Exact Solutions of Some Nonconvex Quadratic Optimization Problems via SDP and SOCP Relaxations

Sunyoung Kim⋆
Department of Mathematics, Ewha Women's University
11-1 Dahyun-dong, Sudaemoon-gu, Seoul 120-750 Korea
email: *skim@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*

B-375, January 2002. Revised February 2003.

Abstract We show that SDP (semidefinite programming) and SOCP (second order cone programming) relaxations provide exact optimal solutions for a class of nonconvex quadratic optimization problems. It is a generalization of the results by S. Zhang for a subclass of quadratic maximization problems that have nonnegative off-diagonal coefficient matrices of quadratic objective functions and diagonal coefficient matrices of quadratic constraint functions. A new SOCP relaxation is proposed for the class of nonconvex quadratic optimization problems by extracting valid quadratic inequalities for positive semidefinite cones. Its effectiveness to obtain optimal values is shown to be the same as the SDP relaxation theoretically. Numerical results are presented to demonstrate that the SOCP relaxation is much more efficient than the SDP relaxation.

Key words. Nonconvex quadratic optimization problem, semidefinite programming relaxation, second order cone programming relaxation, sparsity

⋆ A considerable amount of this work was conducted while this author was visiting Tokyo Institute of Technology, Department of Mathematical and Computing Sciences. Research supported by Kosef R004-000-2001-00200.

1 Introduction

We are concerned with solving quadratic optimization problems (QOPs) with quadratic constraints by semidefinite programming (SDP) relaxation and second order cone programming (SOCP) relaxation. QOPs have been a subject of extensive study for their theoretical and practical importance in optimization. The focus of this paper, in particular, is on nonconvex QOPs which involve indefinite coefficient matrices in the objective function and constraints.

QOPs arise in a broad range of fields such as combinatorial optimization, numerical partial differential equations from engineering, control and finance, and general nonlinear programming problems. Nonconvex QOPs include indefinite symmetric matrices in the objective and constraints, as opposed to convex QOPs whose coefficient matrices are positive semidefinite. Local optimizers of convex QOPs serve as global optimizers, hence, an approximate global optimizer can be found using many publically available codes [1, 8, 9]. For solutions of convex QOPs, formulating the QOPs as SOCP problems is a possible approach before applying the primal-dual interior-point method.

Nonconvex QOPs are known to be NP-hard. Lovász and Schrijver [4] and others showed that certain types of NP-hard combinatorial optimization problems can be approximated using SDPs. Goemans and Williamson [2] proved an approximated result by SDP relaxation for the max cut problem, which is a special QOP. This work was extended to approximately solve QOPs with diagonally homogeneous quadratic constraints and simple bound by Nesterov [6] and Ye [10]; a quality bound was established when the constraints are convex and homogeneous by Nesterov [7]; Zhang [11] showed that an optimal solution of QOPs can be found from the optimal solution of SDP relaxation for special subclasses of QOPs. When numerical solutions are to be computed, we can use software such as [1, 8, 9, etc.], which are extensions of interior-point methods developed for LPs, to SDPs.

SOCP (second order cone programming) relaxations were proposed in [3] as efficient methods for obtaining effective bounds for optimal values of QOPs. As the size of QOPs grows in the number of variables and constraints, computing time needed to solve SDP relaxations of QOPs increases rapidly. This has prevented us from finding bounds for optimal values of large scale QOPs within reasonable amount of computational time. The SOCP relaxation in [3] was formulated based on the lift-and-project LP (linear programming) relaxation by adding convex quadratic valid inequalities for the constraint $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \succeq \boldsymbol{0}$, which indicates an infinite number of convex quadratic inequalities, in the SDP relaxation. The resulting SOCP relaxation method fell between SDP relaxation and the lift-and-project LP relaxation in the sense that it strengthened the lift-and-project LP relaxation method with convex quadratic inequalities. Numerically, the SOCP relaxation was shown to be a reasonable compromise between the effectiveness of the SDP relaxation and the low computational cost of the lift-and-project LP relaxation. A SOCP relaxation for the max cut problem was proposed by Muramatsu [5]. It took into consideration of the triangular inequalities of the max cut problem to reinforce the quality of the added convex quadratic inequalities. In this paper, we generate a new type of quadratic valid inequalities for positive semidefinite cones, by requiring a necessary condition that all the 2×2 principal submatrices of $\boldsymbol{X}$ are nonnegative for an $\boldsymbol{X} \in \mathcal{S}^n$ to be positive semidefinite. Hence the SOCP relaxation obtained here is different from the ones in [3, 5]. In a class of QOPs which we will introduce, the added finite number of quadratic inequalities are equally effective as the positive semidefinite constraint. Considering a smaller number of the variables than SDP relaxation for QOPs with sparse data matrices in this approach, we can expect increased effectiveness as well as numerical efficiency over SDP relaxation.

