Quadratic Integer Programming: Complexity and Equivalent Forms

W. ART CHAOVALITWONGSE[1], IOANNIS P. ANDROULAKIS[2], PANOS M. PARDALOS[3]
[1] Department of Industrial and Systems Engineering, Rutgers University, Piscataway, USA
[2] Department of Biomedical Engineering, Rutgers University, Piscataway, USA
[3] Department of Industrial and Systems Engineering, University of Florida, Gainesville, USA

MSC2000: CE2

Article Outline

Keywords and Phrases

Quadratic zero-one programming; Indefinite quadratic programming; Complexity; Optimality conditions

Introduction

In this paper we consider a quadratic programming (QP) problem of the following form:

$$\begin{aligned} \min \quad & f(x) = \frac{1}{2} x^{\mathrm{T}} Q x + c^{\mathrm{T}} x \\ \text{s.t.} \quad & x \in D \end{aligned} \tag{1}$$

where D is a polyhedron in $\mathbb{R}^n$, $c \in \mathbb{R}^n$. Without any loss of generality, we can assume that Q is a real symmetric $(n \times n)$-matrix. If this is not the case, then the matrix Q can be converted to symmetric form by replacing Q by $(Q + Q^{\mathrm{T}})/2$, which does not change the value of the objective function $f(x)$. Note that if Q is positive semidefinite, then Problem (1) is considered to be a convex minimization problem. When Q is negative semidefinite, Problem (1) is considered to be a concave minimization problem. When Q has at least one positive and one negative eigenvalue (i. e., Q is indefinite), Problem (1) is considered to be an indefinite quadratic programming problem. We know that in the case of convex minimization problem, every Kuhn-Tucker point is a local minimum, which is also a global minimum. In this case, there are a number of classical optimization methods that can obtain the globally optimal solutions of quadratic convex programming problems. These methods can be found in many places in the literature. In the case of concave minimization over polytopes, it is well known that if the problem has an optimal solution, then an optimal solution is attained at a vertex of D. On the other hand, the global minimum is not necessarily attained at a vertex of D for infinite quadratic programming problems. In this case, from second order optimality conditions, the global minimum is attained at the boundary of the feasible domain. In this research, without loss of generality, we are interested in developing solution techniques to solve general (convex, concave and indefinite) quadratic programming problems.

Complexity of Quadratic Programming

In this section we discuss the complexity of quadratic programming problems. The complexity analysis can give an idea of the possibility of developing efficient algorithms for solving the problem. In [10], the QP was shown to be $\mathcal{NP}$-hard in the case of a negative definite matrix Q. The QP was also proven to be $\mathcal{NP}$-hard by reduction to the satisfiability problem [11], and reduction to the knapsack feasibility problem [5]. Moreover, it has also been shown that checking local optimality for the QP itself is an $\mathcal{NP}$-hard problem [11]. In addition, checking for strict convexity (checking local optimality as part of the second order necessary conditions) in the QP was proven to be $\mathcal{NP}$-hard [8]. In fact, finding a local minimum and proving local optimality of such a solution to the QP may take exponential time.

This is true even in the case of a small number of concave variables. For instance, although the matrix Q is of rank one with exactly one negative eigenvalue, the QP is still $\mathfrak{NP}$-hard [9]. However, a large number of negative eigenvalues does not necessarily make the problem harder to solve. For example, consider the following problem:

$$\begin{aligned} \min \quad & \frac{1}{2}x^{\mathrm{T}}Qx + c^{\mathrm{T}}x \\ \text{s.t.} \quad & x \geq 0\,. \end{aligned}$$

If the matrix Q has $(n-1)$ negative eigenvalues, then there must be at least $(n-1)$ active constraints at the optimal solution [3]. Correspondingly, it is sufficient to solve $(n-1)$ different problems, in each case setting $(n-1)$ of the constraints to equalities, to find the optimal solution. In general, if the matrix Q has $(n-k)$ negative eigenvalues, then we are required to solve $\frac{n!}{k!(n-k)!}$ independent problems. In addition, the total computational time required to solve this problem is proportional to $\frac{k^3c^kn!}{k!(n-k)!}$. Thus, if k is an constant and independent of n, then the computational time is bounded by a polynomial in n. On the other hand, if k grows with n, then the computational time can grow exponentially with n [3].

