Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 8, No. 1, pp. 75–87, January 2012

AN IMPROVED CONVEX 0-1 QUADRATIC PROGRAM REFORMULATION FOR QUADRATIC KNAPSACK PROBLEMS*

Shuhui Ji, Xiaojin Zheng and Xiaoling Sun†

Dedicated to the memory of Professor Paul Tseng.

Abstract: We present in this paper a new convex 0-1 QP reformulation for quadratic knapsack problems (QKP). This new reformulation improves the existing reformulation based on diagonal perturbation in the sense that the continuous relaxation of the new reformulation is tighter than or at least as tight as that of the existing reformulation. The improved reformulation is derived from matrix decomposition of the objective function and piecewise linearization of quadratic terms on $\{0,1\}^n$. We show that the optimal parameters in the reformulation can be obtained by solving an SDP problem. Extension to k-item quadratic knapsack problems is also discussed. Computational results are reported to demonstrate the effectiveness of the improved reformulation.

Key words: *quadratic knapsack problem, convex 0-1 QP reformulation, convex relaxation, SDP relaxation*

Mathematics Subject Classification: *90C10, 90C20, 90C22*

1 Introduction

Consider the following 0-1 quadratic knapsack problem:

$$
\begin{aligned}
\text{(QKP)} \qquad & \max\ q(x) := x^T Qx \\
& \text{s.t. } Ax \le b, \\
& \quad x \in \{0,1\}^n,
\end{aligned}
$$

where $Q = (q_{ij})_{n\times n}$ is a symmetric $n \times n$ matrix, $A = (a_{ij})_{m\times n} \ge 0$ is an $m \times n$ matrix and $b \in \mathbb{R}^m$. We assume that $\mathcal{F} = \{x \in \{0,1\}^n \mid Ax \le b\} \ne \emptyset$.

Problem (QKP) is a generalization of the classical quadratic 0-1 knapsack problem (see [10]), where a single capacity constraint is considered. Problem (QKP) with multiple capacity constraints is also called *multi-dimensional* quadratic knapsack problem. Since 0-1 linear knapsack problem is a special case of (QKP) with $q_{ij} = 0$ for $1 \le i < j \le n$, (QKP) is in general NP-hard. The reader is referred to [16] for a survey of the theory and algorithms for linear knapsack problems. Quadratic knapsack programming is a general model for many combinatorial optimization problems such quadratic assignment, graph partitioning and densest k-subgraph (see [10]). Typical examples of real-world applications of quadratic

*Research supported by National Natural Science Foundation of China under Grants 10971034, 11101092 and the Joint NSFC/RGC grant 71061160506.

†Corresponding author.

0-1 knapsack problems arise from task allocation, capital budgeting, VLSI and compiler design [9, 15].

Global optimization methods for (QKP) are of branch-and-bound framework which is based on upper bounding procedures via relaxation schemes and branching rules. Various techniques have been proposed for deriving upper bounds for (QKP), for instance, upper plane methods [10], linearization methods [2, 15], reformulation method [7], Lagrangian relaxation and decomposition [5, 7, 8]. These relaxations and bounds are specially constructed for (QKP) with a single capacity constraint and often require special properties of the objective function such as nonnegativity of q_{ij}. For a survey of methods for 0-1 quadratic knapsack problems with a single constraint, see [21].

Semidefinite programming (SDP) has been a powerful relaxation approach for many NP-hard combinatorial and nonconvex optimization problems due to the development of efficient interior-point method for SDP problems (see, e.g., [20, 26]). SDP relaxation methods have been developed for 0-1 linear programs (see [1, 18, 23]). It has been also shown that SDP relaxations yield tight bounds and good feasible solutions for unconstrained 0-1 quadratic problems and some of its extensions (see, e.g., [11, 19, 27, 28]). For (QKP) with a single capacity constraint, Helmberg *et al.* [13] derived several SDP relaxations based on reformulations of the capacity constraint. Various valid inequalities such as cover inequalities and triangle inequalities are also investigated in [13]. However, no approximation bounds for the SDP relaxations of (QKP) have been known in the literature.

Recently, Burer [6] showed that any binary and continuous nonconvex quadratic program with linear constraints can be expressed as a completely positive program, which is the dual of a copositive program. Sturm and Zhang [25] introduced copositive matrix cone over a domain and showed that any quadratically constrained quadratic program can be represented as a convex conic program over the dual cone of copositive matrices on the feasible region. We remark that expressing an NP-hard problem as a copositive programming does not resolve the inherent computational difficulty of the problem despite of its theoretical interest. Except for some special cases where LMI characterization of the copositive cone can be obtained, the copositive cones on the positive orthant or a domain have to be relaxed in order to derive a tractable relaxation.

Billionnet and Elloumi [3] proposed a 0-1 convex quadratic program reformulation for unconstrained 0-1 quadratic program (P): $\max\{q(x) := x^TQx \mid x \in \{0,1\}^n\}$. Using the equivalence $x_i^2 = x_i \Leftrightarrow x_i \in \{0,1\}$, $q(x)$ can be rewritten as $q_\rho(x) = x^T(Q - \mathrm{Diag}(\rho))x + \rho^T x$ for any $x \in \{0,1\}^n$ and $\rho \in \mathbb{R}^n$. Choosing ρ such that $Q - \mathrm{Diag}(\rho) \preceq 0$, we have the following convex relaxation for (P):

$$\beta(\rho) = \max\{q_\rho(x) \mid x \in [0,1]^n\}.$$

The tightest upper bound generated from the above relaxation is

$$\min_{Q-\mathrm{Diag}(\rho)\preceq 0} \max\{q_\rho(x) \mid x \in [0,1]^n\}.$$

It is shown in [3, 22, 24] that the optimal solution ρ^* to above problem can be reduced to an SDP problem. The corresponding equivalent reformulation of (P) is:

$$(\mathrm{P}_{\rho^*}) \qquad \max\{q_{\rho^*}(x) \mid x \in \{0,1\}^n\}.$$

This reformulation is referred to as *diagonal perturbed reformulation* of (P). Any continuous-based branch-and-bound method such as the mixed integer quadratic programming (MIQP) solver in CPLEX can be used to solve (P_{ρ^*}). The above reformulation technique has been also generalized in [4] to 0-1 quadratic program with linear constraints, in particular, the quadratic knapsack problem (QKP).