The goal of this paper is to demonstrate, first, that SDP and SOCP relaxations can provide exact optimal solutions for a class of QOPs in theory. Second, SOCP relaxation is a much more efficient method than SDP relaxation for the class of QOPs when solutions are sought numerically. More specifically, we find a class of QOPs such that their optimal values are the same as the bounds obtained by their SDP and SOCP relaxations and optimal solutions for the original problems can be obtained from the optimal solutions of the SDP and SOCP relaxations. We call this property as SDP or SOCP relaxation is exact. We should mention that this class is an extension of a special subclass of QOPs by Zhang [11] to more general classes. Numerical experiments are also given to illustrate that SOCP relaxation is much faster in finding solutions than SDP relaxation.

The remaining of the paper is organized as follows: After introducing a standard form of QOPs and brief description of their SDP relaxation in Section 2, we show in Section 3 classes of QOPs whose optimal values can be obtained exactly by SDP and SOCP relaxations. More precisely, we show that the QOPs with nonpositive off-diagonal elements can be solved without gap. We also describe how the class of QOPs can be transformed into SOCP problems. In Section 4, we present numerical results from SDP and SOCP relaxations, and compare the results and performance with respect to computing time. Section 5 is devoted to concluding discussions.

Throughout the paper, we use the following notation: Let $\mathbb{R}^n$, $\mathcal{S}^n$ and $\mathcal{S}^n_+$ 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 $\boldsymbol{Q}_p \in \mathcal{S}^n$, $\boldsymbol{q}_p \in \mathbb{R}^n$, and $\gamma_p \in \mathbb{R}$ $(0 \leq p \leq m)$. For $\boldsymbol{A}, \boldsymbol{B} \in \mathcal{S}^n$, we denote $\boldsymbol{A} \bullet \boldsymbol{B} = \sum_{i=1}^n \sum_{j=1}^n \boldsymbol{A}_{ij}\boldsymbol{B}_{ij}$, and $\boldsymbol{A} \succeq \boldsymbol{B}$ means $\boldsymbol{A} - \boldsymbol{B}$ is positive semidefinite. We use ";" to concatenate vectors into a column.

2 A general QOP and its SDP relaxation

Consider a QOP of the form

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{x}^T\boldsymbol{Q}_0\boldsymbol{x} + 2\boldsymbol{q}_0^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{x}^T\boldsymbol{Q}_p\boldsymbol{x} + 2\boldsymbol{q}_p^T\boldsymbol{x} + \gamma_p \leq 0 \ (1 \leq p \leq m),\ \boldsymbol{x}^2 \in \mathcal{F}. \end{array}\right\} \quad (1)$$

Here $\boldsymbol{Q}_p$ is an $n \times n$ symmetric matrix, $\boldsymbol{q}_p \in \mathbb{R}^n$, $\gamma_p \in \mathbb{R}$ for $0 \leq p \leq m$, $\mathcal{F}$ a closed convex subset of $\mathbb{R}^n$, and $\boldsymbol{x}^2 = (x_1^2, x_2^2, \ldots, x_n^2)^T$. For convenience, we adopt the following matrix notation:

$$\boldsymbol{M}_p = \begin{pmatrix} \gamma_p, & \boldsymbol{q}_p^T \\ \boldsymbol{q}_p, & \boldsymbol{Q}_p \end{pmatrix} \ (0 \leq p \leq m) \ \text{ and } \boldsymbol{M}_{m+1} = \begin{pmatrix} 1, & \boldsymbol{0}^T \\ \boldsymbol{0}, & \boldsymbol{O} \end{pmatrix}.$$

Here we take $\gamma_0 = 0$. Then we can rewrite QOP (1) as a HQOP (homogeneous QOP)

$$\left.\begin{array}{ll} \text{minimize} & (x_0;\boldsymbol{x})^T\boldsymbol{M}_0(x_0;\boldsymbol{x}) \\ \text{subject to} & (x_0;\boldsymbol{x})^T\boldsymbol{M}_p(x_0;\boldsymbol{x})^T \leq 0 \ (p = 1, ..., m), \\ & (x_0;\boldsymbol{x})^T\boldsymbol{M}_{m+1}(x_0;\boldsymbol{x})^T = 1,\ \boldsymbol{x}^2 \in \mathcal{F}. \end{array}\right\} \quad (2)$$

It is easily verified that $(x_0; \boldsymbol{x})$ solves HQOP (2) if and only if $x_0^2 = 1$ and $\boldsymbol{x}/x_0$ solves QOP (1).

