Report n. 12

A new algorithm for the strictly convex quadratic programming problem

Claudio Sodini

Pisa, 1988
This research is partially supported by Public Education Ministry.

A new algorithm for the strictly convex quadratic programming problem

Claudio Sodini

Dipartimento di Statistica e Matematica Applicata all'Economia
Universita' di Pisa

Abstract

A finite algorithm for the strictly convex quadratic programming problem is proposed. The algorithm, which is of the binding constraints type, is similar to that proposed by Houthakker [4], but is more general and uses different optimality conditions.

Key words: quadratic programming, Kuhn-Tucker conditions, binding constraint.

1. Introduction

In this paper we consider the strictly convex quadratic programming problem :

$$\text{(1.1)} \qquad \begin{aligned} &\min z = c^T x + c'^T x + 1/2\, x^T Q x \\ &x \in S = \{ x \in R^n : Ax \geq b \} \end{aligned}$$

where Q is a symmetric positive definite nxn matrix, $c, c' \in R^n$ $b \in R^m$, A is a mxn matrix. Clearly problem (1.1) is convex and thus any local minimum point is also global.

Recently [3,5] some non linear programs have been solved efficiently by means of parametric algorithms. The aim of this paper is to show that via the parametric approach it is possible to devise a new finite algorithm for the strictly convex quadratic programming problem. Many parametric algorithms for the strictly convex quadratic programming problem have been proposed in the literature [6]. The algorithm proposed by Cambini in [2] solves the strictly convex quadratic programming problem by generating a finite sequence of dual solutions. The unconstrained minimum of the objective function is used as starting solution. In the algorithm, at each step, a different constraint is parameterized. Houthakker in [4] introduces the parameter in the right hand side of a constraint of the form $\Sigma_j x_j \leq d$, $d \in R$. When the constraint $\Sigma_j x_j \leq d$ does not appear in the set of the original constraints a variable transformation must be made in order to obtain such a constraint. The algorithm of Houthakker is not a general algorithm, in fact it requires that the point $x=0$ belong to the feasible region. In this paper a parametric algorithm, similar to that of Houthakker, is proposed. The linear form of the objective function is parameterized as following $c^T x = \xi$, $\xi \in R$, and added to the constraints. Optimality conditions related to the parameter ξ are studied. The optimality conditions differ from those used by the algorithm of Houthakker. The proposed algorithm is of the binding constraints type and can use any vertex of the feasible region as a starting point; the optimal solution is obtained in a finite number of iterations.

2. Optimality conditions

In this section we give some optimality conditions for problem (1.1). If we add the constraint $c^T x = \xi$, $\xi \in R$ to problem (1.1) the following problem is obtained :

$$P(\xi): \quad \begin{array}{l} z(\xi) = \xi + \min 1/2\, x^T Q x + c'^T x \\ x \in S \\ c^T x = \xi \end{array}$$

which is equivalent to the problem :

$$P'(\xi): \quad \begin{array}{l} \min 1/2\, x^T Q x + c'^T x \\ x \in S \\ c^T x = \xi . \end{array}$$

Clearly problem (1.1) is equivalent to problem $P(\xi)$, when ξ is the level corresponding to the optimal solution of (1.1). Our aim is to describe a procedure which, starting from a feasible level ξ_0 (i.e. $S \cap \{x \in R^n : c^T x = \xi_0\} \neq \phi$), allows us to verify if ξ_0 is the optimal level and, if not, to find a new level ξ' such that $z(\xi') < z(\xi_0)$. Let x_0 be the optimal solution of problem $P'(\xi_0)$ and let $Mx=h$ be the equations of the constraints binding at x_0 ; we can suppose, without loss of generality, that M has full rank.