The purpose of this paper is to present an improved convex 0-1 quadratic program reformulation for (QKP). This improved reformulation is based on a general decomposition of the nonconvex objective function:

$$q(x) = x^T(Q - M)x + x^T Mx, \tag{1.1}$$

where M is a symmetric matrix in certain matrix cone and $Q-M \preceq 0$. By suitably choosing the matrix M and using the piecewise linearization of quadratic terms on $\{0,1\}^n$, we are able to get a convex 0-1 quadratic programming reformulation which is more efficient than the diagonal perturbed reformulation in [3, 4] in the sense that the continuous relaxation of the new reformulation is always tighter than or at least as tight as that of the diagonal perturbed reformulation of (QKP). We show that the problem of finding the optimal parameters in the improved reformulation can be reduced to an SDP problem. Comparison numerical results show that the improved reformulation is more efficient than the diagonal perturbed reformulation in terms of the computation time and number of nodes explored by the MIQP solver in CPLEX 12.1.

The paper is organized as follows. In Section 2, we present our main results. We first derive the improved 0-1 convex quadratic program reformulation of (QKP) via matrix decomposition and piecewise linear representation of quadratic terms in 0-1 variables. The SDP formulation for finding the optimal parameters in the matrix decomposition is then derived. Extension to k-item quadratic knapsack problems is also discussed. In Section 3, we report comparison numerical results on the effectiveness of the improved reformulation for standard randomly generated test problems from the literature. Some concluding remarks are given in Section 4.

Notations: Throughout the paper, we denote by $v(\cdot)$ the optimal objective value of an optimization problem $(\cdot)$. $\mathbb{R}^n$ denotes the n-dimensional Euclidean space and $\mathbb{R}^n_+$ the set of nonnegative vectors in $\mathbb{R}^n$. Let $e = (1, \ldots, 1)^T$ and let e_i denote the ith unit vector in $\mathbb{R}^n$. For any $a = (a_1, \ldots, a_n)^T \in \mathbb{R}^n$, $\mathrm{Diag}(a)$ denotes the diagonal matrix with diagonal elements $a_1, \ldots, a_n$. Let $\mathcal{S}$ and $\mathcal{S}_+$ denote the sets of $n \times n$ symmetric matrices and positive semidefinite matrices, respectively. The standard inner product in $\mathcal{S}$ is defined by $A \bullet B = \langle A, B\rangle = \mathrm{trace}(AB)$.

2 Improved Reformulation and SDP Relaxation

In this section, we present our main result on constructing the improved convex 0-1 quadratic program reformulation for (QKP). We first present the new reformulation by employing matrix decomposition and piecewise linear representation of quadratic term for 0-1 variables. We then show that the optimal matrix decomposition can be found by solving an SDP problem. Extensions to k-item quadratic knapsack problems are also discussed.

2.1 An Improved Reformulation

Let $\mathcal{P}$ and $\mathcal{N}$ denote the sets of nonnegative and nonpositive $n \times n$ matrices, i.e.,

$$\mathcal{P} := \{P = (P_{ij}) \in \mathcal{S} \mid P \geq 0\},\ \mathcal{N} := \{N = (N_{ij}) \in \mathbb{R}^{n\times n} \mid N \leq 0\}.$$

Consider the following decomposition of $q(x)$:

$$q(x) = x^T(Q - M)x + x^T Mx,$$

where $Q - M \preceq 0$ and $M = \text{Diag}(\rho) + P + N \in \mathcal{S}$ with $\rho \in \mathbb{R}^n$, $P \in \mathcal{P}$ and $N \in \mathcal{N}$.

For any $x_i,\ x_j \in \{0,1\}$, it always holds

$$x_i x_j = \min(x_i, x_j),\ x_i x_j = \max(0, x_i + x_j - 1).$$

Thus, for any $x \in \{0,1\}^n$, we can rewrite $q(x)$ as

$$\begin{aligned} q(x) &= x^T(Q - \text{Diag}(\rho) - P - N)x + x^T(\text{Diag}(\rho) + P + N)x \\ &= x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n} [P_{ij} x_i x_j + N_{ij} x_i x_j] \\ &= x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n} [P_{ij} s_{ij} - N_{ij} t_{ij}], \end{aligned} \tag{2.1}$$

where $s_{ij} = \min(x_i, x_j)$ and $t_{ij} = -\max(0, x_i + x_j - 1)$.