The SDP (semidefinite programming) relaxation of QOP (1) is

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{Q}_0 \bullet \bar{\boldsymbol{X}} + 2\boldsymbol{q}_0^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{Q}_p \bullet \bar{\boldsymbol{X}} + 2\boldsymbol{q}_p^T \boldsymbol{x} + \gamma_p \leq 0 \ (1 \leq p \leq m), \\ & \text{diag}\,(\bar{\boldsymbol{X}}) \in \mathcal{F},\ \boldsymbol{X} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \bar{\boldsymbol{X}} \end{pmatrix} \in \mathbb{S}_+^{1+n}. \end{array}\right\}$$

Using $\boldsymbol{M}_p$, we rewrite the SDP relaxation as

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{M}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{M}_p \bullet \boldsymbol{X} \leq 0 \ (p = 1, ..., m),\ X_{00} = \boldsymbol{M}_{m+1} \bullet \boldsymbol{X} = 1, \\ & \text{diag}\,(\bar{\boldsymbol{X}}) \in \mathcal{F},\ \boldsymbol{X} = \begin{pmatrix} X_{00} & \boldsymbol{x}^T \\ \boldsymbol{x} & \bar{\boldsymbol{X}} \end{pmatrix} \in \mathbb{S}_+^{1+n}. \end{array}\right\} \quad (3)$$

3 A class of quadratic optimization problems that can be solved by SDP and SOCP

In this section, we consider QOPs whose SDP and SOCP relaxations admit no gap with the true optimal values, and whose optimal solutions can be found directly from SDP and SOCP optimal solutions. We provide a condition when HQOP (2) can be solved exactly by the SDP relaxation.

In Zhang [11], a class of QOPs is considered:

$$\text{maximize} \quad \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} \quad \text{subject to} \quad \boldsymbol{x}^2 \in \mathcal{F}. \quad (4)$$

Apparently, the QOP above is a special case of QOP (1) where we take $\boldsymbol{Q}_0 = -\boldsymbol{Q}$, $\boldsymbol{q}_0 = \boldsymbol{0}$ and $m = 0$. To be consistent with our arguments here and with what we usually call convex QOPs having convex quadratic objective functions to be minimized and convex quadratic inequality constraints, we will deal with a minimization problem:

$$\text{minimize} \quad \boldsymbol{x}^T \boldsymbol{Q}_0 \boldsymbol{x} \quad \text{subject to} \quad \boldsymbol{x}^2 \in \mathcal{F} \quad (5)$$

instead of the maximization problem (4). A SDP relaxation of (5) is given as

$$\text{minimize} \quad \boldsymbol{Q}_0 \bullet \bar{\boldsymbol{X}} \quad \text{subject to} \quad \text{diag}\,(\bar{\boldsymbol{X}}) \in \mathcal{F}, \quad \bar{\boldsymbol{X}} \succeq \boldsymbol{0}, \quad (6)$$

which is a special case of (3). Zhang [11] showed that the optimal value of QOP (5) is equal to the optimal value of its SDP relaxation (6) if all off-diagonal elements of $\boldsymbol{Q}_0 = -\boldsymbol{Q}$ are nonpositive (Corollary 1 of [11]) or more generally if $\boldsymbol{Q}_0$ is *almost OD-nonpositive* (or if $\boldsymbol{Q}$ is *almost OD-nonnegative*, Theorem 4 of [11]). See Definition 3.2 below. We will extend this result to more general cases in Theorems 3.1 and 3.4.

