RF signal (reading field) When a tag passes through the reading field of the reader, it is said that the tag is in that zone. An RFID system is mainly composed of four elements: 1) Tags, 2) Readers, 3) Antennas and Radios and 4) Processing Hardware [4] [5]. RFID tags or chips can be passive (without batteries), in which case they are called transponders [3]. Transponders are much cheaper and smaller than active chips (with batteries), but have much less of a reach range. The main RFID technology applications have taken place in industrial, transport environments, etc., but their application in other sectors, including medicine, is increasingly important [3][4] [5].
- Wi-Fi: Localization systems based on WiFi [6] employ wireless network devices to calculation position. A mesh of nodes is employed (fixed transmitters and receivers) which function as a reference for mobile nodes. The system calculates the position of the mobile nodes starting from the signals received by the fixed nodes. A large amount of techniques exist for processing these signals and determining their position, including symbolic or signpost localization, triangulation, trilateration, etc, Localization based on Wi-Fi has three main components: 1) An RFID tag that transmits and receives signals under the regulation 802.11 [7], 2) a WLAN infrastructure, formed by access and controller points, and 3) a localization engine, consisting of software capable of interpreting

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments information provided by the Wi-Fi infrastructure and tags to provide data relating to the location of users [6].

- The ZigBee standard allows operation in the ISM (Industrial, Scientific and Medical) band, which includes 2.4GHz almost all over the world. The underlying IEEE 802.15.4 standard is designed to work with low-power nodes with limited resources. ZigBee adds network and application layers over IEEE 802.15.4 and allows more than 65,000 nodes to be connected in a mesh topology WSN. Another standard for deploying WSNs is Bluetooth. This standard also operates in the 2.4GHz band and allows the creation of star topology WSNs of up to 8 devices, one acting as master and the rest as slaves, but it is possible to create larger WSNs through devices that belong simultaneously to several WSNs. However, it is not easy to integrate devices from different technologies into a single WSN [1]. The lack of a common architecture may lead to additional costs due to the necessity of deploying interconnection elements amongst different WSNs.

The most adequate technology for indoor localization is that based on ZigBee since others such as GPS can only be used outdoors as it is necessary to receive satellite signal. Other networks like GSM allow the implementation of localization but the margin of error is too high so it is not considered to be adequate for use as is the case with WI-FI.

3. Localization Algorithms

There are three main algorithms employed by real time localization systems for determining the position of mobile nodes (tags): Triangulation, Fingerprinting and Multilateration [10]. Triangulation allows us to obtain localization coordinates through the calculation of longitude of the sides of a triangle from the input angles of the received signal in each antenna, for which it is necessary to provide at least 3reference points [10]. Fingerprinting, also known as signpost or symbolic localization, is based on the study of the characteristics of each area of localization, carrying out measurements of radio frequency characteristics and estimating in which area of influence each tag is found [1][11]. Finally, Multilateration is based on the estimation of distances from the readers to the tags by measuring parameters such as RSSI (*Received Signal Strength Indication*) or TDOA (*Time Difference of Arrival*) [12], so that intersecting the estimated differences from each tag to three or more fixed nodes can determine the points where these tags are found. Multilateration allows us to obtain better results outdoors than with triangulation, but its performance lowers notably indoors. This is because indoor RSSI levels will vary in function of the presence of elements (people, objects or animals) and are also based on the calculation of distances, so that it is necessary to carry out a prior estimation of these distances starting from RSSI values which change constantly.

 ISBN 13: 978-84-613-5510-5

For this, we propose a new model which makes use of fuzzy logic and neural networks to improve the calculation of the position of mobile nodes. This model is described below.

3.1. New Fuzzy Model

The model proposed in this article is based on the level of the RSSI signal (*Received Signal Strength Indication*) detected by the nodes (sensors). The absolute value of RSSI is exponentially related to the distance that is found. Therefore, initially it seeks to convert these values in such a way that the connection is more linear although it is not necessary to carry out a precise conversion. In figure 1, there is an example of the operation of a sensor network and a tag. The sensors have been represented in red and the tag in the centre of the image is represented in blue. For each of the sensors, some circles have been represented related to a logarithm of the absolute value of the RSSI detected. The colour of the circles is related to the radio coverage. Thus, it can be seen that the circles intersecting different regions so that the darkest region is the one found closest to the point.

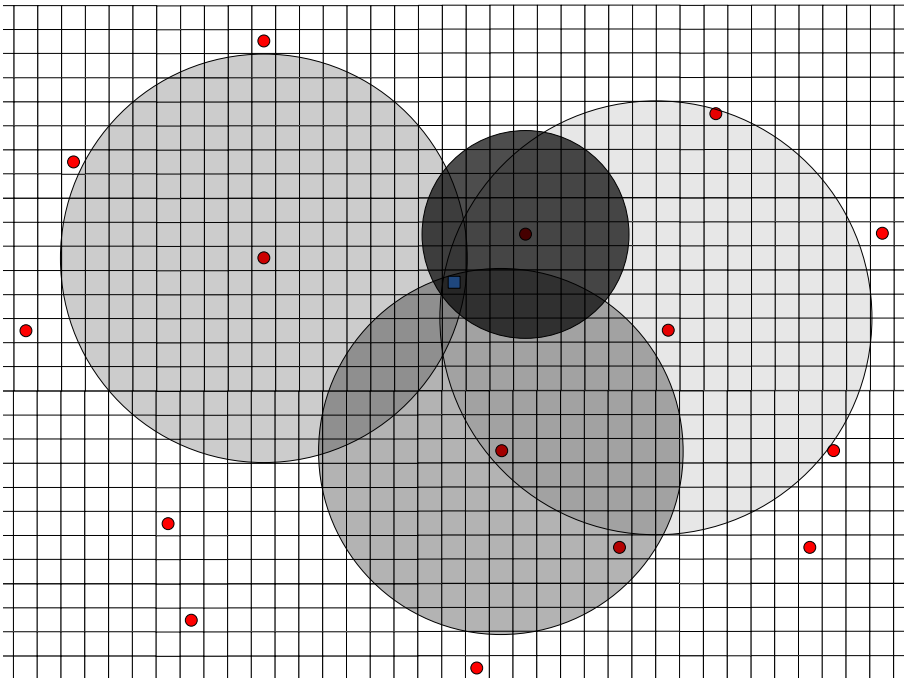


Figure 1. Graphic representation of localization based on RSSI levels.

From the information shown in the previous figure, we can proceed to making an estimation as to the most likely regions in which there is a tag. For this, for each one of the cells a relevance index is calculated based on the circles that are drawn

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments over it, finally having something similar to that shown in Figure 2. The darker the colour of the cell, the greater the possibility of finding a tag in this cell.

