Second Order Cone Programming Relaxation of Nonconvex Quadratic Optimization Problems

Sunyoung Kim*
Department of Mathematics, Ewha Women's University
11-1 Dahyun-dong, Sudaemoon-gu, Seoul 120-750 Korea
email: *skim@mm.ewha.ac.kr*, *skim@is.titech.ac.jp*

Masakazu Kojima
Department of Mathematical and Computing Sciences
Tokyo Institute of Technology
2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552 Japan
email: *kojima@is.titech.ac.jp*

Abstract A disadvantage of the SDP (semidefinite programming) relaxation method for quadratic and/or combinatorial optimization problems lies in its expensive computational cost. This paper proposes a SOCP (second-order-cone programming) relaxation method, which strengthens the lift-and-project LP (linear programming) relaxation method by adding convex quadratic valid inequalities for the positive semidefinite cone involved in the SDP relaxation. Numerical experiments show that our SOCP relaxation is a reasonable compromise between the effectiveness of the SDP relaxation and the low computational cost of the lift-and-project LP relaxation.

Key words. second-order-cone program, lift-and-project convex relaxation method, nonconvex quadratic program, global optimization, primal-dual interior-point method

⋆ This work was conducted while this author has been visiting Tokyo Institute of Technology, Department of Mathematical and Computing Sciences, on a sabbatical leave from Ewha Women's University, Korea. Research of this author was supported in part by KOSEF 97-01-01-01-3 and Brain Korea 21.

1 Introduction

Let $\mathbb{R}^n$, $\mathbb{S}^n$ and $\mathbb{S}^n_+$ denote the n-dimensional Euclidean space, the set of $n \times n$ symmetric matrices and the set of $n \times n$ positive semidefinite symmetric matrices, respectively. Let C_0 be a compact convex subset of $\mathbb{R}^n$, and let $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{Q}_p \in \mathbb{S}^n$, $\boldsymbol{q}_p \in \mathbb{R}^n$, and $\gamma_p \in \mathbb{R}$ $(p = 1, 2, \ldots, m)$. Consider a QOP (quadratic optimization problem) of the form

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0,\ \boldsymbol{x}^T\boldsymbol{Q}_p\boldsymbol{x} + \boldsymbol{q}_p^T\boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m). \end{array}\right\} \tag{1}$$

We denote the feasible region of the QOP above by F;

$$F = \left\{\boldsymbol{x} \in C_0 : \boldsymbol{x}^T\boldsymbol{Q}_p\boldsymbol{x} + \boldsymbol{q}_p^T\boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m)\right\}.$$

We assume that C_0 is a bounded polyhedral set represented by a finite number of linear inequalities in practice, although C_0 can be any compact convex subset of $\mathbb{R}^n$ in theory.

If all the coefficient matrices $\boldsymbol{Q}_p$ $(p = 1, 2, \ldots, m)$ involved in the quadratic inequality constraints are positive semidefinite, the QOP (1) turns out to be a convex program. In this case, every local minimizer is a global minimizer, and we can utilize many existing nonlinear programming codes to compute an approximate global minimizer. In particular, we can reformulate the convex QOP (1) in terms of an SOCP (second order cone programming) problem to which we can apply the primal-dual interior-point method [19].

Throughout the paper, we deal with nonconvex QOPs where some of $\boldsymbol{Q}_p$'s are not positive semidefinite. Nonconvex QOPs are known to be NP-hard, and they cover various difficult nonconvex optimization problems and combinatorial optimization problems; for example, linearly constrained nonconvex quadratic programs, bilevel linear programs, linear programs with equilibrium constraints, maximum clique problems, and 0-1 integer programs. To compute an approximate global minimizer of such a nonconvex QOP, we need to take two distinct techniques into account. The one is to generate a feasible solution (or an approximate feasible solution) with a smaller objective value. The other technique is to derive a tighter lower bound of the minimal objective value. Based on these two techniques, we may regard a feasible solution of the QOP (1) computed by the former technique as an ϵ optimal solution of the QOP if its objective value is within a given small positive number ϵ from a lower bound, computed by the latter technique, of the minimal objective value. These two techniques play essential roles in the branch-and-bound method, which has been serving as one of the most practical and popular computational methods for nonconvex and combinatorial optimization problems.

The aim of the current paper is to explore an SOCP (second order cone programming) relaxation for computing a lower bound of the minimal objective value of the QOP (1). We may regard this relaxation as a special case of more general convex relaxations described as follows. Let C be a closed convex subset of $\mathbb{R}^n$ which includes the feasible region F. Then, we have $\inf\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in C\} \leq \inf\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in F\}$. The problem of minimizing the same linear objective function $\boldsymbol{c}^T\boldsymbol{x}$ over the convex set C serves as a convex relaxation problem of the QOP (1). Here, we implicitly assume that the minimization over the convex set C is easier than the minimization over the original nonconvex feasible region F.

Traditionally, LP (linear programming) has been utilized for convex relaxation of integer programs. However, LP relaxation is not powerful enough to generate a tight lower bound of the minimal objective value in practice, and also it does not cover general QOPs. In recent years, a class of "lift-and-project convex relaxation methods" ([1, 5, 9, 10, 12, 13, 15, 16, 17, 18, 20, 22]) have been proposed and studied extensively. They are characterized as the three steps:

1. Lift the QOP (1) to an equivalent problem in the space $\mathbb{S}^{1+n}$ of $(1+n)\times(1+n)$ symmetric matrices; the resulting problem is an LP with additional rank-1 and positive semidefinite constraints imposed on a matrix variable
$$\boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}^{1+n}.$$
2. Relax the rank-1 and positive semidefinite constraints so that the feasible region of the resulting problem is convex.
3. Project the relaxed lifted problem in $\mathbb{S}^{1+n}$ back to the original Euclidean space $\mathbb{R}^n$.

See Section 2 for more technical details. We can classify the methods into two groups according to a relaxation taken in Step 2.

In one group, only the rank-1 constraint is removed in Step 2, and the relaxed problem turns out to be an SDP (semidefinite program) in the matrix variable $\boldsymbol{Y}$ in $\mathbb{S}^{1+n}$. This group of methods are called the (lift-and-project) SDP relaxation methods [5, 9, 10, 12, 13, 15, 16, 18, 22, etc.]. We will denote the projection of the feasible region of the relaxed lifted problem in $\mathbb{S}^{1+n}$ onto $\mathbb{R}^n$ by F^{SDP}.

Removing both rank-1 and positive semidefinite constraints in Step 2 leads to the other group of the methods whose formulation is represented as an LP in the matrix variable $\boldsymbol{Y}$ in $\mathbb{S}^{1+n}$. This group of methods have been called by various names such as the lift-and-project cutting algorithm [1], the reformulation-linearization technique [17], the matrix cut [14] and the semi-infinite LP relaxation method [12]. In this article, we call them the lift-and-project LP relaxation method. We will denote the projection of the feasible region of the relaxed lifted problem in $\mathbb{S}^{1+n}$ onto $\mathbb{R}^n$ by F^{LP}.

The SDP relaxation F^{SDP} of the nonconvex feasible region F of the QOP (1) is at least as effective as the lift-and-project LP relaxation F^{LP};

$$F \subset \mathrm{c.hull}(F) \subset F^{\mathrm{SDP}} \subset F^{\mathrm{LP}} \subset C_0. \tag{2}$$

Furthermore, it is known that the SDP relaxation is more effective than the lift-and-project LP relaxation in both theory and practice of many combinatorial optimization problems [9, 10, 15, etc.]. In particular, the 0.878 approximation [9] of a maximum cut based on the SDP relaxation is widely known.

