QUADRATIC OPTIMIZATION

CHRISTODOULOS A. FLOUDAS* and V. VISWESWARAN
Department of Chemical Engineering
Princeton University
Princeton, NJ 08544-5263 USA

Abstract. Quadratic optimization comprises one of the most important areas of nonlinear programming. Numerous problems in real world applications, including problems in planning and scheduling, economies of scale, and engineering design, and control are naturally expressed as quadratic problems. Moreover, the quadratic problem is known to be NP-hard, which makes this one of the most interesting and challenging class of optimization problems. In this chapter, we review various properties of the quadratic problem, and discuss different techniques for solving various classes of quadratic problems. Some of the more successful algorithms for solving the special cases of bound constrained and large scale quadratic problems are considered. Examples of various applications of quadratic programming are presented. A summary of the available computational results for the algorithms to solve the various classes of problems is presented.

Key words: Quadratic optimization, bilinear programming, concave programming, indefinite quadratic, quadratic test problems.

1. Introduction

Nonlinear programming problems arise in the mathematical modeling of several problems in real world applications. A large number of these problems can be formulated as quadratic programming problems (**QP**) with a quadratic objective function and a linear set of equality or inequality constraints. Primarily, quadratic programming with linear constraints can be viewed as a generalization of the linear programming problem (**LP**) with a quadratic objective function. Therefore, it encompasses all LP problems, including applications in scheduling, planning and flow computations. Moreover, quadratic programming is known to be NP-hard, which means that some of the most interesting combinatorial optimization problems can be posed in a quadratic programming framework. In addition, quadratically constrained quadratic programs occur frequently in the engineering modeling, design, and control.

Apart from these indirect applications, there are several classes of problems that are naturally expressed as quadratic problems. Examples of such problems can be found in planning and scheduling, game theory, problems involving economies of scale, facility allocation and location problems, quadratic assignment problems (which might also involve mixed 0-1 variables in the quadratic formulation), problems in engineering design, and a number of problems in microeconomics. In the area of chemical engineering process modeling, quadratic models are often used to formulate problems in pooling and blending, multiperiod tankage quality problems,

* Author to whom all correspondence should be addressed

and problems involving separation sequences. Finally, certain aspects of VLSI chip design can be formulated as linearly and quadratically constrained quadratic models.

It is clear, therefore, that the quadratic programming problem has great importance both from the mathematical and application viewpoints. This is reflected in the large number of approaches that have been proposed for the solution of these problems. Traditionally, quadratic problems have been treated as a subclass of the general nonlinear programming problem, and methods of solving these problems have relied on local optimization techniques. However, in recent years, there have been attempts at developing global optimization algorithms for this class of problems. It should be noted that there are several global optimization algorithms that have been developed for the more general nonlinear programming problem. A comprehensive review of these algorithms can be found in Pardalos and Rosen (1986), Pardalos and Rosen (1987) and Horst and Tuy (1993).

Global optimization approaches for the general nonconvex nonlinear problem (including approaches for quadratic optimization) can be primarily classified as being either deterministic or stochastic approaches. In this review, we will restrict ourselves to the deterministic approaches for quadratic optimization. For approaches for more general nonlinear global optimization, the reader is referred to Tuy (1987), Floudas *et al.* (1989), Al-Khayyal (1990), Shor (1990), and Hansen *et al.* (1991). Recent developments in global optimization approaches can be found in Floudas and Pardalos (1992). Discussions on stochastic methods can be found in Rinnooy-Kan and Timmer (1985,1987), Mockus (1989) and Torn and Zilinskas (1989). The reader is also referred to the chapters on concave minimization, d.c. programming, stochastic optimization and Lipschitz optimization in this handbook.

The rest of this chapter is organized as follows. In Section 2, we present the definition of the general quadratic problem with linear constraints along with a classification based upon the structure of the problem. Section 3 discusses the basic properties of this class of problems (including various forms of the necessary conditions for optimality), and provides the known results that characterize their solution. In Section 4, we discuss the quadratic problem from the view point of complexity, and show why these problems are difficult to solve. Sections 5, 6 and 7 discuss the particular nature of bilinear, concave and indefinite quadratic problems respectively, and present an overview of the existing algorithmic approaches. Section 8 considers algorithms for solving the most general form of the quadratic problem, namely problems with quadratic objective functions and quadratic constraints. Section 9 briefly outlines the relations between quadratic programming and linear complementarity problems. Section 10 presents examples from several application areas that can be formulated in the quadratic programming framework. Finally, section 11 discusses available computational results for bilinear, concave, and indefinite quadratic programming problems, as well as quadratically constrained problems.

2. The General Quadratic Programming Problem

The general quadratic problem consists of a quadratic objective function and a set of linear inequality constraints, as shown below:

$$\begin{aligned} \min_x \; Q(x) &= c^T x + \frac{1}{2} x^T D x \\ s.t. \qquad Ax &\leq b \\ x &\geq 0 \end{aligned} \tag{1}$$

where c is an n vector, b is an m vector, A is an $m \times n$ matrix and D is an $n \times n$ matrix. Without any loss of generality, it may be assumed that D is symmetric. If this is not the case, then it can be converted to symmetric form by replacing D by $(D + D^T)/2$, which does not change the value of the objective function $Q(x)$. Similarly, any problem where the variables are not necessarily nonnegative can be converted by a linear transformation to (1).

If the matrix D is positive semidefinite or positive definite, then (1) becomes a convex programming problem. Since any local optimum is equivalent to the global optimum in convex problems, (1) can be solved by any of a number of algorithms for convex quadratic programming. In particular, it is well known that the convex quadratic problem is in P (the class of problems solvable in polynomial time), and therefore there exist polynomial time algorithms that can be applied to solve these problems. For example, when the data for the problem is integers, Chung and Murty (1981) showed that the ellipsoid algorithm of Khachiyan (1979) could be used to solve convex quadratic programming problems in polynomially bounded computation time. A polynomial time algorithm for this problem was also presented by Kozlov *et al.* (1979). Other polynomial algorithms (including some interior-point approaches) for convex quadratic programming have been proposed by Nesterov (1988), Monteiro and Adler (1989), Ye (1989), Ben-Daya and Shetty (1990), Granot and Skorin-Kapov (1990), Monteiro *et al.* (1990), and Goldfarb and Liu (1991). Moreover, there are a large number of algorithms for convex nonlinear programming that can be applied to solve (1). Therefore, the cases of interest, from the viewpoint of global optimization, are when D is not positive semidefinite. In such cases, (1) becomes a nonconvex problem, and the application of local optimization procedures for this problem can no longer guarantee the identification of the global optimum.

2.1. Classification of Quadratic Programming Problems

The main basis for classification of quadratic problems of the form (1) comes from the nature of the quadratic matrix D. Based upon this, quadratic problems can be classified as:

1. Bilinear Problems: Often, the matrix D is such that there exist two subvectors of distinct variables y and z of x such that the problem is linear when one of these vectors is fixed. Such problems are termed bilinear problems, and are considered in Section 5.
2. Concave Quadratic Problems: When the matrix D is negative semidefinite (i.e. all its eigenvalues are nonpositive), then problem (1) reduces to one of concave minimization. This is further discussed in detail in Section 6.
3. Indefinite Quadratic Problems: These problems arise when the matrix D has both positive and negative eigenvalues. From the view point of solution, this class of problems is the most intractable among these subclasses, and consequently

there are not too many methods for solving these problems. The available approaches for solving indefinite quadratic problems are discussed in Section 7.

It should be noted that problems in these categories can often be converted into the other forms (see Konno (1976) and Hansen and Jaumard (1992) for some examples of how this can be achieved). Hence, algorithms that address one of these classes can also be used for problems in the other categories.

3. Optimality Conditions and Solution Characterization

A good starting point for approaching the solution of (1) is to review what can be gathered from general duality theory for nonlinear programming problems. The Lagrange function for (1) is given by

$$L(x,y,u,v) = c^T x + \frac{1}{2} x^T D x + u^T(Ax - b) + v^T(0 - x)$$

where u and v are the Lagrange multipliers for the inequality and nonnegativity constraints respectively. Using this, the first-order Karush-Kuhn-Tucker **(KKT)** optimality conditions for (1) can be written as follows (Bazaraa and Shetty, 1979):

$$-Dx - A^T u + v = c \tag{2}$$

$$u^T(Ax - b) = 0 \tag{3}$$

$$x^T v = 0 \tag{4}$$

$$Ax - b \leq 0 \tag{5}$$

$$x, u, v \geq 0 \tag{6}$$

Here, (2) is the stationarity condition, (3) and (4) are the complementary slackness conditions, and (5) and (6) ensure feasibility of the resulting solution. Any point $x^\star$ which satisfies (2)-(6) is called a KKT stationary point of (1). It can be shown that if $x^\star$ is an optimal solution of (1), then $x^\star$ must be a KKT point of (1) regardless of whether D is positive semidefinite or not. However, the converse is not necessarily true. In other words, the KKT conditions (2)-(6) are necessary for all quadratic problems (whether convex or not) but sufficient only for convex problems.

3.1. Feasible descent directions and optimality conditions

It is also possible to formulate the necessary and sufficient conditions for local optimality for (1) using the concept of a feasible descent direction at a given point. At any feasible point $x^\star$, a vector $d \in \Re^n$ is a feasible direction if there exists an $\epsilon > 0$ such that $x^\star + td$ is feasible for all $t \in [0, \epsilon]$. The vector d is said to be a descent direction if there exists an $\epsilon > 0$ such that $f(x^\star + td) < f(x^\star)$ for all $t \in [0, \epsilon]$. A vector d is a feasible descent direction if it is a feasible direction as well as a descent direction. Then, the following two conditions are equivalent to the KKT conditions for (1):

Theorem 3.1 (Necessary and Sufficient Conditions for Local Optimality) *If g denotes $Dx^\star + c$, the gradient of the objective function at $x^\star$ and A_a denotes the subset of constraints that are active at $x^\star$, then, a necessary and sufficient condition for $x^\star$ to be a local minimizer for (1):*

(i) *For all* d *such that* $A_a d \leq 0, \quad g^T d \geq 0$

(ii) *For all* d *such that* $A_a d \leq 0$ *and* $g^T d = 0$, $d^T D d \geq 0$.

Proof. See Theorems 5.2 and 5.3 of Vavasis (1991).

It can be easily shown that condition (i) of the above theorem, when combined with the feasibility requirement, leads to the KKT conditions (2)-(6). Condition (ii) is simply an additional second order optimality condition. Moreover, it follows as a direct corollary from conditions (i) and (ii) that there is no feasible descent direction at any local optimum $x^\star$.

The following theorem shows the equivalence between (1) and a corresponding linear problem, and can be used as a sufficient test of global optimality.

Theorem 3.2 (Murty, 1988) *If* $x^\star$ *is an optimum solution of (1), then* $x^\star$ *is also an optimum solution of the LP*

$$\begin{aligned} &\min_x \quad (c + x^{\star T} D)x \\ s.t. \quad &Ax \geq b \\ &x \geq 0 \end{aligned} \tag{7}$$

3.2. Active Constraints and Optimality

One of the more interesting results for quadratic problems comes from the consideration of the active set of constraints at the optimal solution. The following theorem, taken from (Hager *et al.*, 1991), is an adaptation of their original theorem for the general nonlinear programming problem:

Theorem 3.3 *If at some point* $x^\star$*, the matrix* Q *has* s *negative eigenvalues (counting multiplicities), then there must be at least* s *active constraints at* $x^\star$.

Theorem 3.3 provides a direct correlation between the geometric structure of the feasible region and the algebraic structure of the Q matrix. It follows directly that if at any $x^\star$, there are n negative eigenvalues, then $x^\star$ is an extreme point of the feasible region. Consequently, for problems where the objective function is bounded over the feasible set and there is at least one optimal solution, the following results follow:

(a) For concave quadratic problems, there exists an optimal solution at an extreme point of the feasible set.

(b) For indefinite quadratic problems, there exists an optimal solution at a boundary point of the feasible set.

(c) If there exists an optimal solution at an interior point on a facet of the polytope defined by the feasible set, then Q must have exactly one negative eigenvalue.

These results can be particularly useful for algorithms that are based upon an active set strategy for solving (1).

3.3. Global Optimality Criteria

While there are a large number of necessary and sufficient conditions of local optimality that are applicable to the solutions of (1), there are very few results of global optimality. One of the few criterion that is specifically directed towards this end is given by the following theorem

Theorem 3.4 (Warga (1992)) *There exist $N^\star \in \{1, 2, \ldots\}$ and $c^\star > 0$ such that for every choice of real numbers $N \geq N^\star$ and $c \geq c^\star$, $x^\star$ is the optimal solution to (1) if and only if $x^\star$ also minimizes $c^T x + \frac{1}{2} x^T D x + c|Ax - b|^{1/N}$ over the nonnegative orthant of $\Re^n$. Moreover, if $\nu, \gamma > 0$, $Ax^\star + b = 0$ and $x^\star$ minimizes $c^T x + \frac{1}{2} x^T D x + \nu|Ax - b|^{1/\nu}$, then $x^\star$ is the optimal solution to (1).*

It is not clear how this theorem can be implemented in practice. A more useful criterion for checking global optimality when Q is negative definite comes from considering the copositivity of Q, as shown by the following theorem.

Theorem 3.5 (Danninger and Bomze (1993)) *Let Q be negative definite, and let $x^\star$ be a feasible point of (1). Let $I(x^\star)$ be the sets of active constraints at $x^\star$, and $u_i = b_i - (Ax^\star)_i > 0$ be the slacks for the inactive constraints. Also define*

$$\begin{aligned}
B_i &= -a_i (Qx^\star + c)^T - (Qx^\star + c)(a_i)^T, \\
Q_0 &= Q, \\
Q_i &= u_i Q - B_i, \quad i = 1, \ldots, m, \\
\Gamma &= \{v \in \Re^n : (Av)_i \leq 0, \ if \ (Ax^\star)_i = b_i\}, \\
\Gamma_0 &= \{v \in \Gamma : (Av)_i \leq 0, \ \forall i \in \{1, \ldots, m\} \setminus I(x^\star)\}, \ and \\
\Gamma_i &= \{v \in \Gamma : (Av)_i \geq 0, \ and \ u_j (Av)_i \geq u_i (Av)_j \quad \forall j \in \{1, \ldots, m\} \setminus I(x^\star)\},
\end{aligned}$$

where $(a_i)^T$ denotes the ith row of A. Then, $x^\star$ is a global solution of (1) if and only if:

(a) *$x^\star$ is a KKT point of (1) and Q_i is Γ_i-copositive for all $i \in \{0, \ldots, m\} \setminus I(x^\star)$.*

(b) *$x^\star$ satisfies $v^T (Qx^\star + c) \leq 0$ for all $v \in \Gamma_0$ and Q_i is Γ_i-copositive for all $i \in \{0, \ldots, m\} \setminus I(x^\star)$.*

In practice, this involves $m - 1$ problems of checking Γ-copositivity. In the worst case, this still has the exponential complexity of checking local optimality.

4. Complexity Issues

What makes the nonconvex quadratic programming problem so hard (and consequently so interesting) to solve? The reason partly can be understood by analyzing the complexity of these problems. Although such an analysis can only provide firm results in worst-case scenarios, it can nevertheless give a good idea of the possibility of developing efficient algorithms for solving the problem.

It has long been known that the general nonconvex nonlinear problem is a tough problem to solve. For the case of quadratic programming, the first result in this direction was presented by Sahni (1974), who showed that for a negative definite matrix Q, the QP is NP-hard. This result was also later proved by Vavasis (1990; 1991) who showed the result by reduction of the QP to the satisfiability problem, and Pardalos (1991), who proved it by reduction to the knapsack feasibility problem.

More recently, several authors have shown that QP is actually in NP, that is, it is an NP-complete problem. Murty (1988) showed this result for special cases of the QP. Vavasis (1990; 1991) proved that the decision version[1] of the QP is NP-complete. This is mainly a theoretical result, since it helps to determine the complexity of QP exactly.

These results are not surprising. In fact, it has also been shown that just checking local optimality for the QP is an NP-hard problem (Murty and Kabadi, 1987; Vavasis, 1991). Pardalos and Schnitger (1988) showed the same result by proving that checking for strict convexity (which is a requirement for checking local optimality as part of the second order necessary conditions) in these problems is NP-hard, with one exception [2]. When this is combined with the fact that there might be a large number of local minima for the problem, the intractability of the general QP becomes obvious. Consider, for example, the following QP:

$$\min - \sum_{i=1}^{n} x_i^2$$

$$-1 \leq x_i \leq 1 \quad i = 1, \ldots, n$$

This problem has a local minimum at each vertex of the feasible region (that is, at $(\pm 1, \pm 1, \ldots, \pm 1)^T$. There are 2^n such points. In general, therefore, the QP cannot be expected to be solved for its global minimum very efficiently. In fact, even finding a local minimum (and proving local optimality of such a solution) may take exponential time.