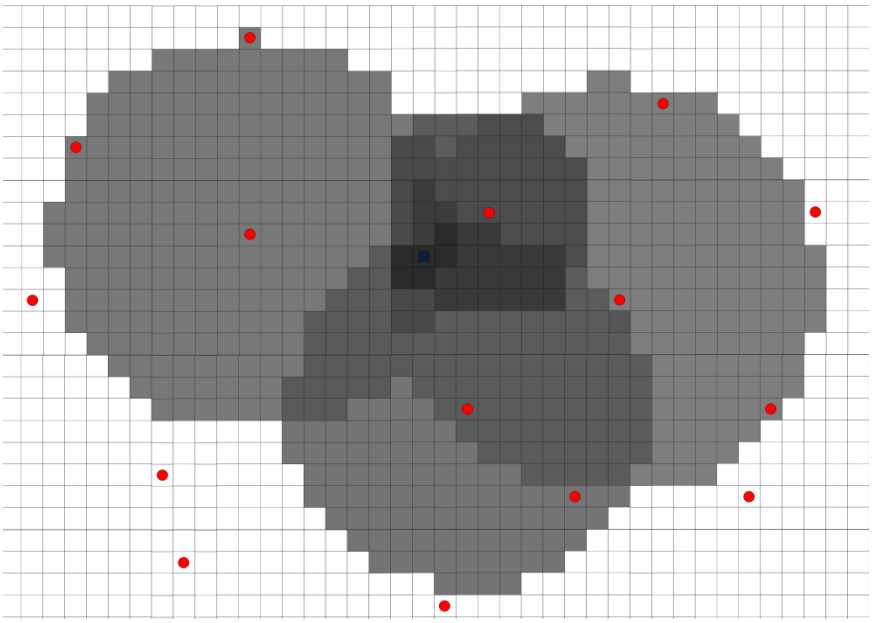


Figure 2. Graphic representation of the possibility of localization in each of the cells. The darker the cell is, the higher the probability.

Finally, once the region with the most possibility is determined, the midpoint, which will represent the estimated location of the node, is calculated.
Due to the fact that the RSSI level does not remain constant and that it will vary according to rebounds from appropriate waves emitted by the sensors, it is necessary to stabilize the positions through the use previously estimated values. For this, the relevance index for each cell is calculated depending on what has previously been observed. Thus, the calculation process of levels follows the algorithm below. In equation (1) the calculation of each relevance index of the tags to each cell is shown.

$$w_{ij}(t+1) = \frac{w_{ij}(t) + f_r(\vec{r}\,(t)) \cdot k}{k+1} \qquad (1)$$

where t represents the sequence, w_{ij} the index of pertinence of the tag to the cell ij , k the rate of update and f_r the function of calculation of the new index based on the vector r of sensitivities. f_r is defined in function of the equation (2).

$$f_r(\vec{r}\,(t)) = \sum_{k=1}^{n} 1/|r_k(t)| \qquad (2)$$

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments $\vec{r}(t)$ is the vector of signal intensities corresponding to the sensors that have detected the tag. In this vector, only the sensor readings for the distances between them are included and the cell is less than the range of the sensor for the sensitivity detected. ****** Namely, the vector only stores the sensor component if the cell that is found within the circle corresponds to the sensor as in Figure 1. Formally $\vec{r}(t)$ is described in the following manner:

$$\vec{r}(t) = \{\log(|\ RSSI_j\ |) * p_j(RSSI) / d(c, s_j) < \log(|\ RSSI_j\ |)\} \qquad (3)$$

where j represents each one of the sensors that detect the tag y $d(c, s_j)$ the Euclidian distance between the cell c and the sensor j. p_j represents a weighting based on the levels of RSSI detected. The weighting is used due to the fact that not all the RSSI levels are equally reliable due to interference. Thus the values obtained between -51 and -80 are less reliable than those obtained from 80 and much less than values near 1 so that a weighting is defined based on these values. The weighting chosen follows the equation below (4).

$$p_j(RSSI) = \begin{cases} k_1 & -1 < RSSI < l_1 \\ \ldots & \\ k_r & l_{r-1} < RSSI < l_r \\ \ldots & \\ k_n & RSSI > l_{n-1} \end{cases} \qquad (4)$$

The final position of the sensor is estimated according to the middle position of the cells with a greater probability rate of belonging to equation (5).

$$l = \frac{\sum_{i=1}^{n} p_{ij}}{n} \qquad (5)$$

3.2. Operation Stage

As the fuzzy model is capturing data, from the signals and estimating positions, it stores these in a memory to subsequently use to carry out the training of an MLP. The neural network allows us to make the fastest estimations and is more responsive to variations resulting from the reflections of the waves emitted. Input data from the neural network corresponds with the intensity values detected by a pre-fixed number of sensors. Output has two coordinates, one for the space coordinate. The number of neurons in the hidden layer is 2n+1, where n is the number of neurons in the input layer. Finally, there is one neuron in the output layer. The activation function selected for the different layers has been the

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments sigmoid. Taking into account the activation function f_j, the calculation of output values is given by the following expression

$$y_j^p = f_j(\sum_{i=1}^{N} w_{ji}(t)\, x_i^p(t) + \theta_j) \qquad (6)$$

The neurons exiting from the hidden layer of the neural network contain sigmoidal neurons. Network training is carried out through the error Backpropagation Algorithm [2].

4. Results and Conclusions

To analyze the system we proceeded to install a network of ZigBee devices in a laboratory. The sensor network was formed by 15 fixed devices distributed in 3 rooms, following the distribution shown in figure 3. The dimensions in meters are 19x19m.

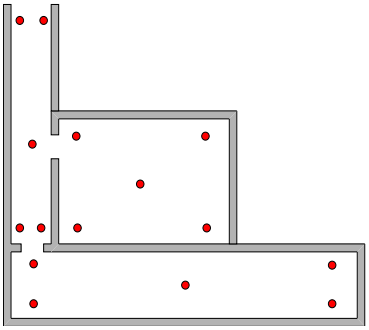


Figure 3. Distribution of the network of ZigBee devices in the laboratory

To analyze the results, an estimation of error was carried out in 19 measurements and the error during the training and estimation phases was calculated. Figure 4 shows the sensors, the real location of the tags and the estimated locations using the multilateration and fuzzy methods.

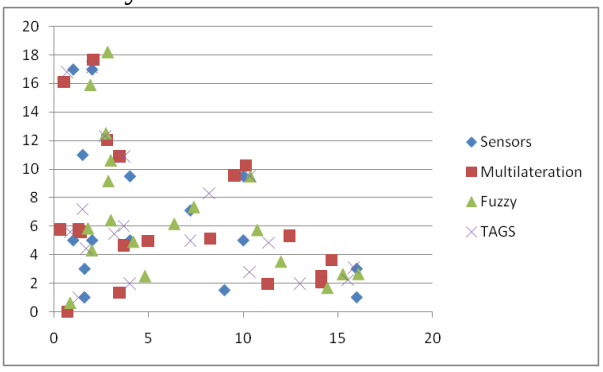


Figure 4. Location of tags using Multilateration and Fuzzy

 ISBN 13: 978-84-613-5510-5

Figure 5 shows the estimated errors obtained using the fuzzy, Multilateration and the localization models described in section 3. The X axis represents the different measurements and the Y axis, the Euclidian distance in meters from the estimated position to the real position of the tag.

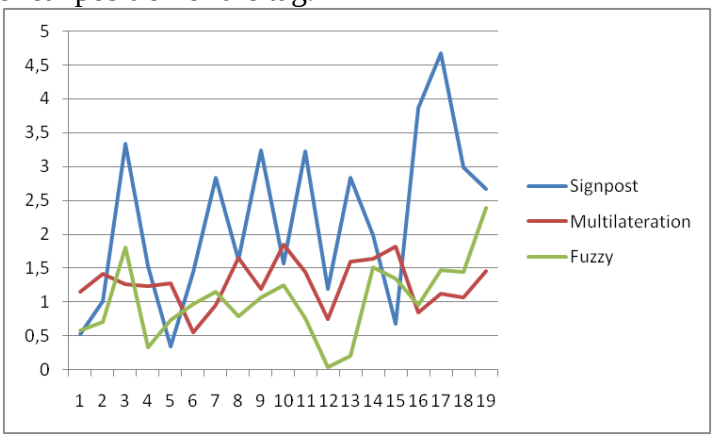


Figure 5. Prediction Errors in meters

The proposed model is capable of carrying out localization of the tags in a more precise way. Furthermore, it allows us to carry out an estimation which subsequently makes the estimation of positions though the use of a neural network possible. The neural network improves the operation of the fuzzy model since it has a greater capacity for adaptation than fuzzy models. Thus, measuring errors due to noise have less effect.

