# A dual method for solving general convex quadratic programs 

Belkacem BRAHMI ${ }^{\mathrm{a}, \mathrm{b}}$ and Mohand Ouamer BIBI ${ }^{\mathrm{a}, \mathrm{c}}$<br>${ }^{\text {a }}$ LAMOS Laboratory, University of Bejaia, 06000 Bejaia, Algeria;<br>${ }^{\text {b }}$ Department of Mathematics, University of Tizi-Ouzou, 01500 Tizi-Ouzou, Algeria;<br>${ }^{\text {c }}$ Department of Operations Research, University of Bejaia, 06000 Bejaia, Algeria.


#### Abstract

In this paper, we present a new method for solving quadratic programming problems, not strictly convex. Constraints of the problem are linear equalities and inequalities, with bounded variables. The suggested method combines the active-set strategies and support methods. The algorithm of the method and numerical experiments are presented, while comparing our approach with the active set method on randomly generated problems.


Keywords-Convex quadratic programming, dual support methods, active set methods.

## I. Introduction

Theory of quadratic programming (QP) handles a special class of problems of nonlinear mathematical programming. In practice, a lot of problems are represented naturally by quadratic models, as in the case of the finance, management of the production, statistics and the optimal control, etc. These models are also used in nonlinear optimization as subproblems in the sequential quadratic programming methods [9].

In the literature, one distinguishes three approaches for solving quadratic optimization problems, such as active-set strategies [7], [6], [9], developed in the beginning of the 70s years. The second approach is constituted from interior points methods, known for their good theoretical complexity that is polynomial and its efficiency for solving large-scale problems [8]. The third is formed by support methods [1], [2], [5] and they are intermediate between the two first approaches. The majority of these methods are primal or primal-dual, but few are of the dual type. In [12], the dual simplexe method has been extended for solving strictly convex problems. The dual active-set method for the strictly convex case has been presented in [11] and Boland [10] generalized it for the general convex case, i.e., the matrix associated to the quadratic form being positive semidefinite. The method presented here is constituted of two phases: the first consists in finding an initial active set by solving a linear program, while the second is iterative and permits to change the active set while improving the objective function until the optimum.

In this work, we generalize the dual support method [1], [5] for the general convex case and with linear general inequalities constraints. The new algorithm combines the active set strategy [6], [7] and support methods [2]. The algorithm of this dual method is iterative and is constituted of two phases: the first consists in constructing an initial coordinator support of the
problem. The second phase permits to get the optimal solution for a bounded problem. In the numerical experiments we compare our method with active set method, implemented in Matlab 7 as Quadprog function, on randomly generated test problems.

## II. Statement of the problem and definitions

We consider the following quadratic model:

$$
\left\{\begin{array}{l}
\min F(x)=\frac{1}{2} x^{\prime} D x+c^{\prime} x  \tag{1}\\
A\left(I_{1}, J\right) x=b\left(I_{1}\right) \\
A\left(I_{2}, J\right) x \leq b\left(I_{2}\right) \\
d^{-} \leq x \leq d^{+}
\end{array}\right.
$$

where indices of constraints and decision variables are respectively noted by: $I=I_{1} \cup I_{2}, I_{1}=\left\{1,2, \cdots, m_{1}\right\}$, $I_{2}=\left\{m_{1}+1, m_{1}+2, \cdots, m\right\}$ and $J=\{1,2, \cdots, n\}$. So, $c=\left(c_{j}, j \in J\right), d^{-}=\left(d_{j}^{-}, j \in J\right), d^{+}=\left(d_{j}^{+}, j \in J\right)$ et $x=\left(x_{j}, j \in J\right)$ are $n$-vectors; $b=\left(b_{i}, i \in I\right)$ is an $m$-vector; $A=A(I, J)$ is an $(m \times n)$ matrix, such that $m_{1} \leq \operatorname{rang}(A)<n$. The square matrix $D=D(J, J)$ is supposed symmetric and positive semidefinite. The symbol ( ${ }^{\prime}$ ) is the transposition operation, and $A_{i}^{\prime}, i \in I$, represent the $i^{\text {th }}$ line of $A$.

## Definition 1:

- Any vector $x$ satisfying all the constraints of the problem (1) is called a feasible solution.
- A feasible solution $x^{*}$ is called optimal if

$$
F\left(x^{*}\right)=\frac{1}{2} x^{* \prime} D x^{*}+c^{\prime} x^{*}=\min _{x} F(x)
$$

where $x$ is taking among all feasible solutions for the problem.

- A general constraint $i \in I$ is said to be active in a point $x$ if it satisfies the identity $A_{i}^{\prime} x=b_{i}$. The active set in $x$ is noted $I_{0}=I_{1} \cup\left\{i \in I_{2}: A_{i}^{\prime} x=b_{i}\right\}$.
- The pair $S_{B}=\left\{I_{a}, J_{B}\right\}$, where $I_{1} \subset I_{a} \subset I_{0}$ and $J_{B} \subset J$ such that $\left|I_{a}\right|=\left|J_{B}\right|$ is called a constraint support of the problem (1) if the matrix $A_{B}=A\left(I_{a}, J_{B}\right)$ is nonsingular. For the initial constraint support, we can choose $S_{B}=\left\{I_{1}, J_{B}\right\}$, with $\left|J_{B}\right|=m_{1}$.
- The couple $\left\{x, S_{B}\right\}$ formed by the feasible solution $x$ and the support $S_{B}$ is called the support feasible solution. It is said to be nondegenerate if

$$
\begin{array}{ll}
d_{j}^{-}<x_{j}<d_{j}^{+}, & \forall j \in J_{B} \\
A_{i}^{\prime} x<b_{i}, & \forall i \in I_{n a}=I \backslash I_{a}
\end{array}
$$

