A Slightly Lifted Convex Relaxation for Nonconvex Quadratic Programming with Ball Constraints

Samuel Burer*

February 28, 2023

Abstract

Globally optimizing a nonconvex quadratic over the intersection of m balls in $\mathbb{R}^n$ is known to be polynomial-time solvable for fixed m. Moreover, when $m = 1$, the standard semidefinite relaxation is exact, and when $m = 2$, it has recently been shown that an exact relaxation can be constructed via a disjunctive semidefinite formulation based on essentially two copies of the $m = 1$ case. However, there is no known explicit, tractable, exact convex representation for $m \geq 3$. In this paper, we construct a new, polynomially sized semidefinite relaxation for all m, and we demonstrate empirically that it is quite strong compared to existing relaxations, although still not exact for all objectives. The key idea is a simple lifting of the original problem into dimension $n+1$. Related to this construction, we also show that nonconvex quadratic programming over $\|x\| \leq \min\{1, g + h^T x\}$, which arises for example as a substructure in the alternating current optimal power flow problem, has an exact semidefinite representation.

1 Introduction

We study the nonconvex optimization problem

$$\min_{x \in \mathbb{R}^n} \left\{ x^T Q x + 2\, q^T x : \ \|x - c_i\| \leq \rho_i \quad \forall\, i = 1, \ldots, m \ \right\}, \tag{QP}$$

where the data are the $n \times n$ symmetric matrix Q, column vectors $q, c_1, \ldots, c_m \in \mathbb{R}^n$, and positive scalars $\rho_1, \ldots, \rho_m \in \mathbb{R}$. In words, (QP) is nonconvex quadratic programming over the intersection of m balls in n-dimensional space. Without loss of generality, we assume $c_1 = 0$ and $\rho_1 = 1$, i.e., the first constraint is the unit ball. Note that, while the feasible set of (QP) is convex, the objective function is generally nonconvex since Q may not be positive

*Department of Business Analytics, University of Iowa, Iowa City, IA, 52242–1994, USA. Email: samuel-burer@uiowa.edu.

semidefinite. We assume that (QP) is feasible and hence has an optimal solution, and we are specifically interested in strong convex relaxations of (QP).

Although (QP) is polynomial-time solvable to within ϵ-accuracy for fixed m [6], there is no known exact convex relaxation for all m. By *exact*, we mean a relaxation with optimal value equal to that of (QP). When $m = 1$, the standard semidefinite (SDP) relaxation is exact [21]. This relaxation is often called the *Shor relaxation* and involves a single $(n+1) \times (n+1)$ positive semidefinite variable. For $m = 2$, Kelly et al. [18] have recently shown that a particular disjunctive semidefinite relaxation is exact. Their construction is based on essentially two copies of the $m = 1$ case, and as a result, it utilizes two $(n+1) \times (n+1)$ positive semidefinite variables. These and other SDP relaxations will be detailed in Section 2. Our goals in this paper are: (i) to construct a new, stronger, non-disjunctive SDP relaxation, which is polynomially sized in n and m; and (ii) to demonstrate its effectiveness empirically. In particular, by *non-disjunctive*, we mean that the SDP optimizes over just one semidefinite variable like the Shor relaxation but with additional constraints on that matrix variable.

Another line of research examines conditions on the data of (QP) under which the Shor relaxation is exact. Note that, without loss of generality, by an orthogonal rotation in the x space, Q may be assumed to be diagonal. In this case, (QP) is a special case of a *diagonal quadratically constrained quadratic program (diagonal QCQP)*, that is, a QCQP in which every quadratic function has a diagonal Hessian. Diagonal QCQPs are well studied in the literature [12, 13, 19, 25, 4], where it is known, for example, that the Shor relaxation of (QP) is exact if $\text{sign}(q_j) = -\text{sign}(c_{1j}) = \cdots = -\text{sign}(c_{mj})$ for all $j = 1, \ldots, n$ [23]. In this paper, however, we seek a relaxation that is strong irrespective of the data.

(QP) can also be solved globally using any of the high-quality global-optimization algorithms and software packages available today. We are also aware of several papers studying global approaches for (QP) that take into account the problem's specific structure [7, 5, 1]. Each of these papers uses some combination of enumeration and lower bounding to find a global optimal solution and verify its optimality. In contrast, we are interested in computing a single strong bound, and we will show that our relaxation is frequently strong enough to deliver a global optimal solution via a rank-1 SDP optimal solution. Moreover, in future research, our relaxation could certainly be used as the lower-bounding technique within an enumerative scheme to solve (QP) globally, but we leave this for future research.

