Journal of Computational and Applied Mathematics 236 (2012) 2990–2997

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A quasi-linear algorithm for calculating the infimal convolution of convex quadratic functions

L. Bayón*, J.M. Grau, M.M. Ruiz, P.M. Suárez

Department of Mathematics, University of Oviedo, Gijón, Spain

ARTICLE INFO

MSC:
90C20
90C25
90C60

Keywords:
Algorithm complexity
Infimal convolution
Quadratic programming

ABSTRACT

In this paper we present an algorithm of quasi-linear complexity to exactly calculate 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.

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 [1–3].

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:

$$(F \bigodot G)(x) := \inf_{y\in\mathbb{R}}\{F(x) + G(y - x)\}.$$

It is known that $(\mathcal{F}(R, \bar{R}), \bigodot)$ is a commutative semigroup. Furthermore, if $A = \{1, \ldots, N\}$, we have that

$$\left(\bigodot_{i\in A} F_i\right)(\xi) = \inf_{\sum\limits_{i\in A} x_i=\xi} \sum_{i\in A} F_i(x_i).$$

When the functions are considered to be constrained to 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:

$$(F_i \bigodot F_j)(K) := \min_{\substack{x_1+x_2=K\\ m_i\le x_i\le M_i}} (F_1(x_2) + F_2(x_2)) = \min_{\substack{m_1\le x\le M_1\\ m_2\le K-x_1\le M_2}} (F_1(x) + F_2(K - x))$$

$$\Psi^A(\xi) := \left(\bigodot_{i\in A} F_i\right)(\xi) = \min_{\substack{\sum\limits_{i\in A} x_i=\xi\\ m_i\le x_i\le M_i}} \sum_{i\in A} F_i(x_i).$$

* Corresponding address: EPI, Campus of Gijón, Spain.
E-mail address: bayon@uniovi.es (L. Bayón).

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.04.011

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 $\{\mathrm{Pr}^A(\xi)\}_{\xi\in\mathbb{R}}$ 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 [4] have been published that deal with applying QP algorithms to diverse problems. Within this extremely wide-ranging field of research, some authors, like for example [5,6], have sought the analytic solution for certain particular cases of QP problems with additional simplifications.

Focusing on our particular problem, $\{\mathrm{Pr}^A(\xi)\}_{\xi\in\mathbb{R}}$, several optimal algorithms have been presented for this bound and equality constrained QP problem. For example, [7] presents an algorithm of linear complexity for the case of a single equality constraint (fixed ξ), including only 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 $\{\mathrm{Pr}^A(\xi)\}_{\xi\in\mathbb{R}}$. This supposes a substantial improvement to a previous paper in [8] in which an algorithm was presented that, as we shall show in this paper, is one of quadratic computational complexity, $O(N^2)$.

The paper is organized as follows. In the next section we first provide some basic definitions and preparatory results to then proceed to present the description of the algorithm that leads to the determination of the optimal solution. The results of the computational complexity of the new algorithm and the previous version [8] are discussed in Section 3. In Section 4 we present a numerical example to demonstrate that the analytic solution obtained with our algorithm is able to deal with large-scale QP problems of this type. Finally, the main conclusions of our research are summarized in Section 5.

2. Algorithm

In this section, we first present the necessary definitions to build our algorithm.

Definition 2. If $\Psi^A := \bigodot_{i\in A} F_j(K)$, the i-th distribution functions

$$\Psi_i^A : \left[\sum_{i\in A} m_i, \sum_{i\in A} M_i\right] \longrightarrow [m_i, M_i]$$

that satisfy $\sum_{i\in A}\Psi_i^A(\xi) = \xi$ and $(\Psi_1^A, \dots, \Psi_N^A) \in \prod_{i=1}^N [m_i, M_i]$ are the solution of $Pr^A(\xi)$.