For numerical computation, we can apply interior-point methods [6, 19, 21, etc.], which are extensions of interior-point methods developed for LPs, to SDPs. Solving an SDP with a large scale matrix variable is still much more expensive than solving an LP with the same number of variables as the SDP, although extensive efforts [2, 7, 8, 11, etc.] continue from various angles. In this article, we propose a (lift-and-project) SOCP relaxation of the QOP (1) as a reasonable compromise between the effectiveness of the SDP relaxation in getting better lower bounds of the QOP (1) and the low computational cost of the lift-and-project LP relaxation. The basic idea behind our SOCP relaxation is:

- Add a finite number of convex quadratic inequalities, which are valid for the positive semidefinite cone involved in the SDP relaxation, to the lift-and-project LP relaxation.
- Reformulate the resulting convex QOP with a linear objective function and a finite number of linear and convex quadratic inequality constraints into an SOCP problem.

The remaining of the paper is organized as follows. Section 2 describes technical details of the SDP relaxation and the lift-and-project LP relaxation for the QOP (1). In Section 3, we discuss how we strengthen the lift-and-project LP relaxation by adding convex quadratic inequalities. As a result, we obtain a convex QOP which serves as a stronger relaxation of the QOP (1) than the lift-and-project LP relaxation of the QOP (1). Section 4 includes conversion techniques of such a convex QOP relaxation problem into an SOCP relaxation problem. In Section 5, we present computational results on the SDP, the SOCP and the lift-and-project LP relaxations, and confirm the effectiveness of the SOCP relaxation. Section 6 is devoted to concluding discussions.

2 Lift-and-project convex relaxation methods

We use the notation $\boldsymbol{A} \bullet \boldsymbol{B} = \sum_{i=1}^n \sum_{j=1}^n A_{ij} B_{ij}$ for the inner product of $\boldsymbol{A}$, $\boldsymbol{B} \in \mathbb{S}^n$. Following the three steps which we have mentioned in the Introduction, we now describe the lift-and-project SDP relaxation method for the QOP (1). First, we rewrite the QOP (1) as

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0, \\ & \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \\ & \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x} \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}_+^{1+n} \ \text{ and rank } \boldsymbol{Y} = 1. \end{array}\right\} \tag{3}$$

It should be noted that the pair of the last two constraints

$$\boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x} \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}_+^{1+n} \ \text{ and rank } \boldsymbol{Y} = 1$$

is equivalent to $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$. Now, removing the constraint rank $\boldsymbol{Y} = 1$ from the problem (3), we have an SDP

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0, \\ & \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \\ & \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}_+^{1+n}. \end{array}\right\} \tag{4}$$

Let F^{SDP} be the orthogonal projection of the feasible region of the SDP (4) in the lifted space $\mathbb{S}^{1+n}$ of the matrix variable $\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}$ onto the space $\mathbb{R}^n$ where the original problem (1) is defined;

$$F^{\mathrm{SDP}} = \left\{ \boldsymbol{x} \in C_0 : \begin{array}{l} \exists \boldsymbol{X} \in \mathbb{S}^n \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}^{1+n}_+ \text{ and} \\ \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m). \end{array} \right\}$$

Then we can rewrite the SDP (4) as

$$\text{minimize } \boldsymbol{c}^T \boldsymbol{x} \ \text{ subject to } \boldsymbol{x} \in F^{\mathrm{SDP}}. \tag{5}$$

We obtain another convex relaxation of the QOP (1), the lift-and-project LP relaxation by neglecting both of the positive semidefinite and rank-1 constraints on $\boldsymbol{Y}$ in the QOP (3):

$$\left. \begin{array}{ll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0, \ \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \end{array} \right\} \tag{6}$$

or equivalently,

$$\text{minimize } \boldsymbol{c}^T \boldsymbol{x} \ \text{ subject to } \boldsymbol{x} \in F^{\mathrm{LP}}, \tag{7}$$

where

$$F^{\mathrm{LP}} = \left\{ \boldsymbol{x} \in C_0 : \ \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m) \ \right\}.$$

By construction, we see that the inclusion relation (2) holds.

3 Adding convex quadratic inequalities to the lift-and-project LP relaxation

3.1 Basic analysis

The positive semidefinite condition

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathbb{S}^{1+n}_+ \tag{8}$$

in (3) (also in (4)) is equivalent to the positive semidefinite condition $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \in \mathbb{S}^n_+$ on the smaller matrix $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T$. The latter condition holds if and only if $\boldsymbol{C} \bullet \left(\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T\right) \geq 0$ is true for $\forall \boldsymbol{C} \in \mathbb{S}^n_+$. Moreover, we can rewrite this inequality as

$$\boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x} - \boldsymbol{C} \bullet \boldsymbol{X} \leq 0. \tag{9}$$

Note that the left hand side of the inequality with a fixed $\boldsymbol{C} \in \mathbb{S}^n_+$ is convex in $\boldsymbol{x}$ and linear in $\boldsymbol{X}$. This allows us to add the convex quadratic inequality (9) to the lift-and-project LP relaxation (6).

The positive semidefinite condition (8) holds if and only if

$$\boldsymbol{Z}^* \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \geq 0 \tag{10}$$

for $\forall \boldsymbol{Z}^* \in \mathbb{S}^{1+n}_+$. We can prove, however, that a linear inequality (10) with a matrix $\boldsymbol{Z}^* \in \mathbb{S}^{1+n}_+$ is weaker than the convex quadratic inequality (9) with the submatrix $\boldsymbol{C} \in \mathbb{S}^n_+$ obtained by deleting the first row and the first column of $\boldsymbol{Z}^* \in \mathbb{S}^{1+n}_+$. In other words, let

$$\boldsymbol{Z}^* = \begin{pmatrix} \beta & \boldsymbol{b}^T/2 \\ \boldsymbol{b}/2 & \boldsymbol{C} \end{pmatrix} \in \mathbb{S}^{1+n}_+, \ \beta \in \mathbb{R}_+, \ \boldsymbol{b} \in \mathbb{R}^n, \ \boldsymbol{C} \in \mathbb{S}^n_+, \tag{11}$$

and suppose that $\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}$ satisfies the convex quadratic inequality (9) with the submatrix $\boldsymbol{C} \in \mathbb{S}^n_+$ of $\boldsymbol{Z}^* \in \mathbb{S}^{1+n}_+$. Then, we have the following from the positive semidefiniteness of $\boldsymbol{Z}^*$

$$0 \leq (1, \boldsymbol{x}^T) \begin{pmatrix} \beta & \boldsymbol{b}^T/2 \\ \boldsymbol{b}/2 & \boldsymbol{C} \end{pmatrix} \begin{pmatrix} 1 \\ \boldsymbol{x} \end{pmatrix} = \beta + \boldsymbol{b}^T \boldsymbol{x} + \boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x}.$$

Combining the inequality above with (9), we obtain that

$$0 \leq \beta + \boldsymbol{b}^T \boldsymbol{x} + \boldsymbol{C} \bullet \boldsymbol{X} = \boldsymbol{Z}^* \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}.$$

Thus we have shown that $\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}$ satisfies the inequality (10).