Let

$$(2.1a) \qquad x(\theta) = x_0 + \theta \alpha ;$$

$$(2.1b) \qquad \mu(\theta) = \mu' + \theta \gamma ;$$

$$(2.1c) \qquad \mu_0(\theta) = \mu'_0 + \theta \beta ;$$

be the solution of the linear system :

$$\begin{array}{lll}
(2.2a) & Qx - c\mu_0 - M^T\mu & = -c' \\
(2.2b) & c^Tx & = \xi_0+\theta \\
(2.2c) & Mx & = h.
\end{array}$$

Set $H(\theta) = \{\theta : x(\theta)\in S\}\cap\{\theta : \mu(\theta) \geq 0\}$. As (2.2) are the Kuhn-Tucker conditions of the parametric problem $P'(\xi_0+\theta)$ then it follows that $x(\theta)$, $\theta\in H(\theta)$, is the optimal solution of $P'(\xi_0+\theta)$. Clearly $\mu(\theta)$ and $\mu_0(\theta)$ are, respectively, the vectors of Lagrange multipliers associated to the constraints (2.2b), (2.2c). Obviously (x_0, μ', μ'_0) is the solution of (2.2) for $\theta=0$; hence μ', μ'_0 are the Lagrange multipliers associated to the optimal solution x_0 of $P'(\xi_0)$.

Set $z_0 = z(\xi_0) = \xi_0 + 1/2\, x_0{}^TQx_0 + c'^Tx_0$, $z(\theta) = z(\xi_0+\theta) = \xi_0 + \theta + 1/2\, x(\theta)^TQx(\theta) + c'^Tx(\theta)$. The following lemma gives an explicit form for the function $z(\theta)$, $\theta\in H(\theta)$.

<u>Lemma 2.1</u>

Suppose $H(\theta)\neq\{0\}$; then it is :

a) $c^T\alpha = 1$, $M\alpha = 0$, $\alpha^TQ\alpha = \beta$, $\alpha^TQx_0 = \mu'_0 - \alpha^Tc'$;

b) $z(\theta) = z_0 + (\mu'_0+1)\,\theta + 1/2\,\beta\theta^2$.

Proof : a) follows directly by substituting (2.1) in (2.2) ; it is $z(\theta)=1/2(x_0+\theta\alpha)^TQ(x_0+\theta\alpha) + \xi_0 + \theta + c'^Tx_0 + \theta\, c'^T\alpha$ and taking into account of a) we obtain b) ; in fact it results :

$z(\theta) = 1/2\, x_0{}^TQx_0 + \alpha^TQx_0\,\theta + 1/2\,\alpha^TQ\alpha\,\theta^2 + \xi_0 + \theta + c'^Tx_0 + \theta\, c'^T\alpha$;

$z(\theta) = z_0 - \xi_0 - c'^Tx_0 + \mu'_0\,\theta - \theta\,\alpha^Tc' + 1/2\,\beta\,\theta^2 + \xi_0 + \theta + c'^Tx_0 + \theta\, c'^T\alpha$;

$z(\theta) = z_0 + (\mu'_0+1)\,\theta + 1/2\,\beta\theta^2$.

The following lemma holds.

<u>Lemma 2.2</u>

If $\mu'_0 > -1$ ($\mu'_0 < -1$) then the function $z(\theta)$ is increasing (decreasing) at $\theta = 0$.

Proof : It is $z'(\theta) = \mu'_0 + 1 + \beta\theta$. Hence we have $z'(0) = \mu'_0 + 1 > 0$ (< 0) according to $\mu'_0 > -1$ ($\mu'_0 < -1$).

In order to state some sufficient optimality conditions for problem (1.1) we set :

$U(\theta) = H(\theta)\cap(-\infty,0]$, if $\mu'_0 > -1$;

$U(\theta) = H(\theta) \cap [0,+\infty)$, if $\mu'_0 < -1$;
$\theta' = -(\mu'_0 + 1)/\beta$.

The following theorem holds :

Theorem 2.1

a) If $\mu'_0 = -1$, then x_0 is the optimal solution for problem (1.1);
b) If $\theta' \in U(\theta)$, then $x(\theta')$ is the optimal solution for problem (1.1).

Proof : a) The equality $\mu'_0 = -1$ implies $z'(0) = 0$; since $z(\theta)$ is convex a) follows. b) It results $z'(\theta') = 0$; since $z(\theta)$ is convex b) follows.