Definition 3. Let us consider in the set $A \times \{m, M\}$ the binary relation $\preccurlyeq$ defined as follows:

$$\begin{aligned} (i,m) \preccurlyeq (j,m) &\iff F_i'(m_i) < F_j'(m_j) \quad \text{or} \quad (F_i'(m_i) = F_j'(m_j) \text{ and } i \le j) \\ (i,m) \preccurlyeq (j,M) &\iff F_i'(m_i) < F_j'(M_j) \quad \text{or} \quad (F_i'(m_i) = F_j'(M_j) \text{ and } i \le j) \\ (i,M) \preccurlyeq (j,m) &\iff F_i'(M_i) < F_j'(m_j) \quad \text{or} \quad (F_i'(M_i) = F_j'(m_j) \text{ and } i \le j) \\ (i,M) \preccurlyeq (j,M) &\iff F_i'(M_i) < F_j'(M_j) \quad \text{or} \quad (F_i'(M_i) = F_j'(M_j) \text{ and } i \le j). \end{aligned}$$

Obviously, $\preccurlyeq$ is a total order relation and $(A \times \{m, M\}, \preccurlyeq)$ is isomorphic with respect to $(\{1, 2, \dots, 2N\}, \le)$.

Definition 4. We denote by g the isomorphism

$$g(n) := (g_1(n), g_2(n)), \quad g : (\{1, 2, \dots, 2N\}, \le) \longrightarrow (A \times \{m, M\}, \preccurlyeq)$$

which at each natural number $n \in \{1, 2, \dots, 2N\}$ corresponds to the n-th element of $A \times \{m, M\}$ following the order established by $\preccurlyeq$.

A proposition was demonstrated in [8] that allows us to interpret that the set $A \times \{m, M\}$ symbolizes the $2N$ possible states of activity/inactivity of the variable constraints and that the activation of the minimal constraints and the activation of the maximal constraints present an order of priority that the solution of the problem must necessarily respect. Thus, a vector $v = (a_1, \dots, a_N)$ which constitutes the solution of the problem $\mathrm{Pr}^A(\xi)$ and satisfies $a_i = m_i$ will necessarily also have to satisfy $a_k = m_k$ if $(i, m) \preccurlyeq (k, m)$ and, likewise, $a_k < M_k$ if $(i, m) \preccurlyeq (k, M)$. This fact, which is not exclusive to quadratic problems, is of extraordinary importance, seeing as it allows the 3^N possible combinations of activity/inactivity of the constraints to be reduced to only $2N + 1$.

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 the minimum are active, and ending with the triad $(\varnothing, \varnothing, A)$, which represents the fact that all the constraints on the maximum are active. We can interpret each triad as the representation of the state of activity of the constraints in the sense that the elements of Ω_n symbolize the variables whose lower constraint is active ($x_i = m_i$), Ξ_n the variables whose upper constraints are active ($x_i = M_i$), and Θ_n the variables whose constraints are both inactive. Each step of the process consists in decreasing the number of active constraints on a minimum by one unit or increasing the number of active constraints on a maximum by one unit, following the order established by the relation $\preccurlyeq$. Let us consider the following recurrent sequence, $X_n := (\Omega_n, \Theta_n, \Xi_n), n = 0, \ldots, 2N$:

$$\begin{array}{lll} \Omega_0 = A & \Theta_0 = \varnothing & \Xi_0 = \varnothing \\ \text{If } g_2(n) = M : \Omega_n = \Omega_{n-1} & \Theta_n = \Theta_{n-1} - \{g_1(n)\} & \Xi_n = \Xi_{n-1} \cup \{g_1(n)\} \\ \text{If } g_2(n) = m : \Omega_n = \Omega_{n-1} - \{g_1(n)\} & \Theta_n = \Theta_{n-1} \cup \{g_1(n)\} & \Xi_n = \Xi_{n-1}. \end{array}$$

We prove the following propositions.

