RAIRO. RECHERCHE OPÉRATIONNELLE

THAÏ QUYNH PHONG
LE THI HOAI AN
PHAM DINH TAO

On globally solving linearly constrained indefinite quadratic minimization problems by decomposition branch and bound method

RAIRO. Recherche opérationnelle, tome 30, nº 1 (1996), p. 31-49
<http://www.numdam.org/item?id=RO_1996__30_1_31_0>

Recherche opérationnelle/Operations Research
(vol. 30, n° 1, 1996, pp. 31-49)

ON GLOBALLY SOLVING LINEARLY CONSTRAINED INDEFINITE QUADRATIC MINIMIZATION PROBLEMS BY DECOMPOSITION BRANCH AND BOUND METHOD (*)

by THAI QUYNH PHONG (1), LE THI HOAI AN (1) and PHAM DINH TAO (1)

Communicated by Michel MINOUX

Abstract. – *The global minimization of an indefinite quadratic function over a bounded polyhedral set using a decomposition branch and bound approach is considered. The objective function consists of an unseparated convex part and a separated concave part. The large-scale problems are characterized by having the number of convex variables much more than that of concave variables. The advantages of the method is that it uses the rectangular subdivision on the subspace of concave variables. Using a easily constructed convex underestimating function to the objective function, a lower bound is obtained by solving a convex quadratic programming problem. Three variants using exhaustive, adaptive and w-subdivision are discussed. Computational results are presented for problems with 10-20 concave variables and up to 200 convex variables.*

Keywords: Global minimization, indefinite quadratic programming, decomposition, branch and bound.

Résumé. – *La minimisation globale d'une forme quadratique indéfinie sur un polyèdre convexe borné par une méthode décomposition-séparation et évaluation est étudiée dans ce papier. La fonction objectif est la somme d'une forme quadratique convexe et d'une forme quadratique concave séparable en ses variables. Les problèmes de grande dimension sont caractérisés par le fait que le nombre des variables convexes est beaucoup plus grand que celui des variables concaves. L'avantage de la méthode réside dans l'utilisation de la subdivision rectangulaire uniquement dans le sous-espace des variables concaves. A l'aide d'une minorante très simple à calculer de la partie concave, une borne inférieure est obtenue en résolvant un programme quadratique convexe. Trois variantes utilisant la subdivision exhautive, la subdivision adaptive et la w-subdivision sont considérées. Les simulations numériques sont présentés pour les problèmes ayant 10–20 variables concaves et jusqu'à 200 variables convexes.*

Mots clés : Minimisation globale, programmation quadratique indéfinie, décomposition – séparation et évaluation.

1. INTRODUCTION

Quadratic programming is an important subject in mathematical programming since it has many applications in economics, planing,

(*) Received January 1994.

(1) LMI – INSA Rouen, B.P. 08, 76131, France.

Recherche opérationnelle/Operations Research, 0399-0559/96/01/$ 4.00

and engineering design. In addition, complicated problems of ninlinear programming are often simplified into quadratic programming problems. Indefinite quadratic programming problems are multiextremal global optimization problems in the sense that they have local optima which fail to be global.

This paper studies the problem of minimizing an indefinite quadratic function over a polytope. As is well-known, a quadratic function can be converted into the separable form by means of affine transformations. Specifically, we are concerned with the problem

$$(IQP) \qquad \min f(x, y) = p(x) + q(y), \quad \text{s.t.} \quad (x, y) \in \Omega$$

where $\Omega = \{(x, y) : Ax + By \leq a, x \geq 0, y \geq 0\}$ is a polytope and $x \in \mathbb{R}^n$, $y \in \mathbb{R}^s$, $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{m \times s}$, $a \in \mathbb{R}^m$. The objective function is a sum of an unseparated convex term

$$p(x) = \frac{1}{2} \langle Cx, x \rangle + \langle c, x \rangle \tag{1}$$

(C is a symmetric positive semi-definite ($n \times n$) matrix), and a separable concave term

$$q(y) = \sum_{i=1}^{s} q_i(y_i) \tag{2}$$

where $q_i(y_i) = d_i y_i - \frac{1}{2} \lambda_i y_i^2$ ($\lambda_i > 0$).

A special case of (IQP) is the constrained concave quadratic global minimization problem for which $C = 0$. This problem is known to be NP-hard, and hence it follows that problem (IQP) is NP-hard. Several algorithms have been proposed for this problem. Exploiting the specific structure of the problem, Rosen [16] developed a parametric decomposition algorithm. In [17] Rosen and Pardalos considered the quadratic function under the separable form over a rectangle that tightly encloses Ω and computed an approximate solution by solving a zero-one mixed integer programming problem associated to the quadratic problem. Another branch and bound approach was proposed by Kalantari and Rosen [9] who used successively refined parallelepipeds defined by conjugate directions of the quadratic function and convex envelopes. More recently a parallel algorithm for linearly constrained large-scale concave quadratic minimization problem has been proposed in Phillips and Rosen [13]. The reported results demonstrate the practical interest of the method. An important property of problem (IQP) in the special case when $C = 0$ (*i.e.* when the objective function is concave) is that the global minimum is always attained at at least a vertex of the convex polytope Ω. This property is no longer true when $C \neq 0$. Hence, problems (IQP) with $C \neq 0$ are likely even more difficult to solve computationally than concave programs.

In convex approaches to nonconvex optimization, Pham D. Tao has extensively studied subgradient methods for solving convex maximization problems [18, 19, 20] and d.c. (difference of two convex functions) optimization problems [1, 21, 23, 22]. Theses algorithms of d.c. optimization (DCA) can not guarantee global optimality of computed solutions. Nevertheless they have been sucessfully applied to many large-scale concrete d.c. optimization problems for which DCA have proved to be more robust and efficient than related standard methods [2, 24].