Equivalence Between Discrete and Continuous Problems

Before we show the equivalence between discrete and continuous programs, it is important to discuss an equivalence property between two extremum problems [2]. Therefore, we refer to the following theorem (see [2] for a proof).

Theorem 1 *Let $\bar{Z}$ and $\bar{X}$ be compact sets in $\mathbb{R}^n$, R be a closed set in $\mathbb{R}^n$, and let the following hypotheses hold.*

H₁) *$f: \mathbb{R}^n \to \mathbb{R}$ is a bounded function on $\bar{X}$, and there exists an open set $A \subset \bar{Z}$ and real number $\alpha,\ L > 0$ such that, for any $x,\ y \in S,\ f$ satisfies the following Hölder condition: $|f(x) - f(y)| \leq L\|x-y\|^{\alpha}$.*

H₂) *It is impossible to find $\varphi: \mathbb{R}^n \to \mathbb{R}$ such that*
- *(i) ϕ is continuous on $\bar{X}$,*
- *(ii) $\varphi(x) = 0,\ x \in \bar{Z};\ \varphi(x) > 0,\ x \in \bar{X} - \bar{Z}$,*
- *(iii) $\forall z \in \bar{Z}$, there exists a neighborhood $S(z)$ and a real $\bar{\varepsilon} > 0$ such that, for any $x \in S(z) \cap (\bar{X} - \bar{Z})$, $\varphi(x) \geq \bar{\varepsilon}\|x - z\|^{\alpha}$.*

Then a real μ_0 exists such that for any real $\mu \geq \mu_0$, $\min f(x),\ x \in \bar{Z} \cap R$ is equivalent to $\min[f(x) + \mu\varphi(x)],\ x \in \bar{X} \cap R$.

Now we can show an equivalence between discrete and continuous programs from the following theorem [2].

Theorem 2 *Let $e^T = (1, 1, \ldots, 1)$, $\bar{Z} = \mathbb{B}^n$, $\bar{X} = \{x \in \mathbb{R}^n;\ 0 \leq x \leq e\}$, $R = \{x \in \mathbb{R}^n;\ g(x) \geq 0\}$. Consider the problem*

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0, \quad x \in B^n\,, \end{aligned} \tag{2}$$

and the problem

$$\begin{aligned} \min \quad & [f(x) + \mu x^T(e - x)] \\ \text{s.t.} \quad & g(x) \geq 0, \quad 0 \leq x \leq e\,. \end{aligned} \tag{3}$$

Then we suppose that f verifies assumption H_1 from Theorem 1 with $\alpha = 1$; that is, it is bounded on $\bar{X}$ and Lipschitz continuous on an open set $A \supseteq \bar{Z}$. Subsequently, there exists some $\mu_0 \in \mathbb{R}$ such that $\forall \mu < \mu_0$ Problems (2) and (3) are equivalent.

Integer Programming Problems and Complementarity Problems

The connections between integer programs and complementarity problems can be exhibited by applying KKT conditions. The results can be generalized in the quadratic programming case [4].

Theorem 3 *Let us first assume*

3a) $f: \mathbb{R}^n \to \mathbb{R}$, $g: \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable functions.

3b) g(x) satisfies a constraint qualification condition at x^0 to ensure that KKT conditions are validated.

Then the nonlinear programming problem

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0, \quad x \geq 0\,, \end{aligned} \tag{4}$$

has an optimal solution x^0 if there exist $u^0 \in \mathbb{R}^n$, $y^0,\ v^0 \in \mathbb{R}^v$ such that (x^0, y^0, u^0, v^0) is an optimal solution to the following problem:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & f'(x) - y^T g'(x) - u = 0, \\ & g(x) - v = 0, \\ & y^T v = 0 \\ & x^T u = 0 \\ & x, y, u, v \geq 0\,. \end{aligned} \tag{5}$$