3.2 Convex quadratic inequalities induced from the inequality constraints of the QOP

In this section, we discuss how we choose a fewer number of convex quadratic inequality constraints

$$\boldsymbol{x}^T \boldsymbol{C}_i \boldsymbol{x} - \boldsymbol{C}_i \bullet \boldsymbol{X} \leq 0 \ (i = 1, 2, \ldots, \ell)$$

to derive an effective convex QOP relaxation of the nonconvex QOP (1):

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0, \ \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \\ & \boldsymbol{x}^T \boldsymbol{C}_i \boldsymbol{x} - \boldsymbol{C}_i \bullet \boldsymbol{X} \leq 0 \ (i = 1, 2, \ldots, \ell). \end{array}\right\} \tag{12}$$

Here our assumption is that $\boldsymbol{C}_i \in \mathbb{S}^n_+$ $(i = 1, 2, \ldots, \ell)$. As we add more convex quadratic inequalities, we can expect to have a better convex QOP relaxation (12) of the QOP (1), but we need more CPU time to solve the problem (12).

We show below how to extract convex quadratic inequalities from the original quadratic inequality constraints of the QOP (1). Let $p \in \{1, 2, \ldots, m\}$ be fixed, and λ_i $(i = 1, 2, \ldots, n)$ denote the eigenvalues of the coefficient matrix $\boldsymbol{Q}_p \in \mathbb{S}^n$ of the pth quadratic inequality constraint. We may assume that

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_\ell \geq 0 > \lambda_{\ell+1} \geq \ldots \geq \lambda_n \tag{13}$$

for some $\ell \in \{0, 1, 2, \dots, n\}$. Let $\boldsymbol{u}_i$ $(i = 1, 2, \dots, n)$ denote the eigenvectors corresponding to λ_i $(i = 1, 2, \dots, n)$, respectively, such that

$$\|\boldsymbol{u}_i\| = 1 \ (i = 1, 2, \dots, n) \ \text{ and } \boldsymbol{u}_i^T \boldsymbol{u}_j = 0 \ (i \neq j). \tag{14}$$

Then we have

$$\boldsymbol{Q}_p = \sum_{j=1}^{n} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T. \tag{15}$$

If $1 \leq s \leq t \leq \ell$, then the matrix $\boldsymbol{C} = \sum_{j=s}^{t} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T$ is positive semidefinite. Similarly, if $\ell + 1 \leq s \leq t \leq n$, then the matrix $\boldsymbol{C} = -\sum_{j=s}^{t} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T$ is positive semidefinite. In both cases, we can add the convex quadratic inequality constraint $\boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x} - \boldsymbol{C} \bullet \boldsymbol{X} \leq 0$ to the lift-and-project LP relaxation (6).

If $\boldsymbol{Q}_p$ itself is positive semidefinite or the pth quadratic inequality constraint of the QOP (1) is convex, then we can take $\boldsymbol{C} = \sum_{j=1}^{n} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T = \boldsymbol{Q}_p$. It follows in this case that the resulting problem includes the constraints

$$\boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ \text{ and } \boldsymbol{x}^T \boldsymbol{Q}_p \boldsymbol{x} - \boldsymbol{Q}_p \bullet \boldsymbol{X} \leq 0,$$

which imply the pth quadratic inequality constraint itself of the original QOP (1).

In Section 4, we will convert the convex QOP relaxation (12) into an SOCP (second-oder cone program). The cost of solving the resulting SOCP depends very much on the ranks of $\boldsymbol{C}_i \in \mathbb{S}^n_+$ $(i = 1, 2, \dots, \ell)$; the larger their ranks are, the more auxiliary variables we need to introduce and the more expensive the cost of solving the resulting SOCP becomes. In an attempt to keep the amount of computation small, low rank $\boldsymbol{C}_i \in \mathbb{S}^n_+$ $(i = 1, 2, \dots, \ell)$ are reasonable. Among many candidates, the simplest and reasonable choice is

$$x_i^2 - X_{ii} \leq 0 \ (i = 1, 2, \dots, n).$$

We can also employ some of the rank-1 convex quadratic inequalities

$$\boldsymbol{x}^T \boldsymbol{u}_i \boldsymbol{u}_i^T \boldsymbol{x} - \boldsymbol{u}_i \boldsymbol{u}_i^T \bullet \boldsymbol{X} \leq 0 \ (i = 1, 2, \dots, n),$$

where each $\boldsymbol{u}_i$ denotes an eigenvector of the coefficient matrix $\boldsymbol{Q}_p$ of the pth quadratic inequality constraint of the QOP (1).

3.3 Reducing the number of variables

One of the disadvantages of the lift-and-project convex relaxation methods described so far is that the relaxed lifted problems (4), (6) and (12) involve an additional $n \times n$ symmetric matrix variable $\boldsymbol{X}$; the number of variables of the relaxed lifted problems grows quadratically in the number of variables of the original QOP (1). We will discuss below how we eliminate the matrix variable $\boldsymbol{X}$ from the convex QOP relaxation (12). Suppose that the coefficient matrix $\boldsymbol{Q}_p \in \mathbb{S}^n$ of the pth quadratic inequality constraint is represented as in (15), where we denote λ_i $(i = 1, 2, \dots, n)$ as the eigenvalues of $\boldsymbol{Q}_p$ satisfying (13) and $\boldsymbol{u}_i$ $(i = 1, 2, \dots, n)$ the corresponding eigenvectors satisfying (14). Let $\boldsymbol{Q}_p^+ = \sum_{j=1}^{\ell} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T$. Then we can rewrite the pth quadratic inequality constraint as

$$\left.\begin{array}{l} \boldsymbol{x}^T \boldsymbol{Q}_p^+ \boldsymbol{x} + \displaystyle\sum_{j=\ell+1}^{n} \lambda_j z_j + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0, \\ \boldsymbol{x}^T (\boldsymbol{u}_j \boldsymbol{u}_j^T) \boldsymbol{x} - z_j = 0 \ (j = \ell+1, \ell+2, \dots, n). \end{array}\right\} \tag{16}$$

Relaxing the last $n - \ell$ equalities, we have a set of convex quadratic inequalities

$$\left.\begin{array}{l} \boldsymbol{x}^T \boldsymbol{Q}_p^+ \boldsymbol{x} + \displaystyle\sum_{j=\ell+1}^{n} \lambda_j z_j + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0, \\ \boldsymbol{x}^T \left(\boldsymbol{u}_j \boldsymbol{u}_j^T\right) \boldsymbol{x} - z_j \leq 0 \ (j = \ell+1, \ell+2, \dots, n). \end{array}\right\} \tag{17}$$

Depending on the constraint $\boldsymbol{x} \in C_0$, it is necessary to add appropriate constraints on the variables z_j $(j = \ell+1, \ell+2, \dots, n)$ to bound them from above. Otherwise, any $\boldsymbol{x} \in C_0$ satisfies the inequalities in (17) for some z_j $(j = \ell+1, \ell+2, \dots, n)$ (recall that $\lambda_j < 0$ $(j = \ell+1, \ell+2, \dots, n)$). In general, the inequality

$$\sum_{j=\ell+1}^{n} z_j \leq \rho_{\max} \equiv \max\{\|\boldsymbol{x}\|^2 : \boldsymbol{x} \in C_0\}$$

holds. In fact, if $\boldsymbol{x}$ and z_j $(j = \ell+1, \ell+2, \dots, n)$ satisfy (16) then

$$\sum_{j=\ell+1}^{n} z_j = \sum_{j=\ell+1}^{n} \boldsymbol{x}^T (\boldsymbol{u}_j \boldsymbol{u}_j^T) \boldsymbol{x} \leq \boldsymbol{x}^T \left(\sum_{j=1}^{n} \boldsymbol{u}_j \boldsymbol{u}_j^T \right) \boldsymbol{x} \leq \|\boldsymbol{x}\|^2,$$

where the last inequality holds since $\left(\sum_{j=1}^{n} \boldsymbol{u}_j \boldsymbol{u}_j^T\right)$ is an orthogonal matrix.