Let x_0 be a vertex of S; in x_0 at least n constraints of S are binding as well as the parametric constraint and thus x_0 is a degenerate basic solution. Clearly, the different bases containing the parametric constraint are n if x_0 is a non degenerate vertex of S ; more than n if x_0 is a degenerate vertex of S. To point out the dependence of $z(\theta)$, $\mu(\theta)$, ... on the basis B, we write $z_B(\theta)$, $\mu_B(\theta)$, ... A basis B is said feasible if $\mu'_B \geq 0$.

The following theorem holds :

Theorem 2.2

a) If there are two different feasible bases B_1 and B_2, such that $\mu'_{0B_1} > -1$, $\mu'_{0B_2} < -1$ or $\mu'_{0B_1} < -1$, $\mu'_{0B_2} > -1$ then x_0 is the optimal solution for problem (1.1) ;

b) If we have $U_B(\theta) = \{0\}$ for any feasible basis B then x_0 is the optimal solution for problem (1.1).

Proof : a) Since $\mu'_{0B_1} > -1$, $\mu'_{0B_2} < -1$ ($\mu'_{0B_1} < -1$, $\mu'_{0B_2} > -1$) , from lemma 2.2, we obtain $z(\theta) \geq z_0$ in a neighbourhood of 0, on the right (left) and on the left (right), respectively. Hence x_0 is a local minimum point for problem (1.1). From the convexity of problem (1.1) it follows that x_0 is also a global minimum point. b) It follows directly from the definition of $U_B(\theta)$.

3. An algorithm for problem (1.1)

The results of the previous sections can be used to propose a finite algorithm for problem (1.1). The algorithm is the following :

Step 0 Determine a vertex x_0 of S optimal for problem $P'(\xi_0 = c^T x_0)$. Go to **Step 2**.

Step 1 If b) of theorem 2.1 is fulfilled, $x_0 + (-(\mu'_0 + 1)/\beta)\alpha$ is the optimal solution. Otherwise let θ^* be the end point of $U(\theta)$ different from zero. Set $x^* = x_0 + \theta^* \alpha$ (x^* is an optimal solution of problem $P'(\xi^* = \xi_0 + \theta^*)$) and delete the constraint i such that $\mu'_i(\theta') = 0$ and add the constraint j such that $a_j^T x^* > b_j$ and $a_j^T x^* = b_j$ (a_j^T denote the j-th row of A). Set $x_0 = x^*$, $\xi_0 = \xi^*$ and determine α, β, γ, μ'_0, μ'.
If $\mu'_0 = -1$ then x_0 is the optimal solution; otherwise go to **Step 2**.

Step 2 If x_0 is not a vertex of S, go to **Step 1** ; otherwise select a feasible basis of x_0 and compute the corresponding α, β, γ, μ'_0, μ'. If $\mu'_0 = -1$ then x_0 is the optimal solution; otherwise go to **Step 3**.

Step 3 If theorem 2.2 holds , x_0 is the optimal solution ; otherwise go to **Step 1**.

Theorem 3.1

The proposed algorithm, given a starting point, determines the optimal solution of problem (1.1) in a finite number of iterations.

Proof : Starting from x_0 the algorithm generates x^* such that :
i) $z(x^*) < z(x_0)$; ii) if x^* is not an optimal solution, then there exists a set of binding constraints at x^* which differs from the set of binding constraints at x_0. Clearly conditions i), ii) guarantee the convergence of the algorithm in a finite number of iterations.

The algorithm requires the capability of determining a vertex $x_0 \in S$ optimal for the problem :

$$P'(c^T x_0): \quad \begin{array}{l} \min 1/2\, x^T Q x + c'^T x \\ x \in S \\ c^T x = c^T x_0 . \end{array}$$

Notice that if $S \cap \{x : c^T x = c^T x_0\} = \{x_0\}$, clearly x_0 is an optimal solution of $P'(c^T x_0)$. Such a vertex can be easily obtained by solving the linear programming problem (3.1) $\min c^T x$, $x \in S$ or (3.2) $\max c^T x$, $x \in S$. In fact if x_0 solves (3.1) or (3.2) and is the unique optimal solution then $c^T x = c^T x_0$ is a supporting hyperplane and $S \cap \{x : c^T x = c^T x_0\} = \{x_0\}$. When x_0 is not the unique optimal solution of (3.1) or (3.2) clearly $S \cap \{x : c^T x = c^T x_0\} \neq \{x_0\}$ and in general x_0 is not the optimal solution of $P'(c^T x_0)$. In this case the point x_0 can be used as a starting point in this way.