Acknowledgements

This work has been supported by the MICINN TIN 2009-13839-C03-03 project and the Professional Excellence Program 2006-2010 IFARHU-SENACYT-Panama.

References

[1] COTE, G., LEC, R., PISHKO, M.: EMERGING BIOMEDICAL SENSING TECHNOLOGIES AND THEIR APPLICATIONS. IEEE SENSORS JOURNAL 3(3), 251–266 (2003)

[2] Y. LECUN, ET AL., EFFICIENT BACKPROP, IN NEURAL NETWORKS: TRICKS OF THE TRADE, (1998) 546.

[3] RAZAVI, R., PERROT, J., & GUELFI, N. (2005). ADAPTIVE MODELING: AN APPROACH AND A METHOD FOR IMPLEMENTING ADAPTIVE AGENTS. LECTURE NOTES IN ARTIFICIAL INTELLIGENCE, 3446, 136-148.

[4] GIUNCHIGLIA, F., MYLOPOULOS, J., PERINI, A.: THE TROPOS SOFTWARE DEVELOPMENT METHODOLOGY: PROCESSES, MODELS AND DIAGRAMS. IN:

 ISBN 13: 978-84-613-5510-5

Mathematical model for a temporal-bounded classifier in security environments AAMAS'02 WORKSHOP ON AGENT ORIENTED SOFTWARE ENGINEERING (AOSE-2002) (2002) 63–74.

[5] ERICKSON, P., WILSON, R., & SHANNON, I. (1995). YEARS OF HEALTHY LIFE. STATISTICAL NOTES (7).

[6] MARTIN, D. ET. AL.: OWL-S: SEMANTIC MARKUP FOR WEB SERVICES, W3C MEMBER SUBMISSION, HTTP://WWW.W3.ORG/SUBMISSION/OWL-S/ (2004)

[7] EMILIANI, P. L. Y STEPHANIDIS, C. (2005). UNIVERSAL ACCESS TO AMBIENT INTELLIGENCE ENVIRONMENTS: OPPORTUNITIES AND CHALLENGES FOR PEOPLE WITH DISABILITIES. IBM SYSTEMS JOURNAL. SEPT, 2005.

[8] FERNÁNDEZ, A.; OSSOWSKI, S. EXPLOITING ORGANISATIONAL INFORMATION FOR SERVICE COORDINATION IN MULTIAGENT SYSTEMS. INT. CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS, ACM-IFAAMAS, PP 257-264. 2008.

[9] KAEMARUNGSI, K. & KRISHNAMURTHY, P. (2004). MODELING OF INDOOR POSITIONING SYSTEMS BASED ON LOCATION FINGERPRINTING. 23RD ANNUAL JOINT CONFERENCE OF THE IEEE COMPUTER AND COMMUNICATIONS SOCIETIES (INFOCOM 2004), VOL. 2, PP. 1012-1022.

[10] DUCATEL, K., BOGDANOWICZ, M., SCAPOLO, F., LEIJTEN, J. & BURGELMAN, J. C. (2001). THAT'S WHAT FRIENDS ARE FOR. AMBIENT INTELLIGENCE (AMI) AND THE IS IN 2010. INNOVATIONS FOR AN E-SOCIETY. CONGRESS PRE-PRINTS, "INNOVATIONS FOR AN E-SOCIETY. CHALLENGES FOR TECHNOLOGY ASSESSMENT". BERLIN, GERMANY.

[11] GLASSNER, A. 1995. PRINCIPLES OF DIGITAL IMAGE SYNTHESIS. MORGAN KAUFMANN.

[12] CAPERA, D., GEORGÉ, J.-P., GLEIZES, M.-P., & GLIZE, P. (2003). EMERGENCE OF ORGANISATIONS, EMERGENCE OF FUNCTIONS. SYMPOSIUM ON ADAPTIVE AGENTS AND MULTI-AGENT SYSTEMS, (PÁGS. 103-108).

[13] M.. CORCHADO,J. BAJO, AND A. ABRAHAM GERAMI: IMPROVING THE DELIVERY OF HEALTH CARE. IEEE INTELLIGENT SYSTEMS. SPECIAL ISSUE ON AMBIENT INTELLIGENCE 3 (2) (2008) 19-25

[14] D.I. TAPIA, J.F. DE PAZ, S. RODRÍGUEZ, J. BAJO AND J.M. CORCHADO MULTI-AGENT SYSTEM FOR SECURITY CONTROL ON INDUSTRIAL ENVIRONMENTS. INTERNATIONAL TRANSACTIONS ON SYSTEM SCIENCE AND APPLICATIONS JOURNAL. 4 (3) (2008) 222-226

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE2010
Almeria, Spain, 27-30 June 2010

Solving Multi Objective Stochastic Programming Problems using Differential Evolution

Radha Thangaraj[1], Millie Pant[2], Pascal Bouvry[1] and Ajith Abraham[3]

[1] *Faculty of Science, Technology and Communications, University of Luxembourg*
[2] *Department of Paper Technology, Indian Institute of Technology Roorkee, India*
[3] *Machine Intelligent Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, USA*

emails: radha.raj@uni.lu, millifpt@iitr.ernet.in, pascal.bouvry@uni.lu, ajith.abraham@ieee.org

Abstract

Stochastic (or probabilistic) programming is an optimization technique in which the constraints and/or the objective function of an optimization problem contains random variables. The mathematical models of these problems may follow any particular probability distribution for model coefficients. The objective here is to determine the proper values for model parameters influenced by random events. In this study, DE and its two recent variants LDE1 and LDE2 are presented for solving multi objective linear stochastic programming (MOSLP) problems, having several conflicting objectives. The numerical results obtained by DE and its variants are compared with the available results from where it is observed that the DE and its variants significantly improve the quality of solution of the given considered problem in comparison with the quoted results in the literature.

Key words: Differential Evoltuion, stochastic programming, multiobjective optimization.

1 Introduction

Stochastic programming (SP) is a mathematical programming where stochastic element is present in the data. In contrast to deterministic mathematical

 ISBN 13: 978-84-613-5510-5

programming where the data (coefficients) are known numbers in stochastic programming these numbers follow a probability distribution. Thus we can say that SP is a framework for modeling optimization problems that involve uncertainty. The goal here is to find some policy that is feasible for all (or almost all) the possible data instances and maximizes the expectation of some function of the decisions and the random variables. More generally, such models are formulated, solved analytically or numerically, and analyzed in order to provide useful information to a decision-maker.

In the recent past, SP has been applied to the problems having multiple, conflicting and non-commensurable objectives where generally there does not exist a single solution which can optimize all the objectives. Several methods for solving Multi-Objective Stochastic Linear Programming (MOSLP) problems and their applications to various fields are available in literature [1] – [7]. Most of the probabilistic models assume normal distribution for model coefficients. Sahoo and Biswal [8] presented some deterministic equivalents for the probabilistic problem involving normal and log-normal random variables for joint constraints. Charles et al. [9] addressed different forms of distributions like Power Function distribution, Pareto distribution, Beta distribution of first kind, Weibull distribution and Burr type XII distribution. In the present study we have followed the models proposed by Charles et al [9] and have solved them using Differential Evolution (DE).