In view of (2.1), we obtain the following equivalent reformulation of (QKP):

$$\begin{aligned} &\max\ x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n} [P_{ij} s_{ij} - N_{ij} t_{ij}] \\ &\text{s.t. } Ax \le b,\ x \in \{0,1\}^n, \\ &\qquad s_{ij} = \min(x_i, x_j),\ t_{ij} = -\max(0, x_i + x_j - 1),\ i,j = 1,\ldots,n. \end{aligned}$$

where $Q - \text{Diag}(\rho) - P - N \preceq 0$, $P \in \mathcal{P}$ and $N \in \mathcal{N}$. Since $P_{ij} \ge 0$, the constraint $s_{ij} = \min(x_i, x_j)$ can be relaxed to two linear inequalities $s_{ij} \le x_i$ and $s_{ij} \le x_j$ without affecting the optimal solution of the above problem. Similarly, we can relax $t_{ij} = -\max(0, x_i + x_j - 1)$ to $t_{ij} \le 0$ and $t_{ij} \le 1 - x_i - x_j$. Therefore, the above reformulation is equivalent to the following convex 0-1quadratic program:

$$\begin{aligned} (\text{QKP}(\rho, P, N)) \qquad &\max\ x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n} [P_{ij} s_{ij} - N_{ij} t_{ij}] \\ &\text{s.t. } Ax \le b,\ x \in \{0,1\}^n, \\ &\qquad s_{ij} \le x_i,\ s_{ij} \le x_j,\ i,j = 1,\ldots,n, \\ &\qquad t_{ij} \le 0,\quad t_{ij} \le 1 - x_i - x_j,\ i,j = 1,\ldots,n. \end{aligned}$$

2.2 Optimal matrix decomposition and SDP relaxation

As $(\text{QKP}(\rho, P, N))$ is a convex 0-1 quadratic program, any continuous-based branch-and-bound method can be applied to $(\text{QKP}(\rho, P, N))$ for searching an exact solution. Relaxing $\{0,1\}^n$ to $[0,1]^n$, the continuous relaxation of $(\text{QKP}(\rho, P, N))$ is

$$(\overline{\text{QKP}}(\rho, P, N)) \qquad \max\ x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n} [P_{ij} s_{ij} - N_{ij} t_{ij}]$$

$$\text{s.t. } Ax \le b,\ 0 \le x \le e, \tag{2.2}$$

$$s_{ij} \le x_i,\ s_{ij} \le x_j,\ i,j = 1,\ldots,n, \tag{2.3}$$

$$t_{ij} \le 0,\quad t_{ij} \le 1 - x_i - x_j,\ i,j = 1,\ldots,n. \tag{2.4}$$

Notice that $(\overline{\text{QKP}}(\rho, P, N))$ is a convex quadratic program and thus is polynomially solvable by interior-point method. For any (ρ, P, N) satisfying $Q - \text{Diag}(\rho) - P - N \preceq 0$, $P \in \mathcal{P}$

and $N \in \mathcal{N}$, solving $(\overline{\mathrm{QKP}}(\rho, P, N))$ gives an upper bound of (QKP). A natural question is: How to choose parameters (ρ, P, N) such that $v(\overline{\mathrm{QKP}}(\rho, P, N))$ provides the tightest upper bound to (QKP)?

It is clear that the tightest upper bound is given by the optimal value of the following problem.

$$\begin{array}{lll} \text{(TUB)} & & \min v(\overline{\mathrm{QKP}}(\rho, P, N)) \\ & & \text{s.t. } Q - \mathrm{Diag}(\rho) - P - N \preceq 0,\ \rho \in \mathbb{R}^n,\ P \in \mathcal{P},\ N \in \mathcal{N}. \end{array}$$

The following theorem shows that problem (TUB) is equivalent to an SDP problem:

Theorem 2.1. *The problem (TUB) is equivalent to the following SDP problem:*

$$\begin{array}{lll} (\mathrm{TUB_s}) & \min & \tau \\ & \text{s.t.} & \begin{pmatrix} -\tau + \vartheta(E, \mu, \xi) & \frac{1}{2}\Gamma(\rho, \mu, \eta, \xi, B, C, E)^T \\ \frac{1}{2}\Gamma(\rho, \mu, \eta, \xi, B, C, E) & \psi(\rho, P, N) \end{pmatrix} \preceq 0, \\ & & B + C = P,\ D + E = -N, \\ & & (\rho, \mu, \eta, \xi) \in \mathbb{R}^n \times \mathbb{R}^m_+ \times \mathbb{R}^n_+ \times \mathbb{R}^n_+, \\ & & P, -N, B, C, D, E \in \mathcal{P}, \end{array}$$

where

$$\psi(\rho, P, N) = Q - \mathrm{Diag}(\rho) - P - N, \tag{2.5}$$

$$\Gamma(\rho, \mu, \eta, \xi, B, C, E) = \rho - A^T\mu + \eta - \xi - \sum_{i=1}^{n}\sum_{j=1}^{n}(E_{ij}e_i + E_{ij}e_j - B_{ij}e_i - C_{ij}e_j), \tag{2.6}$$

$$\vartheta(E, \mu, \xi) = \sum_{i,j=1}^{n} E_{ij} + \mu^T b + e^T\xi. \tag{2.7}$$

Proof. Associate the following multipliers to the constraints in $(\overline{\mathrm{QKP}}(\rho, P, N))$:

- $\mu \in \mathbb{R}^m_+,\ \eta \in \mathbb{R}^n_+$ and $\xi \in \mathbb{R}^n_+$ to the constraints in (2.2);
- $B = (B_{ij}) \in \mathcal{P}$ and $C = (C_{ij}) \in \mathcal{P}$ to the constraints in (2.3);
- $D = (D_{ij}) \in \mathcal{P}$ and $E = (E_{ij}) \in \mathcal{P}$ to the constraints in (2.4).