In Pardalos *et al.* [12] the technique of concave minimization is used to obtain a good approximation for (IQP). Using a linear underestimating function of the concave part a convex problem is solved to obtain a solution. Branch and bound technique is then used to improve the lower and upper bound and to reduce the domain under consideration. If the obtained solution is not a satisfactory approximation to the global minimum a piecewise linear approximation is used to convert the considered problem into a linear zero-one mixed integer program.

It should be noted that problem (IQP) is a special case of the minimization of a d.c. function (difference of two convex functions) over a polytope, for which the method developed by Horst-Phong-Thoai-Vries [7, 15] can be applied. A different branch-and-bound method using a normal subdivision was proposed in Tuy [26] while an outer approximation method was given in Tuy [25]. Recently, an approximation algorithm was proposed in Vavasis [27] for finding an ε-approximate solution. It was shown that such an approximation can be found in polynomial time for fixed ε and t, where t denotes the number of negative eigenvalues of the quadratic term.

The aim of this paper is to develop an efficient algorithm for solving problem (IQP) in the case the number of convex variables is much larger than that of concave variables. Since the objective function is a sum of a convex and concave parts we use a decomposition approach which enables us to work essentially in the small subspace of concave variables. This idea was first used in Phillips and Rosen [13] for dealing with concave quadratic minimization and then was extended to the case of partially separable nonconvex minimization in [14]. More recently, the same approach was used in Muu-Phong-Tao [11, 15] for solving a class of nonconvex programming problems dealing with bilinear and quadratic functions.

Taking advantage of the separable concave part in the objective function, we use a branch and bound method where branching proceeds by a rectangular subdivision in the y-space. Our method can be outlined as follows. The separability of the concave part motivates the use of rectangular subdivision. First a rectangular domain $R_0 \subset \mathbb{R}^s$ is constructed that contains the projection of Ω in the y-space. This rectangle is then divided into smaller and smaller subrectangles. For each rectangle R a convex underestimating function $p(x)+\phi(y)$ of the original objective function $f(x, y)$ is constructed and the convex minimization problem

$$\min \{p(x) + \phi(y) : (x, y) \in \Omega, y \in \mathbb{R}\}$$

is solved. The solution of this convex program gives both a lower and upper bound for the optimal value of the problem

$$\min \{p(x) + q(y) : (x, y) \in \Omega, y \in \mathbb{R}\}.$$

The branch-and-bound procedure is then applied to discard regions which can not contain any global minimizer and eventually to locate an optimal solution.

The efficiency of branch and bound method depends upon the choice of operations such as branching that is, in our case, a rectangular subdivision. To guarantee convergence, in [13] (*see* also Kalantari-Rosen [9]) Phillips-Rosen used an exhaustive subdivision, *i.e.* any nested sequence of rectangles generated by the algorithm will tend to a single point. Another one, called w-subdivision was proposed earlier in Falk-Soland [3]. In Horst-Tuy [8] a concept of "normal rectangular subdivision" was introduced for a class of separable concave minimization problems that includes just mentioned subdivisions. Recently, a so-called adaptive rectangular bisection proposed in Muu-Phong-Tao [11, 15] seems to be more efficient because the exhaustiveness is not necessary for the convergence. In this paper we provide a natural extension of this concept to problem (IQP) and establish the convergence of our method. Especially, we shall show that the adaptive rectangular bisection does belong to the class of normal subdivision, too.

An important question is the choice of the best subdivision strategies. Intuitively, variants of rectangular algorithms using w-subdivision and adaptive subdivision should converge more rapidly that those using exhaustive subdivision, because they take account of the conditions of the current relaxed subproblem. By numerical experiments, we not only confirm this observation but also point out that w-subdivision is the best.

The method presented in this paper should be efficient for large-scale (IQP) problems, when the number of variables that enter the concave part of the objective function is small in comparison with the total number of variables. We provide extensive numerical experiments for randomly generated problems with 10-20 concave variables and up to 200 convex variables.

The next section describes in detail a decomposition branch and bound algorithm for Problem (IQP) which is based on normal rectangular subdivision (NRS). The variants of NRS are discussed in Section 3. The implementation of the algorithm is presented in Section 4, where an illustrative example is given. Finally, the computational results are reported in Section 5.

2. DESCRIPTION OF THE ALGORITHM

To construct the smallest rectangular domain $R_0 \subset \mathbb{R}^s$ which contain the projection of Ω on the y-space, we solve s linear programming problems

$$\max\{y_i \text{ s.t. } (x,\, y) \in \Omega\}, \quad i = 1, \ldots, s$$

to get optimal values L_i^0, $i = 1, \ldots, s$. The rectangular domain can then be expressed as

$$R_0 = \{y : 0 \leq y_i \leq L_i^0\}.$$

2.1. Lower bounding

Let $R = \{y : l_i \leq y_i \leq L_i\}$ be a rectangle in $\mathbb{R}^s$. As usual, we adopt the convention that the infimum of an empty set is $+\infty$.