Proof 1 Necessity. If x^0 is an optimal solution to Problem (4), from KKT conditions we obtain (y^0, u^0) such that

$$\begin{aligned} f'(x^0) - y^{0^{\mathrm{T}}} g(x^0) - u^0 = &\quad 0, \\ g(x^0) \geq &\quad 0, \\ x^{0^{\mathrm{T}}} u^0 = &\quad 0, \\ x^0,\ y^0,\ u^0 \geq &\quad 0\,. \end{aligned}$$

Let $v^0 = g(x^0)$, then (x^0, y^0, u^0, v^0) is an optimal solution to Problem (5).
Sufficiency. The proof is trivial. □

We now generalize the results of Theorem 3 to the quadratic programming case. Consider the following problem

$$\begin{aligned} \min \quad & \frac{1}{2} x^{\mathrm{T}} Q x + c^{\mathrm{T}} x \\ \text{s.t.} \quad & Ax \geq b, \\ & x \in B^n\,, \end{aligned} \tag{6}$$

where Q is a symmetric matrix. Using Theorem 2, Problem (6) is equivalent to

$$\begin{aligned} \min \quad & \left[\frac{1}{2} x^{\mathrm{T}} (Q - 2\mu I) x + (c^{\mathrm{T}} + \mu e^{\mathrm{T}}) x\right] \\ \text{s.t.} \quad & Ax \geq b, \\ & x \leq e, \\ & x \geq 0\,. \end{aligned} \tag{7}$$

Applying Theorem 3 to Problem (7), we then obtain

$$\min \quad \left[\frac{1}{2} x^{\mathrm{T}} (Q - 2\mu I) x + (c^{\mathrm{T}} + \mu e^{\mathrm{T}}) x\right] \tag{8}$$

$$\text{s.t.} \quad c + Qx + \mu(e - 2x) - y^{\mathrm{T}} A + t = u\,, \tag{9}$$

$$b - Ax = v\,, \tag{10}$$

$$e - x = w, \tag{11}$$

$$x^{\mathrm{T}} u = 0\,, \tag{12}$$

$$y^{\mathrm{T}} v = 0\,, \tag{13}$$

$$t^{\mathrm{T}} w = 0\,, \tag{14}$$

$$x, y, t, u, v, w \geq 0\,. \tag{15}$$

Arrange the terms in (9), we then have $Qx - 2\mu x = -(c + \mu e) + y^{\mathrm{T}} A - t + u$. Consequently, (8) becomes $\min[\frac{1}{2}(c^{\mathrm{T}} + \mu e^{\mathrm{T}})x + \frac{1}{2}(b^{\mathrm{T}} y - e^{\mathrm{T}} t)$. From (12), (13), and (14), we have

$$\begin{aligned} x^{\mathrm{T}} u = &\quad 0, \\ 0 = &\quad y^{\mathrm{T}} v = y^{\mathrm{T}} b - y^{\mathrm{T}} A x, \\ 0 = &\quad t^{\mathrm{T}} w = t^{\mathrm{T}} e - t^{\mathrm{T}} x\,; \end{aligned}$$

therefore, $y^{\mathrm{T}} b = y^{\mathrm{T}} A x$ and $t^{\mathrm{T}} e = t^{\mathrm{T}} x$. Taken all together, Problem (6) is equivalent to the following problem.

$$\begin{aligned} \min \quad & \hat{c}^{\mathrm{T}} \hat{x} \\ \text{s.t.} \quad & \hat{A}\hat{x} + \hat{u} = \hat{b}, \\ & \hat{x}\hat{u} = 0, \\ & \hat{x}, \quad \hat{u} \geq 0\,, \end{aligned}$$

where

$$\begin{aligned} \hat{x}^{\mathrm{T}} = &\quad (x^{\mathrm{T}}, y^{\mathrm{T}}, t^{\mathrm{T}}), \\ \hat{u}^{\mathrm{T}} = &\quad (u^{\mathrm{T}}, v^{\mathrm{T}}, w^{\mathrm{T}}), \\ \hat{A} = &\quad \begin{pmatrix} -Q + 2\mu I & A^{\mathrm{T}} & -I \\ A & 0 & 0 \\ I & 0 & 0 \end{pmatrix}, \\ \hat{c}^{\mathrm{T}} = &\quad \frac{1}{2}(c^{\mathrm{T}} + \mu e^{\mathrm{T}} + e^{\mathrm{T}}, b^{\mathrm{T}}, e^{\mathrm{T}}), \\ \hat{b}^{\mathrm{T}} = &\quad (c^{\mathrm{T}}, b^{\mathrm{T}}, e^{\mathrm{T}})\,. \end{aligned}$$