Let $c^* \in R^n$ be a vector such that $S \cap \{x : c^{*T} x = c^{*T} x_0\} = \{x_0\}$; notice that such a vector can be easily obtained by a positive linear combination of the rows corresponding to the binding constraints at x_0.

Let us consider the problem :

$$(3.3) \qquad \begin{aligned} &\min z = 1/2\, x^T Q x + c^{*T} x + c''^T x \\ &Ax \geq b \end{aligned}$$

where $c'' = c + c' - c^*$. Problem (3.3) is exactly problem (1.1). Clearly x_0 is an optimal solution for problem :

$$(3.4) \qquad \begin{aligned} &\min 1/2\, x^T Q x + c''^T x \\ &x \in S \\ &c^{*T} x = c^{*T} x_0 \end{aligned}$$

as $S \cap \{x : c^{*T} x = c^{*T} x_0\} = \{x_0\}$ and then can be used as a starting point for the algorithm with $c = c^*$ and $c' = c''$. In this way any vertex of S can be used as a starting point.

4. Computational aspects

In this section we consider some computational aspects related to the proposed algorithm. In **Step 2** a feasible basis of x_0 must be determined. If x_0 is a non degenerate vertex then M is a non singular nxn matrix ; if x_0 is a degenerate vertex then M is a kxn matrix, k>n, which contains a non singular nxn submatrix.

Set

$$M = \begin{bmatrix} M_1 \\ \text{-----} \\ M_2 \end{bmatrix}$$

where M_1 is a non singular nxn matrix and M_2 the matrix containing the remaining rows (if any). Let us consider the system:

$$
(4.1) \qquad \begin{aligned} & c\,\mu_0 + M_1{}^T\mu_1 + M_2{}^T\mu_2 = c' + Qx_0 \\ & \mu_1\,,\mu_2 \geq 0 \end{aligned}
$$

A feasible basis of x_0 corresponds to a feasible basic solution of system (4.1) containing μ_0 as a basic variable. Given a feasible basis a new feasible basis corresponds to a feasible basic solution of system (4.1), containing μ_0 as a basic variable, different from the current feasible basic solution, and then can be obtained by a pivot operation. In this way all the feasible bases can be enumerated by a finite sequence of pivot operations.

In **Step 1** a new binding constraint is added to the Kuhn-Tucker system or an old binding constraint is deleted from the Kuhn-Tucker system. After this operation the new values of α, β, ... must be calculated. Let us consider the Kuhn-Tucker system:

$$
(4.2) \qquad \begin{aligned} Qx - c\,\mu_0 - M^T\mu &= -c' \\ c^Tx &= \xi_0+\theta \\ Mx &= h. \end{aligned}
$$

System (4.2) has been solved and then the inverse G^{-1} of the matrix

$$
G = \begin{bmatrix} Q & -c & -M^T \\ c^T & 0 & 0 \\ M & 0 & 0 \end{bmatrix}
$$

is known. When a new constraint , say $a_p{}^Tx = h_p$, is added the following Kuhn-Tucker system is obtained:

$$
(4.3) \qquad \begin{aligned} Qx - c\,\mu_0 - M^T\mu - a_p\,\mu_p &= -c' \\ c^Tx &= x_0+\theta \\ Mx &= h \\ a_p{}^Tx &= h_p \end{aligned}
$$