A standard method for lower bounding in branch and bound algorithms it to use convex underestimators of the objective function. Since concave function $q(y)$ is separable, its convex envelope over a rectangle R is simply the sum of affine function $\phi_{Ri}(y_i)$ that agrees with q_i at the endpoints of the segments $[l_i, L_i]$, *i.e.* the function (*cf.* [9, 17, 13], etc.)

$$\phi_R(y) = \sum_{i=1}^{s} \phi_{Ri}(y_i) \tag{3}$$

where $\phi_{Ri}(y_i)$ is given explicitly by

$$\phi_{Ri}(y_i) = \left[d_i - \frac{1}{2}\lambda_i(l_i + L_i)\right] y_i + \frac{1}{2}\lambda_i l_i L_i. \tag{4}$$

So $p(x) + \phi_R(y)$ is a convex underestimating function of $f(x, y)$ over the domain $\{(x, y) \in \mathbb{R}^n \times \mathbb{R}^s : (x, y) \in \Omega, y \in R\}$. The solution to the convex program

$$(RCP) \quad \min\{p(x) + \phi_R(y) : (x, y) \in \Omega, y \in R\}$$

provides a point (x^R, w^R) such that

$$\begin{aligned} p(x^R) + \phi_R(\omega^R) &\le \min\{f(x, y) : (x, y) \in \Omega, y \in R\} \\ &\le f(x^R, w^R) \end{aligned} \tag{5}$$

i.e. $\beta(R) = p(x^R) + \phi_R(w^R)$ is a lower bound for f over R and $f(x^R, w^R)$ is an upper bound for the global optimal value f_*.

Remark 1: A convex underestimator of the concave function q can be chosen in many different ways. In [11], as an application of a decomposition method for bilinear programming problems, the authors proposed the following underestimator

$$\psi_1(y) = \sum_{i=1}^{s}(d_i - \lambda_i L_i) y_i. \tag{6}$$

Another underestimator can be taken simply as

$$\psi_2(y) = \sum_{i=1}^{s}(d_i y_i - \lambda_i L_i^2). \tag{7}$$

It should be noted that ψ_1, ψ_2 are no convex envelopes of q over R. Moreover, the underestimator ϕ_R is the best in the sense

$$\psi_2(y) \le \psi_1(y) \le \phi_R(y), \qquad \forall y \in R.$$

2.2. Normal rectangular subdivision (NRS)

We first recall the concept of a normal rectangular subdivision as introduced by Tuy (*see* e.g. Horst-Tuy [8] (Definition VII.7)).

Let $R = \{y : l_i \leq y_i \leq L_i\}$ be a rectangle and let $\phi_{R(y)}$ be the above defined convex underestimator of $q(y)$ over R. Denote by (x^R, w^R) and $\beta(r)$ an optimal solution and the optimal value, respectively, of the convex program (RCP).

Consider now a rectangular subdivision process in which a rectangle is subdivided into subrectangles by means of a finite number of hyperplanes parallel to certain facets of the orthant $\mathbb{R}^s_+$. Such a process generates a family of rectangles which can be represented by a tree with root R_0 and such that a node is a successor of another one if and only if it represents an elements of the partition of the rectangle corresponding to the latter node. An infinite path in this tree corresponds to an infinite nested sequence of rectangles R_h, $h = 0, 1, \ldots$ For each h let $(x^h, w^h) = (x^{R_h}, w^{R_h})$, $\phi_h(y) = \phi_{R_h}(y)$.

DEFINITION 1: *A nested sequences R_h is said to be normal if*

$$\lim_{h \to \infty} |q(w^h) - \phi_h(w^h)| = 0. \tag{8}$$

A rectangular subdivision process is said to be normal if any infinite nested sequence of rectangles that it generates is normal.

We shall discuss some methods for constructing normal rectangular subdivision (NRS) process in the next section. Suppose now than an NRS process has been defined. Using this subdivision process in conjunction with the lower bounding defined in 2.1 we can construct the following branch and bound algorithm for solving (IQP).

2.3. Algorithm

Initialization: Compute the enclosing rectangle R_0 by solving s linear programs. Compute ϕ_{R_0} and solve the convex program

$$(R_0CP) \quad \min\{p(x) + \phi_{R_0}(y) : (x, y) \in \Omega, \ y \in R_0\}$$

to obtain an optimal solution (x^{R_0}, w^{R_0}) and the optimal value $\beta(R_0)$. Set $\mathcal{R} = \{R_0\}$, $\beta_0 = \beta(R_0)$, $\alpha_0 = f(x^{R_0}, w^{R_0})$ and $(x^0, y^0) = (x^{R_0}, w^{R_0})$.

Iteration k = 0, 1, 2... :

k.1. Delete all $R \in \mathcal{R}_k$ with $\beta(R) \geq \alpha_k$. Let $\mathcal{P}_k$ be the set of remaining rectangles. If $\mathcal{P}_k = \emptyset$ stop: (x^k, y^k) is a global optimal solution.

k.2. Otherwise, select $R_k \in \mathcal{P}_k$ such that

$$\beta_k := \beta(R_k) = \min\{\beta(R) : R \in \mathcal{P}_k\}.$$

and subdivide R_k into R_{k1}, R_{k2} according to the chosen normal rectangular subdivision process.

k.3. For each R_{k1}, R_{k2} compute $\phi_{R_{ki}}$ and solve

$$(R_{ki}CP) \quad \min\{p(x) + \phi_{R_{ki}}(y) : (x, y) \in \Omega, y \in R_{ki}\}$$

to obtain $(x^{R_{ki}}, w^{R_{ki}})$ and $\beta(R_{ki})$.

k.4. Set (x^{k+1}, y^{k+1}) to the best of the feasible solutions known so far and update α_{k+1}.

k.5. Set $\mathcal{R}_{k+1} := (\mathcal{P}_k \backslash R_k) \cup \{R_{k1}, R_{k2}\}$ and go to the next iteration.