Most of the above results concern the general case of QPs. What about cases when there are a small number of concave variables? It would be natural to expect that such problems might be easier to solve. However, it was shown by Pardalos and Vavasis (1990) that even when the matrix Q is of rank one with exactly one negative eigenvalue, the problem is NP-hard. However, a larger number of negative eigenvalues does not necessarily make the problem harder to solve. Consider, for example, the following problem:

$$min \ \frac{1}{2} x^T D x + c^T x$$

[1] The decision version for the quadratic programming problem (1) can be posed as follows: Given a number K, does there exist an x such that

$$x^T D x + c^T x \leq K, \quad \text{and}$$
$$Ax \leq b \ .$$

[2] For concave quadratic problems, each local solution will lie at a vertex of the feasible region. For any such vertex, its adjacent vertices can be computed in polynomial time. To verify local optimality at a vertex, it is therefore sufficient to consider the optimality over the simplex defined by the adjacent vertices. This involves solving a linear number of one-dimensional problems minimizing the objective function over each of the adjacent edges. Hence, for the concave case, local optimality can be verified in polynomial time.

$$x \geq 0$$

Suppose that $(n-1)$ of the eigenvalues of D are negative. Hager *et al.* (1991) showed that there must be at least $(n-1)$ active constraints at the optimal solution. In other words, to find the optimal solution, it is sufficient to solve $(n-1)$ different problems, in each case setting $(n-1)$ of the constraints to equalities.

In general, if there are $n-k$ negative eigenvalues, then we need to solve $\frac{n!}{k!(n-k)!}$ independent problems. It can be shown that the total computational time required to solve this problem is proportional to $\frac{k^3 2^k n!}{k!(n-k)!}$. Thus, if k is a fixed constant independent of n, then the computational time is bounded by a polynomial in n. However, if k grows with n (for example, if $k = n/2$) then the problem solution time can grow exponentially with n.

5. Bilinear Programming Problems

One of the earliest applications involving a quadratic programming formulation comes from the class of bimatrix games. These games involve fixed sets of strategies available to two players, each of whom must then select a mixed strategy involving given payoffs. This problem was first investigated by Nash (1951), who introduced the concept of the Nash Equilibrium Point **(NEP)** for these problems. Given probability vectors x and y for the mixed strategies and payoff matrices A and B for the two players, a NEP $(x^\star, y^\star)$ is the solution to the following problem:

$$\begin{aligned} x^{\star T} A y^\star &= \max_{x \in X} \; x^T A y^\star \\ x^{\star T} B y^\star &= \max_{y \in Y} \; x^{\star T} B y \end{aligned} \tag{8}$$

where X and Y are the sets of permissible strategies for the two players, given by

$$\begin{aligned} X &= \left\{ x \in R^{n_1} : e^T x = 1, x \geq 0 \right\} \\ Y &= \left\{ y \in R^{n_2} : e^T y = 1, y \geq 0 \right\} \\ e &= (1, 1, \ldots, 1)^T \quad \text{is the unit vector.} \end{aligned}$$

It can be shown that finding a NEP in the bimatrix game is equivalent to finding the global solution to the following bilinear programming problem:

$$\max_{x,y,x',y'} \; x^T (A+B) y - x' - y'$$

$$\begin{aligned} s.t. \quad Ay - y' e_{n_2} &\leq 0 \\ B^T y - x' e_{n_1} &\leq 0 \\ x \in X, y &\in Y \end{aligned}$$

If constraints are now imposed upon the set of available strategies to each player, this reduces to the classical bilinear programming problem with disjoint constraints. This problem can be written as:

$$\min_{x,y} \; Q(x,y) = c^T x + x^T D y + d^T y$$

$$\begin{aligned} s.t. \quad & X \equiv \{x | A_1 x \ \le \ b_1, \quad x \ge 0\} \\ & Y \equiv \{y | A_2 y \ \le \ b_2, \quad y \ge 0\} \end{aligned} \tag{9}$$

Note that the QP (1) can be reduced to this form by keeping $c = d = \frac{1}{2}c, A_1 = A_2 = A, b_1 = b_2 = b$ and introducing an additional set of constraints $x = y$. In addition, several other problems, such as dynamic Markovian assignment problems, multi-commodity network flow problems and some complementarity problems, that can be written in this form.

The set $X \times Y$ of feasible solutions to this problem is generally a polyhedral convex set (although it may not be bounded) with each vertex of $X \times Y$ being given by an ordered pair of the vertices of X and Y. It can easily be shown that if (9) has an optimal solution, then it has an extreme point optimal solution. This property has been utilized extensively to develop algorithms for solving (9). These approaches may properly be termed as vertex enumeration methods. Czochralska (1982a) proposed an algorithm for solving (9) that utilizes a method for maximizing a linear function over a convex hull of vertices on a convex polyhedral set. In order to eliminate the vertices for which the linear objective does not increase, the algorithm uses a method for ranking the vertices of the set proposed by Murty (1968). The approach was extended by Czochralska (1982b) for solving the standard QP via reduction to a bilinear formulation.

Some of the earliest methods for solving (9) utilized the use of cutting planes. These methods were inspired by work done on concave problems by Tuy (1964) and Ritter (1966). One of the first such methods was proposed by Konno (1976), who proposed solving (9) by using Ritter cuts to eliminate Karush-Kuhn-Tucker points generated during the algorithm. Konno's method could be considered the natural approach to solving bilinear problems of the form (9). The method consists in starting with an arbitrary fixed $x' \in X$, and solving the related linear problem

$$\min_{y} \ Q(x', y) = c^T x' + {x'}^T D y + d^T y$$

$$s.t. \quad Y \equiv \{y | A_2 y \ \le \ b_2, \quad y \ge 0\}$$

The solution of this problem, y' is then used to solve another linear problem:

$$\min_{x} \ Q(x, y') = c^T x + x^T D y' + d^T y'$$

$$s.t. \quad X \equiv \{x | A_1 x \ \le \ b_1, \quad x \ge 0\}$$

This in turn yields a new value for x'. This procedure is repeated until a pair of values $(\overline{x}, \overline{y})$ is found that solves both these linear programs. Such a point can be shown to be a KKT point of (9). A linear cut is then constructed to delete $\overline{x}$ from X and $\overline{y}$ from Y, and the process is repeated using the reduced feasible domain thus obtained. Although Konno (1976) could not theoretically guarantee convergence for the algorithm, no counterexample has been presented to date to show that the method fails.

Vaish and Shetty (1977) proposed using specialized polar cuts to solve bilinear programs, and showed that these cuts are deeper than those proposed by Konno (1976). Their algorithm converges to an ϵ-approximate solution, and in the limit has asymptotic convergence, although exact convergence to the global optimum cannot be guaranteed. Vaish and Shetty (1976) modified the algorithm of Tuy (1964) so as to generate all the facets of the polytope under consideration, and were able to avoid the problems of cycling that the original algorithm of Tuy (1964) exhibited. However, since the number of facets that need to be generated can be large, this approach is not computationally feasible for larger problems. Sherali and Shetty (1980) improved upon the algorithm of Vaish and Shetty (1977) by simultaneously employing negative edge extension cuts and disjunctive face cuts, and were able to prove finite convergence of the overall procedure. This approach is probably the most efficient algorithm available for the bilinear programming problem with disjoint constraint sets.

The bilinear programming problem (9) can also be approached from the viewpoint of linear duality theory. It can be shown (Falk, 1973) that this problem can be written as an equivalent linear max-min problem. Using this formulation, Falk (1973) proposed a branch and bound algorithm to solve (9) (note that this was the first algorithm that could guarantee convergence in a finite number of steps) while Gallo and Ulkucu (1977) developed two algorithms (one based upon the use of cutting planes and the second upon an enlarging polytope approach) that are modifications of the algorithms proposed by Tuy (1964) for concave programming. However, neither of these algorithms has a proof of convergence, and counterexamples exist for which the second algorithm cycles.

A more general formulation for the bilinear programming problem is to consider the case when there are joint constraints in x and y. In such cases, the problem can be written as:

$$\min_{x,y} \; Q(x,y) = c^T x + x^T D y + d^T y$$

$$\begin{aligned} s.t. \quad Ax + By &\leq b, \\ x &\geq 0 \\ y &\geq 0 \end{aligned} \tag{10}$$

It is easy to show that the general quadratic problem (1) can be converted to the form (10). Another class of problems that can be converted to this form is the set of linear complementarity problems. Problems of the form (10) have received considerably less attention in the literature as compared to problem (9). Primarily, this is because for these problems, the solution can no longer be guaranteed to lie at a vertex of the feasible region (although it will still lie on the boundary defined by the linear constraint set).

The first algorithm for solving (10) was proposed by Al-Khayyal and Falk (1983), who considered a branch-and-bound approach to solving the problem, using convex envelopes of the objective function. The algorithm utilizes the fact that the minimum of the bilinear functional over a convex set is the same as the minimum of its convex envelope over the same set. The problem (10) is first converted to the form

$$\min_{x,y} \; d^T y + x^T y'$$

$$\begin{aligned} s.t. \quad Ax + By \; &\leq \; b, \\ y' \; &= \; Dy + c \\ x \; &\geq \; 0 \\ y \; &\geq \; 0 \end{aligned}$$

Given a bilinear term $x_i y_i'$ and a rectangle $R_i = \{(x_i, y_i') : l_i \leq x_i \leq L_i, m_i \leq y_i' \leq M_i\}$, the convex envelope of the term over R_i is given by

$$C_i(x_i, y_i') = \max\{m_i x_i + l_i y_i' - l_i m_i, M_i x_i + L_i y_i' - L_i M_i\}$$

and therefore, the convex envelope of the bilinear objective term $x^T y'$ is given by

$$C(x, y') = \sum_{i=1}^{n} C_i(x_i, y_i')$$

At each iteration of the algorithm, the current rectangular region is partitioned into four nodes. The convex envelope of the objective function is minimized over the subset of the feasible region given by the current partition, and this helps to provide a bound on the global solution. However, the algorithm can only be shown to converge in the limit. For specific instances, though, the algorithm can be shown to have finite convergence. Al-Khayyal (1990) presented some improvements that enable the direct application of the algorithm for more general problems without resorting to increasing the number of variables and constraints.

Recently, Sherali and Alameddine (1992) proposed a reformulation-linearization algorithm for solving the bilinear problem of the form (10). Their approach consists in generating valid quadratic constraints by using pairwise products of the linear inequalities (or products of inequalities with variables) and subsequently linearizing the quadratic constraints by defining new variables. The bilinear objective term is similarly linearized. This results in a lower bounding linear program. The resulting linear program generates a lower bound that theoretically dominates the bound produced by the algorithm of Al-Khayyal and Falk (1983), and in practice provides tighter bounds. Moreover, the partitioning process for the algorithm can be modified to generate only two node subproblems instead of four. Computational results for both these approaches are provided only for small size problems (5-10 variables).

When the matrix D has very low rank (less than 3), problem (10) reduces to the special form of minimizing a linear multiplicative functional subject to a linear constraint set. For this class of problems, several efficient approaches have been proposed for their solution (Konno and Kuno, 1989a; Konno and Kuno, 1989b; Konno *et al.*, 1990; Konno and Yajima, 1990). These approaches use parametric simplex algorithms to solve the problem, and have computational performances comparable to similar size linear programming problems.

5.1. Transformation of Quadratic Problems to Bilinear Problems

The importance of bilinear programming comes not only from the direct applications, but also because the other classes of quadratic problems (namely indefinite and concave quadratic problems) can be transformed into bilinear formulations. The easiest way to achieve this is by introducing new variables and constraints (Floudas *et al.*, 1989; Aggarwal and Floudas, 1990). Recently, Hansen and Jaumard (1992) have proposed graph theoretic results and an algorithm for the efficient bilinearization of general quadratic problems. The algorithm provides the set of new variables that must be introduced in order to convert the problem into bilinear form. The bilinearization can be achieved with the objective of minimizing either the number of x (or y) variables or the number of new variables that must be introduced in order to make the problem completely bilinear. Moreover, given a bilinear problem with variable subsets x and y, the algorithm can also be used to identify any changes in these subsets that will result in a smaller number of either the x or the y variables. Liu and Floudas (1993a; 1993b) have shown using perturbation theory that a large class of smooth mathematical programming problems can be converted to biconvex and bilinear form.

6. Concave Problems

Concave quadratic problems arise often in fixed charge problems and problems involving economies of scale. Certain aspects of VLSI chip design can also be formulated as concave QPs (Watanabe, 1984). In addition, several other classes of optimization problems can be reformulated as concave QPs, the classical quadratic assignment problem, for example. Another example of such problems is the class of 0-1 integer programming problems, which can be formulated as concave QP with linear constraints (Raghavachari, 1969). Under certain conditions on the boundedness of the objective function and with Lipschitzian continuity, the general nonlinear nonconvex mixed integer program can be reduced to a concave program, although not necessarily quadratic (Giannessi and Niccolucci, 1976). The general bilinear programming problem with a disjoint constraint set is equivalent to concave minimization (Konno, 1976). Frieze (1974) reduced the 3-dimensional assignment problem to a bilinear formulation, and then to a special concave form.

Concave quadratic problems are by far the simplest of QP, although not necessarily the easiest in terms of solution of QP. As mentioned earlier, all solutions (local or global) to the concave QP lie at some vertex of the feasible region[3]. A considerable amount of research has therefore focused on utilizing this property in order to solve the concave QP.

6.1. Extreme Point Ranking Methods

Because the solution of the concave quadratic programming problem lies at a vertex of the polytope, a natural method to solve this problem is by complete enumeration of the extreme points.

[3] On the face of it, this should make these problems relatively easy to solve; however, it is very easy to construct concave QPs where each vertex of the feasible set is a local minimum. Hence, in the worst case, the enumeration of all these vertices can take an exponential amount of time.

The basic idea in these methods is to use an extreme point ranking. In other words, the vertices of the polytope defining the feasible region are ranked in order of importance regarding the global solution. Starting from one of the vertices of the polytope, the nearby vertices are ranked using an extreme point approach. This provides a new vertex to move to, and the process continues until no adjacent vertices can be found with a decreasing objective function value. At each step, usually a linear programming problem is solved to provide a bound on the global optimum. Cabot and Francis (1970) proposed such an approach where a LP is solved (with a linear underestimator of the concave function being used for the objective function of the LP), utilizing the extreme point ranking approach proposed by Murty (1968). Carillo (1977) proposed the use of nonlinear programming duality theory and a lexicographic ordering of the basic feasible solutions (that is, the vertices) in order to obtain lower bounds on the objective function, and used this as the basis for a finitely convergent global optimization algorithm for concave problems.

Several techniques have been proposed for ranking and enumeration of all the vertices of a polyhedral set. Surveys and comparisons of these methods can be found in Matheiss and Rubin (1980) or Dyer (1983). The computational effectiveness of extreme point ranking algorithms is discussed in McKeown (1986).

In general, extreme point ranking methods are not very attractive, because in the worst case, these approaches might degenerate to a complete inspection of all vertices. Consequently, these approaches are not computationally feasible for large problems.

6.2. Cutting Plane Methods

Over the years, a large number of approaches have been proposed for solving concave quadratic programming problems; however, most of these approaches (except for a few recent ones) can be traced back to a seminal paper by Tuy (1964). In that work, Tuy considered the more general case of minimization of an arbitrary concave function (not necessarily quadratic) under linear constraints. His approach was based upon the existence of vertex solutions for these problems. In addition, Tuy recognized the following property of concave functions:

Property 6.1 *With every concave function $f(x)$ defined on a convex set, there exists a concave extension $F(x)$ such that $F(x) \geq \overline{F}(x) \quad \forall x \in \Re^n$ and any other concave extension $\overline{F}(x)$.*

This is a useful property in that for the region outside the polytope, the values of the concave extension can be utilized, thereby restricting the number of vertices of the polytope that need to be considered.

Tuy's basic approach involved starting from a specific vertex of the polytope. The edges of the polyhedron issuing from this vertex are used to define a cone that contains the feasible region. Tuy proposed the use of cuts to successively reduce this cone. These cuts are essentially used to eliminate parts of the feasible region from further consideration. At each step, an auxiliary subproblem is solved in the sub-cone, that gives rise to a new vertex point, and possibly a better candidate for the global solution. The advantage of such an approach is that at each iteration, the auxiliary problem has the same set of constraints (that is, the constraints corresponding to the original polytope) and differ only in the objective function. Moreover, the objective function at each step can be obtained from the previous iteration by a simple linear transformation and column replacement. It should be noted that this method relies on all the vertices being nondegenerate. In the case of degeneracy, a suitable perturbation method needs to be used.

A similar approach using cutting planes to reduce the feasible region was proposed by Ritter (1966). However, it was shown by Zwart (1973) that both these proposed approaches can be nonconvergent. He cited two examples to show that the methods either fail to converge due to cycling, or need an infinite sequence of cutting planes. The reason for the cycling behavior as well as nonconvergence of these approaches lies in the fact that although the approaches generate cones during the algorithm, they failed to explicitly incorporate these cones into the remaining steps. This is essential to avoid the reemergence of vertices that have already been considered. This difficulty with the approaches was recognized by Zwart (1974), who proposed a modified ϵ-convergent approach where, at each iteration, constraints are added to ensure that the solution of the LP at each iteration is contained in the cone with the vertex at the current local minimum and a perturbed set of extreme rays coincident at that vertex. Similar algorithms have been proposed by Bali (1973) and Jacobsen (1981). However, it has been shown recently (Tuy, 1991) that for $\epsilon = 0$, Zwart's algorithm is essentially the same as Bali's algorithm. Moreover, the convergence of Bali's algorithm is still open to question, depending upon a separation property that may not hold for the general case.

Several other variants to Tuy's algorithm have been proposed (Glover, 1973; Krynski, 1979; Bullatov and Kasinskaya, 1982). In general, however, there are several difficulties associated with the Tuy-cut approaches. These come from the fact that to guarantee convergence, the cuts of the feasible region need to be incorporated at each iteration. Moreover, these cuts can generate new vertices that must be accounted for in a way that they do not multiply indefinitely. However, it is possible to incorporate a branch and bound scheme to monitor a sequence of subproblems generated from the local solutions (Thoai and Tuy, 1980). Tuy (1991) has also proposed a normal conical algorithm that avoids the exhaustive subdivision process of his earlier algorithm.