Set $J_{N}=J \backslash J_{B}$. Let's define the potential vector $u$ and the reduced costs vector $E$ :
$u^{\prime}=u^{\prime}(I)=\left(u_{a}^{\prime}, u_{n a}^{\prime}\right)$, such that $u_{n a}=u\left(I_{n a}\right)=0, u_{a}^{\prime}=$ $u^{\prime}\left(I_{a}\right)=g_{B}^{\prime} A_{B}^{-1}, E^{\prime}=E^{\prime}(J)=g^{\prime}-u^{\prime} A=\left(E_{B}^{\prime}, E_{N}^{\prime}\right)$, such that $E_{N}^{\prime}=E^{\prime}\left(J_{N}\right)=g_{N}^{\prime}(x)-u^{\prime} A_{N}, E_{B}^{\prime}=0$, where $g=$ $g(J)=g(x)=D x+c$ represents the gradient of the function $F$ in a point $x$, with $g^{\prime}=\left(g_{B}^{\prime}, g_{N}^{\prime}\right)$ and $A_{N}=A\left(I_{a}, J_{N}\right)$.
The optimality criterion for the problem (1) is given by the following theorem.

Theorem 1: [3]. Let $\left\{x, S_{B}\right\}$ be a feasible support solution for the problem (1). Then the following relations:

$$
\left\{\begin{array}{l}
E_{j} \geq 0, \quad \text { for } \quad x_{j}=d_{j}^{-}, \\
E_{j} \leq 0, \\
\text { for } \quad x_{j}=d_{j}^{+}, \\
E_{j}=0, \\
\text { for } \quad d_{j}^{-}<x_{j}<d_{j}^{+}, j \in J_{N}, \\
u_{i} \geq 0, \\
\text { for } \quad A_{i}^{\prime} x=b_{i}, i \in I_{2} \cap I_{a} \\
u_{i}=0, \\
\text { for } \\
A_{i}^{\prime} x<b_{i}, i \in I_{n a},
\end{array}\right.
$$

are sufficient for the optimality of the point $x$. The same relations are also necessary if $\left\{x, S_{B}\right\}$ is not degenerate.
For the case of bounded quadratic problems, the optimal solution can be an extreme point, a point on the face or in the interior of the convex polyhedron. Thus, it is important in the nonlinear optimization to define a new concept allowing to have some informations on the curvature of the objective function.

## Definition 2:

- The indices set $J_{S} \subset J_{N}=J \backslash J_{B}$ is called an objective function support of the problem (1) if the submatrix $M_{S}=M\left(J_{S}, J_{S}\right)$ of $M$ is nonsingular, where

$$
M=M\left(J_{N}, J_{N}\right)=Z^{\prime} D Z \text { and } Z=\binom{-A_{B}^{-1} A_{N}}{I_{N}}
$$

Here $I_{N}$ is the identity matrix of $\left(n-m_{0}\right)$-dimension and $m_{0}=\left|I_{a}\right|$. Set $J_{P}=J_{B} \cup J_{S}$ and $J_{N N}=J \backslash J_{P}$.

- The indices set $S_{P}=\left\{S_{B}, J_{S}\right\}=\left\{I_{a}, J_{P}\right\}$, formed with the constraints support $S_{B}$ and the functional support $J_{S}$ is called a support of the problem (1).


## III. Dual problem and optimality criterion

The dual problem associated to the primal (1) is the following concave quadratic program:

$$
\left\{\begin{array}{l}
\max \varphi(\lambda)=-\frac{1}{2} \kappa^{\prime} D \kappa+b^{\prime} y+v^{\prime} d^{-}-w^{\prime} d^{+}  \tag{2}\\
D \kappa+c-A^{\prime} y-v+w=0 \\
y\left(I_{2}\right) \geq 0, \quad v \geq 0, \quad w \geq 0
\end{array}\right.
$$

where $\lambda=(\kappa, y, v, w)$.

## Definition 3:

- The quadruplet $\lambda=(\kappa, y, v, w) \in \mathbb{R}^{n \times m \times n \times n}$ verifying all constraints of the problem (2) is called a dual feasible solution. The $n$-vector $\kappa$ verifying $A\left(I_{a}, J\right) \kappa=b\left(I_{a}\right)$ is called a pseudosolution vector. Also, the $n$-vector $\delta=$ $D \kappa+c-A^{\prime} y$ is called a reduced costs vector, associated to the dual feasible solution $\lambda$.
- A dual feasible solution $\lambda^{*}=\left(\kappa^{*}, y^{*}, v^{*}, w^{*}\right)$ is said to be optimal if

$$
\varphi\left(\lambda^{*}\right)=-\frac{1}{2} \kappa^{* \prime} D \kappa^{*}+b^{\prime} y^{*}+v^{* \prime} d^{-}-w^{* \prime} d^{+}=\max _{\lambda} \varphi(\lambda)
$$

where $\lambda=(\kappa, y, v, w)$ is taking among all dual feasible solutions of the problem (2). The corresponding $n$-vector $\delta^{*}=D \kappa^{*}+c-A^{\prime} y^{*}$ is then an optimal reduced costs one.