THEOREM 1: (i) *If the Algorithm terminates at iteration k then* (x^k, y^k) *is a global optimal solution to problem (IQP).*

(ii) *If the Algorithm is infinite then it generates a bounded sequence* (x^k, y^k) *every accumulation point of which is a global optimal solution of* (IQP)*, and*

$$\alpha_k \searrow f_*, \qquad \beta_k \nearrow f_*.$$

Proof: Part (i) is clear from the definition of α_k, β_k and (x^k, y^k).

Assume now the Algorithm is infinite. Then it must generate an infinite nested sequence $\{R_h\}$ of rectangles. It is clear from the construction that the sequence $\alpha_k = f(x^k, y^k)$ is nonincreasing, while the sequence $\beta_k = \beta(R_k)$ is nondecreasing. Hence $\{\alpha_k - \beta_k\}$ is a nonincreasing sequence of positive numbers.

By the normality condition we may assume, by taking subsequences if necessary, that

$$\lim |q(w^{R_h}) - \phi_{R_h}(w^{R_h})| = 0.$$

Since $\beta(R_h) = p(x^{R_h}) + \phi_{R_h}(w^{R_h})$ this implies

$$f(x^{R_h}, w^{R_h}) - \beta(R_h) \to 0.$$

From the updating rule in step k.4. we have

$$0 < \alpha_h - \beta_h \leq f(x^{R_h}, w^{R_h}) - \beta(R_h)$$

hence $\alpha_h - \beta_h \to 0$ as $h \to \infty$. Since $\{\alpha_k - \beta_k\}$ is a nonincreasing, the whole sequence $\alpha_k - \beta_k$ must tend to 0. This and $\beta_k \leq f_* \leq \alpha_k$ for every k imply

$$\alpha_k \searrow f_*, \qquad \beta_k \nearrow f_*.$$

Since $(x^k, y^k) \in \Omega$ and $\alpha_h = f(x^k, y^k)$, any cluster point of the sequence $\{(x^k, y^k)\}$ belongs to Ω and has the function value f_*, *i.e.*, solves problem (IQP). The theorem is proved.

3. CONSTRUCTION OF A NRS

As pointed out in [8], there are different ways to construct a NRS process. Suppose a rectangle $R_k = \{y : l_i^k \leq y_i \geq L_i^k\}$ is selected in step k.2. The following rule for bisection of R_k was used in Kalantari-Rosen [9] and Phillips-Rosen [13]:

3.1. Exhaustive bisection

In general, i_k is chosen as the index of the longest edge of R_k, *i.e.*, such that

$$(L_{i_k}^k - l_{i_k}^k)^2 = \max\{(L_i^k - l_i^k)^2, i = 1, \ldots, s\}.$$

Let $\overline{\alpha} = 1/2\,(L_{i_k} + l_{i_k})$. Then R_k is bisected into two subrectangles:

$$R_{k,1} = \{y \in R_k : y_{i_k} \leq \overline{\alpha}\}, \qquad R_{k,2} = \{y \in R_k : y_{i_k} \geq \overline{\alpha}\}.$$

In particular, for separable concave quadratic function i_k can be also chosen such that

$$\lambda_{i_k}(L_{i_k}^k - l_{i_k}^k)^2 = \max\{\lambda_i(L_i^k - l_i^k)^2, i = 1, \ldots, s\}.$$

It was shown that in both cases any nested sequence of rectangles tends to a single point.

An alternative rule was earlier proposed by Falk and Soland [3] for separable nonconvex programming problems:

3.2. w-subdivision

It follows from the description of the Algorithm that, for the selected R_k, $\beta(R_k) < f(x^k, y^k)$, hence,

$$q(w^k) - \phi_k(w^k) > 0.$$

Choose an index i_k satisfying

$$i_k \in \arg\max_i \{q_i(w_i^h) - \phi_{ki}(w_i^k)\}$$

and subdivide R_k into two subrectangles

$$R_{k,1} = \{y \in R_k : y_{i_k} \le w^k_{i_k}\}, \; R_{k,2} = \{y \in R_k : y_{i_k} \ge w^k_{i_k}\}.$$

Recently, for solving a class nonconvex programming problems dealing with bilinear and quadratic function, in Muu-Phong-Tao [11, 15] the authors introduced the following subdivision rule:

3.3. Adaptive bisection

For each selected R_k, two points are considered. The first one is w^k, the other is a point v^k such that

$$v^k_i \in \arg\min_i \{q_i(l^k_i), \; q_i(L^k_i)\}. \tag{9}$$

In other words

$$v^k \in \arg\min_{R_k} \phi_{R_k}(y). \tag{10}$$

These points are called the *bisection points*. Let i_k be an index such that

$$|(v^k - w^k)_{i_k}| = \max_i |(v^k - w^k)_i|$$

and let $\overline{\alpha} = 1/2\,(w^k_{i_k} + v^k_{i_k})$. Then we bisect R_k into two subrectangles as

$$R_{k,1} = \{y \in R_k : y_{i_k} \le \overline{\alpha}\}, \; R_{k,2} = \{y \in R_k : y_{i_k} \ge \overline{\alpha}\}.$$

The following property has been established in [11].

PROPOSITION 1: *Let $R_1 \supset R_2 \supset \ldots$ be an infinite sequence of nested rectangles so that R_{h+1} is obtained from R_h via the adaptive bisection, and let $\{v^h\}$, $\{w^h\}$ be the sequences of the corresponding bisection points. Then there exists subsequences $\{v^{h_l}\}$ and $\{w^{h_l}\}$ such that*

$$\|v^{h_l} - w^{h_l}\| \to 0 \quad as \quad l \to \infty.$$

It has been shown in [8] that the exhaustive bisection and w-subdivision rules (in conjunction with the lower bounding defined in 2.1, *i.e.*, using convex envelope ϕ_R) generate a normal rectangular subdivision process. We show below that the adaptive subdivision also generates a normal one.