Now we relate (17) to a convex QOP relaxation of the pth quadratic inequality constraint of the QOP (1). Consider the lift-and-project LP relaxation

$$\boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \tag{18}$$

of the pth quadratic inequality constraint of the QOP (1) together with the convex quadratic valid inequalities

$$\left.\begin{array}{l} \boldsymbol{x}^T \boldsymbol{Q}_p^+ \boldsymbol{x} - \boldsymbol{Q}_p^+ \bullet \boldsymbol{X} \leq 0, \\ \boldsymbol{x}^T \left(\boldsymbol{u}_j \boldsymbol{u}_j^T\right) \boldsymbol{x} - \left(\boldsymbol{u}_j \boldsymbol{u}_j^T\right) \bullet \boldsymbol{X} \leq 0 \ (j = \ell+1, \ell+2, \dots, n) \end{array}\right\} \tag{19}$$

for $(\boldsymbol{x}, \boldsymbol{X}) \in \boldsymbol{R}^n \times \mathbb{S}^n$. Suppose that $\boldsymbol{x}$ satisfies (18) and (19) for some $\boldsymbol{X} \in \mathbb{S}^n$. Then $\boldsymbol{x}$ and $z_j = \boldsymbol{u}_j^T \boldsymbol{X} \boldsymbol{u}_j$ $(j = \ell+1, \ell+2, \dots, n)$ satisfy (17). We may regard (17) as a further relaxation of the convex QOP relaxation consisting of (18) and (19).

It is worthy to be noted that not only the vectors $\boldsymbol{u}_j$ $(j = \ell+1, \ell+2, \dots, n)$ but also the variables z_j $(j = \ell+1, \ell+2, \dots, n)$ depend on p. And, if we apply this relaxation technique to each quadratic inequality constraint of the QOP (1), we obtain a convex QOP relaxation of the QOP (1)

$$\text{minimize} \quad \boldsymbol{c}^T \boldsymbol{x} \quad \text{subject to} \quad \boldsymbol{x} \in F^{\text{CQOP}}, \tag{20}$$

where

$$F^{\mathrm{CQOP}} = \left\{ \boldsymbol{x} \in C_0 : \begin{array}{l} \exists z_{pj} \ (j = \ell_p + 1, \ell_p + 2, \ldots, n, \ p = 1, 2, \ldots, m) \text{ such that} \\ \boldsymbol{x}^T \boldsymbol{Q}_p^+ \boldsymbol{x} + \displaystyle\sum_{j=\ell_p+1}^{n} \lambda_{pj} z_{pj} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \\ \boldsymbol{x}^T \left(\boldsymbol{u}_{pj} \boldsymbol{u}_{pj}^T \right) \boldsymbol{x} - z_{pj} \leq 0 \\ (j = \ell_p + 1, \ell_p + 2, \ldots, n, \ p = 1, 2, \ldots, m) \end{array} \right\}.$$

3.4 An illustrative example

The following case demonstrates some differences among the convex relaxations (5), (7) and (20):

$$\begin{array}{l} n = 2, \ m = 3, \ \boldsymbol{c} = (0, -1), \\ C_0(\rho_{\max}) = \{ \boldsymbol{x} \in \mathbb{R}^2 : x_2 \geq 0, \ x_1^2 + x_2^2 \leq \rho_{\max} \}, \\ \boldsymbol{Q}_1 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \ \boldsymbol{q}_1 = (0, 1)^T, \ \gamma_2 = -0.2, \\ \boldsymbol{Q}_2 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \ \boldsymbol{q}_2 = (0, 0)^T, \ \gamma_1 = -1.15, \\ \boldsymbol{Q}_3 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \ \boldsymbol{q}_3 = (0, 0)^T, \ \gamma_3 = -6, \end{array}$$

where $\rho_{\max}$ is a parameter which takes a value of either 2.79 or 3.16. The resulting QOP is:

$$\begin{array}{ll} \text{minimize} & -x_2 \\ \text{subject to} & (x_1, x_2)^T \in C_0(\rho_{\max}), \\ & -x_1^2 + x_2^2 + x_2 - 0.2 \leq 0, \ x_1^2 - x_2^2 - 1.15 \leq 0, \ x_1^2 + 2x_2^2 - 6 \leq 0. \end{array} \tag{21}$$

For the feasible regions of the convex relaxations, we have

$$\begin{array}{rcl} F^{\mathrm{SDP}} & = & \left\{ (x_1, x_2)^T \in C_0(\rho_{\max}) : \begin{array}{l} x_2 - 1.35 \leq 0, \\ 3x_2^2 + x_2 - 6.2 \leq 0, \\ x_1^2 + 2x_2^2 - 6 \leq 0, \\ \dfrac{3}{2} x_1^2 - 4.15 \leq 0 \end{array} \right\}, \\ F^{\mathrm{LP}} & = & \left\{ (x_1, x_2)^T \in C_0(\rho_{\max}) : x_2 - 1.35 \leq 0 \right\}, \\ F^{\mathrm{CQP}} & = & \left\{ (x_1, x_2)^T \in C_0(\rho_{\max}) : \begin{array}{l} \exists z_1, \ z_2 \ \text{ such that} \\ x_2^2 - z_1 + x_2 - 0.2 \leq 0, \ x_1^2 \leq z_1 \leq \rho_{\max}, \\ x_1^2 - z_2 - 1.15 \leq 0, \ x_2^2 \leq z_2 \leq \rho_{\max}, \\ x_1^2 + 2x_2^2 - 6 \leq 0. \end{array} \right\} \end{array}$$

See [5] for computation of F^{SDP} and F^{LP}. Consequently,

$$\begin{array}{rcl} \min\{ \boldsymbol{c}^T \boldsymbol{x} : \boldsymbol{x} \in F^{\mathrm{SDP}} \} & = & \dfrac{1 - \sqrt{75.4}}{6} \approx -1.28, \\ \min\{ \boldsymbol{c}^T \boldsymbol{x} : \boldsymbol{x} \in F^{\mathrm{LP}} \} & = & -1.35, \\ \min\{ \boldsymbol{c}^T \boldsymbol{x} : \boldsymbol{x} \in F^{\mathrm{CQP}} \} & = & \left\{ \begin{array}{ll} -1.30 & \text{if } \rho_{\max} = 2.79, \\ -1.40 & \text{if } \rho_{\max} = 3.16. \end{array} \right. \end{array}$$

It should be noted that the convex QOP relaxation (20) attains a better lower bound -1.30 of the QOP (1) than the lift-and-project LP relaxation (7) when $\rho_{\max} = 2.79$, while the former obtains a worse lower bound -1.40 when $\rho_{\max} = 3.16$.

In order to confirm the above results computationally, we have tested the three convex relaxations (5), (7) and (20) for the QOP (21) with SeDuMi Version 1.03. The SDP relaxation (5) has provided a bound -1.28055 in 0.2 seconds in 9 iterations, while the convex QOP relaxation (20) with $\rho_{\max} = 2.79$ a bound -1.30 in 9 iterations and 0.1 seconds, the convex QOP relaxation (20) with $\rho_{\max} = 3.16$ a bound -1.40 in 9 iterations and 0.1 seconds, and the lift-and-project LP relaxation (7) a bound -1.35 in 2 iterations and 0.01 seconds.