Note that there are no restrictive assumptions made on Q, this transformation is applicable to the convex case as well as the nonconvex case.

Integer Programming Problems and Quadratic Integer Programming Problems

Integer programming is used to model a variety of important practical problems in operations research, engineering, and computer science. Consider the following linear zero-one programming problem:

$$\begin{aligned} \min \quad & c^{\mathrm{T}} x \\ \text{s.t.} \quad & Ax \leq b, \quad x_i \in \{0, 1\}, \quad (i = 1, \ldots, n) \end{aligned}$$

where A is a real $(m \times n)$-matrix, $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. Let $e^{\mathrm{T}} = (1, \ldots, 1) \in \mathbb{R}^n$ denote the vector whose components are all equal to 1. Then the zero-one integer linear programming problem is equivalent to the following concave minimization problem:

$$\begin{aligned} \min \quad & f(x) = c^{\mathrm{T}} x + \mu x^{\mathrm{T}} (e - x) \\ \text{s.t.} \quad & Ax \leq b,\ 0 \leq x \leq e \end{aligned}$$

where μ is a sufficiently large positive integer. We know that the function $f(x)$ is concave because $-x^{\mathrm{T}}x$ is concave.

The equivalence of the two problems is based on the facts that a concave function attains its minimum at a vertex and that $x^{\mathrm{T}}(x-e) = 0$, $0 \le x \le e$, implies $x_i = 0$ or 1 for $i = 1, \ldots, n$. We note that a vertex of the feasible domain is not necessarily a vertex of the unit hypercube $0 \le x \le e$, but the global minimum is attained only when $x^{\mathrm{T}}(e-x) = 0$, provided that μ is a sufficiently large number.

These transformation techniques can be applied to reduce quadratic zero-one problems to equivalent concave minimization problems. For instance, consider a quadratic zero-one problem of the following form:

$$\begin{aligned} \min \quad & f(x) = c^{\mathrm{T}}x + x^{\mathrm{T}}Qx \\ \text{s.t.} \quad & x \in \{0,1\} \end{aligned}$$

where Q is a real symmetric $(n \times n)$ matrix. Given any real number μ, let $\bar{Q} = Q + \mu I$ where I is the $(n \times n)$ unit matrix, and $\bar{c} = c - \mu e$. Because of $\bar{f}(x) = f(x)$, the above quadratic zero-one problem is equivalent to the problem:

$$\begin{aligned} \min f(x) \quad & = \bar{c}^{\mathrm{T}}x + x^{\mathrm{T}}\bar{Q}x \\ \text{s.t.} \quad & x_i \in \{0,1\}, \quad (i = 1, \ldots, n) \end{aligned}$$

In this case, if we choose μ such that $\bar{Q} = Q + \mu I$ becomes a negative semidefinite matrix (e. g., $\mu = -\lambda$, where λ is the largest eigenvalue of Q), then the objective function $\bar{f}(x)$ becomes concave and the constraints can be replaced by $0 \le x \le e$. Thus, this problem is equivalent to the minimization of a quadratic concave function over the unit hypercube [4].

Various Equivalent Forms of Quadratic Zero-One Programming Problems

The problem considered here is a quadratic zero-one program, which has the form

$$\begin{aligned} \min f(x) \quad & = x^{\mathrm{T}}Qx, \\ \text{s.t.} \quad & x_i \in \{0,1\}, \quad i = 1, \ldots, n, \end{aligned} \qquad (16)$$

where Q is an $n \times n$ matrix [6,7]. Throughout this section the following notation will be used.