To solve (4.3) the inverse of the matrix:

$$
G' = \left[\begin{array}{cc} G & -a^*_p \\ \hline a^*_p{}^T & 0 \end{array}\right],
$$

where $a^*_p{}^T = (a_p{}^T, 0, \ldots, 0)$, must be computed. The inverse of G' can be easily obtained by using G^{-1}; in fact it results:

$$G'^{-1} = \begin{bmatrix} G & -a^*_p \\ \hline a^{*T}_p & 0 \end{bmatrix}^{-1} = \begin{bmatrix} G^{-1} - t\,G^{-1}a^*_p a^{*T}_p G^{-1} & G^{-1}a^*_p a^{*T}_p \\ \hline -t\,a^{*T}_p G^{-1} & t \end{bmatrix}$$

where $t=1/a^{*T}_p G^{-1}a^*_p$. Vice versa when the inverse G'^{-1} is known and the constraint $a_p^T x = h_p$ is deleted from the Kuhn-Tucker system (4.3), the Kuhn-Tucker system (4.2) is obtained and then the inverse of the matrix G must be computed. The matrix G^{-1} can be obtained by doing a pivot operation on matrix G'^{-1} ; in fact pivoting on the element t of G'^{-1} we obtain the matrix

$$\begin{bmatrix} G^{-1} & 0 \\ \hline -a^{*T}_p G^{-1} & 1 \end{bmatrix}$$

which contains the inverse of G.

5. Numerical example

Let us consider the problem:

$$\min f(x) = -2x_1 - x_2 + 1/2\,(x_1, x_2) \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$S = \{ x \in R^2 :$ (1) $2x_1 + 2x_2 \geq 3$, (2) $-x_1 + x_2 \geq -2$, (3) $-x_2 \geq -2$,
(4) $x_1 \geq 0$, (5) $x_2 \geq 0$ $\}$.

Applying the algorithm to the problem, the following sequence of steps is obtained:

Step 0. By solving the linear problem $\min -2x_1 - x_2$, $x \in S$, we obtain the vertex $x_0 = (4, 2)$ which can be used as a starting point of the algorithm.

Step 2. x_0 is a vertex of S. In x_0 the constraints (2),(3) and the parametric constraint $-2x_1 - x_2 = -10 + \theta$ are binding. A feasible basis associated to x_0 is given by the parametric constraint and constraint (3). With respect to this basis it results $\alpha=(-1/2,0)$, $\beta=3/4$, $\gamma=-1/4$, $\mu'_0=-7$,

$\mu'_3=1$.

Step 3. Theorem 2.2 does not hold.

Step 1. It results $U(\theta)=[0,4]$, $\theta'=8$. Setting $\theta^*=4$ we obtain the new point $x^*=(4,2)+4(-1/2,0)=(2,2)$ and $\mu'_0(4)=-4$, $\mu'_3(4)=0$. Since $\mu'_3(4)=0$, constraint (3) is deleted and α, β, μ'_0 corresponding to $x_0=x^*=(2,2)$ must be calculated. It results $\alpha=(-1/3,-1/3)$, $\beta=2/3$, $\mu'_0=-4$.

Step 2. x_0 is not a vertex of S.

Step 1. It results $U(\theta)=[0,15/4]$, $\theta'=9/2$. Setting $\theta^*=15/4$ we obtain the new point $x^*=(2,2)+15/4(-1/3,-1/3)=(3/4,3/4)$. Constraint (1) is now binding and then must be added. With respect to the new basis it results $x_0=x^*=(3/4,3/4)$, $\alpha=(-1,1)$, $\beta=2$, $\gamma=1$, $\mu'_0=-3/2$, $\mu'_1=0$.

Step 2. x_0 is not a vertex of S.

Step 1. It results $U(\theta)=[0,3/4]$, $\theta'=1/4$. Since $\theta'=1/4\in U(\theta)=[0,3/4]$ then setting $\theta^*=1/4$ we obtain $x^*=(3/4,3/4)+1/4(-1,1)=(1/2,1)$ which is the optimal solution of the problem.