- Considering a support $S_{P}=\left(I_{a}, J_{P}\right)$ for the problem (1) and let $\lambda=(\kappa, y, v, w)$ a dual feasible solution accompanying $S_{P}$, constructed as follows:
$d_{j}^{-} \leq \kappa_{j} \leq d_{j}^{+}, j \in J_{N N}=J \backslash J_{P}$,
$\kappa_{S}=-M_{S}^{-1} Z^{\prime}\left(J_{S}, J_{P}\right)\left[D\left(J_{P}, J_{B}\right) A_{B}^{-1} b+c\left(J_{P}\right)\right]-$ $M_{S}^{-1} M\left(J_{S}, J_{N N}\right) \kappa_{N N}, \kappa_{B}=A_{B}^{-1}\left(b_{a}-A_{N} \kappa_{N}\right) ;$ $y_{a}=\left[A_{B}^{\prime}\right]^{-1}\left(D\left(J_{B}, J\right) \kappa+c_{B}\right), \quad y_{n a}=0 ;$
$\delta=D \kappa+c-A^{\prime} y$, with $\delta_{P}=\delta\left(J_{P}\right)=0$ by construction of $y_{a}$ and $\kappa_{S} ; v_{j}=\delta_{j}, w_{j}=0$, if $\delta_{j} \geq 0 ; v_{j}=0, w_{j}=$ $-\delta_{j}$, if $\delta_{j}<0$.
The support $S_{P}$ is said to be coordinator if the following relations are verified:

$$
\left\{\begin{array}{lll}
\kappa_{j}=d_{j}^{-} & \text {if } & \delta_{j} \geq 0,  \tag{3}\\
\kappa_{j}=d_{j}^{+} & \text {if } & \delta_{j} \leq 0, j \in J_{N N}=J \backslash J_{P} \\
A_{i}^{\prime} \kappa=b_{i} & \text { if } & y_{i} \geq 0, i \in I_{2} \cap I_{a}
\end{array}\right.
$$

In this case, the couple $\left\{\lambda, S_{P}\right\}$ is called a coordinated dual feasible solution, and $\left\{\delta, S_{P}\right\}$ is also called a coordinated reduced costs vector. It is said to be nondegenerate if

$$
\delta_{j} \neq 0, \forall j \in J_{N N}, \quad y_{i}>0, i \in I_{2} \cap I_{a}
$$

Since $\delta\left(J_{P}\right)=0$ and $y\left(I_{n a}\right)=0$, by construction, then the optimality criterion of the dual problem (2) is expressed as follows:

Theorem 2: The relations

$$
\begin{array}{ll}
d_{j}^{-} \leq \kappa_{j} \leq d_{j}^{+}, & \forall j \in J_{P}=J_{B} \cup J_{S},  \tag{4}\\
A_{i}^{\prime} \kappa \leq b_{i}, & \forall i \in I_{n a}=I \backslash I_{a},
\end{array}
$$

are sufficient for the optimality of the coordinated reduced costs vector $\left\{\delta, S_{P}\right\}$. They are also necessary in the case when $\left\{\delta, S_{P}\right\}$ is not degenerate. The corresponding pseudosolution $\kappa$ of the optimal reduced costs vector is then an optimal solution to the primal problem (1).

## IV. Construction of the initial coordinator SUPPORT

The phase 1 of the proposed method allows to construct an initial coordinator support that is necessary for the formulation of the optimality criterion, and to start the phase 2 of the dual support method. So, this procedure yields an initial coordinated reduced costs vector $\left\{\delta, S_{P}\right\}$. As an initial support, we can choose $S_{P}=\left(S_{B}, J_{S}=\emptyset\right)$, where $S_{B}=\left(I_{1}, J_{B}\right)$ such that $\left|J_{B}\right|=\left|I_{1}\right|=m_{1}$, is the constraints support of the problem. The steps of the phase 1 are as follows:
Step 1. Construction of the reduced costs vector $\delta$ associated to the initial support $S_{P}$ :

- Construct the initial pseudosolution vector $\kappa=$ $\left(\kappa_{B}, \kappa_{N}\right)^{\prime}:$

$$
\left\{\begin{array}{l}
d_{j}^{-} \leq \kappa_{j} \leq d_{j}^{+}, j \in J_{N} \\
\kappa_{B}=A_{B}^{-1}\left(b_{a}-A_{N} \kappa_{N}\right)
\end{array}\right.
$$

- Calculate the vector $\delta=\left(\delta_{B}, \delta_{N}\right)^{\prime}$ :
$\delta_{B}=0, \quad \delta_{N}=D\left(J_{N}, J\right) \kappa+c_{N}-A_{N}^{\prime} y$, with $y=\left[A_{B}^{\prime}\right]^{-1}\left(D\left(J_{B}, J\right) \kappa+c_{B}\right)$.

ISSN: 2517-9934
Vol:3, No:10, 2009

Step 2. Apply the coordination tests:

- If $\kappa_{j}=d_{j}^{-}$for $\delta_{j} \geq 0$ and $\kappa_{j}=d_{j}^{+}$for $\delta_{j}<0, \forall j \in$ $J_{N N}$, then $\left\{\delta, S_{P}\right\}$ is a supporting coordinated reduced costs vector and consequently the procedure is stopped.
- Otherwise, construct a new pseudosolution vector $\bar{\kappa}$ and its corresponding reduced costs vector $\bar{\delta}$ :

$$
\bar{\kappa}=\kappa+\sigma l, \quad \bar{\delta}=\delta+\sigma t, \text { with } t=D l-A^{\prime} q, \bar{y}=y+\sigma q
$$

where the $n$-vectors $l$ and $t$, representing respectively the improvement directions of the pseudosolution and its reduced costs vector, are determined as follows:

$$
\begin{aligned}
& \left\{\begin{array}{l}
l_{j}=d_{j}^{-}-\kappa_{j}, \quad \text { if } \delta_{j} \geq 0 \\
l_{j}=d_{j}^{+}-\kappa_{j}, \quad \text { if } \delta_{j}<0, j \in J_{N N} \\
l_{S}=-M_{S}^{-1} M\left(J_{S}, J_{N N}\right) l_{N N} \\
l_{B}=-A_{B}^{-1} A_{N} l_{N}
\end{array}\right. \\
& \left\{\begin{array}{l}
t_{B}=0, t_{S}=0, \\
t_{N N}=D\left(J_{N N}, J\right) l-A_{N N}^{\prime} q \\
q=\left[A_{B}^{-1}\right]^{\prime} D\left(J_{B}, J\right) l
\end{array}\right.
\end{aligned}
$$

Calculate the optimal stepsize $\sigma=\min \left\{1, \sigma_{j_{0}}\right\}$, where $\sigma_{j_{0}}$ is determined such that $\delta_{N N}$ and $\bar{\delta}_{N N}$ don't change their signs:

$$
\sigma_{j_{0}}=\min _{j \in J_{N N}} \sigma_{j}, \text { with } \sigma_{j}=\left\{\begin{array}{l}
-\delta_{j} / t_{j}, \text { if } \delta_{j} t_{j}<0 \\
0, \text { if } \delta_{j}=0 \text { and } t_{j}<0 \\
+\infty, \text { otherwise }, j \in J_{N N}
\end{array}\right.
$$

- If $\sigma=1$ then $S_{P}=\left\{S_{B}, J_{S}\right\}$ is an initial coordinated support with $\bar{\kappa}=\kappa+l$ its associated pseudosolution vector.
- Otherwise, do the following appropriate index change:

$$
\bar{J}_{S}=J_{S} \cup j_{0}, \quad \bar{J}_{B}=J_{B}, \quad \bar{J}_{P}=\bar{J}_{B} \cup \bar{J}_{S}
$$

Go to step (2).
Remark 1: The proposed procedure is finite, since it allows to construct in the worst case a coordinator support in $\left(n-m_{1}\right)$ iterations, when the method is started with $J_{S}=\emptyset$.

## V. Algorithm of the dual support method

This new dual method combines the support method [1] and the active set strategy [6], [7], [9]. Let $\left\{\delta, S_{P}\right\}$ be an initial coordinated reduced costs vector, obtained by the phase 1 of the method and $\kappa$ its corresponding pseudosolution. The principle of the method consists to transform the pair $\left\{\delta, S_{P}\right\} \longrightarrow\left\{\bar{\delta}, \bar{S}_{P}\right\}$ in two steps; the first is the construction of the new reduced costs vector $\bar{\delta}$ such that $\varphi(\bar{\lambda}) \geq \varphi(\lambda)$, where $\lambda$ and $\bar{\lambda}$ represent respectively the dual feasible solutions associated to the reduced costs vectors $\delta$ and $\bar{\delta}$. The second stage consists to transform $S_{P} \longrightarrow \bar{S}_{P}$ in such a way that $\bar{S}_{P}$ must be a coordinator support.

The algorithm of the method presents the following stages: Step 1. Verification of the optimality conditions of the couple $\left\{\delta, S_{P}\right\}$ :

- If optimality relations (4) are verified, then $\kappa$ is an optimal feasible solution of the primal problem (1). Stop the algorithm.
- Otherwise for all non optimal support indices $J_{P}^{\mathrm{no}} \subset J_{P}$, and those of the violated general constraints at the courant point $\kappa$, noted by $I_{n a}^{\mathrm{no}} \subset I_{n a}=I \backslash I_{a}$, calculate the following numbers:

$$
\begin{aligned}
& \alpha_{j}= \begin{cases}\kappa_{j}-d_{j}^{-}, & \text {if } \kappa_{j}<d_{j}^{-} \\
\kappa_{j}-d_{j}^{+}, & \text {if } \kappa_{j}>d_{j}^{+}, j \in J_{P}^{n o}\end{cases} \\
& r_{i}=A_{i}^{\prime} \kappa-b_{i}>0, \quad i \in I_{n a}^{n o}
\end{aligned}
$$

Choose an index $j_{1} \in J_{P}^{n o}$ and another $i_{1} \in I_{n a}^{n o}$ verifying:

$$
\left|\alpha_{j_{1}}\right|=\max \left\{\left|\alpha_{j}\right|, j \in J_{P}^{n o}\right\}, \quad r_{i_{1}}=\max \left\{r_{i}, i \in I_{n a}^{n o}\right\}
$$

Step 2. Construction of the new pseudosolution and its associated reduced costs vector as follows:

$$
\bar{\kappa}=\kappa+\sigma l, \bar{y}=y+\sigma q, \bar{\delta}=\delta+\sigma t, \sigma \geq 0
$$