Proposition 1. *There exist*

$$\{\phi_i\}_{i=1}^{2N} \subset \mathbb{R}, \qquad \sum_{i=1}^{N} m_i = \phi_1 \le \cdots \le \phi_{2N} = \sum_{i=1}^{N} M_i$$

such that $\forall \xi \,|\, \phi_n \le \xi < \phi_{n+1}$, *the solution of the problem* $v = (\Psi_1^A(\xi), \ldots, \Psi_N^A(\xi))$ *satisfies:*

$$\Psi_k^A(\xi) = \begin{cases} \dfrac{2\widehat{\gamma}_n(\xi - \mu_n) + \widehat{\beta}_n - \beta_k}{2\gamma_k} & \text{if} \quad k \in \Theta_n \\ m_k & \text{if} \quad k \in \Omega_n \\ M_k & \text{if} \quad k \in \Xi_n \end{cases}$$

being

$$\phi_1 = \sum_{i=1}^{N} m_i; \qquad \phi_n = \phi_{n-1} + \frac{1}{2}[s_n - s_{n-1}]\frac{1}{\widehat{\gamma}_{n-1}}$$

$$s_1 = 0; \qquad s_n = \begin{cases} s_{n-1} & \text{if} \quad \Theta_{n-1} = \varnothing \\ F'_{g_1(n)}(m_{g_1(n)}) & \text{if} \quad g_2(n) = m \wedge \Theta_{n-1} \neq \varnothing \\ F'_{g_1(n)}(M_{g_1(n)}) & \text{if} \quad g_2(n) = M \wedge \Theta_{n-1} \neq \varnothing \end{cases}$$

$$\widehat{\gamma}_n := \frac{1}{\sum\limits_{i \in \Theta_n} \frac{1}{\gamma_i}}$$

$$\widehat{\beta}_n := \widehat{\gamma}_n \sum_{i \in \Theta_n} \frac{\beta_i}{\gamma_i}$$

$$\mu_n := \sum_{i \in \Omega_n} m_i + \sum_{j \in \Xi_n} M_j$$

Proof. First, we suppose that $\Theta_n \neq \varnothing, \forall n = 1, \ldots, 2N - 1$. In this case we obtain the expression for each Ψ_k^A by reasoning similarly as in [9,10].

The values of $\{\phi_i\}_{i=1}^{2N}$ are easily established taking into account [8]:

As

$$\phi_n = \begin{cases} \dfrac{1}{2} \sum\limits_{i \in \Theta_n} \dfrac{F'_{g_1(n)}(m_{g_1(n)}) - \beta_i}{\gamma_i} + \sum\limits_{i \in \Omega_n} m_i + \sum\limits_{j \in \Xi_n} M_j & \text{if} \quad g_2(n) = m \\ \dfrac{1}{2} \sum\limits_{i \in \Theta_n} \dfrac{F'_{g_1(n)}(M_{g_1(n)}) - \beta_i}{\gamma_i} + \sum\limits_{i \in \Omega_n} m_i + \sum\limits_{j \in \Xi_n} M_j & \text{if} \quad g_2(n) = M. \end{cases}$$

If $g_2(n) = m$, then we have that

$$\phi_n = \frac{1}{2} \sum_{i \in \Theta_{n-1} \cup \{g_1(n)\}} \frac{F'_{g_1(n)}(m_{g_1(n)}) - \beta_i}{\gamma_i} + \mu_n$$

$$= \frac{1}{2}\left[\sum_{i\in\Theta_{n-1}} \frac{F'_{g_1(n)}(m_{g_1(n)}) - \beta_i}{\gamma_i} + \frac{F'_{g_1(n)}(m_{g_1(n)}) - \beta_{g_1(n)}}{\gamma_{g_1(n)}}\right] + \mu_{n-1} - m_{g_1(n)}$$
$$= \frac{1}{2}\sum_{i\in\Theta_{n-1}} \frac{F'_{g_1(n)}(m_{g_1(n)}) - \beta_i}{\gamma_i} + \mu_{n-1}.$$

Analogously, we obtain ϕ_n for the case in which $g_2(n) = M$. It can, therefore, be easily seen that

$$\phi_n = \begin{cases} \phi_{n-1} + \frac{1}{2}[F'_{g_1(n)}(m_{g_1(n)}) - F'_{g_1(n-1)}(m_{g_1(n-1)})]\frac{1}{\widehat{\gamma}_{n-1}} & \text{if} \quad g_2(n) = m \\ \phi_{n-1} + \frac{1}{2}[F'_{g_1(n)}(M_{g_1(n)}) - F'_{g_1(n-1)}(M_{g_1(n-1)})]\frac{1}{\widehat{\gamma}_{n-1}} & \text{if} \quad g_2(n) = M. \end{cases}$$