Theorem 3.1. *(An extension of Corollary 1 of [11]). Assume that all off-diagonal elements of $\boldsymbol{M}_p$ $(0 \leq p \leq m)$ are nonpositive. Let $\boldsymbol{X}$ be an optimal solution of the SDP relaxation (3). Then $(\hat{x}_0; \hat{\boldsymbol{x}}) = \left(1, \sqrt{X_{11}}, \ldots, \sqrt{X_{nn}}\right)^T$ is an optimal solution of HQOP (2).*

Proof: By definition, we first observe that

$$\hat{\boldsymbol{x}}^2 \in \mathcal{F},\ \hat{x_0} = 1 \quad \text{and} \quad [\boldsymbol{M}_p]_{jj}\hat{x}_j^2 = [\boldsymbol{M}_p]_{jj}X_{jj}\ (0 \leq j \leq n,\ 0 \leq p \leq m).$$

Since $\boldsymbol{X}$ is positive semidefinite, we see that

$$(X_{kj})^2 \leq X_{kk}X_{jj}\ (0 \leq k < j \leq n). \tag{7}$$

Hence, it follows from the nonpositivity of all off-diagonal elements of $\boldsymbol{M}_p$ $(0 \leq p \leq m)$ that

$$[\boldsymbol{M}_p]_{kj}\hat{x}_k\hat{x}_j = [\boldsymbol{M}_p]_{kj}\sqrt{X_{kk}}\sqrt{X_{jj}} \leq [\boldsymbol{M}_p]_{kj}\,|X_{kj}| \leq [\boldsymbol{M}_p]_{kj}X_{kj}$$
$$(0 \leq k < j \leq n,\ 0 \leq p \leq m).$$

Therefore, for every $p = 0, 1, 2, \ldots, m$, we obtain that

$$(1; \hat{\boldsymbol{x}})^T\boldsymbol{M}_p(1; \hat{\boldsymbol{x}}) = \sum_{k=0}^{n}\sum_{j=0}^{n}[\boldsymbol{M}_p]_{kj}\hat{x}_k\hat{x}_j \leq \sum_{k=0}^{n}\sum_{j=0}^{n}[\boldsymbol{M}_p]_{kj}X_{kj} = \boldsymbol{M}_p \bullet \boldsymbol{X}.$$

That is, $(1; \hat{\boldsymbol{x}})$ is a feasible solution of QOP (1) and its objective value is at least as good as $\boldsymbol{M}_0 \bullet \boldsymbol{X}$. ▮

The assumption that all off-diagonal elements of $\boldsymbol{M}_p$ are nonpositive plays an important role in the proof of Theorem 3.1. The class of QOPs satisfying the assumption seems to represent a very small group of QOPs. The assumption, however, can be extended to a slightly larger class of QOPs, in which $[M_p]_{ij}$ are not necessarily of the same sign for $i \neq j$. We observe this with the following definition.

Definition 3.2. *Zhang [11] A symmetric matrix $\boldsymbol{A} \in \mathcal{S}^\ell$ is said to be almost OD-nonpositive (or almost OD-nonnegative, respectively) if there exists a sign vector $\boldsymbol{\sigma} \in \{-1, +1\}^\ell$ such that*

$$[\boldsymbol{A}]_{ij}\sigma_i\sigma_j \leq 0 \text{ (or } [\boldsymbol{A}]_{ij}\sigma_i\sigma_j \geq 0, \text{ respectively) } (1 \leq i < j \leq \ell).$$

Here $[\boldsymbol{A}]_{ij}$ denotes the (i, j)th element of $\boldsymbol{A}$.

Definition 3.3. *A family of symmetric matrices $\boldsymbol{A}_p \in \mathcal{S}^\ell$ $(1 \leq p \leq m)$ is said to be uniformly almost OD-nonpositive (or uniformly almost OD-nonnegative, respectively) if there exists a sign vector $\boldsymbol{\sigma} \in \{-1, +1\}^\ell$ such that*

$$[\boldsymbol{A}_p]_{ij}\sigma_i\sigma_j \leq 0 \text{ (or } [\boldsymbol{A}_p]_{ij}\sigma_i\sigma_j \geq 0, \text{ respectively) } (1 \leq i < j \leq \ell,\ 1 \leq p \leq m).$$

As a result, Theorem 3.1 can be extended to the following.