- Determine the improvement directions $l=\left(l_{B}, l_{S}, l_{N N}\right)^{\prime}$ of the pseudosolution and those of its reduced costs vector $t=\left(t\left(J_{P}\right), t\left(J_{N N}\right)\right)^{\prime}$ in the following way:
- If $\left|\alpha_{j_{1}}\right| \geq r_{i_{1}}$ then put:

$$
\begin{aligned}
& \left\{\begin{array}{l}
l_{N N}=0 \\
l_{S}=t_{j_{1}} M_{S}^{-1} Z^{\prime}\left(J_{S}, j_{1}\right) \\
l_{B}=-A_{B}^{-1} A\left(I_{a}, J_{S}\right) l\left(J_{S}\right)
\end{array}\right. \\
& \left\{\begin{array}{l}
t_{j_{1}}=-\operatorname{sign} \alpha_{j_{1}}, t\left(J_{P} \backslash j_{1}\right)=0, q\left(I_{n a}\right)=0 \\
t_{N N}=D\left(J_{N N}, J_{P}\right) l_{P}-A^{\prime}\left(J_{N N}, I_{a}\right) q_{a} \\
q_{a}=q\left(I_{a}\right)=\left[A_{B}^{-1}\right]^{\prime}\left[D\left(J_{B}, J_{P}\right) l_{P}-t_{B}\right]
\end{array}\right.
\end{aligned}
$$

- Else $\left(\left|\alpha_{j_{1}}\right|<r_{i_{1}}\right)$, set:

$$
\begin{aligned}
& \left\{\begin{array}{l}
l_{N N}=0 \\
l_{S}=-M_{S}^{-1} Z\left(J_{S}, J\right)^{\prime} A^{\prime}\left(J, i_{1}\right) \\
l_{B}=-A_{B}^{-1} A\left(I_{a}, J_{S}\right) l\left(J_{S}\right)
\end{array}\right. \\
& \left\{\begin{array}{l}
t\left(J_{P}\right)=0, q_{i_{1}}=1, q\left(I_{n a} \backslash i_{1}\right)=0 \\
t_{N N}=D\left(J_{N N}, J_{P}\right) l_{P}-A^{\prime}\left(J_{N N}, I_{a} \cup i_{1}\right) q\left(I_{a} \cup i_{1}\right), \\
q\left(I_{a}\right)=\left[A_{B}^{-1}\right]^{\prime}\left[D\left(J_{B}, J_{P}\right) l_{P}-A^{\prime}\left(J_{B}, i_{1}\right)\right]
\end{array}\right.
\end{aligned}
$$

- Calculate the optimal stepsize $\sigma=$ $\min \left\{\sigma_{j_{0}}, \sigma_{j_{1}}, \tau_{i_{0}}, \tau_{i_{1}}\right\}$, where

$$
\begin{aligned}
& \sigma_{j_{0}}=\min _{j \in J_{N N}} \sigma_{j}, \text { with } \sigma_{j}=\left\{\begin{array}{l}
-\delta_{j} / t_{j}, \text { if } \delta_{j} t_{j}<0 \\
0, \text { if } \delta_{j}=0 \text { and } t_{j}<0 \\
\infty, \text { otherwise }, j \in J_{N N}
\end{array}\right. \\
& \text { and } \quad \sigma_{j_{1}}=\left\{\begin{array}{l}
\left(d_{j_{1}}^{-}-\kappa_{j_{1}}\right) / l_{j_{1}}, \text { if } l_{j_{1}}>0 \\
\left(d_{j_{1}}^{+}-\kappa_{j_{1}}\right) / l_{j_{1}}, \text { if } l_{j_{1}}<0 \\
\infty, \text { if } l_{j_{1}}=0
\end{array}\right.
\end{aligned}
$$

The step $\sigma_{j_{0}}$ represent the maximal value for which $\delta_{N N}$ and $\bar{\delta}_{N N}$ keep the same sign, and $\sigma_{j_{1}}$ is the value for which the index $j_{1}$ becomes optimal, i.e., the component $\bar{\kappa}_{j_{1}}=d_{j_{1}}^{-} \vee d_{j_{1}}^{+}$. The number $\tau_{i_{0}}$ is the value of stepsize for which active components of multipliers vectors $y_{a}=$ $y\left(I_{a}\right)$ and $\bar{y}_{a}=\bar{y}\left(I_{a}\right)$ don't change a sign:

$$
\tau_{i_{0}}=\min _{i \in I_{2} \cap I_{a}} \tau_{i}, \text { with } \tau_{i}=\left\{\begin{array}{l}
-y_{i} / q_{i}, \text { if } y_{i} q_{i}<0 \\
0, \text { if } y_{i}=0, q_{i}<0 \\
\infty, \text { otherwise, } i \in I_{2} \cap I_{a}
\end{array}\right.
$$

Finally, $\tau_{i_{1}}$ is the optimal value for which the component $y_{i_{1}}$ becomes critical:

$$
\tau_{i_{1}}= \begin{cases}-r_{i_{1}} / A_{i_{1}}^{\prime} l, & \text { if } A_{i_{1}}^{\prime} l<0 \\ \infty, & \text { otherwise }\end{cases}
$$

- If $\sigma=\infty$, then the dual problem (2) is not bounded below and consequently its primal (1) is infeasible. Stop the resolution process.
- Otherwise $\sigma$ is finite, and calculate $\bar{\delta}=\delta+\sigma t, \bar{\kappa}=$ $\kappa+\sigma l, \bar{y}=y+\sigma q$.
Step 3. Construction of a new coordinator support $\bar{S}_{P}=$ $\left\{\bar{S}_{B}, \bar{J}_{S}\right\}$, where $=\bar{S}_{B}=\left(\bar{I}_{a}, \bar{J}_{B}\right)$ :
This change is operated according to the following cases:
Step 3.1. If $\sigma=\sigma_{j_{1}}, j_{1} \in J_{P}=J_{B} \cup J_{S}$, the component $\kappa_{j_{1}}$ reaches the left or the right bound:
- If $j_{1} \in J_{S}$, then put

$$
\bar{S}_{B}=S_{B}, \quad \bar{J}_{S}=J_{S} \backslash j_{1} .
$$

- Otherwise, change the constraints support while choosing an index $j_{*} \in J_{S}$ such that $z_{j_{1} j_{*}}=Z\left(j_{1}, j_{*}\right) \neq 0$ and put

$$
\bar{I}_{a}=I_{a}, \quad \bar{J}_{B}=\left(J_{B} \backslash j_{1}\right) \cup j_{*}, \quad \bar{J}_{S}=J_{S} \backslash j_{*} .
$$

Go to step (1).
Step 3.2. If $\sigma=\sigma_{j_{0}}$, then component $\kappa_{j_{1}}$ is not critical and consequently the index $j_{1}$ cannot be transferred to $\bar{J}_{N N}$.