Let $\omega = (\mu, \eta, \xi, B, C, D, E)$. The nonnegative restriction on the associated multipliers can be expressed as $\omega \geq 0$. The Lagrangian relaxation of $(\overline{\text{QKP}}(\rho, P, N))$ is

$$\begin{aligned} d(\omega) &= \max\{x^T(Q - \text{Diag}(\rho) - P - N)x + \rho^T x + \sum_{i,j=1}^{n}(P_{ij}s_{ij} - N_{ij}t_{ij}) \\ &\quad + \eta^T x - \xi^T(x - e) - \mu^T(Ax - b) + \sum_{i,j=1}^{n}[-B_{ij}(s_{ij} - x_i) - C_{ij}(s_{ij} - x_j)] \\ &\quad + \sum_{i,j=1}^{n}[-D_{ij}t_{ij} - E_{ij}(t_{ij} + x_i + x_j - 1)] \mid x \in \mathbb{R}^n,\ s_{ij}, t_{ij} \in \mathbb{R},\ i, j = 1, \ldots, n\} \\ &= \sum_{i,j=1}^{n} E_{ij} + \mu^T b + e^T \xi + \max_{x \in \mathbb{R}^n}\{x^T(Q - \text{Diag}(\rho) - P - N)x + (\rho + \eta - \xi - A^T\mu)^T x \\ &\quad - \sum_{i,j=1}^{n}(E_{ij}x_i + E_{ij}x_j - B_{ij}x_i - C_{ij}x_j) \mid B + C = P,\ D + E = -N\}. \end{aligned}$$

Then, the Lagrangian dual of $(\overline{\text{QKP}}(\rho, P, N))$ is $\min_{\omega \geq 0} d(\omega)$. By strong duality of convex quadratic program, we have

$$v(\overline{\text{QKP}}(\rho, P, N)) = \min_{\omega \geq 0} d(\omega).$$

Thus, we have

$$\begin{aligned} v(\text{TUB}) &= \min\{v(\overline{\text{QKP}}(\rho, P, N)) \mid Q - \text{Diag}(\rho) - P - N \preceq 0,\ \rho \in \mathbb{R}^n,\ P \in \mathcal{P},\ N \in \mathcal{N}\} \\ &= \min_{\omega \geq 0}\{d(\omega) \mid Q - \text{Diag}(\rho) - P - N \preceq 0,\ \rho \in \mathbb{R}^n,\ P \in \mathcal{P},\ N \in \mathcal{N}\} \\ &= \min\ \tau \\ &\quad\ \text{s.t.}\ \ \tau \geq x^T\psi(\rho, P, N)x + \Gamma(\rho, \mu, \eta, \xi, B, C, E)^T x + \vartheta(E, \mu, \xi),\ \forall x \in \mathbb{R}^n, \\ &\quad\quad\quad B + C = P,\ D + E = -N, \\ &\quad\quad\quad Q - \text{Diag}(\rho) - P - N \preceq 0, \\ &\quad\quad\quad (\rho, \mu, \eta, \xi) \in \mathbb{R}^n \times \mathbb{R}^m_+ \times \mathbb{R}^n_+ \times \mathbb{R}^n_+, \\ &\quad\quad\quad P, -N, B,\ C,\ D,\ E \in \mathcal{P}, \end{aligned}$$

where $\psi(\rho, P, N)$, $\Gamma(\rho, \mu, \eta, \xi, B, C, E)$ and $\vartheta(\tau, E, \mu, \xi)$ are defined in (2.5)-(2.7), respectively. Obviously, the last problem is exactly (TUB_s) by noting that the first constraint in the last problem is equivalent to the first constraint in (TUB_s). Therefore, (TUB) and (TUB_s) are equivalent. □

Next, let's consider the conic dual of (TUB_s). It is straightforward (yet tedious) to show that the conic dual of the SDP problem (TUB_s) is

$$\begin{aligned} (\text{TUB}_\text{d}) \qquad & \max\ Q \bullet X \\ & \text{s.t.}\ X_{ii} = x_i,\ i = 1, \ldots, n, && (2.8) \\ & \quad\ X_{ij} \leq x_i,\ X_{ij} \leq x_j,\ i, j = 1, \ldots, n, && (2.9) \\ & \quad\ X_{ij} \geq x_i + x_j - 1, X_{ij} \geq 0,\ i, j = 1, \ldots, n, && (2.10) \\ & \quad\ Ax \leq b,\ 0 \leq x \leq e, \\ & \quad\ \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq 0. \end{aligned}$$

Let $\rho^* \in \mathbb{R}^n$, $B^* = (B^*_{ij}) \in \mathcal{P}$, $C^* = (C^*_{ij}) \in \mathcal{P}$, $D^* = (D^*_{ij}) \in \mathcal{P}$, $E^* = (E^*_{ij}) \in \mathcal{P}$ be the optimal multipliers to constraints (2.8)-(2.10), respectively. Let $P^* = B^* + C^*$ and $N^* = -(D^* + E^*)$. By conic strong duality and Theorem 2.1, (ρ^*, P^*, N^*) is an optimal solution to (TUB). Since the size of the SDP problem ($\mathrm{TUB_d}$) is much smaller than ($\mathrm{TUB_s}$), we can solve ($\mathrm{TUB_d}$) by primal-dual interior-point method to obtain (ρ^*, P^*, N^*) . This can be implemented by using the SDP solvers `SeDuMi` and `SDPT3`.

For the sake of convenience, we assume in the sequel that both ($\mathrm{TUB_s}$) and ($\mathrm{TUB_d}$) are strictly feasible. Under this standard assumption, ($\mathrm{TUB_d}$) is polynomially solvable and there is no duality gap between ($\mathrm{TUB_s}$) and ($\mathrm{TUB_d}$).