As we began this project, we were motivated by the idea of constructing an exact relaxation for (QP). Given that (QP) is only known to be polynomial-time for fixed m—not as a function of m—it was unclear whether a given relaxation, which is polynomial in m, could possibly be exact for all m. Nevertheless, we still had hope for the case $m = 2$. Ultimately, we will show by experimentation that our construction is not exact for $m = 2$ and hence for all m; see Section 4. Even still, we believe that our new relaxation makes significant progress towards approximating (QP) as we demonstrate empirically.

Our approach is based on two simple transformations of the feasible set of (QP). First, we write the feasible set in the equivalent form

$$\left\{ x \in \mathbb{R}^n : \|x\| \leq \sqrt{\rho_i^2 - c_i^T c_i + 2c_i^T x} \quad \forall\, i = 1, \ldots, m \right\}.$$

Note that the i-th constraint here is equivalent to the rotated second-order cone condition $x^T x \leq \rho_i^2 - c_i^T c_i + 2c_i^T x$. Second, we introduce an auxiliary variable $\beta \in \mathbb{R}$, which is inserted within each constraint:

$$\left\{ \begin{pmatrix} x \\ \beta \end{pmatrix} \in \mathbb{R}^{n+1} : \begin{array}{l} \|x\| \leq \beta \\ \beta \leq \sqrt{\rho_i^2 - c_i^T c_i + 2c_i^T x} \quad \forall\, i = 1, \ldots, m \end{array} \right\}.$$

Here, $\beta \leq \sqrt{\rho_i^2 - c_i^T c_i + 2c_i^T x}$ can be modeled using a rotated second-order cone of size 3. We can then equivalently minimize $x^T Q x + 2q^T x$ over $\binom{x}{\beta} \in \mathbb{R}^{n+1}$ in this new feasible set. The intuition for these transformations comes from the fact that we are exchanging $m-1$ cone constraints in the original problem, each of size $n+1$, with m cone constraints of size 3. In this sense, we have greatly simplified the structure of the feasible set.

The paper is organized as follows. In Section 2, we introduce the required background necessary to build and evaluate SDP relaxations of (QP). We also discuss the literature on relaxations specifically for (QP), including the exact relaxations mentioned above when $m = 1$ and $m = 2$. Then, in Section 3, we take a slight detour to study the related problem of nonconvex quadratic programming over

$$\left\{ x \in \mathbb{R}^n : \begin{array}{l} \|x\| \leq g_1 + h_1^T x \\ \|x\| \leq g_2 + h_2^T x \end{array} \right\},$$

i.e., when the functions bounding $\|x\|$ are linear. We continue to assume the first constraint is the unit ball, i.e., $g_1 = 1$ and $h_1 = 0$. This is an interesting case in its own right, arising for example as a substructure in the optimal power flow problem [14, 16] and also studied by Kelly et al. [18]. By examining this case, we build intuition for the case (QP), which we then consider fully in Section 4. Both Sections 3–4 contain empirical results demonstrating the strength of our relaxations. Finally, in Section 5, we return to the linear case of Section 3 and prove that our relaxation is exact, thus providing theoretical justification for the techniques and relaxations introduced in this paper.

We briefly mention a problem closely related to (QP), specifically nonconvex quadratic programming over the intersection of m general ellipsoids. When $m = 2$, this problem is known as the *Celis-Dennis-Tapia problem* or the *two trust-region subproblem (TTRS)*. See [15] and references therein for recent work on (TTRS). As with balls, many researchers have studied ways to build tight semidefinite relaxations for (TTRS), but an exact relaxation is unknown, even for the case of concentric ellipsoids. We also tried to apply the ideas of this paper to (TTRS) by introducing multiple artificial variables $\beta, \gamma \in \mathbb{R}$ within the trust-region constraints, say, $\|x\| \le \beta \le 1$ and $\|Hx\| \le \gamma \le 1$ when the ellipsoids are concentric. However, the resulting relaxations were no tighter empirically than those already in the literature, and we were unable to make further progress specifically on (TTRS). We hypothesize that the shared geometry of balls, as represented by the reappearance of the norm $\|x\|$, is critical for the approach of this paper.

We also remark that all computational results in the paper were coded in Python using the Fusion API of MOSEK 10.0.37 [3] and run on a M2 MacBook Air with 24 GB of RAM.