6.3. Convex Envelopes

A different approach to solving the concave quadratic problem can be developed from the use of the convex envelope of the concave function. This concept was first developed by Kleibohm (1967), and can be stated as the following theorem:

Theorem 6.1 (Falk and Hoffman (1976)) *If $F(x)$ is the convex envelope of a concave function $f(x)$ taken over a convex domain S, then any point that globally minimizes $f(x)$ over S also minimizes $F(x)$ over S.*

Proof. The proof of this theorem can be deduced from the fact that the global minimum of the concave function lies at a vertex. See Falk and Hoffman (1976) for more details.

The applicability of this theorem is not immediately obvious. This is because the convex envelope of a function is generally extremely hard to generate (often, this is as tough as solving the original problem). However, for the case of the quadratic objective function with a linear constraint set, it is quite easy to generate the convex envelope, as shown by the following theorem[4]:

Theorem 6.2 *Let $v^0, v^1, \ldots, v^k$ be the vertices of a bounded linear polyhedron P. The convex envelope $F(x)$ of a concave function $f(x)$ defined over P can be expressed as*

$$F(x) = \min_{\alpha_0, \alpha_1, \ldots, \alpha_k} \sum_{i=0}^{k} \alpha_i f(v^i)$$

$$\begin{aligned} s.t. \quad \sum_{i=0}^{k} \alpha_i v^i &= x \\ \sum_{i=0}^{k} \alpha_i &= 1 \\ \alpha_i &\geq 0 \qquad i = 0, 1, \ldots, k \end{aligned}$$

Proof. See Falk and Hoffman (1976).

For the case of concave minimization, this property was utilized by Falk and Hoffman. Their method uses linear underestimating functions (developed using the convex envelopes) for the original concave function over different regions of the feasible domain, and does not involve any cuts of the feasible region. The algorithm converges finitely. However, a disadvantage of the approach is that in order to approximate the convex envelope implicitly, the size of the linear subproblems solved grows in size from one iteration to the next. However, the maximum number of rows in these subproblems (and also the number of such subproblems solved) is limited by the original number of constraints. Hence, the growth of the subproblems is mainly in the addition of new columns.

6.4. Reduction to Bilinear Programming

The concave quadratic program can also be solved by reduction to (and solution of) an associated bilinear programming problem. This was first shown by Konno (1976), who proved that when $Q(x)$ in (1) is negative definite (with the constraints being equalities) then that problem is equivalent to the following bilinear problem:

$$\min_{x,y} c^T x + d^T y + x^T Q y$$

[4] The approach of using convex envelopes of the nonconvex function has also been proposed by several other authors. Originally, Falk and Soland (1969) proposed such an approach to solve the general nonconvex separable programming problem. The branch and bound type approach solves a series of subproblems in which a linear or convex objective function is minimized over successive partitions of the feasible domain. This approach was also used by Falk (1972), who replaced the original function with its piecewise linear convex envelope over different regions, again employing a branch-and-bound approach to minimize these functions and derive new underestimating functions in the case that convergence is not yet achieved.

$$
\begin{aligned}
s.t. \quad Ax &= b \\
Ay &= b \\
x,\ y &\geq 0
\end{aligned} \tag{11}
$$

Then, (11) and (1) are equivalent in the following sense: If $x^\star$ is an optimal solution of (1), then $(x^\star, x^\star)$ solves (11). Conversely, if $(x^\star, y^\star)$ is optimal for (11), then both $x^\star$ and $y^\star$ are optimal for (1). This property enables the application of all the bilinear programming algorithms discussed in Section 5. Moreover, it also enables the development of specialized bilinear programming algorithms for application to the concave quadratic problem.

Konno (1976) used this approach to propose a cutting plane algorithm for bilinear programming, which he then used to solve the concave quadratic problem. See also Konno (1980) and Konno (1981) for other problems and methods of solution using a similar approach. Similarly, Czochralska (1982b) used his previous approach for bilinear problems (Czochralska, 1982a) to address the nonconvex quadratic problem. The resulting bilinear problem is greatly simplified because of the problem structure, and also since the verification of necessary and sufficient conditions for the existence of an optimal solution is reduced to the solution of an LP in this case.

6.5. Solution of Large Concave QPs With Linear Terms

Because of the existence of a large number of local solutions, it cannot be expected that the concave QP could be solved efficiently for anything other than small size problems. Computational results from the literature (Heising-Goodman, 1981; Rosen, 1983) point this out, indicating that it would be impractical to solve problems where the number of variables appearing in quadratic terms is over 50.

Until recently, all approaches for concave QPs had essentially treated the problem assuming that all the variables may appear nonlinearly in the problem formulation. However, as observed in numerous problems of practical interest, a large number of variables appear linearly in the mathematical formulation. The number of such variables usually far exceeds the number of quadratic variables. Hence, it is natural to attempt to take advantage of the fact that only a small fraction of the variables appear nonlinearly. In such cases, a good solver such as MINOS (Murtagh and Saunders, 1988) that treats the linear variables in a different way than those variables appearing nonlinearly, can be used very efficiently. A major portion of the research in recent years in this area has consequently focused on problems of the form:

$$
\min Q(x,y) = c^T x + \frac{1}{2} x^T D x + d^T y
$$

$$
\begin{aligned}
s.t. \quad Ax + By &\leq b \\
x,\ y &\geq 0
\end{aligned} \tag{12}
$$

where $y \in \Re^k$ is a new vector of variables with $k \gg n$.

A common approach to solving (12) is to obtain upper and lower bounds on the optimal solution through linear programming problems and iterate between these bounds until convergence is achieved. An easy formulation for the upper bound problem involves solving a linear programming problem where the terms in x are ignored, that is, the following problem:

$$\min\ d^T y$$

$$\begin{aligned} s.t. \quad Ax + By &\leq b \\ x,\ y &\geq 0 \end{aligned}$$

The solution of this problem will give a vertex (x_l, y_l) and an upper bound $Q(x_l, y_l)$ on the globally optimal solution $Q^\star$ of (12). When the linear terms dominate in the objective function of (12), then the global minimum of (12) is given by $Q(x_l, y_l)$ and these approaches terminate immediately.

Most approaches for large scale concave quadratic minimization (with most of the variables being linear) are based upon the original approach proposed by Rosen (1983). The basic idea of this approach (which was proposed for ordinary concave global minimization) is to take advantage of the ellipsoid like level surfaces of the objective function to find a "good" initial vertex and to eliminate a rectangular domain (enclosed in a level surface) from further consideration. The approaches start by finding the global maximum point $\overline{x}$ of the objective function (in (1)) in the given feasible domain. Then, a set of orthogonal vectors $u_i, i = 1, 2, \ldots, n$ at $\overline{x}$ (these vectors could be the eigenvectors of the Hessian of $Q(x)$ at $\overline{x}$) are obtained. Using u_i, a multiple cost row linear program with $2n$ rows is solved:

$$\max_x\ \pm u_i^T x$$

$$x \in D$$

The vertex solution for x corresponding to the minimum of all of these problems is then a candidate for the global minimum (or at least gives a good upper bound on the global minimum).

Next, a rectangular domain $R \in \Re^n$ is constructed with the level set using the minimum objective value obtained from the multiple cost row programs. For the concave quadratic case, this can be done by using the eigenvalues and eigenvectors of the Hessian of the objective function in such a way that the value of $Q(x)$ at each of the 2^n vertices of R is the same as the minimum objective function value. This involves using the level ellipsoid given by $Q(x) = \overline{Q}$ where $\overline{Q}$ is the minimum value from the multiple cost row linear programs. The main effect of such a partition is that it can be used to exclude R from all further consideration. Moreover, the remaining part of the feasible region is partitioned into at most $2n$ subregions, most of which might actually be empty. Then, a method such as the approach of Falk and Hoffman (1976) can be used to solve the concave problem in that subdomain.

For the case of large scale concave problems of the form (12), the same idea has been extended by several authors. In all the approaches, as a first step, a multiple cost row linear program is solved in order to determine R_x, a subdomain of $\Re^n$ which contains the projection of the feasible region in the space of the x (nonlinear) variables. Linear underestimating functions to the original concave function are then constructed and minimized, giving initial vertices to start the algorithms from. After that, the region R_x is partitioned and a branch-and-bound scheme is followed that successively eliminates partitions and obtains tighter bounds.

Several methods have been suggested for the partitioning of R_x. Kalantari (1984) proposes bisecting R_x using a cut parallel to one of its faces. For each of the two rectangular subdomains, a linear underestimator is found, and an LP is solved for each region. This procedure is repeated in a branch and bound scheme with improved bounds at each step. Zilverberg (1983) proposes partitioning R_x into 2n pyramids, where each pyramid has as its base one of the $2n$ faces of R_x and the predetermined vertex as the apex, with the union of all these pyramids giving the original partition R_x. Kalantari and Rosen (1987) proposed a similar branch and bound algorithm using linear convex envelopes.

These methods are ϵ-approximate, since they successively partition the domain and may therefore end up close to, but not exactly, at a vertex of the original feasible region. A good comparison of these methods is given in Rosen (1984), in which it is shown that the time taken to obtain an ϵ-approximate solution increases linearly with the number of linear variables, and as $O(m^2)$ with the number of constraints. This indicates that when the number of nonlinearly appearing variables is small, these approaches can be reasonably effective.

6.6. Reduction To Separable Form

It is possible to reduce the concave quadratic problem to a separable quadratic form. One way to achieve this was proposed in Rosen and Pardalos (1986), as follows[5]. Given the matrix Q, compute the real eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ of $-Q$ and the corresponding eigenvectors $[u_1, \ldots, u_n]$ so that $Q = UDU^T, U = [u_1, \ldots, u_n], U^T U = I$ and $D = diag[\lambda_1, \ldots, \lambda_n]$ (the eigenvalues and eigenvectors can be easily computed because $-Q$ is a positive semidefinite matrix for concave QP). Once these values are computed, then a multiple-cost row linear program with $2n$ rows is solved. Using the solutions of these programs, the QP can then be reformulated in a separable form:

$$\begin{aligned} \min \ & \sum_{i=1}^{n} q_i(x_i) \\ s.t. \ & A'x \le b' \\ & x \ge 0 \end{aligned}$$

where $q_i(x_i) = c_i x_i - \frac{1}{2}\lambda_i x_i^2$ and A' and b' are produced from A and b through linear transformations using U.

Assuming that the feasible region is bounded, each concave separable function $q_i(x_i)$ can be easily underestimated over the bounds by linear functions. Usually, these underestimators are generated so that they meet $q_i(x_i)$ at the bounds of x_i. For example, the function $\gamma_i = (c_i - \frac{1}{2}\lambda_i\beta_i)x_i$ (where β_i is the upper bound on x_i) is a good underestimator of $q_i(x_i)$ over the region $x_i = [0, \beta_i]$.

Rosen and Pardalos (1986) utilized this approach to solve large-scale concave programs. The first step is to obtain a bound on the relative error of the approx-

[5] This is not the only method for converting the concave QP to a separable form. However, it has the advantage that it preserves the eigenstructure of the quadratic form. This can be useful in error analysis of algorithms that may address the original or the separable form.

imation used (where the relative error is the difference in the approximation from the original objective function). In general, the relative error is guaranteed to be at most 0.25. If the approximate solution obtained is not satisfactory, then a single mixed 0-1 problem is solved. This problem is formulated by a single piecewise linear underestimator of the separable objective function.

More recently, Phillips and Rosen (1988) proposed a parallel algorithm for large scale concave quadratic problems. This algorithm also first reduces the concave objective function to a separable form. At each iteration, it combines a heuristic step that attempts to eliminate part of the feasible region, with a parallel implementation of the $2n$ multiple cost row linear programs. The algorithm has a guaranteed convergence in a finite number of steps to a ϵ-approximate solution, and the number of subproblems that need to be solved is bounded above by $(4n/\epsilon)^{n/2}$. The algorithm performs quite efficiently for large problems, with the maximum size of 50 nonlinear and 400 linear variables being considered. The efficiency of the algorithm primarily comes from the effectiveness of the heuristic step in practice, and the parallel solution of the multiple-cost row linear programs.

7. Indefinite Quadratic Problems

The case when the matrix D has eigenvalues of mixed sign presents the toughest problems of quadratic optimization. While many algorithms have been developed for the more particular cases of bilinear and concave quadratic problems, few approaches have been proposed for globally optimizing (1) for the case of the indefinite quadratic objective. Most efforts at solving this difficult class of problems have focused on first reducing the indefinite quadratic problem to either a bilinear or a concave minimization problem. However, there are a few algorithms for directly solving this class, and we discuss them here.

7.1. Quadratic Problems With Box Constraints

When the only constraints on the variables are lower and upper bounds, then (1) reduces to the following form, which is probably the simplest problem of global nonconvex minimization:

$$\begin{aligned} \min_{x} \ & Q(x) = c^T x + \frac{1}{2} x^T D x \\ s.t. \quad & x^L \leq x \leq x^U \end{aligned} \tag{13}$$

It may be assumed without loss of generality that $x^L = 0$ and $x^U = 1$ (if necessary, a linear transformation can be used to convert the problem to this form). This gives a direct relationship of the QP with box constraints to one of the fundamental problems of combinatorial optimization, namely minimizing a quadratic function of 0-1 variables. As a special case, if the matrix D has zero diagonal entries, then it can be shown (Rosenberg, 1972) that if (13) has a fractional solution, then there also exists an optimal integer solution with the same objective function value, and the integer solution can be easily derived from the fractional one. Moreover, problems of the form (13) appear frequently in the solution of partial differential equations, discretized continuous time optimal control problems, and linear least squares problems, and also as subproblems in successive quadratic programming methods. It is clear, therefore, that the solution of (13) has a special importance.

Due to the presence of only bound constraints in the problem, it is possible to develop an alternate simplified set of necessary conditions and sufficient conditions for local optimality for (13). Let D_{ii} represent the i^{th} diagonal entry of D, D_i represent the i^{th} row of D, and let

$$g_i(x) = D_i x + c_i$$

be the partial derivative of $Q(x)$ with respect to x_i. Then, the following theorem (Hansen *et al.*, 1991) holds[6]:

Theorem 7.1 (Necessary Conditions for optimality for (13)) *If a point $x^\star$ is a local minimum of (13), then*

(a) *For all i such that $D_{ii} \geq 0$,*

$$\begin{aligned} &g_i(x^\star) > 0 \quad \textit{and} \quad x_i^\star = 0, \quad \textit{or} \\ &g_i(x^\star) = 0 \quad \textit{and} \quad 0 \leq x_i^\star \leq 1, \quad \textit{or} \\ &g_i(x^\star) < 0 \quad \textit{and} \quad x_i^\star = 1. \end{aligned}$$

(b) *For all i such that $D_{ii} < 0$,*

$$\begin{aligned} &g_i(x^\star) > 0 \quad \textit{and} \quad x_i^\star = 0, \quad \textit{or} \\ &g_i(x^\star) < 0 \quad \textit{and} \quad x_i^\star = 1. \end{aligned}$$

Proof. See Propositions 1 and 2 of Hansen *et al.* (1991).

If the point $x^\star$ is nondegenerate (that is, there is no component $x_i^\star$ of $x^\star$ such that it lies at either the upper or lower bound and also has the gradient $g_i(x)$ equal to zero), then the above conditions are also sufficient to ensure local optimality. If $x^\star$ is degenerate, then even determining whether it is a local solution is NP-hard (Murty and Kabadi, 1987).

Most algorithms for solving (13) have used the conditions in Theorem 7.1 to develop some kind of active set strategy approach. One such method can be found in Hansen *et al.* (1991). In their branch-and-bound method, branching is done according to the sign of the first order derivatives $g_i(x)$. Branching is possible either dichotomously ($g_i(x) < 0$ or $g_i(x) \geq 0$) or trichotomously ($g_i(x) < 0$, $g_i(x) = 0$ or $g_i(x) < 0$). For example, if $g_i(x) = 0$, then this can be used to eliminate one of the variables (not necessarily x_i). Branching can also be done based upon whether a particular variable lies at a bound or not. The use of these conditions result in auxiliary subproblems that are solved at each node. They also provide various optimality tests at each node of the tree.

The above algorithm is (to the best of our knowledge) the only approach specifically aimed at solving (13). Most other approaches for solving (13) have concentrated

[6] Similar necessary conditions and sufficient conditions for $x^\star$ to be a local minimum can be found in Coleman and Hulbert (1989).

on finding local solutions in the hope that they would be global. Given this circumstance, it is important to be able to check if a particular local minimum is the global one for this problem. Fortunately, it is possible to deduce a sufficient condition for this to occur.