Corollary 2.2. *Let $(\rho^*, B^*, C^*, D^*, E^*)$ be optimal multipliers corresponding to constraints (2.8)-(2.10) in* ($\mathrm{TUB_d}$)*, respectively. Let $P^* = B^* + C^*$ and $N^* = -(D^* + E^*)$. Then,*

$$v(\mathrm{TUB_s}) = v(\mathrm{TUB_d}) = v(\overline{\mathrm{QKP}}(\rho^*, P^*, N^*)).$$

In summary, we have shown that $(\mathrm{QKP}(\rho, P, N))$ is a convex 0-1 quadratic program reformulation of (QKP) for any (ρ, P, N) satisfying $Q - \mathrm{Diag}(\rho) - P - N \preceq 0$, $P \in \mathcal{P}$ and $N \in \mathcal{N}$. Moveover, the parameters (ρ^*, P^*, N^*) that provides the tightest upper bound of the continuous relaxation of the reformulation can be found by solving an SDP problem.

Now, let's compare the reformulation $(\mathrm{QKP}(\rho, P, N))$ with the diagonal perturbed reformulation in [4] when applied to (QKP). The diagonal perturbed reformulation for (QKP) has the following form:

$$(\mathrm{QKP}(\rho)) \qquad \begin{aligned} &\max\ x^T(Q - \mathrm{Diag}(\rho))x + \rho^T x \\ &\text{s.t. } Ax \le b,\ x \in \{0,1\}^n, \end{aligned}$$

where $Q - \mathrm{Diag}(\rho) \preceq 0$. Obviously, the diagonal perturbed reformulation is a special case of the reformulation $(\mathrm{QKP}(\rho, P, N))$ when setting $P = N = 0$. The optimal parameter ρ in $(\mathrm{QKP}(\rho))$ can be obtained by solving the following SDP problem:

$$(\mathrm{DPR_d}) \qquad \begin{aligned} &\max\ Q \bullet X \\ &\text{s.t. } X_{ii} = x_i,\ i = 1, \ldots, n, \\ &\quad Ax \le b,\ 0 \le x \le e, \\ &\quad \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq 0. \end{aligned}$$

It is obvious that $v(\mathrm{TUB_d}) \le v(\mathrm{DPR_d})$. Therefore, $(\mathrm{QKP}(\rho, P, N))$ is an improved reformulation compared with $(\mathrm{QKP}(\rho))$ in the sense that the continuous relaxation of $(\mathrm{QKP}(\rho, P, N))$ is tighter than or at least as tight as that of $(\mathrm{QKP}(\rho))$ when the parameters are optimally chosen.

2.3 Extensions

The reformulation method via matrix decomposition discussed in previous subsections can be easily generalized to 0-1 quadratic programs with equality and inequality constraints. In particular, we consider the following k-item quadratic knapsack problem (see [17]):

$$(\text{k-QKP}) \qquad \begin{aligned} &\max\ q(x) := x^T Q x \\ &\text{s.t. } Ax \le b, \\ &\quad e^T x = K, \\ &\quad x \in \{0,1\}^n. \end{aligned}$$

The only difference between (QKP) and (k-QKP) is the introduction of a cardinality constraint $e^T x = K$ in the problem (k-QKP). As in [4, 17], the objective function in (k-QKP) can be rewritten as

$$\begin{aligned} q(x,\rho,\alpha) &= x^T Q x - \sum_{i=1}^n \rho_i (x_i^2 - x_i) - \sum_{i=1}^n \alpha_i x_i (e^T x - K) \\ &= x^T [Q - \mathrm{Diag}(\rho) - \frac{1}{2}(\alpha e^T + e\alpha^T)]x + (\rho + K\alpha)^T x, \quad \forall x \in \{0,1\}^n. \end{aligned}$$

Suppose that $Q - \mathrm{Diag}(\rho) - \frac{1}{2}(\alpha e^T + e\alpha^T) \preceq 0$. Replacing $q(x)$ by $q(x,\rho,\alpha)$ in (k-QKP), we obtain a convex 0-1 quadratic program reformulation of (k-QKP). Similar to ($\mathrm{DPR_d}$), the optimal parameters (ρ,α) can be obtained by solving the following SDP problem:

$$\begin{aligned} (\text{k-DPR}_\text{d}) \qquad & \max\ Q \bullet X \\ & \text{s.t. } X_{ii} = x_i,\ i = 1,\dots,n, \\ & \quad \sum_{j=1}^n X_{ij} = K x_i,\ i = 1,\dots,n. \\ & \quad Ax \le b,\ 0 \le x \le e, \\ & \quad \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq 0. \end{aligned}$$

Now, we can apply our matrix decomposition scheme in Subsection 2.1 to (k-QKP). The objective function can be further rewritten as

$$\begin{aligned} q(x,\rho,\alpha,P,N) &= x^T [Q - \mathrm{Diag}(\rho) - \frac{1}{2}(\alpha e^T + e\alpha^T) - P - N]x + (\rho + K\alpha)^T x \\ &\quad + \sum_{i,j=1}^n [P_{ij} s_{ij} - N_{ij} t_{ij}], \end{aligned} \tag{2.11}$$

where $s_{ij} = \min(x_i, x_j)$ and $t_{ij} = -\max(0, x_i + x_j - 1)$. Let $Q - \mathrm{Diag}(\rho) - \frac{1}{2}(\alpha e^T + e\alpha^T) - P - N \preceq 0$. Replacing $q(x)$ by $q(x,\rho,\alpha,P,N)$ in (k-QKP), we obtain an improved convex 0-1 quadratic program reformulation of (k-QKP). Again, the optimal parameters (ρ,α,P,N) can be found by solving the following SDP problem:

$$\begin{aligned} (\text{k-TUB}_\text{d}) \qquad & \max\ Q \bullet X \\ & \text{s.t. } X_{ii} = x_i,\ i = 1,\dots,n, \\ & \quad \sum_{j=1}^n X_{ij} = K x_i,\ i = 1,\dots,n. \\ & \quad X_{ij} \le x_i, X_{ij} \le x_j,\ i,j = 1,\dots,n, \\ & \quad X_{ij} \ge x_i + x_j - 1, X_{ij} \ge 0,\ i,j = 1,\dots,n, \\ & \quad Ax \le b,\ 0 \le x \le e, \\ & \quad \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq 0. \end{aligned}$$

3 Computational Results

In this section, we report comparison numerical results on the effectiveness of the improved reformulation for randomly generated instances of problems (QKP) and (k-QKP). The numerical test was implemented in Matlab 7 and run on a PC (2.4G MHz, 3GB RAM). The

SDP problems involved in computing the parameters of the reformulations were modeled by `CVX 1.2` [12] and solved by `SDPT3` within `CVX`.

We consider standard randomly generated instances of (QKP) (see, e.g., [10, 4, 13, 17]). The elements in $Q = (q_{ij})$ are uniformly taken from [1, 100]. The parameters in linear constraints $Ax \leq b$ are set as follows: a_{ij} is uniformly distributed in $[1, 50]$ and b_i is uniformly taken from $[50, \sum_{j=1}^{n} a_{ij}]$. The instances of (k-QKP) are also randomly generated, in which q_{ij} and a_{ij} are uniformly taken from $[1, 100]$ and $b_i = \frac{1}{2}(\sum_{j=1}^{n} a_{ij})$.

To compare the efficiency of different reformulations of (QKP) and (k-QKP), we use the mixed integer quadratic programming (MIQP) solver in IBM CPLEX 12.1 with Matlab interface (see [14]) to solve the reformulations. The MIQP solver in CPLEX 12.1 is based on a branch-and-bound method which uses one of two options for convex relaxations at each node of the searching tree (except the root node): continuous relaxation and linear programming relaxation. At the root node, CPLEX uses continuous relaxation for both options. In our numerical experiment, we will respectively use the two options to solve the convex 0-1 quadratic program reformulations (QKP($\bar{\rho}$)) and (QKP(ρ^*, P^*, N^*)), where $\bar{\rho}$ is obtained by solving the SDP problem (DPR$_\text{d}$) and (ρ^*, P^*, N^*) is obtained by solving (TUB$_\text{d}$). To reduce the size of the reformulation (QKP(ρ, P, N)), we only use partial elements of P and N. More precisely, we set $P_{ij} = N_{ij} = 0$ for $i, j = 1, \ldots, n$ with $|i - j| > 5$.

For each $n = 60, 70, 80, 90, 100$ and $m = 1, 5$, we generate 5 test problems. The maximum CPU time for CPLEX 12.1 is set to be 1800 seconds. The relative error is measured by

$$\text{rel.error} = \frac{(\text{upper bound} - \text{lower bound})}{\text{upper bound}}$$

when the branch-and-bound process is terminated. The average CPU time, the average number of nodes explored and the average relative errors are recorded. The comparison results between the two reformulations (QKP($\bar{\rho}$)) and (QKP($\rho^*, \text{P}^*, \text{N}^*$)), when using MIQP solver in CPLEX 12.1 with LP relaxation and continuous relaxation, are summarized in Tables 1-4, where

- n is the dimension of x and m is the number of linear constraints;
- Columns 2 and 6 of the tables are the CPU time for computing the parameters by solving the SDP problems (DPR$_\text{d}$) and (TUB$_\text{d}$), respectively;
- Columns 3-5 and Columns 7-9 of the tables record the average CPU time, average number of nodes explored and the average relative errors of the two reformulations.

From Tables 1-4, we see that the improved reformulation (QKP(ρ^*, P^*, N^*)) is much more efficient than the diagonal perturbed reformulation (QKP($\bar{\rho}$)) for the randomly generated instances of (QKP) in terms of the CPU time, number of nodes and relative errors. For both relaxation options in the MIQP solver, the number of nodes explored by the MIQP solver for (QKP($\rho^*, \text{P}^*, \text{N}^*$)) is much less than that for (QKP($\bar{\rho}$)). This is mainly because the continuous relaxations of the subproblems of (QKP($\rho^*, \text{P}^*, \text{N}^*$)) are tighter than the subproblems of (QKP($\bar{\rho}$)) during the branch-and-bound process. We also see that the relative error achieved in solving (QKP($\rho^*, \text{P}^*, \text{N}^*$)) is much smaller than that of (QKP($\bar{\rho}$)) when the branch-and-bound method is terminated with the 1800 seconds maximum CPU time. We observe that the CPU time for computing the parameters of (QKP($\rho^*, \text{P}^*, \text{N}^*$))

is longer than that of (QKP($\bar{\rho}$)). However, this additional CPU time is neglectable when compared the total CPU time used by the branch-and-bound method.