1.1 Notation and terminology

Our notation and terminology is mostly standard. $\mathbb{R}^d$ is the space of d-dimensional real column vectors, and $\mathbb{S}^d$ is the space of $d \times d$ real symmetric matrices. The identity matrix in $\mathbb{S}^d$ is denoted I_d, and the trace inner produce on $\mathbb{S}^d$ is defined as $M \bullet N := \operatorname{trace}(MN)$ for any two matrices $M, N \in \mathbb{S}^d$. Define

$$\begin{aligned} \mathcal{SOC}^d &:= \left\{ v \in \mathbb{R}^d : \|(v_2, \ldots, v_d)\| \le v_1 \right\}, \\ \mathcal{PSD}^d &:= \left\{ M \in \mathbb{S}^d : M \text{ is positive semidefinite} \right\} \end{aligned}$$

to be the d-dimensional second-order cone and the $d \times d$ positive semidefinite cone, respectively. We use *SOC* and *PSD* as abbreviations for *second-order cone* and *positive semidefinite*. For any cone S, its *conic hull* is defined as all finite sums of members in S, i.e., $\operatorname{conic.hull}(K) := \{\sum_{k=1}^{K} s_k : K \in \mathbb{N}, s_k \in S\}$.

2 Background on Semidefinite Relaxations

In this section, we recount techniques and constructions from the literature, which we will use for building convex relaxations in Sections 3–4. We also discuss prior research for relaxations of (QP) specifically. Finally, we discuss standard ways to measure the quality of a given relaxation on a particular problem instance, which will be employed in Sections 3–4.

We caution the reader that we will reuse (or "overload") some notation in this section.

For example, the variable x and dimension n in this section are not strictly speaking the same as defined in the Introduction.

2.1 Techniques for building an SDP relaxation

For this subsection as well as Section 2.3, we introduce the generic nonconvex quadratic programming problem

$$\min\left\{x^T Q x : x \in \mathcal{F},\ x_1 = 1\right\}, \tag{1}$$

where

$$\mathcal{F} := \left\{ x \in \mathbb{R}^n : \begin{array}{l} \ell_1^T x \geq 0,\ \ell_2^T x \geq 0 \\ L_3^T x \in \mathcal{SOC}^{d_3},\ L_4^T x \in \mathcal{SOC}^{d_4} \end{array} \right\}$$

is a closed, convex cone with polyhedral and second-order-cone constraints defined by the data vectors $\ell_1, \ell_2 \in \mathbb{R}^n$ and data matrices $L_3 \in \mathbb{R}^{n\times d_3}, L_4 \in \mathbb{R}^{n\times d_4}$. Specifially, $\mathcal{F}$ is defined by two linear constraints and two second-order cone (SOC) constraints of different sizes. We assume that $\mathcal{F}$ is nonempty and bounded in which case (1) has an optimal solution. We also assume that the constraints of $\mathcal{F}$ imply $x_1 \geq 0$.

It is well known that (1) is equivalent to

$$\min\left\{Q \bullet X : X \in \mathcal{G},\ X_{11} = 1\right\}, \tag{2}$$

where

$$\mathcal{G} := \text{conic. hull}\left\{X = xx^T : x \in \mathcal{F}\right\} \subseteq \mathbb{S}^n.$$

Equivalent means that both have the same optimal value and there exists a rank-1 optimal solution $X^* = x^*(x^*)^T$ of (2), where x^* is optimal for (1). Hence, a common approach to solve or approximate (1) is to build strong convex—typically semidefinite—relaxations for $\mathcal{G}$.

One standard option is the *Shor relaxation* of $\mathcal{G}$:

$$\textsc{Shor} := \left\{ X \in \mathcal{PSD}^n : \begin{array}{l} \ell_1^T X e_1 \geq 0,\ \ell_2^T X e_1 \geq 0 \\ J_{d_3} \bullet L_3^T X L_3 \geq 0 \\ J_{d_4} \bullet L_4^T X L_4 \geq 0 \end{array} \right\},$$

where $e_1 \in \mathbb{R}^n$ is the first unit vector and, for any dimension d,

$$J_d := \begin{pmatrix} 1 & 0 \\ 0 & -I_{d-1} \end{pmatrix} \in \mathbb{S}^d.$$

Here, $\ell_i^T X e_1 \geq 0$ reflects the constraint $\ell_i^T x \geq 0$ of (1) and is derived from the implication

$$\ell_i^T x,\ e_1^T x \geq 0 \quad \Longrightarrow \quad \ell_i^T x x^T e_1 \geq 0 \quad \Longrightarrow \quad \ell_i^T X e_1 \geq 0.$$

Morevover, the linear constraint $J_{d_k} \bullet L_k^T X L_k \geq 0$ is derived from the quadratic function defining $L_k^T x \in \mathcal{SOC}^{d_k}$. The Shor relaxation has two important properties. First, one can show that $Xe_1 \in \mathcal{F}$, so that a feasible solution is always embedded in X. Second, when the objective matrix Q is positive semidefinite, Shor is already strong enough to solve the quadratic problem, i.e., the optimal value of (2) equals the optimal value of (1), and the embedded solution is optimal.