Denoting by

$$s_n = \begin{cases} F'_{g_1(n)}(m_{g_1(n)}) & \text{if} \quad g_2(n) = m \\ F'_{g_1(n)}(M_{g_1(n)}) & \text{if} \quad g_2(n) = M \end{cases}$$

we have that

$$\phi_n = \phi_{n-1} + \frac{1}{2}[s_n - s_{n-1}]\frac{1}{\widehat{\gamma}_{n-1}}$$

We now suppose that for a specified n ($n \neq 0, 2N$), $\Theta_n = \varnothing$. Following the order established by the relation $\preccurlyeq$, we have

$$\Theta_{n+1} = \{g_1(n+1)\} \neq \varnothing$$

and

$$\begin{aligned}\phi_{n+1} &= \frac{1}{2}\sum_{i\in\Theta_{n+1}} \frac{F'_{g_1(n+1)}(m_{g_1(n+1)}) - \beta_i}{\gamma_i} + \sum_{i\in\Omega_{n+1}} m_i + \sum_{j\in\Xi_{n+1}} M_j \\ &= \frac{1}{2}\frac{\beta_{g_1(n+1)} + 2\gamma_{g_1(n+1)}m_{g_1(n+1)} - \beta_{g_1(n+1)}}{\gamma_{g_1(n+1)}} + \sum_{i\in\Omega_n} m_i - m_{g_1(n+1)} + \sum_{j\in\Xi_n} M_j \\ &= \phi_n.\end{aligned}$$

We are again in a position to continue with the recurrent process. □

Proposition 2. *The function Ψ^A (infimal convolution) is a continuous function and a piecewise quadratic C^1 function. Specifically, if $\phi_n \le \xi < \phi_{n+1}$, (being ϕ_n the coefficients defined in Proposition 1) we have*

$$\Psi^A(\xi) = \widehat{\alpha}_n + \widehat{\beta}_n(\xi - \mu_n) + \widehat{\gamma}_n(\xi - \mu_n)^2$$

where:

$$\widehat{\gamma}_1 = \gamma_{g_1(1)}; \qquad \widehat{\beta}_1 = \beta_{g_1(1)}; \qquad \widehat{\alpha}_1 = \alpha_{g_1(1)} + \sum_{i\in\Omega_1} F_i(m_i); \qquad \mu_1 = \sum_{i=1}^{N} m_i - m_{g_1(1)}$$

and:

(i) *If $\Theta_n \neq \varnothing \wedge \Theta_{n-1} \neq \varnothing$:*

$$\mu_n = \begin{cases} \mu_{n-1} - m_{g_1(n)} & \textit{if} \quad g_2(n) = m \\ \mu_{n-1} + M_{g_1(n)} & \textit{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\alpha}_n = \begin{cases} \widehat{\alpha}_{n-1} + \alpha_{g_1(n)} - \dfrac{(\widehat{\beta}_{n-1} - \beta_{g_1(n)})^2}{4(\widehat{\gamma}_{n-1} + \gamma_{g_1(n)})} - F_{g_1(n)}(m_{g_1(n)}) & \textit{if} \quad g_2(n) = m \\ \widehat{\alpha}_{n-1} - \alpha_{g_1(n)} - \dfrac{(\widehat{\beta}_{n-1} - \beta_{g_1(n)})^2}{4(\widehat{\gamma}_{n-1} - \gamma_{g_1(n)})} + F_{g_1(n)}(M_{g_1(n)}) & \textit{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\beta}_n = \begin{cases} \dfrac{1}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}}[\widehat{\beta}_{n-1}\cdot\gamma_{g_1(n)} + \beta_{g_1(n)}\cdot\widehat{\gamma}_{n-1}] & \textit{if} \quad g_2(n) = m \\ \dfrac{1}{\widehat{\gamma}_{n-1} - \gamma_{g_1(n)}}[-\widehat{\beta}_{n-1}\cdot\gamma_{g_1(n)} + \beta_{g_1(n)}\cdot\widehat{\gamma}_{n-1}] & \textit{if} \quad g_2(n) = M \end{cases}$$

$$\widehat{\gamma}_n = \begin{cases} \dfrac{\widehat{\gamma}_{n-1} \cdot \gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}} & \text{if} \quad g_2(n) = m \\ -\dfrac{\widehat{\gamma}_{n-1} \cdot \gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} - \gamma_{g_1(n)}} & \text{if} \quad g_2(n) = M \end{cases}$$