- $\{0,1\}^n$: set of n dimensional 0–1 vectors.
- $R^{n \times n}$: set of $n \times n$ dimensional real matrices.
- R^n: set of n dimensional real vectors.

In order to formalize the notion of equivalence we need some definitions.

Definition 1 The problem P is "polynomially reducible" to problem P_0 if given an instance $I(P)$ of problem P, an instance $I(P_0)$ of problem P_0 can be obtained in polynomial time such that solving $I(P)$ will solve $I(P_0)$.

Definition 2 Two problems P_1 and P_2 are called "equivalent" if P_1 is "polynomially reducible" to P_2 and P_2 is "polynomially reducible" to P_1.

Consider the following three problems:

$$\begin{aligned} P: \ \min f(x) = \ & x^{\mathrm{T}}Qx, \quad x \in \{0,1\}^n, \\ & Q \in R^{n \times n}, \\ P_1: \ \min f(x) = \ & x^{\mathrm{T}}Qx + c^{\mathrm{T}}x, \quad x \in \{0,1\}^n, \\ & Q \in R^{n \times n}, c \in R^n. \\ P_2: \ \min f(x) = \ & x^{\mathrm{T}}Qx, \quad x \in \{0,1\}^n, \\ & Q \in R^{n \times n}, \\ & \sum_{i=1}^{n} x_i = k \text{ for some } k \\ \text{s.t.} \qquad & 0 \le k \le n, \\ & \text{where } x = (x_1, x_2, \ldots, x_n). \end{aligned}$$

Next we show that problems P, P_1, and P_2 are all "equivalent". Then, formulation P_2 will be used in the rest of the sections.

Lemma 1 *P is "polynomially reducible" to P_1.*

Proof 2 It is very easy to see that P is a special case of P_1. □

Lemma 2 *P_1 is "polynomially reducible" to P.*

Proof 3 Problem P_1 is defined as follows: $\min f(x) = x^{\mathrm{T}}Qx + c^{\mathrm{T}}x, x \in \{0,1\}^n, Q \in R^{n \times n}, c \in R^n$. If $Q = (q_{ij})$ then let $B = (b_{ij})$ where

$$b_{ij} = \begin{cases} q_{ij} & \text{if } i \ne j \\ q_{ij} + c_i & \text{if } i = j. \end{cases}$$

Since $x_i^2 = x_i$ (because $x_i \in \{0,1\}$), we have $g(x) = x^{\mathrm{T}}Bx = x^{\mathrm{T}}Qx + c^{\mathrm{T}}x$. So the following problem is equivalent to problem P_1 : $\min g(x) = x^{\mathrm{T}}Bx, x \in \{0,1\}^n, B \in R^{n \times n}$. □

Using Lemma 1 and Lemma 2, it is evident that P and P_1 are "equivalent".

Lemma 3 *P_2 is "polynomially reducible" to P.*

Proof 4 Problem P_2 is as follows: $\min f(x) = x^{\mathrm{T}}Qx$, $x \in \{0,1\}^n$, $Q \in R^{n\times n}, \sum_{i=1}^n x_i = k$ for some k s.t. $0 \le k \le n$. If $Q = (q_{ij})$ then let $M = 2[\sum_{j=1}^n \sum_{i=1}^n |q_{ij}|]+1$. Now, define the following problem P: $\min g(x) = x^{\mathrm{T}}Qx + M(\sum_{i=1}^n x_i - k)^2$ s.t. $x \in \{0,1\}^n$, $Q \in R^{n\times n}$. Let $x_b = (x_1^b, \ldots, x_n^b)$ and $x_0 = (x_1^0, \ldots, x_n^0)$ such that $\sum_{i=1}^n x_i^b \neq k$ and $\sum_{i=1}^n x_i^0 = k$, then $g(x_0) \le \frac{M-1}{2}$ as $\sum_{i=1}^n x_i^0 = k$, $g(x_b) \ge \frac{-(M-1)}{2} + M$ or $g(x_b) \ge \frac{M+1}{2}$ as $|\sum_{i=1}^n x_i^b - k| \ge 1$. Therefore, $g(x_0) < g(x_b)$ if $\sum_{i=1}^n x_i^b \neq k$ and $\sum_{i=1}^n x_i^0 = k$. Hence, if $\min g(x) = g(x_0)$ where $x_0 = (x_1^0, \ldots, x_n^0)$ then $\sum_{i=1}^n x_i^0 = k$. So $\min f(x) = \min g(x)$. From the above discussion, it can be easily seen that P_2 is "polynomially reducible" to P. □

The proof of Lemma 3 also illustrates how equality (knapsack) constraints in a quadratic zero-one program can be eliminated.