Suppose that $\overline{x}$ is a nondegenerate local solution to (13). Let λ and μ be the corresponding vectors of multipliers for the lower and upper bound constraints in (13) respectively. Let $y_i = \lambda_i - \mu_i,\ i = 1, \ldots, n$. Given the solution $\overline{x}$, we may assume without loss of generality that

$$\begin{aligned} \overline{x}_i &= x_i^U \ \textit{or}\ x_i^L, \quad i = 1, \ldots, m \\ x_i^L < \overline{x}_i &< x_i^U, \quad i = m+1, \ldots, n \end{aligned}$$

Then, the matrix D can be partitioned as

$$D = \begin{pmatrix} D_A & D_B \\ D_B^T & D_C \end{pmatrix}$$

where $D_A \in \Re^{m \times m}$, $D_B \in \Re^{m \times (n-m)}$, and $D_C \in \Re^{(n-m)\times(n-m)}$. Then, let

$$F = D_A - D_B D_C^{-1} D_B^T$$

and H be a diagonal matrix with positive entries constructed so that $F+H$ is positive semidefinite. If such a matrix can be constructed, then the following theorem can be proved:

Theorem 7.2 *Let $a_i = H_{ii}/2y_i,\ i = 1, \ldots, m$. Then,*

$$(D\overline{x} + c)^T (x - \overline{x}) \geq 0$$

and

$$Q(x) - Q(\overline{x}) \geq (D\overline{x} + c)^T (x - \overline{x})(1 - \sum_{i=1}^{m} a_i (x_i - \overline{x}_i))$$

If

$$\alpha = \sum_{i=1}^{m} |a_i (x_i^U - x_i^L)| \leq 1$$

then $\overline{x}$ is a global minimum of (13). If

$$\alpha = \sum_{i=1}^{m} |a_i (x_i^U - x_i^L)| < 1$$

then $\overline{x}$ is a unique global minimum of (13).

Proof. See Han *et al.* (1992).

Various other active set methods have been proposed for solving (13) to local optimality. Of these, the most efficient is the approach of Coleman and Hulbert (1989) for problems with large sparse matrices. At the start of the algorithm, the matrix D is subject to Cholesky factorization. At each step of the factorization, if a particular diagonal entry becomes positive, then that variable is taken out of the matrix and set to zero. If the diagonal entry is nonpositive, then the variable is set to its lower or upper bound depending on the solution of a trivial quadratic problem in one variable. At the end of the factorization, a free set of variables (that is, variables that are at neither bound), the submatrix of D corresponding to this free set, and starting point for the algorithm is obtained. Starting from this point, a Newton direction is computed and used to switch variables between the free and bound sets until a constrained stationary point of the problem is obtained. Then, a check is done to see if the constrained point is locally optimal. In the event of non-optimality, a new variable that is currently at one of its bounds is switched into the free set, and the process is repeated. The algorithm is guaranteed to converge to the solution only for convex quadratic problems. However, the implementation of this algorithm is probably among the most efficient, and for a large number of indefinite quadratic problems involving very big problem sizes, the algorithm performs very well.

One of the problems with active set strategies for solving (13) is that most such algorithms (like the above one) only allow for the addition or dropping of one constraint per search direction computation. For large problems, this could be prohibitively slow. This can be overcome by using projection steps that allow for several constraints being dropped without much overhead. However, projected methods usually drop constraints only after having solved some reduced problem to optimality, and therefore might end up doing a significant amount of work in a restricted subspace of the original problem. The best method for solving (13) should therefore seek to balance between these two extremes. One such method, which uses projected gradient directions when the active constraint set changes, and conjugate gradient directions otherwise, can be found in Dembo and Tulowitzki (1984).

When (13) contains an additional ellipsoid constraint in the variables, then it can be solved in polynomial time (Ye, 1990). This has been used by Han *et al.* (1990; 1992) to develop an interior-point algorithm for solving (13). The algorithm starts from a point x^0 that is strictly in the interior of the feasible domain. Then, they construct an ellipsoid with center at x^0 that is inscribed in the feasible domain, and solve the quadratic problem with this constraint. The solution of this problem yields a new point x^1. The procedure is repeated, yielding a sequence of points $x^0, x^1, \ldots$. After a sufficient number of steps, a stationary point of (13) is obtained. Although the procedure cannot guarantee convergence to the global solution, it seems to perform well on randomly generated test problems with known global solutions, finding the global solution in most instances.

It is also possible to solve (13) by conversion to an unconstrained minimization problem. Consider the following problem, equivalent to (13):

$$\begin{aligned} \min_x \; & Q(x) = c^T x + \frac{1}{2} x^T D x \\ s.t. \quad & x_i - X_i^U + y_i^2 = 0, \qquad i = 1, \ldots, n \qquad (14) \\ & -x_i + X_i^L + z_i^2 = 0, \qquad i = 1, \ldots, n \end{aligned}$$

$$(15)$$

An augmented Lagrange function for this problem, with the multipliers for the two sets of constraints replaced by the approximation functions $u(x)$ and $v(x)$, can then be written as follows (Grippo and Lucidi, 1991):

$$\begin{aligned} L(x, y, z, \epsilon) &= Q(x) + \sum_{i=1}^{n} \left[u_i(x)(x_i - x_i^U + y_i^2) + v_i(x)(-x_i + x_i^L + z_i^2) \right] \\ &\quad + \frac{1}{\epsilon} \left[\frac{(x_i - x_i^U + y_i^2)^2}{a_i(x)} + \frac{(-x_i + x_i^L + z_i^2)^2}{b_i(x)} \right] \end{aligned}$$

where the bound constraints have been added as penalty terms with ϵ as the penalty parameter, and $a_i(x)$ and $b_i(x)$ are barrier weighting functions for the constraints. The following theorem establishes a direct relationship of the minimum of this penalty function and the solution of (13):

Theorem 7.3 *Let*

$$P(x, \epsilon) = \min_{y,z} L(x, y, z, \epsilon)$$

Then,

(i) *There exists a computable bound* $\epsilon^\star$ *for the penalty parameter such that for all* $\epsilon \in (0, \epsilon^\star]$, *if* $x^\star$ *is a point satisfying* $\nabla P(x, \epsilon) = 0$, *the corresponding triple* $(x^\star, u(x^\star), v(x^\star))$ *is a Karush-Kuhn-Tucker point for (13).*

(ii) *For all* $\epsilon \in (0, \epsilon^\star]$, *any global minimum point of (13) is a global minimum point of* $P(x, \epsilon)$ *and vice-versa.*

Proof. See Propositions 3.3 and 4.1 of Grippo and Lucidi (1991).

From a computational point of view, this equivalent formulation permits the use of standard unconstrained minimization algorithms for differentiable functions to solve (13). In particular, Newton-type algorithms can be defined that make use of consistent approximations to the Newton's direction for the penalty function. Grippo and Lucidi (1991) present one such algorithm that converges in a finite number of iterations to a solution of the constrained problem. Under suitable assumptions, it can also be shown that there exists some neighborhood around the solution where the algorithm converges in a single step.

7.2. Decomposition Techniques

Over the years, several decomposition based approaches have been proposed for solving (1). Although these approaches are usually for the more general global optimization problems, the nature of the quadratic problem makes them quite attractive as a method of solution. In particular, decomposition approaches can circumvent one of the main difficulties associated with the solution of (1), namely problem size. Decomposition enables the solution of the problem through smaller, and possible more tractable, subproblems, whereas the original problem might be of a size that prohibits its direct solution.

Most of the decomposition approaches are based upon the so-called Benders' cuts, which were proposed by Benders (1962) for the solution of mixed integer 0-1 problems. The approach was subsequently generalized to the continuous biconvex case by Geoffrion (1972). To better understand this approach, consider the following special indefinite quadratic problem, first tackled by Kough (1979):

$$\min_{x,y} x^T x - y^T y \tag{16}$$

$$s.t. \quad Ax + By \le b$$

A natural decomposition for this problem can be derived from the separability of the objective function. If the y variables are fixed to y^k, then the following problem (variously called the subsidiary problem or primal problem) is obtained:

$$\min_{x} x^T x - {y^k}^T y^k \tag{17}$$

$$s.t. \quad Ax \le b - By^k$$

This problem provides an upper bound on the global solution of (16), and also gives multipliers μ^k for the constraints. Given the solution x^k to this problem, then the projection $v(y)$ of (16) in the space of y can be defined by

$$v(y^k) = {x^k}^T x^k - {y^k}^T y^k$$

It is then obvious that the solution to (16) is equivalent to

$$\min_{x,y} \; v(y)$$

$$s.t. \quad y \in \{y : Ax + By \le b \quad \text{for some } x\}$$

The solution of this problem is nontrivial, since normally $v(y)$ is an implicitly defined function. However, the solution of (17) can be used to provide a Benders' cut $v^k(y)$ that can be used to approximate $v(y)$:

$$v^k(y) = {x^k}^T x^k - y^T y + {\mu^k}^T (Ax + By - b) \tag{18}$$

If the primal problem were infeasible, then an alternate problem with a artificial objective can be solved and the corresponding multipliers λ^k can be used to generate a *feasibility* cut:

$$g^k(y) = {\lambda^k}^T (Ax + By - b)$$

The use of these two types of cuts leads to a "master" (or dual) problem which provides a lower bound on the global solution:

$$\min_{y} v$$

$$\begin{aligned} s.t. \quad v &\ge v^k(y) \\ 0 &\ge g^k(y) \end{aligned}$$

The typical generalized Benders' approach then involves iterating between the master and primal problems until convergence is reached.

It is the solution of the master problem itself that is the major hurdle in this approach, since this problem can be considered as the minimization of a piecewise concave quadratic function over a linear set. For the separable case above, Kough (1979) proposed solving a set of linear subproblems, each involving the minimization of one of the Benders' cuts. The resulting algorithm provides successively tighter lower bounds, and has finite convergence to an ϵ-approximate solution. However, the number of subproblems to be solved increases as the iterations progress, so that the progress of the algorithm becomes very slow at later iterations. Kough (1979) also proposed a method for developing exact Benders' cuts for this problem by representing the solution of the primal problem x^k in terms of the y variables. More recently, Floudas *et al.* (1989) showed that generalized Benders' decomposition was applicable to a wider class of problems (including the general quadratic problem, polynomial function problems and mixed integer nonlinear programming problems) providing methods for converting these problems to a form to which generalized Benders' decomposition was applicable. They also provided a global search scheme, which although not guaranteed to converge to the global solution, nevertheless proved computationally effective on a large number of problems. This approach has been used by Aggarwal and Floudas (1990) for solving quadratic and mixed 0-1 quadratic problems.

A related approach for solving quadratic problems is the GOP algorithm proposed by Floudas and Visweswaran (1990; 1993), which guarantees convergence (to an ϵ-global solution) for a large class of problems that includes the quadratic problem as a special case. This algorithm can also handle quadratically constrained problems, and it is discussed in more detail in Section 8.1.

7.3. Large-Scale Indefinite Problems

As with the concave quadratic case, it often happens that the nonconvexity in the objective function is due to a small number of variables. This can happen either because (i) only a small number of eigenvalues of D are negative, or (ii) the number of variables appearing in the quadratic terms in the objective function are a small fraction of the total number of variables. For such problems, it is feasible to solve larger problems than would normally be the case.

Consider the case when the matrix D in (1) has relatively few negative eigenvalues. Without loss of generality, we may assume that these eigenvalues are given as

$$-\lambda_1, \ldots, -\lambda_m, \lambda_{m+1}, \ldots, \lambda_n$$

where $\lambda_i > 0$ for $i = 1, \ldots, m$ and $\lambda_i \geq 0$ for $i = m+1, \ldots, n$. We are then concerned with the case where m is relatively small compared to n. In such a case, it can be useful to reduce the objective function in (1) to a separable form (Pardalos *et al.*, 1987). As for the concave case (Section 6.6), a multiple cost-row linear program of the form

$$\min_{x \in \Omega} \pm u^T x$$

can be solved, where Ω is the feasible region for (1), and u_i are the orthogonal eigenvectors of Q. Using the solutions of these linear programs, (1) can be reformulated as

$$\begin{aligned}
\min_x Q(x) &= Q_1(x) + Q_2(x) \\
Ax &\le b \\
x &\ge 0 \\
Q_1(x) &= \sum_{i=1}^{m} \theta_i(x_i) = \sum_{i=1}^{m} \left(c_i x_i - \frac{1}{2}\lambda_i x_i^2 \right) \\
Q_2(x) &= \sum_{i=m+1}^{n} \theta_i(x_i) = \sum_{i=m+1}^{n} \left(c_i x_i + \frac{1}{2}\lambda_i x_i^2 \right)
\end{aligned} \tag{19}$$

Thus, the objective function has a concave part $Q_1(x)$ and a convex part $Q_2(x)$.

The main feature of (19) that makes it interesting is that the real complexity of any algorithm for solving this problem is likely to depend not on the total number of variables n but rather on m. One such algorithm that tries to take advantage of this feature was proposed by Pardalos *et al.* (1987). The basic idea is to obtain good approximations to the concave part $Q_1(x)$ and use them to solve the problem efficiently. For each function $\theta_i(x_i)$, the linear function $\gamma_i(x_i) = \left(c_i - \frac{1}{2}\lambda_i\beta_i\right)$ (where β_i is the upper bound on x_i) provides an underestimating function that meets $\theta_i(x_i)$ at the end points for x_i, that is, at its lower and upper bounds. Hence, the function

$$f(x) = \sum_{i=1}^{m} \gamma_i(x_i) + Q_2(x)$$

is a convex underestimating function to $Q(x)$ in (19). Moreover, it is possible to show that the maximum error in using this approximation is bounded:

Theorem 7.4

$$|Q(x) - f(x)| \le \frac{1}{8} \sum_{i=1}^{m} \lambda_i \beta_i^2$$

Proof. See Pardalos *et al.* (1987).

This theorem indicates that the error in using a linear approximation depends on the curvature of the concave functions (through the eigenvalues) and the geometry of the feasible region (as given by the bounds on the variables).

The solution of (19) using the approximation $f(x)$ instead of $Q(x)$ (which is then a convex quadratic problem) provides a lower bound on the optimal solution of (19). If this solution is satisfactory, then the procedure can be terminated. Otherwise, a branch-and-bound bisection technique is used to continuously bisect the rectangle defined by the bounds on the concave variables x_i, $i = 1, \ldots, m$. In each of the subregions, then, better approximations can be obtained to the concave function and the procedure can be repeated to get tighter lower bounds on the global solution. Finally, it is also possible to obtain a piecewise linear underestimator to the concave functions that involves the use of 0-1 integer variables. This requires the solution of a mixed 0-1 linear program. Such an approach can work very well for large problems (as reported by Pardalos *et al.* (1987)) especially when approximate solutions are sufficient.

Another instance where it might be feasible to solve large scale quadratic problems is when there is a dominance of linear terms in the objective function. Consider the indefinite instance of (12), which is a quadratic problem with a large number of variables (y) appearing linearly in the objective function. Phillips and Rosen (1990) proposed an extension of their previous work for concave quadratic problems (Phillips and Rosen, 1988) to handle indefinite quadratic problems of this form. As with the above approach, the algorithm first reduces the problem to a form with a separable objective function. A convex quadratic underestimator to the whole objective function is then developed, and used to solve an auxiliary problem that provides upper and lower bounds on the global solution. If the bounds are not tight enough, then the feasible domain is partitioned along the directions of the concave variables, and tighter bounds are obtained in these subregions. The procedure continues till an ϵ-approximate solution has been found to the original problem. The computational results presented along with the algorithm (for problems having upto 25 concave and 400 linear variables) indicate that this approach works very well when the number of linear variables dominates at the solution.

7.4. Polynomial Time Algorithms

In Section 4, we mentioned that polynomial time algorithms can be developed for certain classes of quadratic problems. Naturally, this raises the question of whether such algorithms can be developed for the general indefinite quadratic problem. It is obvious that a polynomial algorithm for computing the exact solution of (1) cannot be expected, since that would imply that $P = NP$ (as the general QP is in NP).

It is still open to question, however, whether an approximation[7] algorithm can be developed that solves (1) in polynomial time. One of the more recent results in this direction, due to Vavasis (1992), can be stated in the following theorem:

Theorem 7.5 *Suppose the matrix D is indefinite, and the feasible region for (1) is compact. Let t be the number of negative eigenvalues of D. Then, there exists an algorithm to find an ϵ-approximate solution to (1) in $O((\frac{n(n+1)}{\sqrt{\epsilon}})^t l)$ steps. In this formula, l denotes the time to solve a convex quadratic programming problem of the same size as (1).*

Proof. See Vavasis (1992).

This result indicates that if the number of negative eigenvalues in D is fixed, then a polynomial time algorithm can be developed for the general quadratic problem. One such algorithm was proposed by Vavasis (1992). The approach involves computing a Löwner-John pair for the constraint region of (1), that is, a pair of

[7] One definition of an approximate solution, due to Vavasis (1992), is as follows. Given an optimum $x^\star$ to (1), any feasible point x' is an ϵ-approximate solution if there exists another feasible point $x^\#$ satisfying

$$Q(x') - Q(x^\star) \leq \epsilon[Q(x^\#) - Q(x^\star)].$$

This definition has the advantage that it is insensitive to translations or dilations of the objective function $Q(x)$.

concentric ellipsoids E_1 and E_2 such that $E_1 \subset \Omega \subset E_2$ (where Ω is the feasible region in (1)) and E_1 is obtained from E_2 by shrinking each dimension by $1/n$. Such a pair always exists, and can be computed for a convex feasible region in polynomial time. Assume that the ellipsoids are defined by

$$\begin{aligned} E_1 &= \{x \in \Re^n : (x-b)^T M(x-b) \leq 1\} \\ E_2 &= \{x \in \Re^n : (x-b)^T M(x-b) \leq (n+1)^2 n\} \end{aligned}$$

where M is a positive semi-definite symmetric matrix, and b is some n-vector. Then, the x variables are translated by b so that the Löwner-John pair is centered at the origin (this does not affect the quadratic term in the objective function) and a matrix X is found so that $X^T M X = I$ and $X^T D X$ is diagonal. Replacing x by $X^{-1}x$ leads to a new problem that has the form

$$\begin{aligned} \min_x \ Q(x) &= c^T x + \frac{1}{2} x^T D x \\ s.t. \quad Ax &\leq b \\ x &\geq 0 \end{aligned}$$

where D is now diagonal, and the constraint region Ω now satisfies $S_1 \subset \Omega \subset S_2$, where

$$\begin{aligned} S_1 &= \{x : x^T x \leq 1\} \\ S_2 &= \{x : x^T x \leq n(n+1)^2\} \end{aligned}$$

The matrix D still has t negative eigenvalues (say the first k), and x can be split into vectors y and z corresponding to the first k entries and the remaining $n-k$ entries respectively. This enables the problem to be rewritten in a form with a concave and a convex part. The problem is then projected in the space of the concave (y) variables, and the resulting region is partitioned repeatedly into subcubes, the number of partitions being given by m^t where m is on the order of $n^2/\sqrt{\epsilon}$. Taking a linear underestimator for the concave terms, convex quadratic subproblems are solved in each subregion. The resulting solutions provide a lower bound for the optimal solution, and using error analysis on the difference between the linear approximation and the concave terms, it is possible to show that convergence to within ϵ is obtained. Moreover, the algorithm takes exactly the number of steps given in Theorem 7.5.