- If $\eta_{0}=M\left(j_{0}, j_{0}\right)-M\left(j_{0}, J_{S}\right) M_{S}^{-1} M\left(J_{S}, j_{0}\right) \neq 0$, then the index $j_{0}$ will be transferred to $\bar{J}_{S}$ such that the functional support matrix $M_{S}=M\left(\bar{J}_{S}, \bar{J}_{S}\right)$ remains nonsingular:

$$
\bar{S}_{B}=S_{B}, \quad \bar{J}_{S}=J_{S} \cup j_{0} .
$$

Go to step (2).

- If $\eta_{0}=0$, then the index $j_{0}$ cannot be added to the functional support:
- If $j_{1} \in J_{S}$ then do the appropriate following support change.

$$
\bar{S}_{B}=S_{B}, \quad \bar{J}_{S}=\left(J_{S} \backslash j_{1}\right) \cup j_{0}
$$

- Otherwise, change the constraints support while entering the index $j_{0}$ and take out the index $j_{1}$ :

$$
\bar{I}_{a}=I_{a} \quad \bar{J}_{B}=\left(J_{B} \backslash j_{1}\right) \cup j_{0}, \quad \bar{J}_{S}=J_{S}
$$

- Correct the pseudosolution vector $\tilde{\kappa}=\bar{\kappa}+\tilde{l}$, where the improvement direction $\tilde{l}$ is calculated in such a way that the component $\tilde{\kappa}_{j_{1}}, j_{1} \in \bar{J}_{N N}=J \backslash\left(\bar{J}_{B} \cup \bar{J}_{S}\right)$ becomes coordinated with the support $\bar{S}_{P}$. Consequently, it is determined as follows:

$$
\begin{aligned}
& \tilde{l}_{j_{1}}=d_{j_{1}}^{-}-\bar{\kappa}_{j_{1}}, \text { if } t_{j_{1}}=1 ; \tilde{l}_{j_{1}}=d_{j_{1}}^{+}-\bar{\kappa}_{j_{1}}, \text { if } t_{j_{1}}=-1 ; \\
& \tilde{l}_{j}=0, j \neq j_{1}, j \in \bar{J}_{N N} ; \tilde{l}_{S}=-M_{S}^{-1} M\left(\bar{J}_{S}, \bar{J}_{N N}\right) \tilde{l}_{N N} \\
& \tilde{l}_{B}=Z\left(\bar{J}_{B}, \bar{J}_{N}\right) \tilde{l}_{N},
\end{aligned}
$$

where $M$ and $Z$ are the updating matrices defined by the relation (2). Go to step (1).
Step 3.3. If $\sigma=\tau_{i_{1}}, i_{1} \in I_{n a}$, then the $i_{1}$-th general constraint becomes active. Find an index $j_{*} \in J_{S}$ verifying $h_{j_{*}} \neq 0$, where $h^{\prime}\left(J_{S}\right)=A\left(i_{1}, J_{S}\right)-A\left(i_{1}, J_{B}\right) A_{B}^{-1} A\left(I_{a}, J_{S}\right)$ and put:

$$
\bar{I}_{a}=I_{a} \cup i_{1}, \quad \bar{J}_{B}=J_{B} \cup j_{*}, \quad \bar{J}_{S}=J_{S} \backslash j_{*}
$$

Go to step (1).

Step 3.4. If $\sigma=\tau_{i_{0}}, i_{0} \in I_{2} \cap I_{a}$, then the $i_{0}$-th constraint becomes nonactive and consequently, it will be deleted in the active set by this change:
a) If $\left|\alpha_{j_{1}}\right| \leq r_{i_{1}}$, then the $j_{1}$-th simple constraint will be deleted in $\bar{J}_{P}$ and added to $\bar{J}_{N N}$. Set $\bar{I}_{a}=I_{a} \backslash i_{0}$ and $\bar{J}_{S}=J_{S}$.

- If $j_{1} \in J_{B}$ then $\bar{J}_{B}=J_{B} \backslash j_{1}$.
- If $j_{1} \notin J_{B}$ then search an index $j_{*} \in J_{B}$ such that $A_{i_{0} j_{*}}^{-1} \neq 0$ and put $\bar{J}_{B}=J_{B} \backslash j_{*}$. Go to step (2).
b) Otherwise the $i_{1}$-th general constraint becomes active and the $i_{0}$-th nonactive:

$$
\bar{I}_{a}=\left(I_{a} \cup i_{1}\right) \backslash i_{0}, \quad \bar{J}_{B}=J_{B}, \quad \bar{J}_{S}=J_{S} .
$$

Go to step (1).
Remark 2: For the case where $I_{2}=\emptyset$, the algorithm presented before is the same suggested in [5].