Lemma 4 *P is "polynomially reducible" to P_2.*

Proof 5 Let problem P be defined as follows: $\min f(x) = x^{\mathrm{T}}Qx$, $x \in \{0,1\}^n$, $Q \in R^{n\times n}$. Define a series of $(n+1)$ problems: $P_2(0), P_2(1), P_2(2), \cdots, P_2(n)$, where $P_2(j)$ is the following problem $\min f(x) = x^{\mathrm{T}}Qx$, $x \in \{0,1\}^n$, $Q \in R^{n\times n}$, $\sum_{i=1}^n x_i = j$. Let the minimum of the problem $P_2(j)$ be y_j, then the minimum of problem P is easily seen to be the min $\{y_0, y_1, \ldots, y_n\}$. □

Lemma 3 and Lemma 4 imply that P and P_2 are "equivalent". Since "equivalent" is a transitive relative, P, P_1, P_2 are all "equivalent".

Complexity of Quadratic Zero-One Programming Problems

Quadratic zero-one programming is a difficult problem. We next will show that the quadratic knapsack zero-one problem in (P_2) is a NP hard problem by proving that it is equivalent to the k-clique problem. A k-clique is a complete graph with k vertices.

k-clique Problem

Given a graph G=(V,E) (V is the set of vertices and E is the set of edges), does the graph G have a k-clique as one of its subgraphs?
k-clique problem is known to be NP-complete. We will show that the k-clique problem is "polynomially reducible" to problem P_2 defined in the previous subsection.

Theorem 4 *The k-clique problem is "polynomially reducible" to P_2.*

Proof 6 Problem P_2 was defined as min $f(x) = x^{\mathrm{T}}Qx$, s.t. $x_i \in \{0,1\}, i = 1, \cdots, n, \sum_{i=1}^n x_i = m$ for some $0 \le m \le n$. Given the graph $G = (V, E)$, define $Q = (q_{ij})$ such that

$$q_{ij} = \begin{cases} 0 & \text{if } (v_i, v_j) \in E \\ -1 & \text{if } (v_i, v_j) \notin E\,, \end{cases}$$

where $n = |V|, m = k$ (we are trying to find a k-clique). The meaning attached to the vector $\mathbf{x} \in \{0,1\}^n$ in problem P_2 is as follows

$$x_i = \begin{cases} 1 & \text{means that } v_i \text{ is in the clique,} \\ 0 & \text{means that } v_i \text{ is not in the clique}\,. \end{cases}$$

We can easily prove that the graph G has a k-clique if and only if min $f(x) = -k(k-1)$. So the k-clique problem is "polynomially reducible" to P_2. □

Problem P_2 is "equivalent" to P, so problem P is also NP-hard. Therefore, as the dimension of the problem increases, the necessary CPU time to solve the problem increases exponentially.

Quadratic Zero-One Programming and Mixed Integer Programming

In this section, we consider a quadratic zero-one programming problem in the following form:

$$\begin{aligned} \min f(x) &= x^{\mathrm{T}}Qx\,, \\ \text{s.t.} \quad & \sum_{i=1}^n x_i = k\,, \quad x \in \{0,1\}^n\,. \end{aligned} \tag{17}$$

Let Q be $n \times n$ matrix, whose each element $q_{i,j} \ge 0$. Define $x = (x_1, \ldots, x_n)$, where each x_i represents binary decision variables. We will show that the problem in (17) can be linearized as the following mixed integer programming problems. The first linearization technique is trivial and can be found elsewhere. Recently, more efficient linearization technique was introduced in [1]. In addition, the linearization technique for more general case (where $q_{i,j} \in real$) and multi-quadratic programming was also proposed in [1].