8. Quadratic Problems With Quadratic Constraints

The most general class of quadratic problems arises from the inclusion of quadratic constraints in (1). These problems can be formulated as follows:

$$\begin{aligned} \min_x \ \ & Q(x) = x^T D x \ + \ c^T x, \\ s.t. \quad & x^T A_j x + B_j x \ \leq \ b_j, \qquad j = 1, ..., m \ , \\ & x \ \geq \ 0, \end{aligned} \tag{20}$$

where A_j is an $n \times n$ matrix corresponding to the mth quadratic constraint, and B_j is the j^{th} row of the $m \times n$ matrix B. Note that some of the constraints can be linear due to the absence of the corresponding matrices A_j.

Problem (20) has received far less attention than (1). One of the reasons for this is that even finding a feasible solution for (20) can be a formidable task. Therefore, even for the convex case (when D and A_m are positive semidefinite), there are very few algorithms for solving (20). However, quadratically constrained problems constitute an important part of mathematical programming problems, arising in various practical applications including facility location, production planning, VLSI chip design, optimal design of water distribution networks, and most problems in chemical engineering design. Moreover, (20) encompasses all the other categories of quadratic problems as special instances. It is therefore well worth considering the few algorithms available for solving (20) in detail.

In the early years since it was first introduced in the seminal paper of Kuhn and Tucker (1951), the only case of (20) considered was when there was a single quadratic constraint in the problem (Swarup, 1966; Panne, 1966). The first general approach for solving (20) was proposed by Baron (1972), who considered the following two Lagrange functions for (20):

$$\begin{aligned} L_1(x,\mu) &= x^T D x + c^T x + \sum_{j=1}^{m} \mu_j (x^T A_j x - B_j x - b_j), \quad \text{and} \\ L_2(x,\mu,\lambda) &= L_1(x,\mu) - \lambda_i x_i \end{aligned}$$

Here, μ and λ are the multipliers for the quadratic and bound constraints respectively. Baron (1972) proposed solving (20) by minimizing these Lagrange functions through a cutting plane algorithm. The algorithm solves a sequence of linear master problems that minimize a piecewise linear function constructed from the Lagrange functions for constant x, and a primal problem with either an unconstrained quadratic function (using $L_2(x,\mu,\lambda)$) or a quadratic function over the nonnegative orthant (using $L_1(x,\mu)$). The cuts generated by the problems are used to approximate the two dual forms, and convergence (when both primal and dual problems have equal objective values) is guaranteed if either the objective function is convex, or the set of active constraints at each iteration have positive semidefinite quadratic matrices.

Falk and Soland (1969) proposed a branch and bound algorithm for solving (20) (and other more general problems) when the objective function is separable, and the constraint set is linear. The method involves solving bounding convex envelope approximating problems over successive partitions of the feasible region. The algorithm was later extended by Soland (1971) to handle nonconvex constraints. Although the algorithm can be applied to solve (20), it generates a number of infeasible points during the solution procedure, and does not in general converge in a finite number of iterations.

Another branch and bound method was proposed by Reeves (1975) to solve (20) when the objective function and constraints are separable in the variables. The algorithm uses a similar branching scheme to that of Soland (1971). At each iteration, given a partition of the feasible region, a local minimum of the objective in the partition is found, and a small interval surrounding this minimum is eliminated. The remaining region is then partitioned and the process is repeated for the new partitions. The algorithm differs from that of Soland (1971) in that bounding convex polytope problems are solved only to eliminate regions, not to generate base points for further partition. By deriving local minima at each iteration, the algorithm obtains and verifies in a finite number of iterations, either exactly or to some arbitrarily small precision, a global minimum of the problem.

BenSaad (1989) proposed an algorithm for the solution of linear problems with an additional reverse convex constraint, of the following form:

$$\min_{x \in \Re^n} c^T x$$

$$\begin{aligned} s.t. \quad Ax &\leq b \\ g(x) &\leq 0 \end{aligned}$$

where $g(x)$ is a concave function (it should be noted that by appropriate transformations, any problem of the form (20) can be converted to this form). The algorithm works by partitioning the feasible region into subsets contained in cones originating at an infeasible vertex of the polytope formed by the linear constraints while ensuring that an interior point of the feasible region is contained in each partition. The intersection of the rays forming the cone with the reverse convex constraint are used to determine whether a particular cone is to be made smaller. At each iteration, n cones are generated (where n is the number of variables) and successively tightened until the convex hull thus obtained contains the solution to the reverse convex problem.

Al-Khayyal *et al.* (1990) proposed two algorithms for the solution of problems with concave objective functions and separable quadratic constraints. These problems can be given in the following form:

$$\min_{x \in \Re^n} c^T x$$

$$\begin{aligned} s.t. \quad g_i(x) &:= \sum_{k=1}^{n} \left(\frac{1}{2} p_{ik} x_k^2 + q_{ik} x_k + r_{ik}\right) \leq 0, \quad i = 1, \ldots, m \\ M &\equiv \underline{m}_k \leq x_k \leq \overline{m}_k, \quad k = 1, \ldots, n \end{aligned}$$

The first algorithm is a branch-and-bound algorithm which works by division of the initial rectangular domain M into increasingly smaller rectangles with the size going to zero in the limit (one way to achieve this is to use bisections). In each rectangular subdivision, lower and upper bounds are obtained for the objective function. The partitions are deleted either if the lower bound for the objective in that region is greater than the best obtained solution so far, or if the region is infeasible for the quadratic constraints. The latter condition is checked by use of bounds on $g_i(x)$ in the partition. In addition, to speed up convergence, various linearizations of the quadratic constraints are also added to the problem formulation. The authors also propose a second approach which uses piecewise linear approximation for the quadratic constraints. The resulting problem is then solved as a mixed 0-1 linear problem. This approach is similar to the ones used by Pardalos and Rosen (1986) for concave quadratic problems and Pardalos *et al.* (1987) for indefinite quadratic problems.

8.1. Decomposition Techniques

As mentioned in Section 7.2, decomposition techniques have several features that make them attractive for solving quadratic problems. Foremost among these is that usually the subproblems solved are of far smaller size than the original problem. Most of the decomposition approaches for solving the quadratic problem are based upon the Lagrangian dual formulation. However, one of the main problems with the quadratic problem (indeed, with all nonconvex programs) is that there can exist a duality gap between the nonconvex problem and its Lagrangian dual. Consequently, algorithms that iterate between primal and dual formulations of the original problem (as, for example, the algorithm of Baron (1972)) can fail to converge except under special conditions on the problem structure.

Lazimy (1982; 1985) proposed a method based upon the generalized Benders' decomposition to solve mixed-integer quadratic problems. The approach first converts the problem to a form where the integer variables are absent from the objective function and the resulting master (dual) problem is linear in the integer variables. The original problem is then solved through a series of relaxed integer-linear master problems and continuous convex quadratic primal problems. Wolsey (1981) proposed a resource decomposition algorithm for solving bilinearly constrained problems.

More recently, two decompositions algorithms have been proposed that guarantee convergence to the global solution for problems of the form (20). The first of these approaches, due to Floudas and Visweswaran (1990; 1993), considers the solution of (20) by reducing it to the equivalent problem

$$\begin{aligned} \min_{x,y} \quad & Q(x) = x^T Dy \;+\; c^T x, \\ s.t. \quad & x^T A_j y + B_j x \;\leq\; b_j, \qquad j = 1, ..., m\ , \\ & x_i - y_i \;=\; 0 \qquad i = 1, \ldots, n\ , \\ & x \geq 0, y \;\geq\; 0, \end{aligned}$$

which in turn falls under the framework of the following biconvex problem:

$$\begin{aligned} & \min_{x,y}\ f(x,y) \\ s.t. \quad & g(x,y) \;\leq\; 0 \\ & h(x,y) \;=\; 0 \\ & x \in X, y \;\in\; Y \end{aligned} \tag{21}$$

where f and g are convex, and h is linear, in x for all fixed y and vice-versa. For a fixed value of $y = y^k$, the problem (called a Primal problem) is linear in x, and provides an upper bound on the global solution. Moreover, the solution of this primal problem provides multipliers for the constraints which are used to formulate the Lagrange function:

$$L(x, y, \lambda^k, \mu^k) = f(x, y) + {\lambda^k}^T h(x, y) + {\mu^k}^T g(x, y)$$

It can be shown (Geoffrion, 1972) that a *relaxed* dual formulation for (21) can be written as

$$\begin{aligned} \min_{\substack{y \in Y \\ \mu_B}} \ & \mu_B, \\ s.t. \quad \mu_B \ \geq \ & \min_x L(x, y, \lambda^k, \mu^k), \qquad \forall \mu^k \geq 0\ ,\ \lambda^k\ , \end{aligned} \tag{22}$$

where μ_B is a scalar. This problem provides a lower bound on the global solution of (21). Because of the presence of the inner minimization problems, this problem can be very tough to solve. However, because the primal problem is solved in the space of the x variables and the relaxed dual problem is essentially solved in the space of y, they can be used as the basis for an iterative algorithm that alternates between providing bounds as well as solutions for x and y.

Floudas and Visweswaran (1990; 1993) proposed the following method for solving the relaxed dual problem, which involves linearizing the Lagrange function with respect to the x variables around the solution x^k (of the primal problem corresponding to $y = y^k$). The linearization ensures that each of the Lagrange functions independently achieves its minimum at a bound of the x variables. This enables the solution of the relaxed dual problem as a set of linear problems, with the x variables being replaced by a combination of their bounds in each of the problems. The basic algorithm (GOP) then can be summarized as follows:

Step 0 Initialize storage parameters, select a starting point y^1.

Step 1 Set $k = 1$. Solve the primal problem for $y = y^k$, generate the Lagrange function using the optimal multipliers from this problem. Update the best upper bound on the original problem. If the primal is infeasible, then solve a relaxed primal problem and use the solution to generate a feasibility cut for the relaxed dual problem.

Step 2 Generate a set of constraints from previous iterations that provide valid underestimators for the current region, and add them to the relaxed dual problem.

Step 3 Solve the relaxed dual problem as a set of linear subproblems, storing all feasible solutions. Select the best stored solution as the lower bound, and the corresponding solution for y as y^{k+1}.

Step 4 Check for convergence between the bounds. Set $k = k + 1$ and return to Step 2.

A few points are worth mentioning about the algorithm. At each iteration, the set of subproblems solved in Step 3 are restricted to a subregion of the y variables.

This region is defined by a cone formed by a set of constraints from the previous iterations. These constraints are generated by setting the x variables in the Lagrange function for that iteration to a particular set of bounds. The actual set of bounds to be used is determined by the sign of the gradients of the Lagrange function at the current y^k, and are selected in such a way that the underestimators used are always valid for the current cone. At each iteration, therefore, Step 3 is equivalent to partitioning Y into a set of subregions. Thus, the algorithm functions both as a branch and bound as well as a cutting plane method.

The lower bound found at the end of Step 3 is guaranteed to be nondecreasing as the iterations proceed. This property, coupled with the presence of constraints from previous iterations that prevent cycling, give a theoretical guarantee of convergence of the algorithm to an ϵ-global solution in a finite number of iterations.

Visweswaran and Floudas (1990) discussed the application of GOP algorithm to several classes of problems (including quadratically constrained quadratic problems) and provided preliminary computational experience for some examples in each class. Visweswaran and Floudas (1993) proved several properties for quadratic and polynomial problems that significantly improve the computational performance of the algorithm. More recently, Floudas *et al.* (1993) have proposed solving the relaxed dual subproblems at each iteration by a single mixed 0-1 linear problem. It has also been shown by Liu and Floudas (1993a) that the GOP algorithm is applicable to a large class of smooth optimization problems. The convergence for the algorithm for these problems has been shown by Liu and Floudas (1993b).

Problem (20) can often be written in the following form (Ben-Tal and Gershovitz, 1992; Ben-Tal *et al.*, 1993):

$$\min_{\substack{x\in\Re^n\\ y\in\Re^p}} \quad f_0(x,y)$$

$$\begin{aligned} s.t. \qquad x\in X(y) &= \{x\in\Re^n : f_i(x,y)\le 0, \quad i=1,\ldots,m\} \\ y\in Y &\subseteq \Re^p \end{aligned} \tag{23}$$

where $p \ll n$ and the functions $f_i(x,y)$ are such that for every fixed x $f_0(.,y)$ is convex and $X(y)$ is a convex set. Note that the bilinear problem (and therefore all quadratic problems by reduction to bilinear form) satisfies this condition. A natural approach for solving this problem is by decomposition:

$$\min_{y\in Y}\left\{ \quad \phi(y) =: \min_{x\in X(y)} \quad f_0(x,y)\right\} \tag{24}$$

Although the function $\phi(y)$ may not be smooth (even though the inner problem is convex), the availability of good algorithms for non-smooth optimization can overcome this drawback. Therefore, it is the nonconvexity of the outer problem that poses the main challenge.

The dual form of (23) can be written as

$$\max_{\mu\ge 0}\left\{ h(y) =: \min_{y\in Y,x} \; L(x,y,\mu) = f_0(x,y) + \sum_{i=1}^{m}\mu_i f_i(x,y)\right\} \tag{25}$$

By weak duality, then, the solution of (25) provides a lower bound to (23). In particular, when there is no duality gap, then the solutions of (23) and (25) are identical. However, the presence of a duality gap means that (25) may not provide a tight lower bound. To overcome this, Ben-Tal *et al.* (1993) proposed a branch and bound approach based upon reducing the duality gap by successive domain partitioning. Consider a partition of the set Y into I subsets $Y_i, i = 1, \ldots, I$. Then, for the i^{th} partition, the corresponding primal and dual problems are respectively

$$\min_{\substack{x \in \Re^n \\ y \in \Re^p}} \ \{ \ f_0(x, y) \ : y \in Y_i, x \in X(y) \} \tag{26}$$

and

$$\max_{\mu \geq 0} \ \left\{ \ h(y) =: \min_{y \in Y_i, x} \ L(x, y, \mu) \right\} \tag{27}$$

Then, the following theorem can be proved:

Theorem 8.1 *If $Q^\star$ is the optimal value of (23), $D^\star$ is the optimal value of (25), and $D_i^\star$ is the optimal value for (26), then*

$$Q^\star \geq \min_{i \in I} \{ \ D_i^\star \ \} \geq D^\star$$

This theorem shows that as the set Y is partitioned, the gap between the primal and dual problems decreases. For problems of the form (23) which also satisfy an appropriate constraint qualification, it can be shown that the gap varies directly as the "size" of the partitions (as measured by, say, the radius of the smallest hypersphere containing Y_i); when the size of the partitions is zero, then there is no duality gap, and strict equality holds in the left hand side of the theorem.

In general, the dual problem solved in each partition involves a linear objective function minimized over a convex set of constraints. However, when the functions $a_j(y)^T \mu + c_j$ are quasiconcave (as for bilinear problems, for example) each of these dual problems can be solved as a linear program. Ben-Tal *et al.* (1993) have applied this approach to solve several instances of the pooling problem (see Section 10.2 for one formulation of this problem).

One particular quadratically constrained problem that has received considerable attention is the following problem, which involves minimizing a quadratic constraint over a sphere:

$$\begin{aligned} \min \ & \frac{1}{2} x^T D x + c^T x \\ s.t. \quad & x^T x \leq 1 \end{aligned} \tag{28}$$

This problem often comes up as a subproblem in general optimization algorithms. Often, the objective function in nonlinear programming is approximated locally by a quadratic function. In such cases, the approximation is restricted to a small region around the current iterate. If the 2-norm is used to define this region, then we end up with (28). Such methods are referred to as model trust region methods.

Because of the presence of the sphere constraint, the solution of (28) is likely to be irrational, which implies that it is not possible to exactly compute the solution. However, there exist polynomial time algorithms to efficiently compute the approximate solution to this problem given a desired precision. Sorensen (1982) proposed a general algorithm using model trust regions, while Ye (1990), Ye (1992) and Karmarkar (1990) proposed strongly polynomial algorithms for solving (28).