(ii) *If* $\Theta_n \neq \varnothing \wedge \Theta_{n-1} = \varnothing$*:*

$$\mu_n = \mu_{n-1} - m_{g_1(n)}; \qquad \widehat{\alpha}_n = \widehat{\alpha}_{n-1} + \alpha_{g_1(n)} - F_{g_1(n)}(m_{g_1(n)}); \qquad \widehat{\beta}_n = \beta_{g_1(n)}; \qquad \widehat{\gamma}_n = \gamma_{g_1(n)}$$

(iii) *If* $\Theta_n = \varnothing$*:*

$$\widehat{\alpha}_n = \widehat{\alpha}_{n-1} - \alpha_{g_1(n)} + F_{g_1(n)}(M_{g_1(n)}) = \sum_{i\in\Omega_n} F_i(m_i) + \sum_{i\in\Xi_n} F_i(M_i); \widehat{\beta}_n := 0; \widehat{\gamma}_n := 0$$

Proof. Its continuous and piecewise quadratic C^1 character, is easily proven by simply using the technique employed in [9,10].

(i) First we suppose that $\Theta_n \neq \varnothing \wedge \Theta_{n-1} \neq \varnothing$.

The values of the coefficients are easily established taking into account the values of [8] and the constructed recurrent sequence. We provide the demonstration for the case in which $g_2(n) = m$, the process being analogous for the case $g_2(n) = M$.

As $\mu_n := \sum_{i\in\Omega_n} m_i + \sum_{j\in\Xi_n} M_j$, it is evident that

$$\mu_n = \mu_{n-1} - m_{g_1(n)}.$$

As $\widehat{\gamma}_n := \frac{1}{\sum_{i\in\Theta_n} \frac{1}{\gamma_i}}$, then

$$\widehat{\gamma}_n = \frac{1}{\sum\limits_{i\in\Theta_{n-1}\cup\{g_1(n)\}} \frac{1}{\gamma_i}} = \frac{1}{\sum\limits_{i\in\Theta_{n-1}} \frac{1}{\gamma_i} + \frac{1}{\gamma_{g_1(n)}}} = \frac{1}{\frac{1}{\widehat{\gamma}_{n-1}} + \frac{1}{\gamma_{g_1(n)}}} = \frac{\widehat{\gamma}_{n-1} \cdot \gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}}.$$

As $\widehat{\beta}_n := \widehat{\gamma}_n \sum_{i\in\Theta_n} \frac{\beta_i}{\gamma_i}$, then

$$\begin{aligned} \widehat{\beta}_n &= \widehat{\gamma}_n \sum_{i\in\Theta_{n-1}\cup\{g_1(n)\}} \frac{\beta_i}{\gamma_i} = \frac{\widehat{\gamma}_{n-1}\cdot\gamma_{g_1(n)}}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}} \left[\sum_{i\in\Theta_{n-1}} \frac{\beta_i}{\gamma_i} + \frac{\beta_{g_1(n)}}{\gamma_{g_1(n)}}\right] \\ &= \frac{1}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}} \left[\widehat{\gamma}_{n-1} \sum_{i\in\Theta_{n-1}} \frac{\beta_i}{\gamma_i} \cdot \gamma_{g_1(n)} + \widehat{\gamma}_{n-1} \cdot \beta_{g_1(n)}\right] \\ &= \frac{1}{\widehat{\gamma}_{n-1} + \gamma_{g_1(n)}} \left[\widehat{\beta}_{n-1} \cdot \gamma_{g_1(n)} + \beta_{g_1(n)} \cdot \widehat{\gamma}_{n-1}\right]. \end{aligned}$$