Theorem 3.4. *(An extension of a part of Theorem 4 of [11]). Assume that the family of symmetric matrices $\boldsymbol{M}_p \in \mathcal{S}_+^{1+n}$ $(0 \leq p \leq m)$ is uniformly almost OD-nonpositive with a sign vector $\boldsymbol{\sigma} \in \{-1, +1\}^{1+n}$. Let X be an optimal solution of the SDP relaxation (3). Then $(\hat{x_0}; \hat{\boldsymbol{x}}) = \left(1, \sigma_0\sigma_1\sqrt{X_{11}}, \ldots, \sigma_0\sigma_n\sqrt{X_{nn}}\right)^T$ is an optimal solution of HQOP (2).*

Proof: By assumption, $[\boldsymbol{M}_p]_{ij}\sigma_i\sigma_j \leq 0$ $(0 \leq i < j \leq n,\ 0 \leq p \leq m)$. Replace the variable x_j by $\sigma_0\sigma_j x_j$ in HQOP (2). Then the resulting HQOP satisfies the assumption of Theorem 3.1. Hence the conclusion follows. ∎

In the proof of Theorem 3.1, the set of quadratic inequalities (7), which is merely a necessary (but not sufficient) condition for $\boldsymbol{X}$ to be positive semidefinite, is essential. In other words, even if we replace the positive semidefinite requirement in the SDP relaxation (3) by the weaker condition (7), the conclusion of Theorem 3.1 remains true. In general, the resulting problem involves $n(n+1)/2$ quadratic inequality constraints as in (7). When the data matrices $\boldsymbol{M}_p$ $(0 \leq p \leq m)$ are sparse, however, we can reduce the number of the quadratic inequality constraints. Let

$$\Lambda = \{(k,j)\ :\ 0 \leq k < j \leq n \ \text{ and } [\boldsymbol{M}_p]_{kj} \neq 0 \text{ for some } p\}.$$

Thus we consider the following relaxation of (1).

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{M}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{M}_p \bullet \boldsymbol{X} \leq 0\ (1 \leq p \leq m),\ X_{00} = \boldsymbol{M}_{m+1} \bullet \boldsymbol{X} = 1, \\ & (X_{11}, X_{22}, \ldots, X_{nn})^T \in \mathcal{F},\ X_{jj} \geq 0\ (1 \leq j \leq n), \\ & (X_{kj})^2 \leq X_{kk}X_{jj}\ ((k,j) \in \Lambda)) \end{array}\right\} \qquad (8)$$

Theorem 3.5. *Assume that the family of symmetric matrices* $\boldsymbol{M}_p \in \mathcal{S}^{1+n}_+$ $(0 \leq p \leq m)$ *is uniformly almost OD-nonpositive with a sign vector* $\boldsymbol{\sigma} \in \{-1,+1\}^{1+n}$. *Let* $\boldsymbol{X}$ *be an optimal solution of the problem (8). Then* $(\hat{x}_0; \hat{\boldsymbol{x}}) = \left(1, \sigma_0\sigma_1\sqrt{X_{11}}, \ldots, \sigma_0\sigma_n\sqrt{X_{nn}}\right)^T$ *is an optimal solution of HQOP (2).*

Proof: If all off-diagonal elements of $\boldsymbol{M}_p$ $(0 \leq p \leq m)$ are nonpositive, apply the same proof to the theorem as that of Theorem 3.1. Then use the same argument as in the proof of Theorem 3.4 for the general case. ∎

We now convert (8) to a SOCP. It is known that a constraint of the form

$$w^2 \leq \xi\eta,\ \xi \geq 0 \ \text{ and } \eta \geq 0$$

is equivalent to the SOCP constraint

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

Thus we obtain the following SOCP which is equivalent to the problem (8).

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{M}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{M}_p \bullet \boldsymbol{X} \leq 0\ (1 \leq p \leq m),\ X_{00} = \boldsymbol{M}_{m+1} \bullet \boldsymbol{X} = 1, \\ & (X_{11}, X_{22}, \ldots, X_{nn})^T \in \mathcal{F}, \\ & \left\| \begin{pmatrix} X_{kk} - X_{jj} \\ 2X_{kj} \end{pmatrix} \right\| \leq X_{kk} + X_{jj}\ ((k,j) \in \Lambda)). \end{array}\right\} \qquad (9)$$

It should be noted that only the variables X_{jj} $(0 \leq j \leq n)$ and X_{kj} $((k,j) \in \Lambda))$ are relevant in the problem (9). Therefore, as the number of the elements in Λ decreases, the efficiency of the problem (9) can increase. Especially, when all $\boldsymbol{Q}_p$ $(0 \leq p \leq m)$ are diagonal in QOP (1), the corresponding $\Lambda = \{(1,j) : 1 \leq j \leq n\}$ consists of n elements. We will illustrate the advantage of the SOCP relaxation (9) against the SDP relaxation (3) through numerical results in the next section.