9. Quadratic Optimization and the Linear Complementarity Problem

Closely associated with quadratic problems is the class of linear complementarity problems (LCP). Given a real matrix $M \in \Re^{n\times n}$, the pair of vectors $(I_{.j}, -M_{.j})$ is called a *complementary pair of vectors* for M. If we denote a vector from this pair by $A_{.j}$, then the cone defined by $\{y : y = \alpha_1 A_{.1} + \ldots + \alpha_n A_{.n},\quad \alpha_i \geq 0\ \forall\ i = 1,\ldots,n$ $\}$ is called a *complementary cone* of M. Then, given any vector $q \in \Re^n$, the LCP involves finding a complementary cone of M that contains the point q, that is, finding a complementary set of vectors $(A_{.1},\ldots,A_{.n})$ such that q can be expressed as a nonnegative linear combination of these vectors. This is equivalent to solving the following problem[8]:

Find a feasible solution vector x to the set of constraints

$$\begin{aligned} y &= Mx + q \\ x,\ y &\geq 0, \\ x^T y &= 0 \end{aligned} \tag{29}$$

or prove that such a feasible solution x does not exist.

Note that the LCP as given by (29) is not an optimization problem in the strict sense; there is no objective function to minimize. However, these problems have a close relationship to several of the most important classes of optimization problems, including linear programming, convex quadratic programming, bimatrix games and mixed integer programming. The following quadratic problem, for example, is equivalent to (29):

$$\begin{aligned} &\min_x\ Q(x) = x^T(Mx+q) \\ &s.t. \qquad Mx + q \geq 0 \\ &\qquad\qquad\qquad x \geq 0 \end{aligned} \tag{30}$$

Moreover, the LCP is closely related to the optimality conditions for the solution of (1). Consider, the KKT conditions for the quadratic problem (1), given by (2)-(6). Then, it can be easily shown that the KKT conditions for (1) reduce to the LCP (29) with

$$M = \begin{bmatrix} 0 & -A \\ A^T & D \end{bmatrix} \quad q = \begin{bmatrix} b \\ c \end{bmatrix}$$
$$y = \begin{pmatrix} y \\ v \end{pmatrix} \quad x = \begin{pmatrix} u \\ x \end{pmatrix}$$

[8] Note that when the vector q is nonnegative, then the problem is immediately solved by setting $x = 0$ and $y = q$. Therefore, the only cases of interest are when at least one component of q is negative.

where y and x are the original vectors augmented by v and u respectively. Thus, the LCP can be considered as a problem of finding a KKT point for the quadratic problem (1). It can be proved (Theorem 1.13 of Murty (1988)) that any solution $\overline{x}$ to the LCP (29) (with the data as given above) is a KKT point for (1). For the case of positive semidefinite D (convex quadratic programming), the LCP (29) can thus be used to provide the necessary and sufficient optimality conditions.

For the more general case of indefinite (nonconvex) quadratic problems, since the KKT conditions are not sufficient to ensure optimality, the LCP cannot directly be used to obtain a solution to (1). However, even for these problems, the LCP has an indirect relevance. Consider, for example, the following problem:

$$\begin{aligned} \Psi(x,u) &= \frac{1}{2}(c^T x - b^T u) \\ s.t. &\quad (2)-(6). \end{aligned} \tag{31}$$

It is well known that at any point (x, y, u, v) that satisfies the KKT conditions for (1), the value of $Q(x)$ coincides with the value of the objective function $\Psi(x, u)$ of the above problem. This leads to the following theorem (Gianessi and Tomasin, 1974):

Theorem 9.1 *Let the QP (1) have an optimal solution. Then, any point $\boldsymbol{x}$ solves (1) if and only if there exist vectors $\boldsymbol{y}, \boldsymbol{u}$ and $\boldsymbol{v}$ such that $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{u}, \boldsymbol{v})$ solves (31).*

It should be noted that this theorem holds only when $Q(x)$ is bounded from below. If this is the case, then the above theorem indicates that it is possible to solve (1) by minimizing a linear function over the solution set of a linear complementarity problem. Judice and Mitra (1988) present several other equivalences between quadratic problems and related linear complementarity problems.

It is clear, therefore, that the nature of the linear complementarity problem and its methods of solution can help provide considerable insight into the solution of the QP (1). While a detailed treatment of the LCP is beyond the scope of this chapter, the reader is referred to Mitra (1979), Lemke (1980), Berschanskii and Meerov (1983), Murty (1988) and Cottle *et al.* (1992).

10. Applications

In this section, we describe some interesting problems in mathematical programming, engineering design and control that can be formulated as quadratic programming problems. Several applications and test problems for quadratic programming, as well as general nonconvex optimization problems can be found in Floudas and Pardalos (1990).

10.1. Economies of Scale

Problems involving economies of scale (in production and sales) often can be formulated as concave quadratic programming problems. Consider a case when n products are being produced, with x_i being the number of units of product i and c_i being the cost of production per unit of product i. Usually, as the number of units produced increases, the unit cost decreases. Often, this can be correlated by a linear

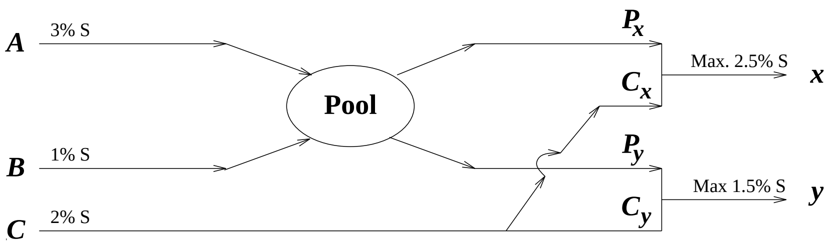

Fig. 1. The Haverly Pooling Problem (Haverly, 1978)

functional:

$$c_i = c_{i_0} + e_i x_i$$

where e_i is a negative quantity. Then, given constraints on the production demands and availabilities of each product, the problem of minimizing the total production cost can be written as

$$\begin{aligned} \min_x \quad & \sum_{i=1}^{n} (c_{i_0} + e_i x_i) x_i \\ s.t. \quad & x \in D \end{aligned}$$

where D represents the demand and availability constraints. This is a concave minimization problem. Similarly, if the profits are maximized, the economies of scale dictate that the profits per unit rise linearly with the number of units produced. Hence, in this case, the problem becomes one of maximization of a convex functional.

10.2. Pooling and Blending Problems

Pooling and blending problems are a feature of the models of most chemical processes. In particular, for problems relating to refinery and petrochemical processing, it is often necessary to model not only the product flows but the properties of intermediate streams as well. These streams are usually combined in a tank or pool, and the pool is used in downstream processing or blending. The presence of these streams in the model introduces nonlinearities, often in a nonconvex manner. The nonconvexities arise from the interactions between the qualities of the input streams and the blended products.

A typical pooling problem is shown in Figure 1. It consists of three feed streams (A, B, and C) being combined in a pool to form two products x and y. In the absence of the pooling restriction, the problem can be formulated and solved as an LP. However, when the streams need to be pooled (as, for example, when there is only one tank to store A and B), the formulation becomes a quadratic one, as shown below:

$$\min \quad 6A + 16B + 10(C_x + C_y) - 9x - 15y$$

s.t.

$$\begin{aligned}
P_x + P_y - A - B &= 0 \;\} \quad \textit{pool balance} \\
\left.\begin{aligned} x - P_x - C_x &= 0 \\ y - P_y - C_y &= 0 \end{aligned}\right\} &\quad \textit{component balance} \\
p.(P_x + P_y) - 3A - B &= 0 \;\} \quad \textit{pool quality} \\
\left.\begin{aligned} p.P_x + 2.C_x - 2.5x &\le 0 \\ p.P_y + 2.C_y - 1.5y &\le 0 \end{aligned}\right\} &\quad \textit{product quality constraints} \\
\left.\begin{aligned} x &\le x^U \\ y &\le y^U \end{aligned}\right\} &\quad \textit{upper bounds on products}
\end{aligned}$$

where p is the sulfur quality of the pool; its lower and upper bounds are 1 and 3 respectively.

Traditionally, these problems are solved using successive linear programming techniques (Haverly, 1979; Lasdon *et al.*, 1979). These methods suffer from the drawback of being highly dependent on the starting point. More recently, this problem has been studied by Floudas and Aggarwal (1990) who applied the global optimum search method to solve the problem, and by Visweswaran and Floudas (1990), who successfully applied the GOP algorithm to solve this problem. Similar pooling problems have been studied by Ben-Tal and Gershovitz (1992) and Ben-Tal *et al.* (1993). Multiperiod tankage quality problems that lead to similar formulations have been studied by Visweswaran and Floudas (1990).

10.3. Multicomponent Separation Problems

Like the pooling and blending problems, quadratic models arise in separation of multicomponent process streams in chemical refineries. These problems typically involve multicomponent feed streams that need to be separated into one or more products, each of which may have a specified composition of the various components. One method of solving such problems is the superstructure approach. This involves first constructing a superstructure containing all possible options for splitting, bypassing and blending to achieve the desired products. The resulting model is then optimized to give the actual configuration that will perform the separation. A typical example of such a problem (Floudas and Aggarwal, 1990) is shown in Figure 2, which concerns the separation of a 3-component feed stream into two products. The resulting bilinear formulation for the optimization model is shown below:

$$\min \quad 0.9979 + 0.00432F_5 + 0.01517F_{13}$$

subject to

(Overall Mass Balances)

$$F_1 + F_2 + F_3 + F_4 = 300$$

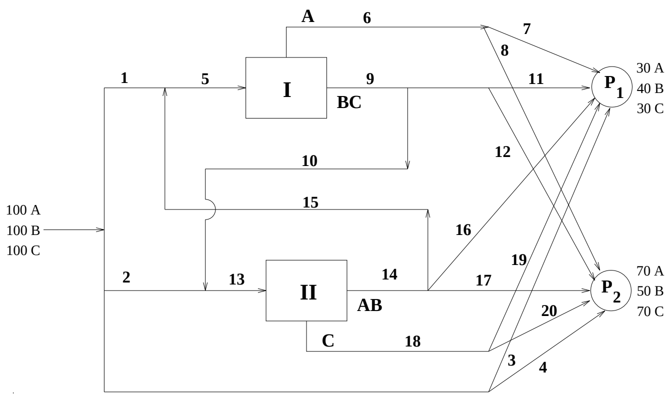

Fig. 2. Multicomponent Separation Example (Floudas and Aggarwal, 1990)

$$\begin{aligned}
F_6 - F_7 - F_8 &= 0 \\
F_9 - F_{10} - F_{11} - F_{12} &= 0 \\
F_{14} - F_{15} - F_{16} - F_{17} &= 0 \\
F_{18} - F_{19} - F_{20} &= 0
\end{aligned}$$

(Splitter Component Balances)

$$\begin{aligned}
F_5 x_{j,5} - F_6 x_{j,6} - F_9 x_{j,9} &= 0 \quad j = A, B, C \\
F_{13} x_{j,13} - F_{14} x_{j,14} - F_{18} x_{j,18} &= 0 \quad j = A, B, C
\end{aligned}$$

(Inlet Mixer Balances)

$$\begin{aligned}
0.333F_1 + F_{15} x_{j,14} - F_5 x_{j,5} &= 0 \quad j = A, B, C \\
0.333F_2 + F_{10} x_{j,9} - F_{13} x_{j,13} &= 0 \quad j = A, B, C \\
0.333F_3 + F_7 x_{A,6} + F_{11} x_{A,9} + F_{16} x_{A,14} + F_{19} x_{A,18} &= 30 \\
0.333F_3 + F_7 x_{B,6} + F_{11} x_{B,9} + F_{16} x_{B,14} + F_{19} x_{B,18} &= 50 \\
0.333F_3 + F_7 x_{C,6} + F_{11} x_{C,9} + F_{16} x_{C,14} + F_{19} x_{C,18} &= 30
\end{aligned}$$

(Compositions)

$$x_{A,i} + x_{B,i} + x_{C,i} = 1 \quad i = 5, 16, 9, 13, 14, 16$$

(Sharp Split)

$$x_{B,6} = x_{C,6} = x_{A,9} = x_{C,14} = x_{A,18} = x_{B,18} = 0$$

where all variables are nonnegative.

10.4. The Quadratic Knapsack Problem

The quadratic knapsack problem is given as

$$\min_x \; c^T x + \frac{1}{2} x^T Q x$$

$$s.t. \quad a^T x = \gamma$$
$$x^L \leq x \leq x^U.$$

This problem occurs frequently in resource allocation, multicommodity network flow problems, and problems in logistics among others. In resource allocation, for example, γ represents the total amount of resource, and a_i is the consumption of this resource by the i^{th} activity. The presence of the "knapsack" constraint in the formulation makes this problem much easier to solve. Various approaches have been proposed for this problem (Pardalos and Kovoor, 1990; Ye, 1990; Vavasis, 1992). Of these approaches, the interior point algorithm of Ye (1990) is the only one guaranteed to converge to the global minimum in polynomial time.

10.5. IC Circuit Layout and Compaction

One of the most challenging and error prone parts of IC design is low level cell layout. Given a logical design, there are several possible ways to map the layout geometrically. Often, it is desirable to consider this problem in the context of compaction, that is, as the problem of finding the minimum of the area function subject to linear and nonlinear constraints. Given a two-dimensional layout, variables in the formulation represent the position of a particular symbol in the logical diagram. Consider, for example, any two components on the circuit, whose coordinates may be given by (x_1, y_1) and (x_2, y_2). The geometric restrictions on layout can then occur due to separation rules that require that the two components be at least a certain distance apart. Such constraints then take the form

$$x_2 - x_1 \geq d_1$$

or

$$y_2 - y_1 \geq d_2$$

where d_1 and d_2 are minimum distance requirements. Another form of constraints can be imposed due to connectivity requirements, which again take the form of linear inequalities. These and other constraints constitute the feasible region for a set of points (x_i, y_i). The area occupied by a cell composed of several such components can then be given as

$$f(x, y) = (Tx - Sx)(Ty - Sy)$$

Thus, the compaction problem becomes one of quadratic minimization:

$$\min_{x,y \in \Omega} f(x, y)$$

where Ω is the feasible region defined by the set of linear inequalities. For further discussion on this problem, see Kedem and Watanabe (1983).

A similar model arises in the layout problem, which requires a combination of space and communication costs to be minimized. This corresponds to the rectangular dualization problem. An example of such a model, which has quadratic constraints, is shown below:

$$\min \quad x_a y_a + x_b y_b + x_c y_c + x_d y_d + x_e y_e + x_f y_f$$

$$
\begin{aligned}
s.t. \qquad -x_d + x_e &= 0 \\
-x_a + x_b - x_d + x_e &= 0 \\
-x_a + x_c - x_d + x_f &= 0 \\
-y_c + y_f &= 0 \\
-y_a - y_b - y_c + y_d + y_e + y_f &= 0 \\
x_b - x_c &\geq 1 \\
y_a - y_d &\geq 1 \\
x_a y_a &\geq 30 \\
x_b y_b &\geq 20 \\
x_c y_c &\geq 20 \\
x_d y_d &\geq 25 \\
x_e y_e &\geq 15 \\
x_f y_f &\geq 20 \\
x_a \geq 5, x_b \geq 5, x_c \geq 2, x_d \geq 4, x_e \geq 4, x_f &\geq 5 \\
y_a \geq 5, y_b \geq 2, y_c \geq 5, y_d \geq 4, y_e \geq 5, y_f &\geq 5
\end{aligned}
$$

For more details, the reader is referred to Maling *et al.* (1982).

10.6. Optimal Design of trusses

The structural design of trusses typically involves the minimization of the total weight of the structure. Such problems have a large field of practical applications and often result in bilinear models. A typical problem is shown below (Simões, 1987):

$$
min\ x_1 + x_2 + x_3
$$

$$
\begin{aligned}
s.t. \quad x_1 x_4 + x_3 x_6 &= 0 \\
3x_1 x_4 + 1.2 x_2 x_5 - x_3 x_6 &= 10 \\
5x_4 + x_5 + x_6 &\leq 2.5 \\
0.1 \leq x_1 &\leq 5.0 \\
0.1 \leq x_2 &\leq 5.0 \\
0.1 \leq x_3 &\leq 5.0 \\
0.0 \leq x_4 &\leq 2.5 \\
0.0 \leq x_5 &\leq 2.5 \\
-2.5 \leq x_6 &\leq 0.0
\end{aligned}
$$

10.7. Robust Stability Analysis of Feedback Control Systems

Feedback controllers are widely used to stabilize plant operations in various processes. Because of mismatches between the plant model being used and the physical system, it is important that the controller be designed for robustness, i.e. to satisfy stability and performance requirements for a range of values for the parameters of the plant model. A typical example of a problem (de Gaston and Sofonov (1988))

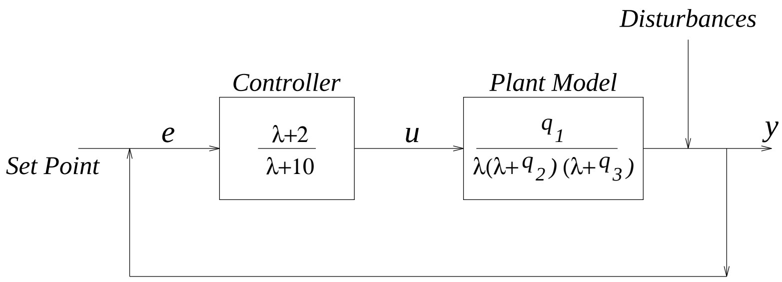

Fig. 3.

involving the analysis of the robust stability of feedback control structures with real parametric uncertainty is shown in Figure 3. The y variables are the measured variables from the plant output, u are the manipulated variables, and e are the errors between the measured values and desired set points for the measured variables. λ refers to the variables in the Laplace domain. The variables q_1, q_2 and q_3 are three parameters in the plant model, with nominal values of 800, 4 and 6, but these values can vary due to uncertainty. The requirement that the controller be robust leads to the following quadratically constrained optimization problem (see Psarris and Floudas (1993)):

$$\min \ x_6$$

$$\begin{array}{lrcl}
s.t. & x_5 y_1 - (x_4 + 10x_2 + 10x_3)y_1 + 2x_1 & = & 0 \\
& (x_2 + x_3 + 10)y_1 - 10x_4 - x_1 & = & 0 \\
& x_4 - y_2 x_3 & = & 0 \\
& x_5 - y_1 & = & 0 \\
& x_2 - y_2 & = & 0 \\
& 800 - 800x_6 \leq & x_1 & \leq 800 + 800x_6 \\
& 4 - 2x_6 \leq & x_2 & \leq 4 + 2x_6 \\
& 6 - 3x_6 \leq & x_3 & \leq 6 + 3x_6
\end{array}$$

$$x \in X, \quad y \in Y$$

where $x_1 = q_1$, $x_2 = q_2$, $x_3 = q_3$, $x_4 = q_2 q_3$, $x_5 = \lambda$ and x_6 is the stability margin for the control problem.