## VI. Numerical experiments

In this section, we present numerical results of comparison between dual support method (DSM) and active set method (ASM), implemented in Matlab 7.0 R14, under Windows Vista, at a CPU 3Ghz and 512 MO of RAM. The comparison is based on quadratic programming problems randomly generated. The characteristic of these problems is that the optimal solution $x^{*}$, and the value of the objective function $f^{*}$ at the optimum are known previously.

Input parameters of the procedure that generate randomly test quadratic problems are $n, m, n_{0}, m_{0}$, where respectively they represent the number of variables, the number of general constraints, the number of active variables at the optimum ( $x_{j}^{*}=d_{j}^{-} \vee d_{j}^{+}, j=1 \cdots n_{0}$ ) and the number of general active constraints $\left(A_{i}^{\prime} x^{*}=b_{i}, i=1 \cdots m_{0}\right)$. Its stages are summarized as follows:

1) Generate randomly (or fixed) the optimal solution $x^{*}$.
2) Calculate the reduced costs vector $\delta^{*}$ by the following rules:

$$
\delta_{j}^{*} \neq 0, \forall j=1 \cdots n_{0} ; \quad \delta_{j}^{*}=0, \forall j=n_{0}+1 \cdots n
$$

3) Generate the multiplier vector $y^{*}$ :

$$
\begin{aligned}
& y_{i}^{*} \neq 0, i=\overline{1, m_{1}} ; \quad y_{i}^{*}>0, i=\overline{m_{1}+1, m_{0}} \\
& y_{i}^{*}=0, i=\overline{m_{0}+1: m}
\end{aligned}
$$

4) Construct the $n$-vectors $d^{-}$and $d^{+}$, such that $d_{j}^{-}<d_{j}^{+}$, for all $j=1, \cdots, n$ :

$$
\begin{array}{ll}
d_{j}^{-}=x_{j}^{*}, & d_{j}^{+}>x_{j}^{*}, \\
d_{j}^{-}<x_{j}^{*}, & \delta_{j}^{+}=x_{j}^{*}, \\
\delta_{j}^{*}<0, \\
d_{j}^{-}<x_{j}^{*}, & d_{j}^{+}>x_{j}^{*},
\end{array} \delta_{j}^{*}=0, j \in J .
$$

5) Generate the constraints matrix $A$ and the symmetric semidefinite matrix $D$ in the form $D=G^{\prime} G$ where $G$ is an $r \times n$ matrix, with $\operatorname{rank}(G) \leq r$.
6) Calculate the $m$-vector $b$ :

$$
b_{i}=A_{i}^{\prime} x^{*}, \forall i=1 \cdots m_{0} ; A_{i}^{\prime} x^{*}<b_{i}, \forall i=m_{0}+1 \cdots m
$$

7) For the $n$-vector $c$, set $c=-D x^{*}+A^{\prime} y^{*}+\delta^{*}$.

# International Journal of Engineering, Mathematical and Physical Sciences 

ISSN: 2517-9934
Vol:3, No:10, 2009

Random values of different vectors and matrices are uniformly distributed in the range $[-1,1]$. For computational experiments we compare our dual support method (DSM) and the primal active-set method (ASM) implemented in Matlab 7 as function quadprog. This comparison is done for the standard case $I_{2}=\emptyset$ and on two kinds of test problems: strictly convex and convex quadratic programs. Concerning the general case $I_{2} \neq \emptyset$, numeric results are under realization.

Table 1 reports numerical results for strictly convex problems, where $T$ and Iters represent respectively the average CPU time in seconds and the average number of iterations required for solving 10 QPs with the same entries. For DSM, we adapt two strategies: the first when we start the method with the empty functional support ( $J_{S}=\emptyset$ ) and the second with the full functional support ( $J_{S}=J_{N}$ ). For DSM, we note that the first strategy requires more iterations compared to the second strategy for all test problems that have been solved. When the number of active constraints at the optimum is small, DSM with the second strategy is required to DSM with empty support and also to ASM. But when the number of active constraints is large we conclude that Active-set method is relatively efficient to our approach for solving strictly convex quadratic programs. For general convex quadratic programs,

TABLE I
NUMERICAL RESULTS FOR STRICTLY CONVEX QUADRATIC TEST PROBLEMS

| $n$ | $m$ | $n_{0}$ | $\begin{aligned} & \text { DSM with } \\ & J_{S}=\emptyset \end{aligned}$ |  | DSM with$J_{S}=J_{N}$ |  | ASM |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | T(s) | Iters | T(s) | Iters | T(s) | Iters |
| 100 | 10 | 50 | 0.39 | 111.6 | 0.34 | 51.2 | 0.41 | 51 |
| 100 | 50 | 50 | 0.43 | 101.8 | 0.30 | 51.8 | 0.23 | 50 |
| 200 | 10 | 100 | 2.47 | 222.6 | 2.68 | 101.6 | 3.02 | 101 |
| 200 | 10 | 150 | 3.33 | 243.2 | 4.32 | 157.4 | 3.66 | 151 |
| 200 | 50 | 150 | 3.72 | 254.4 | 3.77 | 163.8 | 2.80 | 150 |
| 200 | 100 | 150 | 2.51 | 206.6 | 1.98 | 106.6 | 1.56 | 100 |
| 300 | 10 | 250 | 14.55 | 370.8 | 20.94 | 264.4 | 15 | 251 |
| 300 | 50 | 100 | 7.37 | 314.8 | 6.90 | 100.4 | 7.45 | 101 |
| 300 | 50 | 250 | 14.91 | 391.8 | 17.99 | 277.2 | 12.15 | 250 |
| 300 | 100 | 100 | 6.99 | 300.2 | 5.12 | 100.2 | 5.51 | 101 |
| 300 | 100 | 200 | 12.14 | 366.6 | 11.48 | 218 | 8.40 | 200 |
| 400 | 10 | 350 | 44.107 | 494.8 | 65.62 | 372.4 | 42.73 | 351 |
| 400 | 50 | 100 | 18.26 | 417 | 15.32 | 100.20 | 17.30 | 101 |
| 400 | 50 | 350 | 47.27 | 525.6 | 60.83 | 383.6 | 36.99 | 350 |
| 400 | 100 | 100 | 17.48 | 400 | 12.63 | 100 | 14.35 | 101 |
| 400 | 100 | 300 | 43.02 | 515.2 | 45.05 | 332 | 28.97 | 300 |
| 500 | 10 | 450 | 106.24 | 632.2 | 155.95 | 476.6 | 95.73 | 451 |
| 500 | 50 | 200 | 57.33 | 553.4 | 60.25 | 204.4 | 55.74 | 201 |
| 500 | 50 | 450 | 114.82 | 676.6 | 148.58 | 492.4 | 85.66 | 450 |
| 500 | 100 | 100 | 36.15 | 500 | 26.07 | 100 | 28.74 | 101 |
| 500 | 100 | 400 | 110.16 | 664.2 | 117.37 | 436.4 | 69.20 | 400 |