Table 1: Comparison results for (QKP) with LP relaxation ($m = 1$)

n	(DPR$_\text{d}$)	(QKP($\bar{\rho}$))			(TUB$_\text{d}$)	(QKP(ρ^*, P^*, N^*))		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	613.33	1467872	1.46	5.00	14.00	9429	0.00
70	1.60	1638.20	3718590	3.60	6.60	119.20	12009	0.00
80	1.78	1747.67	3975569	7.33	9.00	585.33	402603	0.00
90	2.00	1800.00	3311959	2.67	12.00	686.67	277329	0.23
100	2.37	1800.00	3581474	20.00	13.00	801.00	128992	0.53

Table 2: Comparison results for (QKP) with continuous relaxation ($m = 1$)

n	(DPR$_\text{d}$)	(QKP($\bar{\rho}$))			(TUB$_\text{d}$)	(QKP(ρ^*, P^*, N^*))		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	1800.00	71350	5.33	5.00	822.67	4784	1.00
70	1.60	1800.00	68077	8.40	6.60	1694.20	7643	1.20
80	1.78	1800.00	76147	21.67	9.00	1659.00	7395	2.63
90	2.00	1800.00	35954	6.00	12.00	1800.00	4930	2.00
100	2.37	1800.00	39933	31.67	13.00	1800.00	4427	5.33

Tables 5-6 summarize the numerical results of the MIQP solver in CPLEX 12.1 with LP relaxation and continuous relaxation respectively for the two reformulations of (k-QKP). We observe that the average performance of the improved reformulation (k-QKP($\rho^*, \alpha^*, P^*, N^*$)) is slightly better than the reformulation (k-QKP($\bar{\rho}, \bar{\alpha}$)) in terms of the CPU time, the number of nodes and the relative errors. This is partly because that the difference between the continuous relaxations of (k-QKP($\rho^*, \alpha^*, P^*, N^*$)) and (k-QKP($\bar{\rho}, \bar{\alpha}$)) is not significant due to the presence of the additional constraints: $\sum_{j=1}^{n} X_{ij} = Kx_i$ $(i = 1, \ldots, n)$ in (k-DPR$_\text{d}$) and (k-TUB$_\text{d}$).

4 Conclusions

We have presented in this paper an improved convex 0-1 quadratic program reformulation for quadratic knapsack problems. This new reformulation is based on matrix decomposition of the objective function and the piecewise linear representation of quadratic terms in 0-1 variables. We have shown that the optimal parameters in the improved reformulation can be obtained by solving an SDP problem. Comparison numerical results suggest that the improved reformulation is more efficient than the existing reformulation based on diagonal perturbation in terms of the CPU time, the number of nodes explored and the relative errors when the MIQP solver in CPLEX 12.1 is used to solve the reformulations.

We point out that the matrix decomposition method can be extended to any 0-1 quadratic program. Although the continuous relaxation of the new reformulation is always tighter than or at least as tight as that of the diagonal perturbed reformulation, the size of the new reformulation also increases significantly. There is a trade-off between the quality of the bound and its computation time in a branch-and-bound method. For this reason, we can use partial elements of the matrix P and N when constructing the new reformulation, as is the case in our numerical experiments.

Table 3: Comparison results for (QKP) with LP relaxation ($m = 5$)

n	($\mathrm{DPR_d}$)	($\mathrm{QKP}(\bar{\rho})$)			($\mathrm{TUB_d}$)	($\mathrm{QKP}(\rho^*, P^*, N^*)$)		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	1457.00	3934684	12.67	5.00	329.67	273263	0.00
70	1.00	1800.00	4295601	18.00	6.33	1018.67	496554	4.33
80	1.00	1438.67	3538670	11.33	8.00	209.00	103542	0.00
90	2.33	1800.00	4234728	21.33	10.33	678.67	198235	3.00
100	3.00	1800.00	4014356	33.67	12.33	1338.67	338247	8.00

Table 4: Comparison results for (QKP) with continuous relaxation ($m = 5$)

n	($\mathrm{DPR_d}$)	($\mathrm{QKP}(\bar{\rho})$)			($\mathrm{TUB_d}$)	($\mathrm{QKP}(\rho^*, P^*, N^*)$)		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	1800.00	108682	38.33	5.00	1308.00	8819	16.33
70	1.00	1800.00	80091	39.67	6.33	1800.00	9025	16.67
80	1.00	1800.00	75413	69.33	8.00	1800.00	7434	28.67
90	2.33	1800.00	50101	34.33	10.33	1800.00	5412	12.67
100	3.00	1800.00	40722	50.00	12.33	1800.00	4473	25.67

Table 5: Comparison results for (k-QKP) with LP relaxation ($m = 1$, $K = \lfloor \frac{n}{8} \rfloor$)

n	(k-$\mathrm{DPR_d}$)	(k-$\mathrm{QKP}(\bar{\rho}, \bar{\alpha})$)			(k-$\mathrm{TUB_d}$)	(k-$\mathrm{QKP}(\rho^*, \alpha^*, P^*, N^*)$)		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	33.30	114835	0.00	6.00	24.00	40900	0.00
70	2.00	220.80	745746	0.00	7.20	167.60	242717	0.00
80	2.00	1106.80	2643030	0.00	9.40	751.00	880486	0.00
90	2.20	1787.60	3465774	4.30	12.00	1697.80	1844663	1.50
100	3.00	1800.00	3229040	6.20	15.20	1800.00	1677809	5.00

Table 6: Comparison results for (k-QKP) with continuous relaxation ($m = 1$, $K = \lfloor \frac{n}{8} \rfloor$)

n	(k-$\mathrm{DPR_d}$)	(k-$\mathrm{QKP}(\bar{\rho}, \bar{\alpha})$)			(k-$\mathrm{TUB_d}$)	(k-$\mathrm{QKP}(\rho^*, \alpha^*, P^*, N^*)$)		
	time	time	nodes	rel.error(%)	time	time	nodes	rel.error(%)
60	1.00	1800.00	69836	5.84	6.00	1800.00	18800	4.17
70	2.00	1800.00	53224	8.90	7.20	1800.00	14551	7.64
80	2.00	1800.00	44465	13.05	9.40	1800.00	12746	12.11
90	2.20	1800.00	38035	13.52	12.00	1800.00	9397	11.21
100	3.00	1800.00	29276	15.74	15.20	1800.00	7569	12.86

References

[1] E. Balas, S. Ceria and G. Cornuéjols, A lift-and-project cutting plane algorithm for mixed 0-1 programs, *Math. Program.* 58 (1993) 295–324.