4 Conversion of convex QOPs into SOCPs

We notice that each inequality constraint in the relaxation problems (12) and (20) can be written as

$$\boldsymbol{x}^T \boldsymbol{C} \boldsymbol{x} + \boldsymbol{q}^T \boldsymbol{x} + \boldsymbol{g}^T \boldsymbol{y} + \gamma \leq 0 \tag{22}$$

for some $\boldsymbol{C} \in \mathbb{S}^n_+$, $\boldsymbol{q} \in \mathbb{R}^n$, $\boldsymbol{g} \in \mathbb{R}^s$ and $\gamma \in \mathbb{R}$. Some of the inequalities are linear so that we can take $\boldsymbol{C} = \boldsymbol{O}$. Suppose that rank $\boldsymbol{C} = k \geq 1$. Then there exists an $n \times k$ matrix $\boldsymbol{L}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$; we can compute such an $\boldsymbol{L}$ by applying the Cholesky factorization or the eigenvalue decomposition to the matrix $\boldsymbol{C}$. And, we can rewrite the inequality (22) as

$$(\boldsymbol{L}^T \boldsymbol{x})^T (\boldsymbol{L}^T \boldsymbol{x}) \leq -\boldsymbol{q}^T \boldsymbol{x} - \boldsymbol{g}^T \boldsymbol{y} - \gamma. \tag{23}$$

It is known and also easily verified that $\boldsymbol{w} \in \mathbb{R}^t$, $\xi \in \mathbb{R}$ and $\eta \in \mathbb{R}$ satisfy

$$\boldsymbol{w}^T \boldsymbol{w} \leq \xi\eta, \ \xi \geq 0 \ \text{ and } \eta \geq 0$$

if and only if they satisfy

$$\left\| \begin{pmatrix} \xi - \eta \\ 2\boldsymbol{w} \end{pmatrix} \right\| \leq \xi + \eta.$$

If we take $\boldsymbol{w} = \boldsymbol{L}^T \boldsymbol{x}$, $\xi = 1$ and $\eta = -\boldsymbol{q}^T \boldsymbol{x} - \boldsymbol{g}^T \boldsymbol{y} - \gamma$, we can convert the inequality (23) into a linear inequality with an additional SOC (second order cone) condition

$$\begin{pmatrix} v_0 \\ \boldsymbol{v} \end{pmatrix} = \begin{pmatrix} 1 - \boldsymbol{q}^T \boldsymbol{x} - \boldsymbol{g}^T \boldsymbol{y} - \gamma \\ 1 + \boldsymbol{q}^T \boldsymbol{x} + \boldsymbol{g}^T \boldsymbol{y} + \gamma \\ 2\boldsymbol{L}^T \boldsymbol{x} \end{pmatrix} \in \mathbb{R}^{2+k} \ \text{ and } \|\boldsymbol{v}\| \leq v_0. \tag{24}$$

Now, applying the conversion from the convex quadratic inequality (22) into the linear inequality (24) with an additional SOC condition to each convex quadratic inequality $\boldsymbol{x}^T \boldsymbol{C}_i \boldsymbol{x} - \boldsymbol{C}_i \bullet \boldsymbol{X} \leq 0$, we can convert the convex QOP (12) into an SOCP

$$\left.\begin{array}{ll}
\text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\
\text{subject to} & \boldsymbol{x} \in C_0, \\
& \boldsymbol{Q}_p \bullet \boldsymbol{X} + \boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (p = 1, 2, \ldots, m), \\
& \begin{pmatrix} v_{i0} \\ \boldsymbol{v}_i \end{pmatrix} = \begin{pmatrix} 1 + \boldsymbol{C}_i \bullet \boldsymbol{X} \\ 1 - \boldsymbol{C}_i \bullet \boldsymbol{X} \\ 2\boldsymbol{L}_i^T \boldsymbol{x} \end{pmatrix} \ (i = 1, 2, \ldots, \ell), \\
& \|\boldsymbol{v}_i\| \leq v_{i0} \ (i = 1, 2, \ldots, \ell),
\end{array}\right\}$$

where $\boldsymbol{C}_i = \boldsymbol{L}_i \boldsymbol{L}_i^T \ (i = 1, 2, \ldots, \ell)$.

Similarly we can convert the convex QOP (20) into an SOCP, using $\boldsymbol{Q}_p^+ = \boldsymbol{L}_p^+ (\boldsymbol{L}_p^+)^T$,

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in C_0, \\ & \begin{pmatrix} w_{p0} \\ \boldsymbol{w}_p \end{pmatrix} = \begin{pmatrix} 1 - \displaystyle\sum_{j=\ell_p+1}^{n} \lambda_{pj} z_{pj} - \boldsymbol{q}_p^T x - \gamma_p \\ 1 + \displaystyle\sum_{j=\ell_p+1}^{n} \lambda_{pj} z_{pj} + \boldsymbol{q}_p^T x + \gamma_p \\ 2(\boldsymbol{L}_p^+)^T \boldsymbol{x} \end{pmatrix} \ (p = 1, \ldots, m), \\ & \|\boldsymbol{w}_p\| \leq w_{p0} \ (p = 1, \ldots, m), \\ & \begin{pmatrix} v_{pj0} \\ \boldsymbol{v}_{pj} \end{pmatrix} = \begin{pmatrix} 1 + \boldsymbol{z}_{pj} \\ 1 - \boldsymbol{z}_{pj} \\ 2\boldsymbol{u}_{pj}^T \boldsymbol{x} \end{pmatrix} \ (j = \ell_p + 1, \ldots, n, \ p = 1, \ldots, m), \\ & \|\boldsymbol{v}_{pj}\| \leq v_{pj0} \ (j = \ell_p + 1, \ldots, n, \ p = 1, \ldots, m). \end{array}\right\} \tag{25}$$

5 Numerical results

We present computational results on the SDP relaxation (4), the SOCP relaxation (25) with the use of the technique of reducing the number of variables and the lift-and-project LP relaxation (6). All the computation was implemented using a MATLAB toolbox, SeDuMi Version 1.03 [19] on Sun Enterprise 4500 (400MHz CPU with 6 GB memory).

The set of test problems in our numerical experiments consists of

(a) Box constraint QOPs.

(b) 0-1 integer QOPs.

(c) Box and linear inequality constraint QOPs.

(d) Box and quadratic inequality constraint QOPs.

The problems in (b) and (c) are from literature [4] and [3], respectively. In both cases, the optimal solution of the generated problem is known in advance. The problems in (d) are randomly generated in the form of (1). We use the notations described in Table 1 in the discussion of computational results.

5.1 Box constraint QOPs

The general form of box constraint QOPs (a) is

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{q}^T \boldsymbol{x} \\ \text{subject to} & x_j^2 \leq 1 \ (j = 1, 2, \ldots, n) \end{array}\right\}$$

n	the number of variables
m	the number of quadratic inequality constraints
m_l	the number of linear constraints
$\#\lambda$	the number of negative eigenvalues of $\boldsymbol{Q}_p$
SDP	the SDP relaxation (4)
SOCP	the SOCP relaxation (25)
LP	the lift-and-project LP relaxation (6)
obj.val.	the value of objective function obtained
rel.err.	the relative error, *i.e.*, rel.err. $= \dfrac{\lvert \text{obj.val.} - \zeta^* \rvert}{\max\{\lvert\zeta^*\rvert, 1.0\}}$,
	where ζ^* denotes the optimal value of the test problem under consideration.
CPU	the CPU time in seconds
it.	the number of iterations that the corresponding relaxation takes

Table 1: Notations

We rewrite the problem as

$$\left.\begin{array}{ll} \text{minimize} & x_{n+1} \\ \text{subject to} & \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} - x_{n+1} \leq 0, \\ & -\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{q}^T\boldsymbol{x} + x_{n+1} \leq 0, \\ & x_j^2 \leq 1 \ (j = 1, 2, \ldots, n). \end{array}\right\} \tag{26}$$

The matrix $\boldsymbol{Q}$ and the vector $\boldsymbol{q}$ in (26) are generated randomly. The random number generator in MATLAB is used to produce a random number on the interval $(0.0, 10.0)$ for each element in $\boldsymbol{q}$ and an $n \times n$ matrix $\boldsymbol{A}$. Then, a symmetric matrix $\boldsymbol{Q}$ is attained from $(\boldsymbol{A} + \boldsymbol{A}^T)/2$. The number of the negative eigenvalues of $\boldsymbol{Q}$ generated this way was $n/2$ in all of our test problems.