n	the number of variables
m	the number of quadratic inequality constraints
Sparsity	the density of nonzeros in off-diagonal elements of $\boldsymbol{Q}_p$ and elements of $\boldsymbol{q}_p$
SDP	the SDP relaxation
SOCP	the SOCP relaxation
cpu	the cpu time in seconds
it.	the number of iterations that the corresponding relaxation takes
cpu.ratio	cpu time consumed by SDP relaxation/ that by SOCP relaxation

Table 1: Notation

4 Numerical results

We present computational results on the SDP relaxation (3) and the SOCP relaxation (9) to compare the results of Theorems 3.4 and 3.5. All the computation was implemented using a MATLAB toolbox, SeDuMi Version 1.03 [8] on Sun Enterprise 4500 (CPU 400MHz with 6 GB memory). The set of test problems in our numerical experiments consists of

(a) QOPs with nonpositive off-diagonal $\boldsymbol{Q}_p$ and $\boldsymbol{q}_p \leq \boldsymbol{0}$ $(0 \leq p \leq m)$.

(b) QOPs with diagonal $\boldsymbol{Q}_p$ and $\boldsymbol{q}_p \leq \boldsymbol{0}$ $(0 \leq p \leq m)$.

In both cases, we took $\mathcal{F} = \mathbb{R}^n$. We use the notation described in Table 1 in the discussion of computational results.

4.1 QOPs with nonpositive off-diagonal $\boldsymbol{Q}_p$'s and nonpositive $\boldsymbol{q}_p$'s

One of the main purposes of creating QOPs of this type was to vary its sparsity. To control the sparsity of QOPs, a sparse vector $\boldsymbol{s}$ of the size of $n(n-1)/2$ and a sparse vector $\boldsymbol{t}$ of length n with a given density were generated using MATLAB random number generator. The number of nonzero elements in the vector $\boldsymbol{s}$ was decided by the preassigned density. Then, we generated a nonzero element in $\boldsymbol{Q}_p$ $(p = 0, 1, \ldots, m)$, according to a nonzero element of $\boldsymbol{s}$. More precisely, we first examined a component in $\boldsymbol{s}$, and if the component was nonzero, a nonzero element in $\boldsymbol{Q}_p$ was produced in the interval $(-10, 0)$ and the element in its symmetric position in $\boldsymbol{Q}_p$ was also assigned with the same number. We continued this process until we reached the last element of $\boldsymbol{s}$. Using the vector $\boldsymbol{t}$, we created a nonpositive element from the interval $(-1, 0)$ in $\boldsymbol{q}_p$ $(\boldsymbol{p} = 0, 1, \ldots, m)$; if t_j was not zero, $(\boldsymbol{q}_p)_j$ was assigned to a nonzero element. The diagonal elements of $\boldsymbol{Q}_p$ were generated randomly in the interval $(-1, 1)$. And, γ_p was set to -1 to have an interior of the feasible region. We present the numerical results of the problems generated this way.

We also tested whether the positioning of the nonzero elements in $\boldsymbol{Q}_p$ $(p = 0, 1, \ldots, m)$ made differences in performance of the SDP and SOCP relaxations. For this test, we generated nonzero elements in the interval $(-10, 0)$ to place the upper left corner of each $\boldsymbol{Q}_p$ according to the given sparsity. $\boldsymbol{q}_p$ and γ_p were chosen as above. The numerical results

n	m	Sparsity	SDP		SOCP		cpu.ratio
			cpu	it.	cpu	it.	
200	100	10%	251.4	17	32.6	22	7.7
200	100	10%	290.4	19	28.8	20	10.1
200	100	10%	287.5	19	31.5	21	9.1
200	100	10%	297.2	19	30.8	21	9.6
200	100	10%	282.8	19	31.6	21	9.0

Table 2: 5 tries with $n = 200$ and $m = 100$

n	m	Sparsity	SDP		SOCP		cpu.ratio
			cpu	it.	cpu	it.	
200	100	5%	198.8	19	15.1	18	13.1
200	100	10%	290.4	19	28.8	20	10.1
200	100	50%	1430.7	31	173.1	27	8.3
200	100	70%	1858.3	33	212.4	26	8.7
200	100	100%	2282.8	33	342.5	32	6.7

Table 3: QOPs with $n = 200$, $m = 100$ and varying sparsity

were very similar to those from the problems generated with the vectors $\boldsymbol{s}$ and $\boldsymbol{t}$ above. This leads us to say that the locations of nonzero elements in $\boldsymbol{Q}_p$ do not effect the performance of both SDP and SOCP relaxations.