Detailed discussions of the importance of real parameter uncertainty as a tool for robustness analysis in control systems can be found in Ackermann (1993). See Psarris and Floudas (1993) for more details on the solution of such quadratic problems.

Problem	Problem size			CPU Times	
	n_x	n_y	m	S-A	A-F
1	2	4	8	4.1	89+
2	2	4	16	4.4	4.5
3	3	5	20	5.4	105+
4	3	5	24	3.7	3.7
5	6	3	24	7.8	81
6	7	4	24	14.1	30.0
7	4	7	20	20.0	25.0
8	5	4	24	11.0	84+
9	4	5	24	14.3	14
10	7	6	20	27.0	28.0
11	7	7	24	69.0	163+

TABLE I
Random runs of bilinear problems using the algorithms of Al-Khayyal and Falk (1983) and Sherali and Alameddine (1992)

11. Summary of Computational Results

The real test of any algorithm comes in how well it performs in an efficient implementation. For most of the algorithms discussed in the preceding sections, very little computational results have been reported. In this section, we summarize the best known algorithms for the various classes of quadratic problems in terms of the computational effort required. Almost all of the results reported consider randomly generated test problems. Often, the solution times for the algorithms can vary a lot depending on the type of problems generated. Therefore, while the results presented here are indicative of the general performance of the algorithm, they should not be taken as criteria for evaluating one algorithm vis a vis another algorithm.

11.1. Bilinear Programs and Quadratically Constrained Problems

For problems of the form (10), computational results for three algorithms are available. The performance of two of these algorithms, due to Al-Khayyal and Falk (1983) and Sherali and Alameddine (1992) is shown in Table I. In Table II is given the performance of the GOP algorithm as described in Floudas *et al.* (1993).

Several instances of the pooling problem (which has bilinear constraints) have been studied by Visweswaran and Floudas (1993) and Ben-Tal and Gershovitz (1992). The results for these problems are given in Table III.

11.2. Concave Quadratic Problems

For concave quadratic problems, the branch-and-bound algorithm of Phillips and Rosen (1988) and the GOP algorithm of Visweswaran and Floudas (1993) have been

Problem Size			Iterations and CPU	
n_x	n_y	m	Iter.	Cpu.
5	10	15	1.800	0.356
5	15	20	2.400	0.386
5	20	25	4.400	5.192
5	25	30	4.200	1.046
5	30	35	5.400	5.662
5	35	40	4.750	34.000
5	40	45	7.500	306.720
5	45	50	6.333	120.217
10	10	20	3.800	3.566
10	15	25	6.750	11.957
10	25	35	4.000	21.060

TABLE II
Randomly Generated Bilinear Problems Solved Using the GOP Algorithm (Floudas *et al.*, 1993)

Problem No.	*Problem Size*			GOP	Ben-Tal & Gershovitz
	n_x	n_y	m	Iterations.	Dual Problems
1.	8	1	7	7	25
2.	15	2	19	19	339
3.	32	6	24	47	438
4.	32	6	24	42	283

TABLE III
Pooling Problems From Ben-Tal and Gershovitz (1992)

applied to similar categories of problems. Two classes of problems were considered - a set of small example problems and a large number of randomly generated test problems. For both cases, the problems considered have the following form:

$$\min_{x,y\in\Omega} \ \psi(x,y) = \theta_1\varphi(x) + \theta_2 d^t y$$

where

$$\begin{aligned} \varphi &= 0.5\sum_{i=1}^{n}\lambda_i(x_i-\overline{\omega_i})^2, \\ \Omega &= \{(x,y) : A_1x + A_2y \leq b, x \geq 0, y \geq 0\}, \\ x,\ \lambda,\ \overline{\omega} &\in \Re^n, \end{aligned}$$

$$\begin{aligned} y,\ d &\in \Re^k, \\ A_1 &\in \Re^{m\times n} \\ A_2 &\in \Re^{m\times k} \\ \theta_1,\ \theta_2 &\in \Re. \end{aligned}$$

Here, m is the number of linear constraints, n is the number of concave variables (x), and k is the number of linear variables (y). The parameters θ_1 and θ_2 are -1 and 1 respectively, and the relative tolerance for convergence between the upper and lower bounds (ϵ) is 0.001.

Problem	*Problem Size*			GOP Algorithm	P&R
	m	n	k	CPU (HP730)	CPU (CRAY2)
example	5	2	0	1.09	0.026
prob1	5	6	0	0.54	0.022
prob2	5	6	0	0.55	0.020
prob3	5	6	0	0.45	0.026
prob10	4	2	0	1.17	0.017
prob11	4	3	0	1.48	0.015
prob12	4	3	0	1.50	0.014
prob13	10	3	0	0.68	0.022
prob14	10	3	0	0.82	0.020
prob15	4	4	0	2.03	0.029
prob20	9	2	1	8.98	0.023

m = Number of constraints
n = Number of concave variables
k = Number of linear variables

TABLE IV
Concave Quadratic Problems from Phillips and Rosen (1988) using the algorithms of Phillips and Rosen (1988) and Visweswaran and Floudas (1993)

Table IV contains the results for 11 example problems from Phillips and Rosen (1988). Solution times for the randomly generated test problems are reported in Table V. In this table, the times of the algorithm of Phillips and Rosen for sequential and (4) parallel processors are given in columns P&R1 and P&R2. Note that the cpu times reported for the Phillips-Rosen algorithm are for the CRAY2, while the times for the GOP algorithm are on an HP730.

11.3. Indefinite Quadratic Problems

For the case of the indefinite quadratic problem with box constraints, computational results are presented for two algorithms, the global optimization algorithm of Hansen

Run	Problem size			CPU Time		
	m	n	k	P&R1	P&R2	GOP
1	20	25	100	4.5	1.5	16.508
2	20	25	200	27.5	9.0	33.149
3	20	25	400	58.0	19.0	82.026
4	20	50	100	28.0	7.0	46.761
5	20	50	200	45.0	15.0	108.968

TABLE V
Random runs for concave quadratic problems using the algorithms of Phillips and Rosen (1988) and Visweswaran and Floudas (1993)

Density (%)	CPU for Problem size						
	20	30	40	50	60	70	80
10	0.08	0.13	0.36	1.14	92.32	81.13	488.06
20	0.13	0.70	8.90	132.30	—	—	—
30	0.23	3.39	143.0	—	—	—	—
40	0.51	9.12	—	—	—	—	—
50	1.07	50.85	—	—	—	—	—
60	1.15	116.2	—	—	—	—	—
70	3.81	160.9	—	—	—	—	—
80	8.82	—	—	—	—	—	—
90	8.12	—	—	—	—	—	—
100	17.87	—	—	—	—	—	—

(— indicates problems could not be solved).

TABLE VI
Results of runs of the algorithm of Hansen *et al.* (1991) for quadratic problems with box constraints

et al. (1991), and the local optimization algorithm of Coleman and Hulbert (1989). These results are given in Tables VI, and VII. These results indicate that both the density and type of eigenvalues play crucial roles in the performance. As the number of negative eigenvalues increases, the problem behaves more like a concave quadratic problem, with the solution likely to lie at a vertex. Consequently, the solution times are less for these problems. It can be seen that due to the sparsity of the problems considered in Table VII (which consider about 4% dense problems) it is possible to solve much larger problems in reasonable time. As the density of the matrix D increases to 100%, it becomes impossible to solve any but the smallest size problems.

No. variables	CPU for % Negative Eigenvalues		
	10	50	90
503	193.76	62.86	34.38
1005	523.36	503.34	461.04
2680	1009.5	1092.96	873.78

TABLE VII
Results of runs of the algorithm of Coleman and Hulbert (1989) for large sparse quadratic problems with box constraints

Run	Problem size			GOP Algorithm	
	m	n	k	Iter	CPU
1	20	25	0	2.61	21.552
2	20	25	50	6.87	50.703
3	20	25	100	11.32	121.718

m = Number of constraints
n = Number of concave variables
k = Number of linear variables

TABLE VIII
Random runs of indefinite quadratic problems for the GOP algorithm (Visweswaran and Floudas, 1993)

Table VIII gives the results of runs of the GOP algorithm of Visweswaran and Floudas (1993) for large indefinite quadratic problems. As the number of linear variables increases, the size of the linear relaxed dual problems solved at each iteration increases.

Conclusions

In this chapter, we have reviewed the proposed global optimization approaches for quadratic programming and quadratically constrained problems. Starting with the definition of the general quadratic programming problem and its classification, we discussed the local and global optimality conditions, as well as complexity issues. Subsequently, we discussed the theoretical and algorithmic developments in bilinear, concave, and indefinite quadratic programming. The important class of problems, from the application point of view, of quadratically constrained problems and decomposition based global optimization approaches for such problems were discussed next. After a brief discussion on the relation between quadratic programming and linear complementarity problems, we presented several examples of important application areas for quadratic programming and quadratically constrained problems. Finally, we discussed the computational results for a number of global optimization approaches.

From this review, it becomes clear that there has been a significant progress, especially in the last decade, towards the theoretical and algorithmic development of global optimization approaches for quadratic programming and quadratically constrained problems. At the same time however, the available computational studies of proposed global optimization algorithms on either randomly generated test problems or actual applications of engineering design and control described in section 11, indicate the current limitations and the need for future work. The difficulty of the reviewed classes of global optimization problems can be attributed to two main features. The first characteristic is the degree of nonconvexity which is associated with the number and balance of the mixed-sign eigenvalues of the Hessian Q for the general quadratic programming problem for instance. The second characteristic is related to the sparsity of the the objective function and constraints, as well as the relative contribution of the nonconvex terms versus the linear or convex terms. Based on the existing computational experience, there now exist a number of efficient global optimization algorithms that can address medium to large size sparse concave quadratic programming problems. The classes of bilinear and indefinite quadratic programming appear to be more challenging, and the existing global optimization approaches are effective only for small to medium size problems that have desirable sparsity characteristics. The important class of quadratically constrained problems, for which a few existing global optimization algorithms can be applied, is also very challenging even though very encouraging computational results have been reported for application problems whose mathematical structure is exploited effectively.

The classes of bilinear and indefinite quadratic programming, and quadratically constrained problems with linear or quadratic objective function deserve serious attention with respect to both new theoretical, algorithmic, and computational studies so as to succeed in addressing medium to large size global optimization problems effectively. Furthermore, it is expected that global optimization approaches that take advantage of the special mathematical structure exhibited in a number of important application areas may become more efficient and capable of addressing these problems at the expense however of being less general.

Acknowledgements

Financial support from the National Science Foundation under Grants CBT-8857013 and CTS-9221411 is gratefully acknowledged.

References

J. Ackermann. *Robust Control: Systems with Uncertain Physical Parameters.* Springer–Verlag, Berlin, 1993.

A. Aggarwal and C. A. Floudas. A decomposition approach for global optimum search in QP, NLP and MINLP problems. *Annals of Oper. Res.*, 25:119–146, 1990.

F. A. Al-Khayyal. Jointly Constrained Bilinear Programs and Related Problems: An Overview. *Computers in Mathematical Applications*, 19:53–62, 1990.

F. A. Al-Khayyal and J. E. Falk. Jointly constrained biconvex programming. *Math. of Oper. Res.*, 8(2):273 – 286, 1983.

F. A. Al-Khayyal, R. Horst, and P. M. Pardalos. Global Optimization of Quadratic Functions Subject to Quadratic Constraints. *Annals of Operations Research*, 1990.

S. Bali. *Minimization of a Concave function on A Bounded Convex Polyhedron*. PhD thesis, University of California, Los Angeles, 1973.

B. R. Barmish. New Tools for Robustness Analysis. 1981.

D. P. Baron. Quadratic Programming with Quadratic Constraints. *Naval Research Logistics Quarterly*, 19:253–260, 1972.

M.S. Bazaraa and C.M. Shetty. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, Inc., New York, 1979.

Mohamed Ben-Daya and C. M. Shetty. Polynomial barrier function algorithms for convex quadratic programming. *Arabian J. Sci. Eng.*, 15(4B):657–670, 1990.

A. Ben-Tal, G. Eiger, and V. Gershovitz. Global Minimization by Reducing the Duality Gap. *Mathematical Programming, Accepted*, 1993.

A. Ben-Tal and V. Gershovitz. Computational Methods for the Solution of the Pooling/Blending Problem. Technical report, Technion-Israel Institute of Technology, Haifa, Israel, 1992.

J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.*, 4:238–252, 1962.

S. BenSaad. *An algorithm for a class of nonlinear nonconvex optimization problems*. PhD thesis, University of California, Los Angeles, 1989.

Y. M. Berschanskii and M. V. Meerov. The Complementarity Problem: Theory and Methods of Solution. *Automation and Remote Control*, 44(6):687–710, 1983.

V.P. Bullatov and L.I. Kasinskaya. Some methods of concave minimization on a convex polyhedron and their applications (Russian). Methods of Optimization and their applications. *"Nauka" Sibirsk. Otdel, Novosibirsk*, pages 71–80, 1982.

A.V. Cabot and R.L. Francis. Solving nonconvex quadratic minimization problems by ranking the extreme points. *Oper. Res.*, 18:82–86, 1970.

M. J Carillo. A Relaxation For the Minimization of a Quasiconcave Function On a Convex Polyhedron. *Mathematical Programming*, 13:69–80, 1977.

S.J. Chung and K.G. Murty. *Polynomially Bounded Ellipsoid Algorithm For Convex Quadratic Programming*, pages 439–485. Nonlinear Programming 4. Academic Press, New York, 1981.

T. F. Coleman and L. A. Hulbert. A Direct Active Set Algorithm for Large Sparse Quadratic Programs with Simple Bounds. *Mathematical Programming*, 45:373–406, 1989.

Rchard W. Cottle, Jong-Shi Pang, and Richard E. Stone. *The Linear Complementarity Problem*. Academic Press, Inc., London, 1992.

I. Czochralska. Bilinear programming. *Zastosow. Mat.*, 17:495–514, 1982a.

I. Czochralska. The method of bilinear programming for nonconvex quadratic problems. *Zastosow . Mat.*, 17:515–525, 1982b.

Gabriele Danninger and Immanuel Bomze. Using Copositivity for global optimality criteria in concave quadratic programming problems. *Mathematical Programming*, 62:575–580, 1993.

R. R. E. de Gaston and M. G. Sofonov. Exact calculation of the multiloop stability margin. *IEEE Transactions on Automatic Control*, 2:156–171, 1988.

R. S. Dembo and U. Tulowitzki. On the Minimization of Quadratic Functions Subject to Box Constraints. *SIAM J. Sci. Stat. Comp.*, 1984.

M.E. Dyer. The complexity of vertex enumeration methods. *Math. Oper. Res.*, 8:381–402, 1983.

J.E. Falk. An Algorithm For Locating Approximate Global Solutions of Nonconvex Separable Problems. Technical Report T-262, Program in Logistics, George Washington University, Washington, D.C, 1972.

J.E. Falk. A linear max-min problem. *Math. Progr.*, 5:169–188, 1973.

J.E. Falk and K.L. Hoffman. A successive underestimating method for concave minimization problems. *Math. of Oper. Res.*, 1:251–259, 1976.

J.E. Falk and R.M. Soland. An algorithm for separable nonconvex programming problems. *Manag. Sci.*, 15(9):550–569, 1969.

C. A. Floudas, B. Jaumard, and V. Visweswaran. Decomposition Techniques for Global Optimization. In Preparation., 1993.

C. A. Floudas and P. M. Pardalos. *A Collection of Test Problems for Constrained Global Optimization Algorithms*, volume 455 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany, 1990.

C. A. Floudas and P. M Pardalos. *Recent Advances in Global Optimization*. Princeton Series in Computer Science. Princeton University Press, Princeton, New Jersey, 1992.

C. A. Floudas and V. Visweswaran. A Primal-Relaxed Dual Global Optimization Approach. *Journal of Optimization Theory and Applications*, 78(2), 187–225, 1993.

C.A. Floudas and A. Aggarwal. A Decomposition Strategy for Global Optimum Search in the Pooling Problem. *ORSA Journal on Computing*, 2(3):225–235, 1990.

C.A. Floudas, A. Aggarwal, and A.R. Ciric. Global optimum search for nonconvex NLP and MINLP problems. *Computers and Chemical Engineering*, 13(10):1117–1132, 1989.