The numerical results on the problem (26) are shown in Table 2. As n increases, SDP takes CPU time about 10 times as much as SOCP, and SOCP twice as much as LP (e.g., $n = 50, 100$). The discrepancies in the object values of SOCP with SDP and LP remain relatively unchanged; SOCP provides the objective values around the middle of those of SDP and LP, (e.g., for $n = 400$, SOCP-SDP = 2425, LP-SOCP= 2014, for $n = 200$, SOCP-SDP= 858, LP-SOCP= 1071), with the speed comparable to LP.

Remark 5.1. We may replace the convex quadratic inequalities $x_j^2 \leq 1$ $(j = 1, 2, \ldots, n)$ by the linear inequalities $-1 \leq x_j \leq 1$ $(j = 1, 2, \ldots, n)$ in the QOP (26) to obtain an equivalent but simpler QOP. It is known that the SDP relaxation and the lift-and-project LP relaxation of the resulting simpler problem are not as effective as the SDP relaxation and the lift-and-project relaxation of the original QOP (26), respectively. See [5] for more details.

n	$\#\lambda$	SDP			SOCP			LP		
		obj.val.	CPU	it.	obj.val.	CPU	it.	obj.val.	CPU	it.
10	5	-101.3	0.3	13	-122.1	0.2	10	-172.0	0.1	7
50	25	-1245.5	2.5	15	-1396.9	0.5	13	-1612.8	0.3	8
100	50	-3609.2	20.0	17	-4167.6	2.1	16	-4657.3	1.0	9
200	100	-10330.4	133.9	16	-11288.4	11.2	14	-12259.1	7 .1	10
400	200	-29967.9	1228.2	19	-32392.8	124.3	16	-34406.5	58.3	11

Table 2: Box constraint QOP

5.2 0-1 integer QOPs

Consider the 0-1 integer QOP

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} \\ \text{subject to} & x_j \in \{0,1\}\ (j=1,2,\ldots,n). \end{array}\right\}$$

We used the program given in [4] to generate the coefficient matrices and vectors of the objective functions of 3 test problems with $n = 40, 60$, and 80. To apply our SDP, lift-and-project LP, and SOCP relaxations, we rewrite the problem above as

$$\left.\begin{array}{ll} \text{minimize} & x_{n+1} \\ \text{subject to} & \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} - x_{n+1} \leq 0, \\ & -\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{q}^T\boldsymbol{x} + x_{n+1} \leq 0, \\ & x_j(x_j-1) \leq 0\ (j=1,2,\ldots,n), \\ & -x_j(x_j-1) \leq 0\ (j=1,2,\ldots,n). \end{array}\right\}$$

The numerical results are summarized in Table 3. We have generated ten problems of the same size for each n and obtained computational results. The motivation for these experiments is to provide test results with a large number of problems of the same size. The minimum, average and maximum relative error, computational time, and number of iterations from solving the ten problems are shown in Table 3. The gap in CPU time between SDP and SOCP has widened as much as 20 times on average with $n = 80$. The computational time of SOCP has stayed closely to that of LP relaxation, about 3 times when $n = 80$ in maximum case. However, SOCP maintains the effectiveness in achieving a relatively good lower bound compared to LP, as indicated in the columns of rel.err.. The relative errors of SOCP are much smaller than LP.

Remark 5.2. Suppose that $\boldsymbol{q} = \boldsymbol{0}$ in the 0-1 integer QOP above. The well-known max cut problem on graphs is an exemplified case. Under the assumption, our convex QOP

n		SDP			SOCP			LP		
		rel.err.	CPU	it.	rel.err.	CPU	it.	rel.err.	CPU	it.
	min.	0.067	1.40	10	0.273	0.4	14	0.809	0.2	4
40	aver.	0.541	1.62	11	1.555	0.4	17	9.564	0.25	4.8
	max.	1.042	12.4	67	3.138	0.7	19	21.667	0.4	6
	min.	0.005	4.20	10	0.151	0.4	15	0.819	0.4	4
60	aver.	0.470	6.03	15.1	1.543	0.6	16.5	10.478	0.45	4.2
	max.	0.844	20.3	51	4.000	2.0	22	31.125	0.6	6
	min.	0.082	9.10	9	0.482	0.4	14	2.652	0.7	4
80	aver.	0.537	15.2	16.7	2.071	0.7	15.6	6.632	0.7	4.2
	max.	1.445	54.3	61	5.957	3.4	19	14.000	1.1	6

Table 3: 0-1 integer QOPs

relaxation discussed in Section 3.3 takes the following form.

$$\left.\begin{array}{ll}
\text{minimize} & x_{n+1} \\
\text{subject to} & \boldsymbol{x}^T \left(\sum_{j=1}^{\ell} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T \right) \boldsymbol{x} + \sum_{j=\ell+1}^{n} \lambda_j z_j - x_{n+1} \leq 0, \\
& -\boldsymbol{x}^T \left(\sum_{j=\ell+1}^{n} \lambda_j \boldsymbol{u}_j \boldsymbol{u}_j^T \right) \boldsymbol{x} - \sum_{j=1}^{\ell} \lambda_j z_j + x_{n+1} \leq 0, \\
& \boldsymbol{x}^T \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{x} - z_j \leq 0 \ (j = 1, 2, \ldots, n), \\
& \sum_{j=1}^{n} z_j \leq \rho_{\max} \equiv \max\{\|\boldsymbol{x}\|^2 : 0 \leq x_j \leq 1 \ (j = 1, 2, \ldots, n)\} = n, \\
& x_j(x_j - 1) \leq 0 \ (j = 1, 2, \ldots, n).
\end{array}\right\}$$

Here λ_j and $\boldsymbol{u}_j$ $(j = 1, 2, \ldots, n)$ are eigenvalues and the corresponding eigenvectors of $\boldsymbol{Q}$ satisfying (13) and (14). The optimal solution $(\boldsymbol{x}^*, x_{n+1}^*, \boldsymbol{z}^*) \in \mathbb{R}^{2n+1}$ of the problem is given by

$$\boldsymbol{x}^* = \boldsymbol{0}, \ x_{n+1}^* = n\lambda_n, \ z_j^* = 0 \ (j = 1, 2, \ldots, n-1), \ z_n^* = \rho_{\max} = n.$$

We can see that the lower bound $n\lambda_n$ obtained coincides with the trivial lower bound

$$\min\{\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} : \|\boldsymbol{x}\|^2 \leq n\}$$

of the 0-1 integer QOP with $\boldsymbol{q} = \boldsymbol{0}$.