We show the numerical results for various n, m and sparsity from Table 2 to Table 5. Since all the test problems were generated using MATLAB random number generator, we first experimented to see whether there existed a difference in cpu time and the number of iterations of test results each time that the program was executed. The results from five runs of the program for $n = 200, m = 100$ and sparsity=10% are summarized in Table 2. Cpu time and the number of iterations are similar in the five tries. In particular, the ratio of cpu time of SDP to that of SOCP remains in the range of 7 to 10. Based on the test, we can say that the results from one execution of the program are not much different from those of others. We proceeded rest of the tests on varying n, m and sparsity with this information.

Table 3 shows that the results from varying sparsity of $\boldsymbol{Q}_p$ $(p = 0, 1, \ldots, m)$ from 5% to 100%, fully dense matrix $\boldsymbol{Q}_p$. The objective values of the two relaxations were the same for all the experiments, which was shown in Section 3, therefore, actual values were omitted in the Tables. We observe that SOCP was much faster to obtain an optimal solution than SDP. SOCP was as fast as from 6 times to 13 times in the case of $n = 200$, $m = 100$. The speed of SOCP increased as the number of nonzero elements decreased as shown in the column of cpu.ratio. It should be noted that even in the case of fully matrices, *i.e.,* 100% sparsity in $\boldsymbol{Q}_p$, the SOCP relaxation provided better performance in finding a solution.

Numerical results for $n = 100$, sparsity= 10%, and from $m = 50$ to $m = 400$ are shown in Table 4. As m becomes large, computational advantage of SOCP shown in the cpu.ratio column varies from 8 to 6.5 to 7.7. The ratio has not worsen as m increases.

n	m	Sparsity	SDP		SOCP		cpu.ratio
			cpu	it.	cpu	it.	
100	50	10%	18.3	16	2.3	15	8.0
100	100	10%	42.1	20	6.5	19	6.5
100	200	10%	125.4	18	17.7	20	7.1
100	400	10%	733.1	19	95.6	19	7.7

Table 4: QOPs with $n = 100$ and varying m

n	m	Sparsity	SDP		SOCP		cpu.ratio
			cpu	it.	cpu	it.	
50	100	10%	12.6	15	1.3	13	9.7
100	100	10%	42.1	20	6.5	19	6.5
200	100	10%	290.4	19	28.8	20	10.1
400	100	10%	3910.4	25	236.7	36	16.5

Table 5: QOPs with $m = 100$ and varying n

If all $\boldsymbol{Q}_p$ $(p = 0, 1, \ldots, m)$ are of the same structure, increasing number of constraints m does not make the performance of SOCP relaxation worse. We can say from this that the computing time of the SDP and SOCP relaxations does not depend heavily on m. This is partly because the number of variables in both SDP and SOCP does not change when m increases.

Table 5 shows the computing time of the SDP and SOCP relaxations of QOPs with fixed $m = 100$ and sparsity = 10%. The cpu.ratio ranges from 6.5 for $n = 100, m = 100$ to 16.5 for $n = 400, m = 100$. This indicates that as the size of n becomes large, the SDP relaxation requires much cpu time to solve the same size of the problem than the SOCP relaxation. The efficiency of the SOCP relaxation increases with n. Note that the matrix variable $\boldsymbol{X} \in \mathcal{S}^{1+n}$ of the SDP relaxation involves $n(n+1)/2$ real variables while the SOCP relaxation involves merely 10% of those variables. Hence the difference in the number of variables between SDP and SOCP increases as n becomes large, and SDP's large number of variables comparing to SOCP requires much cpu time. Therefore, we observe that the problems with large n can benefit from numerical efficiency of the SOCP relaxation.

4.2 QOPs with diagonal $\boldsymbol{Q}_p$'s and nonpositive $\boldsymbol{q}_p$'s

We generated QOPs in the form of (1) with $\mathcal{F} = \mathbb{R}^n$, diagonal $\boldsymbol{Q}_p$ $(p = 0, 1, \ldots, m)$ and $\boldsymbol{q}_p \leq \boldsymbol{0}$ $(p = 0, 1, \ldots, m)$. Random numbers in $(-1, 1)$ were assigned to the diagonal elements of $\boldsymbol{Q}_p$. The vectors $\boldsymbol{q}_p$ were also created with random numbers in the range of $(-1, 0)$ and γ_p with a random number in $(-1, 0)$.