Let $\{R^h\}$ be any infinite nested sequence of rectangles. By Proposition 1, by taking a subsequence if necessary we may assume that $\|v^h - w^h\| \to 0$ as $h \to \infty$. From the definition (10) v^h is a vertex of R_h therefore $\phi_h(v^h) = q(v^h)$. Thus

$$\begin{aligned} |q(w^h) - \phi_h(w^h) &\le |q(w^h) - q(v^h)| + |q(v^h) - \phi_h(w^h)| \\ &\le |q(w^h) - q(v^h)| + |\phi_h(v^h) - \phi_h(w^h)|. \end{aligned}$$

hence, $|q(w^h) - q(v^h)| \to 0$. By (4) there exists a constant η such that

$$|\phi_{hi}(v_i^h) - \phi_{hi}(w_i^h)| \le \eta |v_i^h - w_i^h|$$

for all h so

$$|\phi_h(v^h) - \phi_h(w^h)| \le \eta s |v_{i0}^h - w_{i0}^h|$$

where $i_0 \in \arg\max_i |v_i^h - w_i^h|$. Therefore when $\|v^h - w^h\| \to 0$ we have $|\phi_h(v^h) - \phi_h(w^h)| \to 0$ that implies (8).

Crucial points in the above proof are the following:

(i) $\phi_h(v^h) = q(v)$ where v is defined by (10).

(ii) ϕ_{hi} is Lipschitzian on $[l_i, L_i]$ such that

$$K_{hi} \le K, \qquad \forall h, \quad \forall i$$

where K_{hi} is the Lipschitz constant of the function ϕ_{hi}.

Arguing analogously we can establish the following more general result:

PROPOSITION 2: *Let ϕ_h be an underestimator for q over R satisfying the conditions* (i)-(ii). *Then for any nested sequence $\{R_h\}$ generated by the adaptive bisection, one has (8).*

Comment. The adaptive rectangular subdivision was successfully used in [11, 15] for global minimizing a concave quadratic function under linear constraints and for solving quadratic mixed integer programming problems. We pointed out that, indeed, this is a particular case of the normal subdivision so the convergence of the methods in [11, 15] follows from Theorem 1. Specifically, both underestimators ψ_1, ψ_2 defined in Remark 1 satisfy the conditions (i)-(ii) so by Proposition 2, the adaptive subdivisions in conjunction with the lower bounding which use ψ_1, ψ_2 instead of ϕ_R also generate a NRS process.

Intuitively, it can be expected that rectangular algorithms using the w-subdivision and adaptive subdivision should converge more rapidly than those using exhaustive bisection, because the former depends on the solution w^k of the current relaxed subproblem. This has been confirmed by our numerical experiments (*see* Section 5). Moreover, the numerical results have shown that for the problem under consideration, the w-subdivision is more efficient than the adaptive bisection.

4. IMPLEMENTATION AND ILLUSTRATIVE EXAMPLE

A matter of utmost importance in the computer implementation of an branch and bound algorithm is the construction of a convenient computer scheme for storing and updating the information about partition elements. This scheme should permit an efficient use of computer resource and must be able to:

1) store the information decribing the current list of rectangles at each step

2) select a rectangle to be subdivided further

3) remove a number of rectangles from the list when necessary

4) record the information about the newly generated rectangles.

It should be noted that the number of rectangles to be stored may increase very quickly and that a number of vertices of different rectangles may coincide. This fact should be taken into account in order to save memory storage.

Since each rectangle is defined by (at least) two opposite vertices, for storing rectangles we use two linked lists: list POINT contains the coordinates of vertices of all rectangles and list REC each elements of which contains the information about the rectangle: two indexes of two vertices, lower bound,... In passing from step k to step $k+1$ the rectangle with the smallest lower bound is selected from the list REC to subdivided. This elements will be replaced by (at most) two new elements. It may happen that the new vertex coincides with one previously generated and already stored in list POINT. One can also free an element of list POINT if the number of elements of list REC having is as a vertex is null.

The main subroutine in the Algorithm described in the preceding sections is for solving a convex programming problem

$$(RCP) \quad \min \{p(x) + \phi_R(y) : (x, y) \in \Omega, y \in R\}.$$

For a given rectangle R, the convex underestimator defined by (4) can be rewritten as

$$\phi_R(y) = \langle e, y\rangle + \gamma$$

where

$$e_i = d_i - 1/2\,\lambda_i\,(l_i + L_i), \qquad i = 1, \ldots, s$$
$$\gamma = 1/2 \sum_{i=1}^{s} \lambda_i\, l_i\, L_i.$$

Thus problem (RCP) is reduced to solving

$$\min \frac{1}{2}\langle Cx, x\rangle + \langle c, x\rangle + \langle e, y\rangle$$

subject to

$$Ax + By \leq b, \qquad x \geq 0, \qquad l \leq y \leq L.$$

Several methods are available for minimizing a convex quadratic function over a polytope, in particular active set methods and Lemke's method (*cf.* e.g. [4, 6]). We choose to use Lemke's method for its simplicity. Our implementation also takes account of the fact that the above problems differ only by the linear term $\langle e, y\rangle$ and the box constraints $l \leq y \leq L$.

The algorithm was coded in Pascal under Unix system. Optionally, the user can choose one of the three types of normal rectangular subdivision discussed in Section 3.