As $\widehat{\alpha}_n := \sum_{i\in\Theta_n} \alpha_i + \frac{\widehat{\beta}_n^2}{4\widehat{\gamma}_n} - \sum_{i\in\Theta_n} \frac{\beta_i^2}{4\gamma_i} + \sum_{i\in\Omega_n} F_i(m_i) + \sum_{i\in\Xi_n} F_i(M_i)$, then

$$\begin{aligned} \widehat{\alpha}_n &= \sum_{i\in\Theta_{n-1}\cup\{g_1(n)\}} \alpha_i + \frac{\widehat{\beta}_n^2}{4\widehat{\gamma}_n} - \sum_{i\in\Theta_{n-1}\cup\{g_1(n)\}} \frac{\beta_i^2}{4\gamma_i} + \sum_{i\in\Omega_{n-1}-\{g_1(n)\}} F_i(m_i) + \sum_{i\in\Xi_{n-1}} F_i(M_i) \\ &= \sum_{i\in\Theta_{n-1}} \alpha_i + \alpha_{g_1(n)} + \frac{\widehat{\beta}_n^2}{4\widehat{\gamma}_n} - \sum_{i\in\Theta_{n-1}} \frac{\beta_i^2}{4\gamma_i} - \frac{\beta_{g_1(n)}^2}{4\gamma_{g_1(n)}} + \sum_{i\in\Omega_{n-1}} F_i(m_i) - F_{g_1(n)}(m_{g_1(n)}) + \sum_{i\in\Xi_{n-1}} F_i(M_i) \\ &= \sum_{i\in\Theta_{n-1}} \alpha_i - \sum_{i\in\Theta_{n-1}} \frac{\beta_i^2}{4\gamma_i} + \sum_{i\in\Omega_{n-1}} F_i(m_i) + \sum_{i\in\Xi_{n-1}} F_i(M_i) + \alpha_{g_1(n)} + \frac{\widehat{\beta}_n^2}{4\widehat{\gamma}_n} - \frac{\beta_{g_1(n)}^2}{4\gamma_{g_1(n)}} - F_{g_1(n)}(m_{g_1(n)}). \end{aligned}$$

Considering $\frac{\widehat{\beta}_n^2}{4\widehat{\gamma}_n}$ in terms of $\widehat{\beta}_{n-1}$ and $\widehat{\gamma}_{n-1}$ and operating, we obtain that

$$\widehat{\alpha}_n = \widehat{\alpha}_{n-1} + \alpha_{g_1(n)} - \frac{(\widehat{\beta}_{n-1} - \beta_{g_1(n)})^2}{4(\widehat{\gamma}_{n-1} + \gamma_{g_1(n)})} - F_{g_1(n)}(m_{g_1(n)}).$$

(ii) If $\Theta_n \neq \varnothing \wedge \Theta_{n-1} = \varnothing$, bearing in mind the order established for the binary relation $\preccurlyeq$, we have

$$\Theta_n = \{g_1(n)\} \text{ and } g_2(n) = m.$$

Moreover, from Proposition 1, we have that

$$\mu_n = \mu_{n-1} - m_{g_1(n)}; \qquad \widehat{\beta}_n = \beta_{g_1(n)}; \qquad \widehat{\gamma}_n = \gamma_{g_1(n)}$$

and from the definition of $\widehat{\alpha}_n$ in (i), it is easy to see that

$$\widehat{\alpha}_n = \widehat{\alpha}_{n-1} + \alpha_{g_1(n)} - F_{g_1(n)}(m_{g_1(n)}).$$

(iii) If $\Theta_n = \varnothing$, we take

$$\widehat{\beta}_n := 0; \qquad \widehat{\gamma}_n := 0.$$

Bearing in mind the order established for the binary relation $\preccurlyeq$, we have that Θ_{n-1} is a unit set, and from $\Theta_n = \varnothing$ it is evident that

$$g_2(n) = M.$$

Finally, from the definitions of μ_n and $\widehat{\alpha}_n$, we obtain that

$$\mu_n = \mu_{n-1} + M_{g_1(n)}$$
$$\widehat{\alpha}_n = \widehat{\alpha}_{n-1} - \alpha_{g_1(n)} + F_{g_1(n)}(M_{g_1(n)}) = \sum_{i\in\Omega_n} F_i(m_i) + \sum_{i\in\Xi_n} F_i(M_i) \quad \square$$

Remark. If $\Theta_n \neq \varnothing$, $\forall n = 1, \dots, 2N-1$, also using the technique employed in [9,10] provides the character C^1 of Ψ^A.