5.3 Box and linear inequality constraint QOPs

The test problem in this section is the linear inequality constraint QOP from the literature [3]:

$$\left.\begin{array}{ll}
\text{minimize} & \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{q}^T \boldsymbol{x} \\
\text{subject to} & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b},
\end{array}\right\}$$

n	m_l		SDP			SOCP			LP		
			rel.err.	CPU	it.	rel.err.	CPU	it.	rel.err.	CPU	it.
		min.	1.24	3.8	58	2.99	0.4	11	3.29	0.2	7
20	30	aver.	1.53	4.4	66	2.99	0.52	12.8	4.51	0.2	7.5
		max.	1.70	4.9	69	3.01	0.7	16	8.82	0.2	9
		min.	1.01	6.1	45	1.35	0.6	14	8.85	0.3	8
30	45	aver.	1.13	8.4	57.5	2.81	0.7	17.5	10.56	0.34	9.5
		max.	1.66	9.2	66	3.00	1.5	32	11.20	0.4	11
		min.	1.01	11.9	24	1.01	0.7	13	9.91	0.5	10
40	60	aver.	1.06	13.9	45.6	2.80	0.8	17	10.65	0.6	10.9
		max.	1.38	19.7	63	3.00	1.1	24	10.75	0.6	11
		min.	0.99	26.1	52	2.90	0.8	14	4.216	0.6	7
50	75	aver.	1.97	29.5	58.8	2.94	0.89	15.5	6.707	0.75	9.3
		max.	2.73	35.2	74	3.00	1.1	20	10.94	1.1	13

Table 4: Box and linear inequality constrained QOPs

where $\boldsymbol{Q} \in \mathbb{S}^n$, $\boldsymbol{q} \in \mathbb{R}^n$, $\boldsymbol{A} \in \mathbb{R}^{m_l \times n}$, and $\boldsymbol{b} \in \mathbb{R}^{m_l}$. While we generate a test problem by the code provided by [3], the optimal value *opt* and the optimal vector $\boldsymbol{x}^{op}$ of the problem are obtained. We add the box constraint

$$lb_j \leq x_j \leq ub_j \ (j = 1, 2, ...n),$$

where we use $x_j^{op} - |x_j^{op}|$ and $x_j^{op} + |x_j^{op}|$ for lb_j and ub_j $(j = 1, 2, \ldots, n)$, respectively to ensure a feasible region large enough to include the optimal solution of the original problem. We rewrite the problem with a quadratic constraint obtained from the box constraint as

$$\left.\begin{array}{ll} \text{minimize} & x_{n+1} \\ \text{subject to} & \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} - x_{n+1} \leq 0, \\ & -\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{q}^T\boldsymbol{x} + x_{n+1} \leq 0, \\ & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b} \\ & (ub_j - x_j)(lb_j - x_j) \leq 0 \ (j = 1, 2, ...n+1) \end{array}\right\}$$

Ten test problems of the same size are generated for each n and m_l. Table 4 shows the minimum, average and maximum relative error, computational time, and number of iterations from solving the ten problems. As shown in Table 4, SOCP is as fast as LP, yet very effective in obtaining lower bounds.

5.4 Box and quadratic inequality constraint QOPs

The box constraint of $-1 \leq x_j \leq 1$ $(j = 1, 2, \ldots, n)$ is added to have a bounded feasible region for the QOP (1). In this case, C_0 is the region represented by the box constraint. The vector $\boldsymbol{c}$ in the objective function is chosen to be $(1, 1, \ldots, 1)^T \in \mathbb{R}^n$. Random numbers from a uniform distribution on the interval $(-1.0, 0.0)$ are assigned to the real number γ_p and each component of the vector $\boldsymbol{q}_p$ $(p = 1, 2, \ldots, m)$. We provide the number of positive and negative eigenvalues of $\boldsymbol{Q}_p$ prior to generating $\boldsymbol{Q}_p$, which represent the convex and concave parts of $\boldsymbol{Q}_p$, respectively, and then produce $\boldsymbol{Q}_p$ accordingly. More precisely, we first generate a diagonal matrix $\boldsymbol{D}_p = \text{diag}[1, \lambda_2, \ldots, \lambda_n]$ with a predetermined number of negative/positive diagonal entries, where each λ_i denotes a random number uniformly distributed either in the interval $\in (0.0, 1.0)$ if $\lambda_i > 0$ or in the interval $\in (-1.0, 0.0)$ if $\lambda_i < 0$ $(i = 2, \ldots, n)$. And, the lower triangular matrix $\boldsymbol{L}_p$ and a matrix $\boldsymbol{E}$ are created as

$$\boldsymbol{L}_p = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & 0 & 0 \\ \vdots & \vdots & & \ddots & 0 \\ l_{n1} & l_{n2} & \cdots & \cdots & l_{nn} \end{bmatrix}, \quad \boldsymbol{E} = \begin{bmatrix} -1 & -1 & \cdots & -1 & -1 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where l_{ij} is a random number uniformly distributed in the interval $(-1.0, 1.0)$. Finally we generate each $\boldsymbol{Q}_p \in \mathbb{S}^n$ such that

$$\boldsymbol{Q}_p = 2n\boldsymbol{E}\boldsymbol{L}_p\boldsymbol{D}_p\boldsymbol{L}_p^T\boldsymbol{E}^T.$$

By the construction, $\boldsymbol{x} = \boldsymbol{0}$ always gives an interior feasible solution of the QOP (1) since $\gamma_p < 0$ $(p = 1, 2, \ldots, m)$, and the objective function $\boldsymbol{c}^T\boldsymbol{x}$ of the QOP (1) takes its minimal value over the box constraint $C_0 = \{\boldsymbol{x} \in \mathbb{R}^n : 0 \leq x_j \leq 1 \ (j = 1, 2, \ldots, n)\}$ at $\boldsymbol{x} = (-1, -1, \ldots, -1)^T \in \mathbb{R}^n$. But this point is cut off by each quadratic inequality constraint generated. In fact, we observe that if $\boldsymbol{x} = (-1, -1, \ldots, -1)^T \in \mathbb{R}^n$ then

$$\begin{aligned} \boldsymbol{x}^T\boldsymbol{Q}_p\boldsymbol{x} + \boldsymbol{q}_p^T\boldsymbol{x} + \gamma_p &\geq \boldsymbol{x}^T\left(2n\boldsymbol{E}\boldsymbol{L}_p\boldsymbol{D}_p\boldsymbol{L}_p^T\boldsymbol{E}^T\right)\boldsymbol{x} - n - 1 \\ &= 2n - n - 1 = n - 1 > 0. \end{aligned}$$

Therefore $C_0 \ni (-1, -1, \ldots, -1)^T \notin F$ and the optimal value of the QOP (1) is greater than $\min\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{x} \in C_0\} = -n$. This does not guarantee, however, that the point $(-1, -1, \ldots, -1)^T \in \mathbb{R}^n$ is cut off by its convex relaxations F^{SDP}, F^{SOCP} and F^{LP}.

The numerical results for various n, m, and $\#\lambda$ on each relaxation are shown in Table 5. We have chosen $n \geq 50$, $m \leq n$ and $\#\lambda = 2, n/2, n-2$ to observe the effects on different numbers of variables and constraints, and increasing/decreasing convexity in $\boldsymbol{Q}_p$ for large n. Various combinations of n and m with small number of negative eigenvalues $\#\lambda$, namely 2, clearly indicate that SOCP is from 3.5 (n=100, m=100) up to 12 times ($n = 50$, $m = 10$) faster than SDP, while resulting effective lower bounds near SDP except for the case $n = 100$, $m = 100$, and $\#\lambda = 2$. Decreasing convexity in $\boldsymbol{Q}_p$, which is caused by introducing more negative eigenvalues, places the performance of SOCP in the middle of SDP and LP in terms of CPU time and abilities in obtaining lower bounds.