TABLE II
Numerical results for semidefinite quadratic test problems

|  |  |  |  | DSM with $J_{S}=\emptyset$ |  |  |  |
| :---: | :---: | :---: | :---: | :--- | :---: | :--- | :---: |
| $n$ | $m$ | $n_{0}$ | $\operatorname{rank}(\mathrm{D})$ | ASM |  |  |  |
|  | T(s) | Iters | $\mathrm{T}(\mathrm{s})$ | Iters |  |  |  |
| 100 | 10 | 50 | 80 | 0.375 | 107.2 | 0.656 | 51 |
| 200 | 10 | 150 | 150 | 4.734 | 281.4 | 5.469 | 151 |
| 200 | 40 | 150 | 150 | 4.687 | 279.7 | 3.391 | 151 |
| 300 | 10 | 250 | 250 | 25.609 | 429 | 36.734 | 251 |
| 300 | 30 | 250 | 250 | 18.338 | 434.3 | 19.323 | 251 |
| 400 | 20 | 200 | 350 | 44.953 | 529.5 | 81.797 | 201 |
| 500 | 10 | 450 | 450 | 135.734 | 675.1 | 272.141 | 451 |
| 500 | 30 | 300 | 450 | 91.063 | 646 | 140.766 | 301 |
| 500 | 50 | 450 | 450 | 126.078 | 669.4 | 88.406 | 450 |
| 700 | 10 | 600 | 650 | 432.843 | 939.2 | 755.282 | 601 |
| 700 | 50 | 500 | 650 | 427.390 | 911.5 | 322.922 | 501 |

quadratic problems, contrary to other methods which consider only the strictly convex case [12], [11], [13].

## References

[1] R. Gabasov, F. M. Kirillova and V. M. Raketsky. On methods for solving the general problem of convex quadratic programming. Soviet. Math Dokl. 23 (1981) 653-657.
[2] R. Gabasov, F. M. Kirillova, V. M. Raketsky and O.I. Kostyukova Constructive Methods of Optimization, Volume 4: Convex Problems. University Press, Minsk (1987).
[3] E. A. Kostina and O. I. Kostyukova. An algorithm for solving quadratic programming problems with linear equality and inequality constraints. Computational Mathematics and Mathematical Physics, 41(7) (2001) 960-973.
[4] E. A. Kostina and O. I. Kostyukova. A primal-dual active-set method for convex quadratic programming. Preprints of the University of Karlsruhe, Germany (2003).
[5] B. Brahmi and M. O. Bibi. Dual support method for solving convex quadratic programs. To appear in Optimization (2009).
[6] R. Fletcher. Practical Methods of Optimization. J. Wiley and Sons, Chichester, England, second edition (1987).
[7] P. E. Gill, W. Murray, and M. H. Wright. Practical Optimization. Academic Press Inc, London (1981)
[8] Y. Nesterov and A. Nemirovskii. Interior-Point Polynomial Algorithms in Convex Programming. SIAM Studies in Applied Mathematics: Vol 13, Philadelphia (1994).
[9] J. Nocedal and S. J. Wright. Numerical Optimization. Springer-Verlag, New York, second edition (2006).
[10] N. L. Boland. A dual-active-set algorithm for positive semi-definite quadratic programming. Mathematical Programming, 78(1)(1997)1-27
[11] D. Goldfarb and A. U. Idnani. A numerically stable dual method for solving strictly convex quadratic problems. Mathematical Programming, 27 (1983) 1-33.
[12] C. Van de Panne and A. Whinston. The simplex and dual method for quadratic programming. Operational Research Quarterly Journal, 15 (1964) 355-388.
[13] R. A. Bartlett and L. T. Biegler. QPSchur: A dual, active-set, Schurcomplement method for large-scale and structured convex quadratic programming. Optimization and Engineering, 7 (2006) 5-32.
the results are drawn up in table 2 . We constat that when $m \leq 30$ our method is very efficient compared to the ASM.

## VII. Conclusion

In this paper, we have developed a new method for solving general convex quadratic problems that combines active set strategy [7] and support methods [1], [2]. Its advantage is that it handles constraints such as initially presented, without any preliminary transformation. This permits to have a gain in memory space and CPU time. It also allows, following the example that of Boland [10], to deal with semidefinite