If $\Theta_n = \varnothing$, there is no guarantee that Ψ^A belongs to class C^1, because it may be not be derivable in ϕ_i. This is what occurs in the example we shall see in Section 4.

3. Computational complexity of the algorithm

In this section we analyze the complexity of the algorithm presented in this paper and compare it to the one presented in [8]. In the former case we shall have complexity of a quasi-linear order, $O(N\log(N))$, and in the latter, quadratic complexity, $O(N^2)$. As both cases share the underlying idea and cannot avoid the ordering of the set $A \times \{m, n\}$, the quasi-linear complexity cannot thus be improved.

Given the family of strictly convex quadratic functions $F_i(x_i) = \alpha_i + \beta_i x_i + \gamma_i x_i^2$ with $i = 1, \dots, N$ 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:

$$\{\{m_1, M_1, \alpha_1, \beta_1, \gamma_1\}, \{m_2, M_2, \alpha_2, \beta_2, \gamma_2\}, \dots, \{m_N, M_N, \alpha_N, \beta_N, \gamma_N\}\}.$$

The output, which we symbolize as:

$$\{\{\phi_1, \phi_2, \widehat{\alpha}_1, \widehat{\beta}_1, \widehat{\gamma}_1\}, \dots, \{\phi_n, \phi_{n+1}, \widehat{\alpha}_n, \widehat{\beta}_n, \widehat{\gamma}_n\}, \dots, \{\phi_{2N-1}, \phi_{2N}, \widehat{\alpha}_{2N}, \widehat{\beta}_{2N}, \widehat{\gamma}_{2N}\}\}$$

shall represents the infimal convolution

$$\left(\bigodot_{i\in A} F_j\right)(K) = \widehat{\alpha}_i + \widehat{\beta}_i K + \widehat{\gamma}_i K^2$$

which symbolizes the $2N$ polynomials with their respective intervals of action.

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 $X_n := (\Omega_n, \Theta_n, \Xi_n),\ n = 0, \dots, 2N$.
(D) Construction of the sequence $s_n,\ n = 0, \dots, 2N$.
(E) Construction of the sequences $\widehat{\alpha}_n, \widehat{\beta}_n, \widehat{\gamma}_n,\ n = 1, \dots, 2N-1$.
(F) Construction of the sequences $\phi_n, n = 1, \dots, 2N$.

For the aforementioned algorithm, we prove the following proposition.

Proposition 3. *The complexity of the aforementioned algorithm is quasi-linear:*

$$O(N\log(N))$$

Proof. Phases (A) and (C) The complexity is $O(N)$.

Phase (B) Using merge sort, we have complexity $O(N\log(N))$.

Phases (D), (E) and (F) Bearing in mind that the construction is recurrent, with a constant complexity for the loop, with a recurrence

$$g(n) = O(1) + g(n-1)$$

the complexity of this phase is $O(N)$.

In short, the complexity of the entire algorithm is, in fact, that of phase (B): $O(N\log(N))$ which dominates the others, which are linear in order. $\square$

Table 1
CPU time (s).

	n	400	800	1600	3200	6400	12800
t CPU	New algorithm	0.05	0.2	1.1	3.6	18.7	86.23
t CPU	Old algorithm [8]	10.46	66.8	433	3855.346	35201	319227

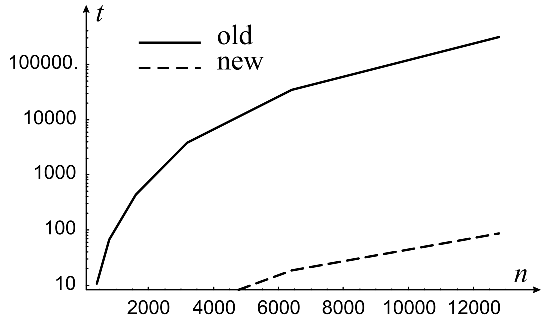

Fig. 1. CPU time.