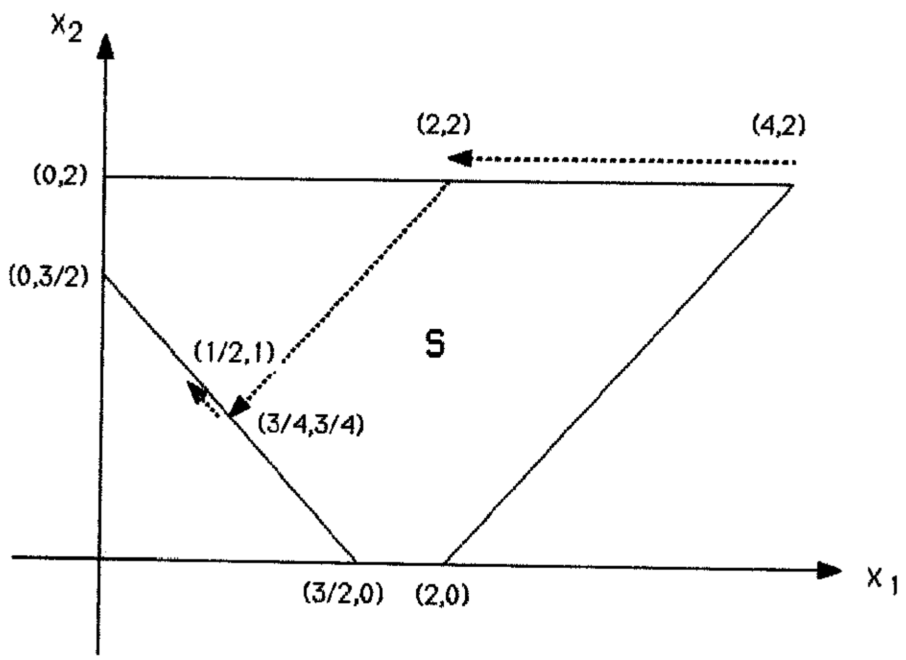

Fig. 1

In Fig.1 the feasible region S of the problem is depicted. The path followed by the algorithm to obtain the optimal solution is represented by the dotted lines.

References

[1] J.C.G. BOOT, "Quadratic Programming", North-Holland, Amsterdam, (1964).

[2] A. CAMBINI, "A sequential method for convex quadratic programming", Report A-16, Dept. of Op. Res., Univ. of Pisa, (1975).

[3] A. CAMBINI, L. MARTEIN, C. SODINI, "An algorithm for two particular non linear fractional programs", *Methods of Operations Research* **45**, 61-70 (1983).

[4] H.S. HOUTHAKKER,"The Capacity Method of Quadratic Programming", *Econometrica* **28**, 62-87 (1960).

[5] C. SODINI, "Minimizing the sum of a linear function and the square root of a convex quadratic form", *Methods of Operations Research* **53**, 171-182 (1985).

[6] C. VAN DE PANNE, "Methods for linear and quadratic programming", North-Holland, Amsterdam, (1975).

PUBBLICAZIONI
del
Dipartimento di Statistica e Matematica Applicata all Economia

Report n. 1 - Some Optimality Conditions in Vector Optimization.(A.Cambini- L.Martein), 1987

Report n. 2 - On Maximizing a Sum of Ratios. (A.Cambini-L.Martein-S.Schaible), 1987

Report n.3 - On the Charnes-Cooper Transformation in Linear Fractional Programming. (G.Gasparotto), 1987

Report n. 4 - Non-linear Separation Theorems, Duality and Optimality. (A.Cambini), 1987

Report n. 5 - Indicizzazione parziale: aspetti metodologici e riflessi economici. (G.Boletto), 1987

Report n. 6 - On Parametric Linear Fractional Programming. (A.Cambini-C.Sodini), 1987

Report n. 7 - Alcuni aspetti meno noti delle migrazioni in Italia. (A.Bonaguidi), 1987

Report n. 8 - On Solving a Linear Program with one Quadratic Constraint. (L.Martein-S.Schaible), 1987

Report n. 9 - Alcune osservazioni sull'equazione funzionale $\phi(x,y,z) = \phi(\phi(x,y,t),t,z)$. (E.Lari), 1988

Report n.10 - Une étude par ménage des migrations des personnes âgées: comparaison des résultats pour l'Italie et les Etats-Unis. (F.Bartiaux), 1988

Report n.11 - Metodi di scomposizione del tasso di inflazione (G.Boletto), 1988

Report n.12 - A New Algorithm for the Strictly Convex Quadratic Programming Problem. (C.Sodini), 1988