In Section 3.3, we have introduced the techniques on reducing the number of variables. Instead of dealing with n^2 variables in $\boldsymbol{X}$, we have implemented the SOCP relaxation (25) for the reduced number of variables. The efficiency of this approach can be measured with reducing the number of constraints m for a given n, since the number of variables is dependent on the size of m and the number of negative eigenvalues in $\boldsymbol{Q}_p$. In Table 5, the tests on the three problems with $n = 50$, varying m from 10, 25 to 50, and 50 negative eigenvalues show that as we decrease the size of m, the differences in CPU time between SDP and SOCP become large. We observe similar results for the problems with $n = 100$, $m = 20$, 50, and 100. This confirms the observation in Section 3.3 that QOPs with a smaller number of inequality constraints m are better suited to the approach (25).

n	m	$\#\lambda$	SDP			SOCP			LP		
			obj.val.	CPU	it.	obj.val.	CPU	it.	obj.val.	CPU	it.
50	10	2	-0.392	15.5	59	-0.609	1.2	13	-50.0	0.6	2
50	10	25	-0.452	13.3	54	-46.205	5.5	18	-50.0	0.5	2
50	10	48	-0.069	11.5	45	-6.898	5.2	18	-29.618	3.0	18
50	25	2	-0.115	29.1	72	-0.178	2.6	14	-50.0	1.0	2
50	25	25	-0.259	31.2	73	-38.077	12.0	36	-50.0	0.9	2
50	25	48	-0.065	17.9	43	-3.931	13.0	22	-27.021	8.5	26
50	50	2	-0.162	48.0	62	-0.266	6.5	16	-50.0	2.2	2
50	50	25	-0.150	49.5	64	-37.809	34.7	41	-50.0	2.0	2
50	50	48	-0.063	34.5	44	-2.301	27.5	21	-21.784	24.6	34
100	20	2	-0.347	192.2	75	-1.455	12.3	13	-100.0	4.6	2
100	20	50	-0.074	129.2	50	-10.079	42.5	24	-53.754	35.6	26
100	20	98	-0.094	141.6	55	-8.563	71.0	23	-61.683	33.3	26
100	50	2	-0.162	426.7	78	-0.815	41.3	14	-100.0	11.7	2
100	50	50	-0.062	286.2	52	-4.909	165.6	27	-32.852	86.3	24
100	50	98	-0.090	290.7	56	-4.995	274.5	29	-51.294	133.7	38
100	100	2	-1.27e-06	1227.2	112	-46.1102	355.3	47	-100.0	26.2	2
100	100	50	-1.22e-06	1132.4	109	-99.9748	910.3	81	-100.0	30.5	2
100	100	98	-8.63e-02	589.4	55	-85.6803	502.0	25	-100.0	63.0	7

Table 5: General QOPs

6 Concluding discussions

In most of the test problems that we listed in the previous section, the SOCP relaxation attained much better lower bounds than the lift-and-project LP relaxation without spending much additional CPU time. Although a lower bound computed by the SOCP relaxation is not as good as a bound by the SDP relaxation in theory, they are often close to each other and savings in CPU time grows as the size of the QOP increases, as we have observed through the numerical experiments.

All the three convex relaxations provide merely a lower bound of the optimal objective value of a QOP, which is not tight in general. One way of tightening of the lower bound computed is to subdivide the QOP into smaller subproblems, and then apply the convex relaxations to each subproblem. This technique is commonly used in the branch-and-bound method. Another way of tightening the lower bound is to apply them repeatedly in the framework of successive convex relaxation methods [12, 20, etc.]. In both methods, it is critical to have better lower bounds with less computational costs. Therefore the SOCP relaxation proposed in this article can serve as a reasonable candidate to be incorporated into those methods.

References

[1] Balas, E., Ceria, S., Cornuéjols, G. (1993): A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming* **58**, 295–323

[2] Burer, S., Monteiro, R. D. C., Zhang, Y. (1999): Solving semidefinite programs via nonlinear programming part I: Transformations and derivatives. School of Mathematics, Georgia Tech., Atlanta, Georgia 30332, USA.

[3] Calamai, P. H., Vincente, L. N., Judice, J. J. (1993): A new technique for generating quadratic programming test problems, *Math. Programming* **61**, no. 2, Ser. A, 215-231

[4] Floudas, C. A., Pardalos, P. M.(1990): *A collection of test problems for constrained global optimization algorithms.* Lecture Notes in computing sciences, Springer, New York.

[5] Fujie, T., Kojima, M. (1997): Semidefinite relaxation for nonconvex programs. *Journal of Global Optimization* **10**, 367–380

[6] Fujisawa, K., Kojima, M., Nakata, K. (1995): SDPA (SemiDefinite Programming Algorithm) user's manual, Version 5.0. Research Report B-308, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh-Okayama, Meguro, Tokyo 152-8552, Japan.

[7] Fujisawa, K., Kojima, M., Nakata, K. (1997): Exploiting Sparsity in Primal-Dual Interior-Point Methods for Semidefinite Programming. *Mathematical Programming* **79**, 235–254

[8] Fukuda, M., Kojima, M., Murota, K., Nakata, K.: Exploiting sparsity in semidefinite programming via matrix completion I: General framework. To appear in *SIAM Journal on Optimization* **11** (2001) 647–674.

[9] Goemans, M. X., Williamson, D. P. (1995): Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of Assoc. Comput. Mach.* **42**, 1115–1145

[10] Grötschel, M., Lovász, L., Schrijver, A. (1988): *Geometric algorithms and combinatorial optimization.* Springer, New York, NY.

[11] Helmberg, C. and Rendl, F. (1997): "A spectral bundle method for semidefinite programming," *SIAM J. on Optimization* **10**, 673–696.

[12] Kojima, M., Tunçel, L. (2000): Cones of matrices and successive convex relaxations of nonconvex sets. *SIAM J. on Optimization* **10**, 750–778.

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

[14] Lovász, L.(1998): "Applications in Combinatorial Optimization," presented at Workshop *Semidefinite Programming: Applications and Algorithms,*" ZIB-Berlin, November 15-17.

[15] Nesterov, Yu. E. (1998): Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software* **9**, 141–160

[16] Poljak, S., Rendl, F., Wolkowicz, H. (1995): A recipe for semidefinite relaxation for (0,1)-quadratic programming. *Journal of Global Optimization* **7**, 51–73

[17] Sherali, H. D., Tuncbilek, C. H. (1992): A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. *Journal of Global Optimization* **2**, 101–112

[18] Shor, N. Z. (1990): Dual quadratic estimates in polynomial and boolean programming. *Annals of Operations Research* **25**, 163–168

[19] Sturm, F. J. (1999): Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software* **11-12**, 625–653

[20] Takeda, A., Dai, Y., Fukuda, M., Kojima, M.: Towards the implementation of successive convex relaxation method for nonconvex QOP. To appear in *Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems* (P. M. Pardalos, Editor), Kluwer Academic Press.

[21] Toh, K., Todd, M. J., Tütüntü, R. H. (1998): SDPT3 — a MATLAB software package for semidefinite programming. Dept. of Mathematics, National University of Singapore, Singapore.

[22] Ye, Y. (1999): Approximating quadratic programming with bound and quadratic constraints. *Mathematical Programming* **84**, 219–226