For the prior algorithm, we prove the following proposition.

Proposition 4. *The complexity of the algorithm* [8] *is quadratic:*

$$O(N^2)$$

Proof. Phases (A), (B), (C) are common to both versions and their joint complexity is $O(N\log(N))$.
Phase (D) is not explicitly required.
Phases (E) and (F) are of complexity $O(N^2)$ as the calculation of each

$$\{\phi_n, \widehat{\alpha}_n, \widehat{\beta}_n, \widehat{\gamma}_n\}$$

requires $O(2N)$ operations. □

4. Example

In this section we present an example of a large-scale QP problem. Both the new and the old algorithm presented in [8] were implemented using the symbolic calculus program Mathematica. We shall generate an example, which is very easy to reproduce, considering the quadratic model: $F_i(x) = \alpha_i + \beta_i x + \gamma_i x^2$, generating the coefficients with the formulas:

$$\alpha_i = 0; \qquad \beta_i = i; \qquad \gamma_i = \frac{1}{2i}; \qquad m_i = \frac{1}{i}; \qquad M_i = \frac{1}{i} + 1, \quad i = 1, \ldots, n$$

Table 1 presents the CPU time employed (in seconds, measured on a Pentium IV, 3.4 GHz PC) by the two algorithms analyzed in this paper. We considered different values of n and present the times corresponding to phases A, C, D, E and F, the ones in which these differ. Phase B (ordering of the set $A \times \{m, M\}$ following the ordering relation $\preccurlyeq$) is common to both algorithms and was performed using the *Sort* command available in the Mathematica package.

Fig. 1 shows, in logarithmic scale, the time employed by both algorithms versus n.

As can be appreciated, the new algorithm supposes a substantial improvement to its predecessor. For values of tens of thousands of variables, it shows itself to be both a powerful and robust tool.

5. Conclusions

In this paper we have provided a complete analytic solution to a family of separable convex quadratic programming problems with bound and equality constraints. This study constitutes a substantial improvement to a prior paper in which the computational complexity of the algorithm was much greater. We have demonstrated that our algorithm is able to deal with large-scale QP problems. Finally, we underline the fact that these algorithms do not solve a single concrete problem of separable quadratic programming, but rather an uniparametric family of problems resulting from varying the equality constraint.

Acknowledgment

This work was supported by the Government of Principality of Asturias (PCTI: FICYT IB09-085) and by the Spanish Government (MICINN, project: MTM2010-15737).

References

[1] J.J. Moreau, Inf-convolution, sous-additivit e, convexit eriques, J. Math. Pures et Appl. 49 (1970) 109–154.
[2] R.T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, 1970.
[3] T. Stromberg, The operation of infimal convolution, Diss. Math. 352 (1996).
[4] N.I.M. Gould, PH.L. Toint, A quadratic programming bibliography, 2001, http://www.optimization-online.org/DB_HTML/2001/02/285.html.
[5] C. Audet, P. Hansen, S. Le Digabel, Exact solution of three nonconvex quadratic programming problems, in: In Frontiers in Global Optimization, in: Nonconvex Optimization and Applications, vol. 20, Kluwer Acad. Publ., Dordrecht, 2004, pp. 25–45.
[6] S. Kim, M. Kojima, Exact solutions of some nonconvex quadratic optimization problems via SDP and SOCP relaxations, Comput. Optim. Appl. 26 (2) (2003) 143–154.
[7] 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.
[8] L. Bayon, J.M. Grau, M.M. Ruiz, P.M. Suarez, An analytic solution for some separable convex quadratic programming problems with equality and inequality constraints, J. Math. Inequ. 4 (3) (2010) 453–465.
[9] L. Bayon, J.M. Grau, P.M. Suarez, A new formulation of the equivalent thermal in optimization of hydrothermal systems, Math. Probl. Eng. 8 (3) (2002) 181–196.
[10] L. Bayon, J.M. Grau, M.M. Ruiz, P.M. Suarez, New developments on equivalent thermal in hydrothermal optimization: an algorithm of approximation, J. Comput. Appl. Math. 175 (1) (2005) 63–75.