We illustrate the described algorithm by the following example taken from Floudas and Pardalos [5].

$$\min f(x, y) = 0.5 \sum_{i=1}^{10} \mu_i\,(x_i - \beta_i)^2 - 0.5 \sum_{i=1}^{10} \lambda_j\,(y_j - \alpha_j)^2$$

subject to

$$Ax + By \leq b, \qquad x \geq 0, \quad y \geq 0, \quad x \in \mathbb{R}^{10}, \quad y \in \mathbb{R}^{10}$$

where

$$\begin{aligned}
\lambda &= (63,\ 15,\ 44,\ 91,\ 45,\ 50,\ 89,\ 58,\ 86,\ 82),\\
\mu &= (42,\ 98,\ 48,\ 91,\ 11,\ 63,\ 61,\ 61,\ 38,\ 26),\\
\alpha &= (-19,\ -27,\ -23,\ -53,\ -42,\ 26,\ -33,\ -23,\ 41,\ 19)\\
\beta &= (-52,\ -3,\ 81,\ 30,\ -85,\ 68,\ 27,\ -81,\ 97,\ -73),
\end{aligned}$$

$$A = \begin{pmatrix}
2 & 5 & 5 & 6 & 4 & 4 & 5 & 6 & 4 & 4\\
5 & 4 & 5 & 4 & 1 & 4 & 4 & 2 & 5 & 2\\
1 & 5 & 2 & 4 & 7 & 3 & 1 & 5 & 7 & 2\\
3 & 2 & 6 & 3 & 2 & 1 & 6 & 1 & 7 & 3\\
6 & 6 & 6 & 4 & 5 & 2 & 2 & 4 & 3 & 2\\
5 & 5 & 2 & 1 & 3 & 5 & 5 & 7 & 4 & 3\\
3 & 6 & 6 & 3 & 1 & 6 & 1 & 6 & 7 & 1\\
1 & 2 & 1 & 7 & 8 & 7 & 6 & 5 & 8 & 7\\
8 & 5 & 2 & 5 & 3 & 8 & 1 & 3 & 3 & 5\\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix},$$

$$B = \begin{pmatrix}
8 & 2 & 4 & 1 & 1 & 1 & 2 & 1 & 7 & 3\\
3 & 6 & 1 & 7 & 7 & 5 & 8 & 7 & 2 & 1\\
1 & 7 & 2 & 4 & 7 & 5 & 3 & 4 & 1 & 2\\
7 & 7 & 8 & 2 & 3 & 4 & 5 & 8 & 1 & 2\\
7 & 5 & 3 & 6 & 7 & 5 & 8 & 4 & 6 & 3\\
4 & 1 & 7 & 3 & 8 & 3 & 1 & 6 & 2 & 8\\
4 & 3 & 1 & 4 & 3 & 6 & 4 & 6 & 5 & 4\\
2 & 3 & 5 & 5 & 4 & 5 & 4 & 2 & 2 & 8\\
4 & 5 & 5 & 6 & 1 & 7 & 1 & 2 & 2 & 4\\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}$$

and

$$b = (380,\ 415,\ 385,\ 405,\ 470,\ 415,\ 400,\ 460,\ 400,\ 200)^T.$$

The convex part can be rewritten as $0.5 < Cx, x > + < c, x > +911425.50$ where

$$\begin{aligned}
C &= \mathrm{diag}\,(42,\ 98,\ 48,\ 91,\ 11,\ 63,\ 61,\ 38,\ 26),\\
c &= \mathrm{diag}\ (2184,\ 294,\ -3888,\ -2730,\ 85,\ -4284,\ -1647,\\
&\qquad 4941,\ -3686,\ 1898).
\end{aligned}$$

We choose

$$R_0 = \{x : 0 \leq y_1 \leq 50,\, 0 \leq y_2 \leq 67,\, 0 \leq y_3 \leq 67,\\ 0 \leq y_4 \leq 64,\, 0 \leq y_5 \leq 55,\, 0 \leq y_6 \leq 50,\, 0 \leq y_7 \leq 67.5,\\ 0 \leq y_8 \leq 60,\, 0 \leq y_9 \leq 55, 0 \leq y_{10} \leq 66\}$$

and solve the convex program

$$\begin{aligned} &\min \frac{1}{2}\langle Cx,\, x\rangle + \langle c,\, x\rangle + \langle e,\, y\rangle + 547663.5 \\ &\text{s.t. } Ax + By \leq b, \qquad x \geq 0, \quad y \in R_0 \end{aligned}$$

where

$$e = (-2772,\ -907.5,\ -2486,\ -7735,\ -3127.5,\ 50,\ -5940.75,\\ -\ 3074,\ 1161,\ -1148)$$

to obtain $\beta(R_0) = 37923.5$ and

$$x^{R_0} = (0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 6.666667,\, 0.,\, 0.,\, 0.,\, 0.),$$
$$w^{R_0} = (6.666667,\, 0.,\, 0.,\, 60.,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.).$$