The rest of the paper is organized as follows: Section 2 briefly describes the classical DE, LDE1 and LDE2 algorithms. The problem definition is given in section 3. In section 4; the experimental settings and numerical results are discussed. Finally the paper concludes with section 5.

2 Differential Evolution Algorithms

2.1 Classical Differential Evolution (DE)

Differential Evolution (DE) [10] is a population based metaheuristics that has been consistently ranked as one of the best search algorithm for solving benchmark as well as real life problems in several case studies. The algorithm mainly has three advantages; finding the true global minimum regardless of the initial parameter values, fast convergence, and uses a few control parameters [11]. DE has been successfully applied to solve a wide range of real life application problems such as clustering [12], unsupervised image classification [13], digital filter design [14], optimization of non-linear functions [15], global optimization of non-linear chemical engineering processes [16] and multi-objective optimization [17] etc. Also it has reportedly outperformed other optimization techniques [18] – [20].

 ISBN 13: 978-84-613-5510-5

A general DE variant may be denoted as *DE/X/Y/Z*, where *X* denotes the vector to be mutated, *Y* specifies the number of difference vectors used and *Z* specifies the crossover scheme which may be binomial (bin) or exponential (exp). Throughout the study we shall consider the mutation strategy *DE/rand/1/bin* [10] which is perhaps the most frequently used version of DE.

For a D-dimensional search space, each target vector $x_{i,g}$, a mutant vector is generated by

$$v_{i,g+1} = x_{r_1,g} + F*(x_{r_2,g} - x_{r_3,g}) \quad (1)$$

where $r_1, r_2, r_3 \in \{1,2,....,NP\}$ are randomly chosen integers, must be different from each other and also different from the running index *i*. *F* (>0) is a scaling factor which controls the amplification of the differential evolution $(x_{r_2,g} - x_{r_3,g})$. In order to increase the diversity of the perturbed parameter vectors, crossover is introduced. The parent vector is mixed with the mutated vector to produce a trial vector $u_{ji,g+1}$,

$$u_{j,i,g+1} = \begin{cases} v_{j,i,g+1} & if \quad rand_j \le C_r \vee j = k \\ x_{j,i,g+1} & otherwise \end{cases} \quad (2)$$

where j = 1, 2,……, D; $rand_j \in [0,1]$; CR is the crossover constant takes values in the range [0, 1] and $j_{rand} \in (1,2,.....,D)$ is the randomly chosen index.

The final phase of DE algorithm is selection. Here the population for the next generation is selected from the individual in current population and its corresponding trial vector according to the following rule:

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & if \quad f(u_{i,G+1}) \le f(x_{i,G}) \\ x_{i,G} & otherwise \end{cases} \quad (3)$$

Thus, each individual of the advance (trial) population is compared with its counterpart in the current population. The one with the lower objective function value will survive from the tournament selection to the population of the next generation. As a result, all the individuals of the next generation are as good as or better than their counterparts in the current generation.

2.2 Laplace Differential Evolution (LDE)

The LDE algorithms are proposed by Thangaraj et al. [21]. These algorithms differ from the classical DE in the mutation phase in a twofold manner. These schemes make use the absolute weighted difference between the two vector points in place of the usual vector difference as in classical DE and secondly, in LDE schemes amplification factor, F (of the usual DE), is replaced by L, a random variable following Laplace distribution.

The mutation schemes of LDE1 and LDE2 algorithms are defined as follows:

 ISBN 13: 978-84-613-5510-5

2.2.1 LDE1 Scheme

$$v_{i,g+1} = x_{best,g} + \mathrm{L}^* | x_{r_1,g} - x_{r_2,g} | \quad (4)$$

In LDE1 scheme, the base vector is the one having the best fitness function value; whereas the other two individuals are randomly selected.

2.2.2 LDE2 scheme

If (U(0,1) < 0.5) then

$$v_{i,g+1} = x_{best,g} + \mathrm{L}^* | x_{r_1,g} - x_{r_2,g} |$$

Else

$$v_{i,g+1} = x_{r_1,g} + F^*(x_{r_2,g} - x_{r_3,g})$$

In LDE2 scheme, mutant vector using equation (4) and the basic mutant vector equation are applied probabilistically using a predefined value. A random variable following normal distribution U(0,1) is generated. If it is less than 0.5 then LDE1 scheme is applied otherwise Eqn. (1) is applied.
Both the modified versions, LDE1 and LDE2 have reportedly given good performances for solving benchmark as well as real life problems [21].

3 Problem Definition

Mathematical model of a constrained MOLSP may be given as [9]:

$$\text{Maximize } z_k = \sum_{j=1}^{n} c_j^k x_j, \; k = 1,2,...,K$$

$$\text{Subject to } P\left(\sum_{j=1}^{n} a_{1j} x_j \le b_1, \sum_{j=1}^{n} a_{2j} x_j \le b_2, ..., \sum_{j=1}^{n} a_{mj} x_j \le b_m \right) \ge p$$

$$x_j \ge 0, \; j = 1,2,...,n$$

Where $0 < p < 1$ is usually close to 1. It has been assumed that the parameters a_{ij} and c_j are deterministic constants and b_i are random variables. For more details the interested reader may please refer to [9]. In the present study, we have considered the two test problems which are used in [9]. These problems are multi-objective stochastic linear programming problems (MOSLP) involving random variables following different distributions.

Test problem 1: MOSLP1:

$$Maximize \; z_1 = 5x_1 + 6x_2 + 3x_3$$

$$Maximize \; z_2 = 6x_1 + 3x_2 + 5x_3$$

$$Maximize \; z_3 = 2x_1 + 5x_2 + 8x_3$$

Subject to

$$P(3x_1 + 2x_2 + 2x_3 \le b_1) \ge 0.90$$

 ISBN 13: 978-84-613-5510-5

$$P(2x_1 + 8x_2 + 5x_3 \le b_2) \ge 0.98$$
$$P(5x_1 + 3x_2 + 2x_3 \le b_3) \ge 0.95$$
$$P(0.5x_1 + 0.5x_2 + 0.25x_3 \le b_4) \ge 0.90$$
$$P(8x_1 + 3x_2 + 4x_3 \le b_5) \ge 0.99$$
$$x_1, x_2, x_3 \ge 0$$

Here, b_1 follow Power Function distribution, b_2 follow Pareto distribution, b_3 follow Beta distribution, b_4 follow Weibull distribution; b_5 follow Burr type XII distribution. The problem is converted to deterministic model as follows:

$$Maximize\ z = \lambda_1(5x_1 + 6x_2 + 3x_3) + \lambda_2(6x_1 + 3x_2 + 5x_3) + \lambda_3(2x_1 + 5x_2 + 8x_3)$$

Subject to

$$3x_1 + 2x_2 + 2x_3 \le 6.3096,\ 2x_1 + 8x_2 + 5x_3 \le 8.0812$$
$$5x_1 + 3x_2 + 2x_3 \le 4.7115,\ 0.5x_1 + 0.5x_2 + 0.25x_3 \le 0.9379$$
$$8x_1 + 3x_2 + 4x_3 \le 10.0321,\ \lambda_1 + \lambda_2 + \lambda_3 = 1$$
$$x_1, x_2, x_3, \lambda_1, \lambda_2, \lambda_3 \ge 0$$

Test problem 2: MOSLP2:

$$Maximize\ z_1 = 3x_1 + 8x_2 + 5x_3$$
$$Maximize\ z_2 = 7x_1 + 4x_2 + 3x_3$$
$$Maximize\ z_3 = 6x_1 + 7x_2 + 10.5x_3$$