Conventional Linearization Approach

For each product $x_i x_j$ in the objective function of the problem (17) we introduce a new continuous variable, $x_{ij} = x_i x_j (i \neq j)$. Note that $x_{ii} = x_i^2 = x_i$ for $x_i \in \{0,1\}$. The equivalent mixed integer programming problem (MIP) is given by:

$$
\begin{aligned}
\min \quad & \sum_i \sum_j q_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_i = k, \\
& x_{ij} \le x_i, && \text{for } i, j = 1, \ldots, n (i \neq j) \\
& x_{ij} \le x_j, && \text{for } i, j = 1, \ldots, n (i \neq j) \\
& x_i + x_j - 1 \le x_{ij}, && \text{for } i, j = 1, \ldots, n (i \neq j) \\
& 0 \le x_{ij} \le 1, && \text{for } i, j = 1, \ldots, n (i \neq j)
\end{aligned}
\tag{18}
$$

where $x_i \in \{0,1\}, i, j = 1, \ldots, n$.

The main disadvantage of this approach is that the number of additional variables we need to introduce is $O(n^2)$, and the number of new constraints is also $O(n^2)$. The number of 0–1 variables remains the same.

A New Linearization Approach

Consider the following mixed integer programming problem:

$$
\begin{aligned}
\min_{x,y,s} g(s) \quad & = \sum_{i=1}^{n} s_i = e^{\mathrm{T}} s \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_i = k, \\
& Qx - y - s = 0, \\
& y \le \mu(e - x), \\
& x_i \in \{0,1\}, \quad \text{for } i = 1, \ldots, n \\
& y_i,\ s_i \ge 0, \quad \text{for } i = 1, \ldots, n\,.
\end{aligned}
\tag{19}
$$

where Q is an $n \times n$ matrix, whose each element $q_{i,j} \ge 0$.

In [1], the mixed integer 0–1 programming problem in (19) was proved equivalent to the quadratic zero-one programming in (17). The main advantage of this approach is that we only need to introduce $O(n)$ additional variables and $O(n)$ new constraints, where the number of 0–1 variables remains the same. This linearization technique proved more robust and more efficiently solving quadratic zero-one and multi-quadratic zero-one programming problems [1].

References

1. Chaovalitwongse WA, Pardalos PM, Prokoyev OA (2004) Reduction of multi-quadratic 0–1 programming problems to linear mixed 0–1 programming problems. Oper Res Lett 32(6):517–522
2. Giannessi F, Niccolucci F (1976) Connections between nonlinear and integer programming problems. Istituto Nazionale di Alta Matematica, Symposia Mathematica 19:161–176
3. Hager WW, Pardalos PM, Roussos IM, Sahinoglou HD (1991) Active constraints, indefinite quadratic programming, and test problems. J Optim Theor Appl 68(3):499–511
4. Horst R, Pardalos PM, Thoai NV (1995) Introduction to global optimization. Kluwer Academic Publishers CE3
5. Pardalos PM (1991) Global optimization algorithms for linearly constrained indefinite quadratic problems. Comput Math Appl 21:87–97
6. Pardalos PM, Rodgers G (1989) Parallel branch and bound algorithms for unconstrained quadratic zero-one programming. In: Sharda R et al. (eds) Impact of recent computer advances on operations research. North-Holland CE4
7. Pardalos PM, Rodgers G (1990) Computational aspects of a branch and bound algorithm for quadratic zero-one programming. Comput 45:131–144
8. Pardalos PM, Schnitger G (1988) Checking local optimality in constrained quadratic programming is $\mathcal{NP}$-hard. Oper Res Lett 7(1):33–35
9. Pardalos PM, Vavasis S (1991) Quadratic programming with one negative eigenvalue is $\mathcal{NP}$-hard. J Glob Optim 1:15–23
10. Sahni S (1974) Computationally related problems. SIAM J Comput 3:262–279
11. Vavasis S (1991) Nonlinear optimization: complexity issues. Oxford Univeristy Press CE3