Set

$$\mathcal{R}_0 = \{R_0\},\, \beta_0 = 37923.5,\, (x^0,\, w^0) = (x^{R_0},\, y^{R_0})$$

and $\alpha_0 = f(x^0,\, y^0) = 57943.5$. Applying w-subdivision rule we choose $i_0 = 4$ and subdivide R_0 into two subrectangles

$$R_1 = \{x : 0 \leq y_1 \leq 50,\, 0 \leq y_2 \leq 67,\, 0 \leq y_3 \leq 67,\, 0 \leq y_4 \leq 60,\\ 0 \leq y_5 \leq 55,\, 0 \leq y_6 \leq 50,\, 0 \leq y_7 \leq 67.5,\, 0 \leq y_8 \leq 60,\\ 0 \leq y_9 \leq 55, 0 \leq y_{10} \leq 66\},$$
$$R_2 = \{x : 0 \leq y_1 \leq 50,\, 0 \leq y_2 \leq 67,\, 0 \leq y_3 \leq 67,\, 60 \leq y_4 \leq 64,\\ 0 \leq y_5 \leq 55, 0 \leq y_6 \leq 50,\, 0 \leq y_7 \leq 67.5,\, 0 \leq y_8 \leq 60,\\ 0 \leq y_9 \leq 55, 0 \leq y_{10} \leq 66\}.$$

For R_1 we compute

$$e = (-2772,\ -907.5,\ -2486,\ -7553,\ -3127.5,\ 50,\ -5940.75,\\ -\ 3074,\ 1161,\ -1148),$$

and solve

$$\begin{aligned} &\min \frac{1}{2}\langle Cx,\, x\rangle + \langle c,\, x\rangle + \langle e,\, y\rangle + 547663.5 \\ &\text{s.t. } Ax + By \le b, \qquad x \ge 0, \quad y \in R_1 \end{aligned}$$

obtaining $\beta(R_1) = 48833.50543$ and

$$\begin{aligned} x^{R_1} &= (0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 7.22995,\, 0.,\, 0.,\, 0.,\, 0.), \\ w^{R_1} &= (9.008739,\, 0.,\, 0.,\, 54.307875,\, 0.,\, 0.,\, 5.781065,\, 0.,\, 0.,\, 0.). \end{aligned}$$

Analogously for R_2 we get

$$\begin{aligned} e = (&-2772,\, -907.5,\, -2486,\, -10465,\, -3127.5,\, 50,\, -5940.75, \\ &-3074,\, 1161,\, -1148) \end{aligned}$$

$\beta(R_2) = 48843.5$ and

$$\begin{aligned} x^{R_2} &= (0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 6.666667,\, 0.,\, 0.,\, 0.,\, 0.), \\ w^{R_2} &= (6.666667,\, 0.,\, 0.,\, 60.,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.). \end{aligned}$$

We update $\beta_1 = 48833.50543$, $\alpha_1 = 57943.5$ and set $\mathcal{R}_1 = \{R_1,\, R_2\}$.

The algorithm terminated after 5 iterations at the global optimal solution

$$\begin{aligned} x &= (0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 4.347826,\, 0.,\, 0.,\, 0.,\, 0.), \\ y &= (0.,\, 0.,\, 0.,\, 62.608696,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.,\, 0.) \end{aligned}$$

with the optimal value –49318.01796. It is interesting to note that the same solution was provided in [5] but it has not been proved to be globally optimal.

5. COMPUTATIONAL RESULTS FOR LARGE-SCALE PROBLEMS

It should be noted that the algorithm still converges if instead of bisection, a rectangle is subdivided into a relatively high number of small rectangles. At first glance, this usually leads to better lower bounds and so needs less iterations as compared to bisection. However, the most expensive part in our algorithm is the solution of subproblems at each iteration and an algorithm is faster if the total number of subproblems to solve is lower. From reported experimental results, we see that the algorithm using bisection converges very rapidly, in general after 11 iterations, *i.e.* one needs to solves 23 subproblems.

We have tested the Algorithms on a number of problems, where all elements of matrices A, B and vectors c, d, λ are randomly generated together with their signs, so that the feasible region was nonempty and bounded. A positive definite matrix C is constructed following Moré and Sorensen [10]. More precisely, we set $C = QDQ^T$ for some orthogonal matrix Q and a diagonal matrix D. The orthogonal matrix Q of the form $Q_1 Q_2 Q_3$ where

$$Q_j = I - 2\,\frac{w_j\, w_j^T}{\|w_j\|^2}, \qquad j = 1, 2, 3$$

and the components w_j are random numbers in $(-1, 1)$. The diagonal elements of matrix D are random numbers in $(0, 5)$.

The deletion rule $\beta(R) \geq \alpha_k$ was replaced by $\beta(R) \geq (\alpha_k - \varepsilon)$ so that the algorithm terminates whenever and $\varepsilon-$ optimal solution $\overline{x}$ has been obtained, *i.e.* when $|f(\overline{x}) - f(x^*)| \leq \varepsilon$.

All the test problems were ran on SPARC station (processor 2). In Tables I and II the quantities "min iter", "avg iter" and "max iter" represent, respectively, the minimal average and maximal number of iterations, while "time" means the average excecution time over 3 problems with the same size. We took $\varepsilon = 0.001$. The relative error (in percentage) ranged from 0.000002 to 0.0001.

TABLE I

Comparison of three variants using exhaustive, adaptive and w-subdivision.

Subdiv rule	min iter	avg iter	max iter	time
Exhaustive	45	49.67	56	78.94
Adaptive	12	15.7	19	25.25
w-subdivision	5	7.3	12	11.71

TABLE II

Numerical results for large-scale problems using w-subdivision.

n	s	m	min iter	avg iter	max iter	time
50	10	10	8	10.0	11	17.26
100	10	20	8	8.7	10	88.37
150	10	20	4	5.0	6	86.57
150	15	20	4	5.3	8	195.23
150	20	20	5	15.0	32	429.25
200	10	20	6	8.0	11	380.10

Table I presents the numerical results for $n = 50$, $s = 5$, $m = 10$. We see that the variants of the Algorithm using adaptive bisection and w-subdivision converge more rapidly than that using exhaustive bisection. Moreover, the w-subdivision is more efficients than the adaptive bisection.