Subject to

$$P(5x_1 + 4x_2 + 2x_3 \le b_1) \ge 0.95$$
$$P(7x_1 + 3x_2 + x_3 \le b_2) \ge 0.95$$
$$P(2x_1 + 7x_2 + 3x_3 \le b_3) \ge 0.95$$
$$P(0.5x_1 + 0.5x_2 + 0.25x_3 \le b_4) \ge 0.95$$
$$P(5x_1 + 2x_2 + 1.5x_3 \le b_5) \ge 0.95$$
$$x_1, x_2, x_3 \ge 0$$

Here b_1 follow Power Function distribution; b_2 follow Pareto distribution; b_3 follow Beta distribution of first kind; b_4 follow Weibull distribution and b_5 follow Burr type XII distribution. The deterministic model of the problem is given as:

$$Maximize\ z = \lambda_1(3x_1 + 8x_2 + 5x_3) + \lambda_2(7x_1 + 4x_2 + 3x_3) + \lambda_3(6x_1 + 7x_2 + 10.5x_3)$$

 ISBN 13: 978-84-613-5510-5

Subject to

$$\left[\frac{y_1^{\,2}}{9}\right]\left[\frac{y_2^{\,2}-100}{y_2^{\,2}}\right]\left[\frac{y_3-5}{10}\right]\left[\frac{e^{2y_4}-1}{e^{2y_4}}\right]\left[\frac{3y_5^{\,2}}{1+3y_5^{\,2}}\right]\geq 0.95$$

$$5x_1+4x_2+2x_3=y_1,\ 7x_1+3x_2+x_3=y_2$$

$$2x_1+7x_2+3x_3=y_3,\ 2x_1+3x_2+2.5x_3=y_4$$

$$5x_1+2x_2+1.5x_3=y_5,\ \lambda_1+\lambda_2+\lambda_3=1$$

$$x_1,x_2,x_3,y_1,y_2,y_3,y_4,y_5,\lambda_1,\lambda_2,\lambda_3\geq 0$$

4 Experimental Settings and Numerical Results

4.1 Parameter Settings

DE has three main control parameters; population size, crossover rate Cr and Scaling factor F which are fixed as 50, 0. 5 and 0.5 respectively. For LDE schemes the scaling factor is a random variable, L, following Laplace distribution. For each algorithm, the stopping criterion is to terminate the search process when the maximum number of generations is reached (assumed 1000 generations). Constraints are handled according to the approach based on repair methods suggested in [22]. A total of 50 runs for each experimental setting were conducted and the best solution throughout the run was recorded as global optimum. Results obtained by basic DE and LDE versions are also compared with previously quoted results [9].

4.2 Numerical Results

We have considered four test cases in each of the test problems. Since, $\lambda_1+\lambda_2+\lambda_3=1$, one of λ_i, i = 1, 2, 3 could be eliminated to reduce the number of dependent variables from the expression of objective function. So, we assigned equal weights to two terms at a time in the objective expression. The resultant test cases are as follows:

(i) $\lambda_1=W,\lambda_2=\lambda_3=\dfrac{1-W}{2},0\leq W\leq 1$

(ii) $\lambda_2=W,\lambda_1=\lambda_3=\dfrac{1-W}{2},0\leq W\leq 1$

(iii) $\lambda_3=W,\lambda_1=\lambda_2=\dfrac{1-W}{2},0\leq W\leq 1$

(iv) λ_1, λ_2, and λ_3 are dependent variables.

The numerical results of the given two test problems MOSLP1 and MOSLP2 are recorded in Tables 1 and 2 respectively. The best solution obtained by DE and

 ISBN 13: 978-84-613-5510-5

LDE algorithms for MOSLP1 in terms of optimal decision variable values and objective function value are given in Table 1. For the test case (i), the performance of LDE1 is better than all the other algorithms. For the remaining 3 test cases, LDE2 performs better than other compared algorithms. If we compare the LDE algorithms with classical DE algorithm then from the numerical results we can see that LDE algorithms are superior with classical DE algorithm. There is an improvement of 52% in objective function value when the problem is solved by LDE2 in comparison with the quoted result [9], where the problem is solved by Genetic Algorithm. The results of test problem MOSLP2 are given in Table 2. From this table also we can see that LDE2 algorithm is superior with others in all the test cases. The improvement of LDE2 algorithm in comparison with the results in the literature is 141%. Figure 1 shows the performance of DE and LDE algorithms in terms of objective function value.

Table 1 Results of MOSLP1

	DE	LDE1	LDE2	GA [9]
$\lambda_1 = W, \lambda_1 = \lambda_2 = (1-W)/2, 0 \le W \le 1$				
Z	10.9905	**10.997**	10.996	--NA--
x1	0.349128	0.351905	0.35171	
x2	0	0	0	
x3	1.47618	1.47538	1.47539	
$\lambda_2 = W, \lambda_1 = \lambda_3 = (1-W)/2, 0 \le W \le 1$				
z	9.48974	**9.48975**	**9.48975**	--NA--
x_1	0.35214	0.35215	0.352142	
x_2	0	0	0	
x_3	1.47538	1.47537	1.47538	
$\lambda_3 = W, \lambda_1 = \lambda_2 = (1-W)/2, 0 \le W \le 1$				
z	12.9277	12.9288	**12.9292**	--NA--
x_1	0	0	0	
x_2	0	0	0	
x_3	1.61611	1.61612	1.61617	
Problem described as in [9]				
z	9.48978	11.3988	**12.9299**	8.5089
x_1	0.352147	0.334378	0	0.3727
x_2	2.12479e-007	0.00514505	0	0.2319
x_3	1.47538	1.47426	1.61624	1.0761

 ISBN 13: 978-84-613-5510-5

Table 2 Results of MOSLP2

	DE	LDE1	LDE2	GA [9]
$\lambda_1 = W, \lambda_1 = \lambda_2 = (1-W)/2, 0 \le W \le 1$				
z	5.5452	6.3844	**6.86328**	--NA--
x_1	0.170342	0.275175	0.297729	
x_2	0.0367932	0.0654974	0.00485206	
x_3	0.759151	0.627495	0.726168	
y_1	2.5158	2.89285	2.96039	
y_2	2.06291	2.7502	2.82483	
y_3	2.862	2.89131	2.80793	
y_4	2.36484	2.31558	2.42544	
y_5	2.06754	2.44811	2.5876	
$\lambda_2 = W, \lambda_1 = \lambda_3 = (1-W)/2, 0 \le W \le 1$				
z	5.3215	7.01255	**7.72732**	--NA--
x_1	0.170342	0.12258	0	
x_2	0.0367932	0.0575791	0.00166162	
x_3	0.759151	0.777962	0.995503	
y_1	2.5158	2.39914	1.99765	
y_2	2.06291	1.80875	1.00048	
y_3	2.862	2.98209	2.99814	
y_4	2.36484	2.36281	2.49374	
y_5	2.06754	1.895	1.49658	
$\lambda_3 = W, \lambda_1 = \lambda_2 = (1-W)/2, 0 \le W \le 1$				
z	6.60213	9.3271	**10.4638**	--NA--
x_1	0.170342	0.126015	0	
x_2	0.0367932	0	0.00166304	
x_3	0.759151	0.816303	0.995504	
y_1	2.5158	2.26268	1.99765	
y_2	2.06291	1.69841	1.00049	
y_3	2.862	2.70093	2.99815	
y_4	2.36484	2.29278	2.49374	
y_5	2.06754	1.85453	1.49659	
Problem described as in [9]				
z	6.87235	7.13425	**7.73912**	3.2081
x_1	2.65138e-006	0.000944931	0.000308158	0.1939
x_2	0.000127494	0.061029	0.127573	0.2810
x_3	0.664552	0.738963	0.688939	0.1968
y_1	1.32963	1.72678	1.88971	2.4872
y_2	0.664947	0.928675	1.07383	2.3971
y_3	1.99454	2.64598	2.96046	2.9454
y_4	1.66177	2.03239	2.10569	1.7229
y_5	0.9971	1.0	1.0	1.8267