Since the SOCP relaxation (9) has only $2n$ variables and $(n+1)n/2$ variables are involved in the SDP relaxation (3), we can expect greater advantages in the SOCP relaxation than the SDP relaxation. Numerical results are shown in Table 6 for various n and m, from

$n = 100, m = 50$ to $n = 500, m = 500$. The cpu.ratio changes from 4.1 ($n = 100, m = 100$) to 26.2 ($n = 500, m = 250$ and $n = 500, m = 500$). The SOCP relaxation shows much faster performance than Section 4.1. If the number m of constraints is smaller than n (*e.g.*, $n = 100, m = 50$ and $n = 200, m = 100$), the performance of the SOCP relaxation is even better than the SOCP relaxation for the problems of equal n and m (*e.g.*, $n = 100, m = 100$ and $n = 200, m = 200$), compared to the SDP relaxation. That is, we can have smaller amount of computing time for the SOCP relaxation for the cases of m is smaller than n, though the rate deteriorates as n increases (*e.g.*, $n = 400, m = 200$ and $n = 400, m = 400$). We also see that the savings in cpu time of SOCP increases with n; SOCP shows better performance for large n than SDP.

n	m	SDP		SOCP		cpu.ratio
		cpu	it.	cpu	it.	
100	50	25.7	19	3.8	16	6.8
100	100	42.1	21	10.4	24	4.1
200	100	247.9	25	17.5	18	14.2
200	200	376.7	25	43.4	18	8.7
300	150	759.1	22	45.7	17	16.6
300	300	1709.3	31	125.6	21	13.6
400	200	2004.3	16	91.5	17	21.9
400	400	4462.8	33	242.6	19	18.4
500	250	4040.8	24	154.0	17	26.2
500	500	15245.2	16	582.6	23	26.2

Table 6: Diagonal QOPs with varying n and m

5 Concluding discussions

(A) We have shown that the SDP and SOCP relaxations provide exact optimal solutions for the class of QOPs with uniformly almost OD-nonpositive coefficient matrices. Many practical problems may not satisfy the uniformly almost OD-nonnegativity condition since the condition is too restrictive. Even in such cases, however, subproblems with some of their variables fixed may satisfy the condition. We also know that when subproblems become convex QOPs, we can solve them as SOCPs.

(B) Numerically, the proposed SOCP relaxation has proven to be more efficient than the SDP relaxation in all of the test problems in the previous section, and we have concluded that the proposed SOCP relaxation is a much better approach to the class of of QOPs with uniformly almost OD-nonpositive coefficient matrices than the SDP relaxation. From these numerical results, the SOCP relaxation is expected to work efficiently on more general QOPs in terms of computing performance although bounds which the SOCP relaxation generates for their optimal values may be inferior to bounds generated by the SDP relaxation.

(C) The discussions in (A) and (B) above suggest an effective incorporation of SOCPs into branch-and-bound methods for solving general (0-1) QOPs. An important issue to be studied further for such methods is to fix as few variables of a given QOP as possible, so that more subproblems become convex QOPs or QOPs with uniformly almost OD-nonpositive coefficient matrices.

Acknowledgment. The authors would like to thank Prof. Yinyu Ye of University of Iowa for helpful discussions on this paper during his visit to Tokyo Institute of Technology in 2001.

References

[1] K. Fujisawa, M. Kojima, K. Nakata, M. Yamashita (1995) SDPA (SemiDefinite Programming Algorithm) user's manual — Version 6.0, Research Report B-308, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh-Okayama, Meguro, Tokyo 152-8552, Japan. Revised July 2002.

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

[3] S. Kim and M. Kojima (2001) Second order cone programming relaxations of quadratic optimization problems, *Optimization Methods and Software*, **15**, 201-224.

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

[5] M. Muramatsu (2001) *A New Second Order Cone Programming Relaxation for Max-cut Problems*, Department of Computer Science, Technical Report CS-01-01, The University of Electro-Communication, Tokyo, Japan.

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

[7] Yu. E. Nesterov (1998) Global quadratic optimization via conic optimization, Working paper, CORE, Université Catholique de Louvain, Louvain-la-Neuve, Belgium, November.

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

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

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

[11] S. Zhang (2000) Quadratic optimization and semidefinite relaxation, *Mathematical Programming* **87** 453-465.