Table II presents the numerical results for large-scale problems using the w-subdivision. The number of variables entering the convex part of the objective function was increased up to 200. As expected the number of iterations appears to increase only linearly with the number of concave variables. This reflects the theoretical aspect of the decomposition approach that the branching procedure is actually performed in the space of the concave variables. Numerical experiments reported in this paper are only preliminary. Nevertheless, we hope that the proposed algorithm will be practicable for the class of problems under consideration.

ACKNOWLEDGEMENT

The authors very are grateful to the referees for several helpful comments and suggestions which have improved the presentation of the revised paper.

REFERENCES

1. Le T. H. An, *Analyse numérique des algorithmes de l'Optimisation d.c. Approches locales et globales. Codes et simulations numériques en grande dimension. Applications.* Thèse de Doctorat de l'Université de Rouen, France, 1994.
2. Le T. H. An, Pham D. Tao and L. D. Muu, Numerical solution for Optimization over the Efficient set by d.c. optimization algorithm, To appear in *Operations Research Letters.*
3. J. E. Falk and R. M. Soland, An algorithm for separable non convex programming problems, *Management Science*, 1969, *15*, pp. 550-569.
4. R. Fletcher, *Practical methods of Optimization (second edition)*, John Wiley & Sons, New York, 1991.
5. C. A. Floudas and P. M. Pardalos, A collection of test problems for constrained global optimization algorithms, In G. Goos and J. Hartmanis, editors, *Lecture notes in Computer Science*, 445, Springer-Verlag, 1987.
6. P. E. Gill, W. Murray and M. H. Wright, *Practical optimization*, Academic Press, 1981.
7. R. Horst, T. Q. Phong, N. V. Thoai and J. de Vries, On solving a d.c. programming problem by a sequence of linear programs, *J. of Global Optimization*, 1991, *1*, pp. 183-203.
8. R. Horst and H. Tuy, *Global Optimization: Deterministic Approaches*, Springer-Verlag, Berlin, New York, 2e edition, 1993.
9. B. Kalantari and J. B. Rosen, Algorithm for global minimization of linearly constrained concave quadratic functions, *Mathematics of Operations Research*, 1987, *12*, pp. 544-561.

10. J. J. Moré and D. C. Sorensen, Computing a trust region step, *SIAM J. Sci. Statist. Comput.*, 1981, *4*, pp. 553-572.

11. L. D. Muu, T. Q. Phong and Pham Dinh Tao, Decomposition methods for solving a class of nonconvex programming problems dealing with bilinear and quadratic functions, *Computational Optimization and Application*, 1995, *4*, pp. 203-216.

12. P. M. Pardalos, J. H. Glick and J. B. Rosen, Global optimization of indefinite quadratic problems, *Computing*, 1987, *39*, pp. 281-291.

13. A. T. Phillips and J. B. Rosen, A parallel algorithm for constrained concave quadratic global minimization, *Mathematical Programming*, 1988, *42*, pp. 412-448.

14. A. T. Phillips and J. B. Rosen, A parallel algorithm for partially separable non-convex global minimization: linear constraints, *Annals of Operations Research*, 1990, *25*, pp. 101-118.

15. T. Q. Phong, *Analyse numérique des algorithmes d'Optimisation globale. Codes et simulations numériques. Applications.* Thèse de Doctorat de l' lUniversité de Rouen, France, 1994.

16. J. B. Rosen, Minimization of linearly constrained concave function by partition of feasible domain, *Math. Operations Research*, 1983, *8*, pp. 215-230.

17. J. B. Rosen and P. M. Pardalos, Global minimization of large scale constrained quadratic problems by separable programming, *Mathematical Programming*, 1986, *34*(2), pp. 163-174.

18. Pham D. Tao, *Contribution à la théorie de normes et ses applications à l'analyse numérique.* Thèse de doctorat d'etat es sciences, USMG, Grenoble, France, 1981.

19. Pham D. Tao, Convergence of subgradient method for computing the bound norm of matrices, *Linear Alg. and its Appl.*, 1984, *62*, pp. 163-182.

20. Pham D. Tao, Algorithmes de calcul du maximum d'une forme quadratique sur la boule unité de la norme du maximum, *Numerische Mathematik*, 1985, *45*, pp. 377-440.

21. Pham D. Tao, Algorithms for solving a class of nonconvex optimization problems. subgradient methods, *Fermat days 85. Mathematics for Optimization*, Elsevier Science Publishers B. V. North-Holland, 1986.

22. Pham D. Tao, *Some new results in nonconvex nondifferentiable optimization. 6th French-German Conference on Optimization*, Lambrech, Germany, 2/6-8/6 1991.

23. Pham D. Tao and El Bernoussi, Duality in d.c. (difference of convex functions) optimization. subgradient methods, *Trends in Mathematical Optimization*, volume 84 of *International Series of Numerische Mathematik*, Birkhauser, 1988.

24. Pham D. Tao and Le T. H. An, D. C. (difference of convex functions) optimization algorithms (DCA) for globally minimizing nonconvex quadratic forms on euclidean balls and spheres, To appear in *Operations Research Letters.*

25. H. Tuy, Global minimization of a difference of two convex functions, *Mathematical Programming Study*, 1987, *30*, pp. 150-182.

26. H. Tuy, The complementary convex structure in global optimization, *J. of Global Optimization*, 1992, *2*, pp. 21-40.

27. S. A. Vavasis, Approximation algorithms for indefinite quadratic programming, *Mathematical Programming*, 1992, *57*, pp. 279-311.