 ISBN 13: 978-84-613-5510-5

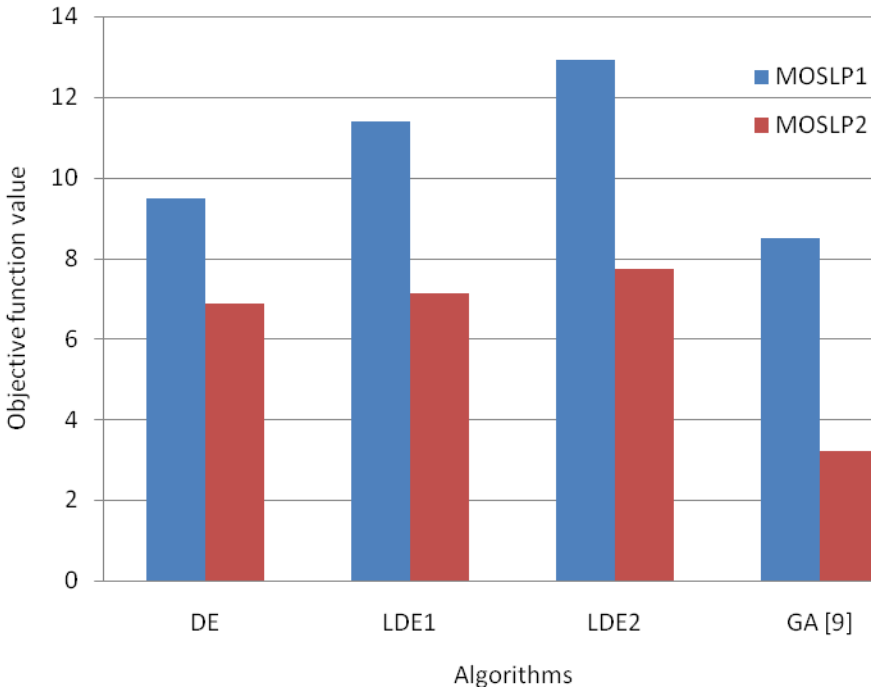


Figure 1 Performance of DE and LDE algorithms in terms of objective function value

5 Conclusion

The Stochastic Programming is an optimization technique in which the constraints and/or the objective function of an optimization problem contains certain random variables following different probability distributions. In the present study DE and two of its recent variants LDE1 and LDE2 are used to solve two constrained multiobjective stochastic linear programming problems. Four test cases were considered with respect to the weighing factors and the results were produced in terms of objective function value and decision variable values. From the experimental results it was observed that the DE algorithm and its variants significantly improve the quality of solution of the considered problems in comparison with the quoted results in the literature. As expected the modified versions LDE1 and LDE2 performed better than the basic version of DE because of the presence of the Laplace mutation operator. In conclusion we can say that DE's present an attractive option for solving stochastic programming problems.

Acknowledgement

This work was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme.

 ISBN 13: 978-84-613-5510-5

6 References

[1] ABDELAZIZ, F.B. AOUNI, B. and RIMEH, F. EI "Multi-objective programming for portfolio selection", European Journal Operational Research 177 (3), 2007, 1811–1823.

[2] BABA, N., and MORIMOTO, A., "Stochastic approximations methods for solving the stochastic multi-objective programming problem", International Journal of Systems Sciences, Vol. 24, 1993, 789–796.

[3] CABALLERO, R., CERDÁ, E., MUNOZ, M.M., REY, L., and STANCU-MINASIAN, I.M., "Efficient solution concepts and their relations in stochastic multi-objective programming", Journal of Optimization Theory and Applications, Vol. 110, 2001, 53–74.

[4] CHARLES, V., and DUTTA, D., "Bi-weighted multi-objective stochastic fractional programming problem with mixed constraints", R. Natarajan and G. Arulmozhi (Eds), Second National Conference on Mathematical and Computational Models. Allied Publishers, Chennai, 2003.

[5] GOICOECHEA, A., HANSEN, D.R., and DUCKSTEIN, L., "Multi-objective Decision Analysis with Engineering and Business Application", New York: John Wiley, 1982.

[6] LECLERCQ, J.P., "Stochastic Programming: An Interactive Multiple Approach", European Journal of Operations Research, Vol. 10, 1982, 33-41.

[7] SUWARNA, H., BISWAL, M.P. and SINHA, S.B., "Fuzzy programming approach to multiobjective stochastic linear programming problems, Fuzzy Sets and Systems, Vol. 88, 1997, 173-181.

[8] SAHOO, N.P. and BISWAL, M.P., "Computation of Probabilistic linear programming problems involving normal and log-normal random variables with a joint constraint", International Journal of Computer Mathematics, Vol. 82 (11), 2005, 1323-1338.

[9] V. CHARLES, S. I. ANSARI and M. M. KHALID, "Multi-Objective Stochastic Linear Programming with General form of Distributions", http://www.optimization-online.org/DB_FILE/2009/11/2448.pdf.

[10] R. STORN and K. PRICE, "Differential Evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces", Technical Report, International Computer Science Institute, Berkley, 1995.

[11] R. STORN and K. PRICE, "Differential Evolution – a simple and efficient Heuristic for global optimization over continuous spaces", Journal Global Optimization. 11, 1997, 341 – 359.

[12] PATERLINI. S, KRINK. T.: High performance clustering with differential evolution." In: Proceedings of the IEEE Congress on Evolutionary Computation, vol. 2, 2004, 2004–2011.

 ISBN 13: 978-84-613-5510-5

[13] OMRAN. M, ENGELBRECHT. A and A. SALMAN.: Differential evolution methods for unsupervised image classification," In: Proceedings of the IEEE Congress on Evolutionary Computation, vol. 2, 2005, 966–973.

[14] STORN. R.: Differential evolution design for an IIR-filter with requirements for magnitude and group delay. Technical Report TR-95-026, International Computer Science Institute, Berkeley, CA, 1995.

[15] BABU. B and ANGIRA. R., "Optimization of non-linear functions using evolutionary computation", In Proc. of the 12th ISME International Conference on Mechanical Engineering, India, 2001, 153–157.

[16] ANGIRA. R and BABU. B., "Evolutionary computation for global optimization of non-linear chemical engineering processes", In Proc. of International Symposium on Process Systems Engineering and Control, Mumbai, 2003, 87–91.

[17] ABBASS. H., "A memetic pareto evolutionary approach to artificial neural networks", Lecture Notes in Artificial Intelligence, Vol. 2256. Springer, 2002, 1–12.

[18] VESTERSTROEM. J and THOMSEN. R., "A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems", In Proc. of Congress on Evolutionary Computation, Vol. 2, 2004, 1980–1987.

[19] ANDRE. J, SIARRY. P, and DOGNON. T., "An improvement of the standard genetic algorithm fighting premature convergence in continuous optimization", Advance in Engineering Software, Vol. 32, 2001, 49–60.