[2] A. Billionnet and F. Calmels, Linear programming for the 0-1 quadratic knapsack problem, *European J. Oper. Res.* 92 (1996) 310–325.

[3] A. Billionnet and S. Elloumi, Using a mixed integer quadratic programming solver for the unconstrained quadratic 0-1 problem, *Math. Program.* 109 (2007) 55–68.

[4] A. Billionnet, S. Elloumi and M.C. Plateau, Improving the performance of standard solvers for quadratic 0-1 programs by a tight convex reformulation: The QCR method, *Discrete Appl. Math.* 157 (2009) 1185–1197.

[5] A. Billionnet, A. Faye and E. Soutif, A new upper bound for the 0-1 quadratic knapsack problem, *European J. Oper. Res.* 112 (1999) 664–672.

[6] S. Burer, On the copositive representation of binary and continuous nonconvex quadratic programs, *Math. Program.* 120 (2009) 479–495.

[7] A. Caprara, D. Pisinger and P. Toth, Exact solution of the quadratic knapsack problem, *INFORMS J. Comput.* 11 (1999) 125–137.

[8] P. Chaillou, P. Hansen and Y. Mahieu, Best network flow bounds for the quadratic knapsack problem, *Lecture Notes in Math.* 1403 (1986) 226–235.

[9] C.E. Ferreira, A. Martin, C.C. de Souza, R. Weismantel and L.A. Wolsey, Formulations and valid inequalities for the node capacitated graph partitioning problem, *Math. Program.* 74 (1996) 247–266.

[10] G. Gallo, P. Hammer and B. Simeone, Quadratic knapsack problems, *Math. Program. Study* 12 (1980) 132–149.

[11] M.X. Goemans and D.P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM* 42 (1995) 1115–1145.

[12] M. Grant and S. Boyd, CVX: Matlab software for disciplined convex programming, version 1. 21, 2010. http://cvxr. com/cvx/

[13] C. Helmberg, F. Rendl and R. Weismantel, A semidefinite programming approach to the quadratic knapsack problem, *J. Comb. Optim.* 4 (2000) 197–215.

[14] IBM CPLEX: Users Manual, 2010.
http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/

[15] E.L. Johnson, A. Mehrotra, and G.L. Nemhauser, Min-cut clustering. *Math. Program.* 62 (1993) 133–151.

[16] H. Kellerer, U. Pferschy and D. Pisinger, *Knapsack Problems*, Springer Verlag, Berlin, 2004.

[17] L. Létocart, M.C. Plateau and G. Plateau, Dual heuristics for the 0-1 exact k-item quadratic knapsack problem, Tech. rep., Laboratoire d'Informatique de Paris-Nord, 2010.

[18] L. Lovász and A. Schrijver, Cones of matrices and set-functions and 0-1 optimization, *SIAM J. Optim.* 1 (1991) 166–190.

[19] Y. Nesterov, Semidefinite relaxation and nonconvex quadratic optimization, *Optim. Methods Softw.* 9 (1998) 141–160.

[20] Y. Nesterov and A. Nemirovsky, *Interior-Point Polynomial Methods in Convex Programming*, SIAM, Philadelphia, PA, 1994.

[21] D. Pisinger, The quadratic knapsack problem–a survey, *Discrete Appl. Math.* 155 (1995) 623–648.

[22] S. Poljak, F. Rendl and H. Wolkowicz, A recipe for semidefinite relaxation for (0, 1)-quadratic programming, *J. Global Optim.* 7 (1995) 51–73.

[23] H.D. Sherali and W.P. Adams, *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*, Kluwer Academic Publishers, Dordrecht, 1999.

[24] N.Z. Shor, Class of global minimum bounds of polynomial functions, *Cybernetics* 23 (1987) 731–734.

[25] J.F. Sturm and S. Zhang, On cones of nonnegative quadratic functions, *Math. Oper. Res.* 28 (2003) 246–267.

[26] L. Vandenberghe and S. Boyd, Semidefinite programming, *SIAM Rev.* 38 (1996) 49–95.

[27] Y. Ye, Approximating quadratic programming with bound and quadratic constraints, *Math. Program.* 84 (1999) 219–226.

[28] S. Zhang, Quadratic maximization and semidefinite relaxation, *Math. Program.* 87 (2000) 453–465.

Manuscript received 16 March 2010
revised 26 January 2011, 14 February 2011
accepted for publication 12 May 2011

Shuhui Ji
Department of Management Science, School of Management, Fudan University
Shanghai 200433, P. R. China
E-mail address: shji@fudan.edu.cn

Xiaojin Zheng
School of Economics and Management, Tongji University
Shanghai 200092, P. R. China
E-mail address: xjzheng@fudan.edu.cn

Xiaoling Sun
Department of Management Science, School of Management, Fudan University
Shanghai 200433, P. R. China
E-mail address: xls@fudan.edu.cn