C.A. Floudas and V. Visweswaran. A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs: I. Theory. *Comp. &Chem. Eng.*, 14:1397–1417, 1990.

A. M. Frieze. A Bilinear Programming Formulation of the 3-Dimensional Assignment Problem. *Mathematical Programming*, 7:376–379, 1974.

G. Gallo and A. Ulkucu. Bilinear Programming : An Exact Algorithm. *Math. Prog.*, 12:173, 1977.

A.M. Geoffrion. Generalized Benders Decomposition. *Journal of Optimization Theory and Applications*, 10(4):237–260, 1972.

F. Gianessi and G. Tomasin. *Nonconvex Quadratic Programming, Linear Complementarity Problems and Integer Programming*, pages 162–201. North-Holland, Amsterdam, Holland, 1974.

F. Giannessi and F. Niccolucci. *Connections Between Nonlinear and Integer Programming Problems*, pages 161–176. Academic Press, New York, NY, 1976.

F. Glover. Convexity Cuts and Cut Search. *Oper. Res.*, 21:123–134, 1973.

D. Goldfarb and S. Liu. An $O(n^3L)$ primal interior point algorithm for convex quadratic programming. *Mathematical Programming*, 49(3):325–340, 1991.

Frieda Granot and Jadranka Skorin-Kapov. Towards a strongly polynomial algorithm for strictly convex quadratic programs: An extension of Tardos' algorithm. *Math Programming*, 46(2):225–236, 1990.

L. Grippo and S. Lucidi. A differentiable exact penalty function for bound constrained quadratic programming problems. *Optimization*, 22(4):557–578, 1991.

W.W. Hager, P.M. Pardalos, I.M. Roussos, and H.D. Sahinoglou. Active constraints, indefinite quadratic test problems, and complexity. *Journal of Optimization Theory and Applications*, 68(3):499–511, 1991.

C. G. Han, P. M. Pardalos, and Ye. Y. Computational Aspects of An Interior Point Algorithm For Quadratic Programming Problems with Box Constraints. In *Large-Scale Numerical Optimization*. SIAM, 1990.

C. G. Han, P. M. Pardalos, and Ye. Y. On the Solution of Indefinite Quadratic Problems using an Interior-Point Algorithm. *Informatica*, 3(4):474–496, 1992.

P. Hansen and B. Jaumard. Reduction of Indefinite Quadratic Programs to Bilinear Programs. *Journal of Global Optimization*, 2:41–60, 1992.

P. Hansen, B. Jaumard, and S.H. Lu. An Analytical Approach To Global Optimization. *Mathematical Programming*, 52:227–241, 1991.

P. Hansen, B. Jaumard, M. Ruiz, and J. Xiong. Global Minimization of Indefinite Quadratic Functions Subject to Box Constraints. Technical Report G-91-54, GERAD, École Polytechnique, Université McGill, Montreal, CA, 1991.

C.A. Haverly. Studies of the Behaviour of Recursion for the Pooling Problem. *ACM SIGMAP Bulletin*, 25:19, 1978.

C.A. Haverly. Behaviour of Recursion Model - More Studies. *SIGMAP Bulletin*, 26:22, 1979.

C. D Heising-Goodman. A Survey of Methodology For the Global Minimization of Concave Functions Subject to Convex Constraints. *OMEGA, Intl. Jnl. of Mgmt. Sct.*, 9:313–319, 1981.

R. Horst and H. Tuy. Global Optimization: Deterministic Approaches. Springer-Verlag, Berlin, 1993.

S.E. Jacobsen. Convergence of a Tuy-type algorithm for concave minimization subject to linear inequality constraints. *Appl. Math. Optim.*, 7:1–9, 1981.

J. J. Judice and G. Mitra. Reformulation of Mathematical Programming Problems as Linear Complementarity Problems and Investigation of their Solution Methods. *Journal of Optimization Theory and Applications*, 57:123–149, 1988.

B. Kalantari. *Large Scale Global Minimization of Linearly Constrained Concave Quadratic Func-*

tions and Related Problems. PhD thesis, Computer Science Department, University of Minnesota, Minneapolis, MN, 1984.
K. Kalantari and J.B. Rosen. An Algorithm for Global Minimization of Linearly Constrained Concave Quadratic Functions. *Math. Oper. Res.*, 12(3):544, 1987.
N. Karmarkar. An interior-point approach to NP-complete problems. I. *Contemporary Mathematics*, 114:297–308, 1990.
G. Kedem and H. Watanabe. Optimization Techniques for IC Layout and Compaction. In *Proc. IEEE Intern. Conf. Computer Design: VLSI in Computers*, pages 709–713, 1983.
L. Khachiyan. A Polynomial Algorithm in Linear Programming. *Soviet Math. Doklady*, 20:191–194, 1979.
K. Kleibohm. Bemerkungen zum problem der nichtkonvex Programmierung (Remarks on the Nonconvex Programming Problem). *Unternehmensf.*, 11:49–60, 1967.
H. Konno. A Cutting Plane Algorithm For Solving Bilinear Programs. *Mathematical Programming*, 11:14–27, 1976.
H. Konno. Maximization of a convex quadratic function subject to linear constraints. *Math. Progr.*, 11:117–127, 1976.
H. Konno. Maximizing a convex quadratic function over a hypercube. *J. Oper. Res. Soc. Japan*, 23(2):171–189, 1980.
H. Konno. An algorithm for solving bilinear knapsack problems. *J. Oper. Res. Soc. Japan*, 24(4):360–373, 1981.
H. Konno and T. Kuno. Generalized Linear Multiplicative and Fractional Programming. Manuscript, 1989a.
H. Konno and T. Kuno. Linear Multiplicative Programming. Technical Report IHSS Report 89-13, Institute of Human And Social Sciences, Tokyo Institute of Technology, 1989b.
H. Konno and Y. Yajima. Solving Rank Two Bilinear Programs By Parametric Simplex Algorithms. Technical Report IHSS Report 90-17, Institute of Human And Social Sciences, Tokyo Institute of Technology, 1990.
H. Konno, Y. Yajima, and T. Matsui. Parametric Simplex Algorithm For Solving Special Cases of Nonconvex Minimization Problems. Technical Report IHSS Report 90-16, Institute of Human And Social Sciences, Tokyo Institute of Technology, 1990.
P.L. Kough. The indefinite quadratic programming problem. *Oper. Res.*, 27(3):516–533, 1979.
M. K. Kozlov, S. P. Tarasov, and L. G Hačijan. Polynomial Solvability of Convex Quadratic Programming. *Soviet Math. Doklady*, 20:1108–1111, 1979.
S. L. Krynski. Minimization of A Concave Function Under Linear Constraints (Modification of Tuy's Method). In *Survey of Mathematical Programming (Proc. 9'th Intl. Math. Prog. Symp. Budapest, 1976)*. North-Holland, Amsterdam, 1979.
H. W. Kuhn and A. W. Tucker. Nonlinear Programming. In J. Neyman, editor, *Proc. of the Second Berkeley Symposium on Math. Stat. and Prob.*, pages 481–492. U. of California Press, Berkeley, California, 1951.
L.S. Lasdon, A.D. Waren, S. Sarkar, and F. Palacios-Gomez. Solving the Pooling Problem Using Generalized Reduced Gradient and Successive Linear Programming Algorithms. *ACM SIGMAP Bulletin*, 27:9, 1979.
R. Lazimy. Mixed-Integer Quadratic Programming. *Math. Prog.*, 22:332, 1982.
R. Lazimy. Improved Algorithm for Mixed-Integer Quadratic Programs and a Computational Study. *Math. Prog.*, 32:100, 1985.
C. E. Lemke. *A Survey of Complementarity Theory*, pages 213–239. Wiley, N. Y., 1980.
W. B. Liu and C. A. Floudas. A Remark on the GOP Algorithm for Global Optimization. *Journal of Global Optimization*, 3(3), 519-521, 1993a.
W. B. Liu and C. A. Floudas. Convergence of the GOP Algorithm for a Large Class of Smooth Optimization Problems. Submitted for publication, 1993b.
K. Maling, S.H. Mueller, and W.R. Heller. On finding most optimal rectangular package plans. In *Proceeding of the 19th Design Automation Conference*, pages 663–670, 1982.
T. H. Matheiss and D. S Rubin. A Survey and Comparison of Methods for Finding all the Vertices of Convex Polyhedral Sets. *Mathematics of Operations Research*, 5:167–185, 1980.
P. C McKeown. Extreme Point Ranking Algorithms: A Computational Survey. in Computers and Mathematical Programming, U. S. Government Printing Office, Washington, D.C, 1986.
G. Mitra. An Exposition of the (Linear) Complementarity Problem. *Intl. Jnl. Math. Edu. Sci. Tech.*, 10(3):401–416, 1979.

J. Mockus. *Bayesian Approach to Global Optimization*. Kluwer Academic Publishers, Amsterdam, Holland, 1989.

Renato D. C. Monteiro and Ilan Adler. Interior path following primal-dual algorithms. II: Convex quadratic programming. *Mathematical Programming*, 44(1):43–66, 1989.

Renato D. C. Monteiro, Ilan Adler, and Maricio G. C. Resende. A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension. *Math. Oper. Res.*, 15(2):191–214, 1990.

B. A. Murtagh and M. A. Saunders. *MINOS5.0 Users Guide*. Systems Optimization Laboratory, Dept. of Operations Research, Stanford University, CA., 1988. Appendix A: MINOS5.2, Technical Report SOL 83-20.

K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Heldermann Verlag, Berlin, 1988.

K.G. Murty. Solving The Fixed-Charge Problem By Ranking the Extreme Points. *Operations Research*, 16:268–279, 1968.

K.G. Murty and S.N. Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Math. Progr.*, 39:117–129, 1987.

J. F. Nash. Noncooperative Games. *Ann. Math.*, 54:286–295, 1951.

Yu. E. Nesterov. Polynomial methods in the linear and quadratic programming. *Sov. J. Comput. Syst. Sci.*, 26(5):98–101, 1988.

P. M. Pardalos and N. Kovoor. An algorithm for a singly constrained class of quadratic programs subject upper and lower bounds. *Mathematical Programming*, 46:321–328, 1990.

P. M. Pardalos and S. A. Vavasis. Quadratic Programming with One Negative Eigenvalue is NP-Hard. Submitted for Publication, 1990.

P.M. Pardalos. Global Optimization Algorithms for Linearly Constrained Indefinite Quadratic Problems. *Comput. Math. Appl.*, 21:87–97, 1991.

P.M. Pardalos, J.H. Glick, and J.B. Rosen. Global Minimization of Indefinite Quadratic Problems. *Computing*, 39:281–291, 1987.

P.M. Pardalos and J.B. Rosen. Methods for global concave minimization: A bibliographic survey. *SIAM Review*, 28(3):367–379, 1986.

P.M. Pardalos and J.B. Rosen. *Constrained global optimization: Algorithms and applications*, volume 268 of *Lecture Notes in Computer Science*. Springer Verlag, Berlin, Germany, 1987.

P.M. Pardalos and G. Schnitger. Checking Local Optimality in Constrained Quadratic Programming is NP-Hard. *Operations Research Letters*, 7(1):33–35, 1988.

A.T. Phillips and J.B. Rosen. A parallel algorithm for constrained concave quadratic global minimization. *Math. Progr.*, 42:421–448, 1988.

A.T. Phillips and J.B. Rosen. Guaranteed ϵ-approximate Solution for Indefinite Quadratic Global Minimization. *Naval Research Logistics*, 37:499–514, 1990.

Polycarpos Psarris and C. A. Floudas. Robust Stability Analysis of Linear and Nonlinear Systems with Real Parameter Uncertainty. *Submitted for publication*, 1994.

M. Raghavachari. On Connections Between Zero-One Integer Programming and Concave Programming Under Linear Constraints. *Operations Research*, 17:680–684, 1969.

G.R. Reeves. Global minimization in non-convex all-quadratic programming. *Manag. Sci.*, 22:76–86, 1975.

A. H. G. Rinnooy-Kan and G. T. Timmer. Stochastic Methods for Global Optimization. In *Computational Mathematical Programming*. Springer Verlag, Berlin, 1985.

A. H. G. Rinnooy-Kan and G. T. Timmer. Stochastic Global Optimization Methods. Part I: Clustering Methods. *Mathematical Programming*, 39:27, 1987.

K. Ritter. A method for solving maximum problems with a nonconcave quadratic objective function. *Z. Wahrscheinlihkeitstheorie Geb.*, 4:340–351, 1966.

J.B. Rosen. Global minimization of a linearly constrained concave function by partition of feasible domain. *Math. Oper. Res.*, 8:215–230, 1983.

J.B. Rosen. Performance of approximate algorithms for global minimization. *Math. Progr. Study*, 22:231–236, 1984.

J.B. Rosen and P.M. Pardalos. Global Minimization of Large-Scale Constrained Concave Quadratic Problems by Separable Programming. *Math. Prog.*, 34:163, 1986.

I. Rosenberg. Zero-One Optimization and Nonlinear Programming. *Revue Francaise d'Informatique et de Recherche Opérationnelle*, 6:95–97, 1972.

S. Sahni. Computationally Related Problems. *SIAM J. Comp.*, 3:262–279, 1974.

H. Sherali and A. Alameddine. Reformulation-Linearization Technique for Bilinear Programming-Problems. *Journal of Global Optimization*, 2(4):379–410, 1992.

H.D. Sherali and C.M. Shetty. A finitely convergent algorithm for bilinear programming problems using polar and disjunctive face cuts. *Math. Progr.*, 19:14–31, 1980.

N. Z. Shor. Dual Quadratic Estimates in Polynomial and Boolean Programming. *Annals of Operations Research*, 25:163–168, 1990.

L. M. C. Simões. Search for the Global Optimum of Least Volume Trusses. *Engineering Optimization*, 11:49–67, 1987.

R.M. Soland. An Algorithm for Separable Nonconvex Programming Problems II : Nonconvex Constraints. *Mgmt. Sci.*, 17(11):759, 1971.

D. C. Sorensen. Newton's Method With A Model Trust Region Modification. *SIAM J. Numer. Anal.*, 19:409–426, 1982.

K. Swarup. Indefinite Quadratic Programming with a Quadratic Constraint. *Ekonomickomatem. Obzor*, 4:69–75, 1966.

N.V. Thoai and H. Tuy. Convergent algorithms for minimizing a concave function. *Math. of Oper. Res.*, 5(4):556–566, 1980.

A. Torn and A. Zilinskas. *Global Optimization*, volume 350 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany, 1989.

H. Tuy. Concave programming under linear constraints. *Dokl. Akad. Nauk. SSSR*, 159:32–35, 1964. Translated: Soviet Math. Dokl.5 1964 pp.1437–1440.

H. Tuy. Global minimization of a difference of two convex functions. *Math. Prog. Study*, 30:150–182, 1987.

H. Tuy. Normal conical algorithm for concave minimization over polytopes. *Math. Prog.*, 51:229–245, 1991.

H. Vaish and C.M. Shetty. The bilinear programming problem. *Naval Res. Logist. Quart.*, 23:303–309, 1976.

H. Vaish and C.M. Shetty. A cutting plane algorithm for the bilinear programming problem. *Naval Res. Logist. Quart.*, 24:83–94, 1977.

C. Van de Panne. Programming with a Quadratic Constraint. *Management Science*, 12:709–815, 1966.

S. Vavasis. Quadratic Programming is in NP. *Info. Proc. Lett.*, 36:73–77, 1990.

S. Vavasis. *Nonlinear Optimization: Complexity Issues*. Oxford University Press, New York, 1991.

S. Vavasis. Local minima for indefinite quadratic knapsack problems. *Math. Program.*, 54:127–153, 1992.

V. Visweswaran and C. A. Floudas. New Properties and Computational Improvement of the GOP Algorithm For Problems With Quadratic Objective Function and Constraints. *Journal of Global Optmization*, 3(3), 439–462, 1993.

V. Visweswaran and C.A. Floudas. A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs: II. Application of Theory and Test Problems. *Comp. &Chem. Eng.*, 14:1419–1434, 1990.

Jack Warga. A Necessary and Sufficient Condition for a Constrained Minimum. *SIAM J. Optimization*, 2(4):665–667, 1992.

H. Watanabe. *IC layout generation and compaction using mathematical programming*. PhD thesis, Computer Sc. Dept. Univ. of Rochester, 1984.

L. A. Wolsey. A Resource Decomposition Algorithm for General Mathematical Programs. *Mathematical Programming Study*, 14:244–257, 1981.

Y. Ye. Interior-point algorithms for global optimization. *Ann. Oper. Res.*, 25:59–74, 1990.

Y. Ye. On Affine Scaling Algorithms for Nonconvex Quadratic Programming. *Mathematical Programming*, 56:285–300, 1992.

Yinyu Ye. An extension of Karmarkar's algorithm and the trust region method for quadratic programming. In *Proceedings of the Conference on Progress in Mathematical Programming. Interior-Point and Related Methods, Pacific Grove, California*, pages 49–63, 1989.

N.D. Zilverberg. *Global minimization for large scale linear constrained systems*. PhD thesis, Computer Sc. Dept., Univ. of Minnesota, 1983.

P.B. Zwart. Nonlinear programming: Counterexamples To Two Global Optimization Algorithms. *Oper. Res.*, 21(6):1260–1266, 1973.

P.B. Zwart. Global Maximization of a Convex Function with Linear Inequality Constraints. *Oper. Res.*, 22:602, 1974.