[20] HRSTKA. O and KUˇCEROVÁ. A., "Improvement of real coded genetic algorithm based on differential operators preventing premature convergence", Advance in Engineering Software, Vol. 35, 2004, 237–246.

[21] R. THANGARAJ, M. PANT and A. ABRAHAM, "New Mutation Schemes for Differential Evolution Algorithm and their application to the Optimization of Directional Overcurrent Relay Settings", Applied Mathematics and Computation, Elsevier Science, Vol. 216 (2), 2010, 532 – 544.

[22] M. PANT, R. THANGARAJ and V. P. SINGH, "Optimization of Mechanical Design Problems using Improved Differential Evolution Algorithm", Int. Journal of Recent Trends in Engineering, Vol. 1(5), 2009, 21 – 25.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Short Abstracts

New Approaches to Characterizing the Information Theory of MIMO Wireless Channel

Yang Chen[1] and Matthew McKay[2]

[1] *Department of Mathematics, Imperial College London,UK*

[2] *Electronic and Computer Engineering Department Hong Kong University of Science and Technology Clear Water Bay, Hong Kong*

Abstract

In this talk we compute the Shannon capacity of multi-antenna Gaussian channel through its moment generating function. It transpires that such a quantity can be described as a particular Hankel determinant of a (n x n) moment matrix generated by a perturbed Laguerre weight, with a parameter t. We show that the logarithmic derivative of the Hankel determinant with respect to t, satisfies the Jimbo-Miwa-Okamoto sigma form of a Painlevé V.

We show how to reconcile the Coulomb Fluid approach (valid for large n) with the moment generating function with this PV and obtain 1/n and higher order corrections.

On a degenerative version of the Favard's theorem

R.S. Costas-Santos and J.F. Sánchez-Lara

Abstract

We state a degenerate version of Favard's theorem that allow us to extend the orthogonality properties valid up to some integer degree N to Sobolev type orthogonality properties. We also present the process to obtain the factorization and the non- standard Sobolev-type orthogonality property for those families of classi- cal orthogonal polynomials which satisfy a finite orthogonality property, i.e. it consists in sum of finite number of masspoints.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Public-Key Cryptography based on Modular Lattices

Marcus Greferath and Jens Zumbrägel (University College Dublin)

Abstract

This contribution seeks to generalize pairing-based public-key cryptography to more general algebraic structures. It focuses on modular lattices, presents a pairing on such lattices, and studies a projective-geometry based cryptosystem.

A class of asymptotic preserving schemes for kinetic equations and related problems with stiff sources

Shi Jin

University of Wisconsin-Madison, USA

Abstract

We propose a general time discrete framework to design asymptotic preserving schemes for initial value problem of the Boltzmann kinetic and related equations. Numerically solving these equations are challenging due to the nonlinear stiff collision (source) terms induced by small mean free or relaxation time. We propose to penalize the nonlinear collision term by a BGK-type relaxation term, which can be solved explicitly even if discretized implicitly in time. Moreover, the BGK-type relaxation operator helps to drive the density distribution toward the local Maxwellian, thus naturally imposes an asymptotic-preserving scheme in the Euler limit. The scheme so designed does not need any nonlinear iterative solver or the use of Wild Sum. It is uniformly stable in terms of the (possibly small) Knudsen number, and can capture the macroscopic fluid dynamic (Euler) limit even if the small scale determined by the Knudsen number is not numerically resolved. It is also consistent to the compressible Navier-Stokes equations if the viscosity and heat conductivity are numerically resolved. The method is applicable to many other related problems, such as hyperbolic systems with stiff relaxation, and high order parabolic equations.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

New results on Laguerre-type orthogonal polynomials

Edmundo J. Huertas[1], Francisco Marcellán[1] and Herbert Dueñas[2]

[1] *Departamento de Matemáticas, Escuela Politécnica Superior, Universidad Carlos III, Leganés-Madrid, Spain.*

[2] *Departamento de Matemáticas, Universidad Nacional de Colombia, Ciudad Universitaria, Bogotá, Colombia.*

emails: **ehuertas@math.uc3m.es, pacomarc@ing.uc3m.es, haduenasr@unal.edu.co**

Abstract

This contribution is devoted to the study of the Laguerre-type monic orthogonal polynomial sequences (MOPS, in short) defined by an Uvarov?s canonical spectral transformation of the Laguerre weight supported on the positive semi-axis of the real line. In such a way, we state a comparative analysis with the behavior of the standard Laguerre-type polynomials, taking into account that, in our case, we are dealing with a mass point located outside the support of the measure. The outline of the talk is the following. In the first part we introduce the representation of the perturbed MOPS in terms of the classical ones, we deduce the three term recurrence relation that they satisfy, as well as the behavior of their coefficients. Next, we obtain the lowering and raising operators associated with these polynomials, and thus the corresponding holonomic equation follows in a natural way. The second part of the talk is devoted to the study of the behavior of the zeros of these polynomials in terms of the mass M. We also provide an electrostatic interpretation of them. Finally, we analyze the outer relative asymptotics as well as the Mehler-Heine formula for these polynomials.

Key words: Orthogonal polynomials, Zeros of polynomials, Christoffel transforms, Uvarov transforms, Connection Formula, Structure relation.
MSC 2000: 33C47.

References

[1] R. Alvarez-Nodarse, F. Marcellán, and J. Petronilho, *WKB Approximation and Krall-type Orthogonal Polynomials*, Acta Appl. Math. **54** (1998) 27-58.

[2] J. Dini and P. Maroni, *La multiplication d'une forme linéaire par une forme rationnelle. Application aux polynomes de Laguerre-Hahn*, Ann. Polon. Math. **52** (1990) 175-185.

[3] H. Dueñas, and F. Marcellán, *Laguerre-Type orthogonal polynomials. Electrostatic interpretation*, Int. J. Pure and Appl. Math. **38** (2007) 345-358.

[4] F. Marcellan and P. Maroni, *Sur l'adjonction d'une masse de Dirac á une forme reguliére art semi-classique*, Annal. Mat. Pura ed Appl. **CLXII** (1992) 1-22.

[5] A. Zhedanov, *Rational spectral transformations and orthogonal polynomials*, J. Comput. Appl. Math. **85** (1997) 67-83.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

The Seysen reduction algorithm and its application to MIMO systems

Gerard Maze[1]

[1] *Mathematical Institute, University of Zurich, Switzerland* email: **gmaze@math.uzh.ch**

Abstract

Given a lattice L, a basis B of L together with its dual B^*, the orthogonality measure $S(B) = \sum_i ||b_i||^2||b_i^*||^2$ of B was introduced by M. Seysen [5] in 1993. This measure is at the heart of the Seysen lattice reduction algorithm and is linked with different geometrical properties of the basis [2, 3, 6, 7]. In this talk, we will derive different expressions for this measure as well as new inequalities related to the Frobenius norm and the condition number of a matrix. This approach allows us to improve known upper bounds for the Seysen measure and the orthogonality defect. We will then review the Seysen reduction algorithm [5, 1] and describe its application to the field of MIMO systems [4]. We will concentrate on the conceptual differences between the LLL algorithm and the Seysen reduction algorithm. The LLL algorithm focuses on local optimization (i.e., on 2 dimensional sublattices) but Seysen's algorithm performs global angle optimization to produce a reduced lattice. As a consequence, Seysen's scheme can achieve a better BER performance. We will also present the work of [4] showing that it requires less computational time than the LLL algorithm in the linear detection case.

References

[1] LaMacchia, B.A., *Basis reduction algorithms and subset sum problems.*, SM Thesis, Dept. of Elect. Eng. and Comp. Sci., Massachusetts Institute of Technology, Cambridge, MA, May 1991.

[2] Ling, C., *Towards characterizing the performance of approximate lattice decoding in MIMO communica tions.*, Proceedings of International Symposium on Turbo Codes/International ITG Conference Source Channel Coding06, Munich, Germany, April 2006.

[3] Ling, C., *On the proximity factors of lattice reduction aided decoding.*,Submitted to IEEE Trans. on Information Theory, May 2007. Available at http://www.commsp.ee.ic.ac.uk/cling/Lattice.pdf

[4] Niu, J. and Lu, I.T., *A Comparison of Two Lattice-Reduction-Based Receivers for MIMO Systems.*, Sarnoff Symposium, 2008 IEEE, April 2008.

 ISBN 13: 978-84-613-5510-5

[5] Seysen, M., *Simultaneous reduction of a lattice basis and its reciprocal basis.*, Combinatorica **13**(3) (1993) 363-376.

[6] Seysen, M., *A measure for the non-orthogonality of a lattice basis.*, Combinatorics, Probability and Computing **8** (1999) 281-291.

[7] Zhang, W., Arnold, F., and Mai, X., *An analysis of Seysen's lattice reduction algorithm.*, Signal Processing **88**(10) (2008) 2573-2577.

On numerical integration of perturbed rigid body problems

A. Pascual and J. M. Ferrándiz

University of Alicante, Department of Applied Mathematics, Spain

Abstract

The accurate numerical integration of rigid and non-rigid solid bodies is an important issue in Space Geodesy. Its main difficulty is that a very high level of accuracy is required in many cases, as well as long-term validity of the solution. For instance, in the case of the Earth rotation the accuracy must be better than 0.15 milli arcseconds during time periods spanning several decades. Besides, the problem includes dissipations which prevents the use of the of symplectic or other geometric integrators and the variables that have been successful when deriving asymptotic analytical solutions give rise to virtual singularities. This presentation reports on the last progress we have made in the derivation of systems of variables and of equations of motion convenient for the numerical integration of such problems. An application to a simplified Earth rotation problem is included.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Building Public Key Crypto-Systems

Joachim Rosenthal

University of Zürich, Mathematics Institute, Switzerland

Abstract

Cryptography has a long history and its main objective is the transmission of data between two parties in a way which guarantees the privacy of the information. There are other interesting applications such as digital signatures, the problem of authentication and the concept of digital cash to name a few. The proliferation of computer networks resulted in a large demand for cryptography from the private sector.

A basic building block in public key cryptography are the one-way trapdoor functions. These are one-one functions which can be efficiently computed. The inverse function can however only be computed if some additional trapdoor is known. The best known one-way trapdoor function is the RSA function whose difficulty of inverting is related to the difficulty of factoring. Other one-way trapdoor functions use the arithmetic of elliptic curves and more general abelian varieties,

In this talk we will first provide a survey for the non-specialists. We then explain some new ideas on how to bulid one-way trapdoor functions from actions of finite simple semi-rings on finite semi-modules. The presented results constitute joint work with Elisa Gorla, Gerard Maze and Chris Monico and Jens Zumbrägel.

Multiphysics Simulation

Pablo Vallejos

Applications Department, COMSOL Multiphysics Sweden

Abstract

Simulation is a necessary task for every researcher and design engineer. Multiphysics simulation takes this task to the next level by introducing everything required to build precise comprehensive models. That is why multiphysics simulation is one of the fastest growing research fields in industrial engineering and academic research. In this presentation, you will be introduced to COMSOL Multiphysics. It is a simulation environment that facilitates all steps in the modeling process - defining your geometry, meshing, specifying your physics, solving, and then visualizing your results.

We will present the mechanics of the new (COMSOL Multiphysics 4.0) model-builder-based user interface which not only is much more efficient and quick to use, but also provides new functionality to the user to modify and quickly adapt models. We will work through a 3-physics coupled example to demonstrate the speed and efficiency of the new work flow. This will be of interest to both new and existing users of COMSOL Multiphysics.

 ISBN 13: 978-84-613-5510-5

Proceedings of the 10th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2010
27–30 June 2010.

Coupled Heat Transfer in Simulations

Pablo Vallejos

Applications Department, COMSOL Multiphysics Sweden

Abstract

In almost every manufacturing or product design process one must consider the effects of thermal fluctuations. A combination of capabilities to model heat transfer via conduction, convection, and radiation, as well as the ability to couple these to other physics is presented. In addition a case story of Ugitech S.A., a manufacturer of stainless steel in France is presented. It runs its continuing casting machines as fast as possible while maintaining quality. Yet, If it cuts off individual pieces from the square bloom coming out of the casts prematurely, the inside of the steel section will not completely solidify, and a molten metal well with as much as 1.5 tons of liquid steel can empty into the bottom sections of the vertical concast machine, causing major damage. Through modeling, Ugitech has optimized the proper temperatures and process speeds for each of the 150 different steel grades the company produces.

Randomization Techniques on Lattice Reduction Algorithms

U. Wagner

Abstract

Lattice reduction algorithms are of crucial importance in many cryptographic protocols. The goal of reduction algorithms is hereby to output lattice bases that consist of short and nearly orthogonal vectors. The notion of reducedness is not uniquely defined and several measurements of reducedness like the Seysen measure [1],[6] and the Gram-Schmidt log [2] exist. Often the output of the shortest vector in a lattice is desired. However it is hard in general to find short vectors in lattices in higher dimensions and known reduction algorithms such as LLL can tackle the problem of finding the shortest vector in a lattice only to a certain (low) dimension [3], [4]. Our work builds on the fact that most algorithms are not deterministic for the given lattice,

 ISBN 13: 978-84-613-5510-5

i.e. the basis to apply the reduction algorithm on in uences the performance of the algorithm. Hence randomization of the lattice basis randomizes the whole algorithm. Our interest lies in ran- domization techniques in order to find suitable bases, on which the reduction algorithms perform better than on the average basis. In order to recognize a considerable improvement the average behaviour of the reduction algorithms have to be known. Fortunately results in this direction exist ([2] and [5]), where extensive tests on the behaviour of LLL and BKZ have been done. In special the Hermite factor of the reduced bases is computed, which gives information on the quality of the shortest vector found by the reduction algorithm. Hence our goal is to have a comparison of the diferent randomization techniques by means of the Hermite factor.

References

[1] B. A. Lamacchia, *Basis Reduction Algorithms and Subset Sum Problems*, SM thesis, Mas-sachusetts Inst. Techno. (1991)

[2] N. Gama and P. Nguyen, *Predicting Lattice Reduction*, Lecture Notes in Computer Science, Springer Verlag, Berlin, (2008) Advances in Cryptology, Proc. Eurocrypt 08.

[3] D. Micciancio and S. Goldwasser, *Complexity of Lattice Problems: a cryptographic per- spective*, Kluwer Academic Publishers, Boston, Massachusetts. (2002)

[4] C. P. Schnorr and M. Euchner, *Lattice Basis Reduction: Improved Practical Algorithms ans Solving Subset Sum Problems*, Math. Programming. (1993) 181–191.

[5] P. Nguyen and D. Stehl, *LLL on the Average, Lecture Notes in Computer Science*, Annal. Mat. Pura ed Appl. **4076**,Springer Verlag, Berlin (2006) Proceeding of ANTS VII.

[6] G. Maze, *Some Inequalities Related to the Seysen Measure of a Lattice*, Submitted for publication. Preprint available at arXiv.

 ISBN 13: 978-84-613-5510-5