Next, we introduce an *RLT constraint* [22] using the implication

$$\ell_1^T x \geq 0,\ \ell_2^T x \geq 0 \quad \Longrightarrow \quad \ell_1^T x x^T \ell_2 \geq 0 \quad \Longrightarrow \quad \ell_1^T X \ell_2 \geq 0,$$

and *SOCRLT constraints* [24, 9] using the implication

$$\ell_i^T x \geq 0,\ L_k^T x \in \mathcal{SOC}^{d_k} \quad \Longrightarrow \quad L_k^T x x^T \ell_i \in \mathcal{SOC}^{d_k} \quad \Longrightarrow \quad L_k^T X \ell_i \in \mathcal{SOC}^{d_k}.$$

Then defining

$$\begin{aligned} \mathrm{RLT} &:= \left\{ X \in \mathbb{S}^n : \ell_1^T X \ell_2 \geq 0 \right\}, \\ \mathrm{SOCRLT} &:= \left\{ X \in \mathbb{S}^n : L_k^T X \ell_i \in \mathcal{SOC}^{d_k} \ \ \forall\ i = 1,2,\ k = 3,4 \right\}, \end{aligned}$$

we arrive at the strengthened relaxation

$$\textsc{Shor} \cap \mathrm{RLT} \cap \mathrm{SOCRLT}.$$

Another valid constraint can be derived from the fact that $L_k^T x \in \mathcal{SOC}^{d_k}$ is equivalent to a positive semidefinite constraint (or *linear matrix inequality*) [2]. To explain, we first write $y := L_3^T x$ and $z := L_4^T x$, so that the SOC constraints in (1) are $y \in \mathcal{SOC}^{d_3}$ and $z \in \mathcal{SOC}^{d_4}$. Then it is well-known that

$$y \in \mathcal{SOC}^{d_3} \quad \Longleftrightarrow \quad \mathrm{Arr}_{d_3}(y) := \begin{pmatrix} y_1 & y_2 & \cdots & y_{d_3} \\ y_2 & y_1 & & \\ \vdots & & \ddots & \\ y_{d_3} & & & y_1 \end{pmatrix} \in \mathcal{PSD}^{d_3}.$$

For any dimension d, $\mathrm{Arr}_d : \mathbb{R}^d \to \mathbb{S}^d$ is called the *arrow operator*. Likewise $z \in \mathcal{SOC}^{d_4}$

if and only if $\mathrm{Arr}_{d_4}(z) \in \mathcal{PSD}^{d_4}$. Then, using the fact that the Kronecker product of PSD matrices is PSD, we have $\mathrm{Arr}_{d_3}(y) \otimes \mathrm{Arr}_{d_4}(z) \in \mathcal{PSD}^{d_3 d_4}$. Because the left-hand side of this expression is quadratic in y and z, we can equivalently write

$$\mathrm{Arr}_{d_3}(y) \otimes \mathrm{Arr}_{d_4}(z) = (\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4})(zy^T)$$

where $\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4} : \mathbb{R}^{d_4 \times d_3} \to \mathbb{S}^{d_3 d_4}$ is an operator that is linear in zy^T and defined by

$$(\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4})(zy^T) := \begin{pmatrix} \mathrm{Arr}_{d_3}(y_1 z) & \mathrm{Arr}_{d_3}(y_2 z) & \cdots & \mathrm{Arr}_{d_3}(y_{d_3} z) \\ \mathrm{Arr}_{d_3}(y_2 z) & \mathrm{Arr}_{d_3}(y_1 z) & & \\ \vdots & & \ddots & \\ \mathrm{Arr}_{d_3}(y_{d_3} z) & & & \mathrm{Arr}_{d_3}(y_1 z) \end{pmatrix}.$$

Substituting back $y = L_3^T x$ and $z = L_4^T x$, we arrive at the implication

$$(\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4})(L_4^T x x^T L_3) \in \mathcal{PSD}^{d_3 d_4} \quad \Longrightarrow \quad (\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4})(L_4^T X L_3) \in \mathcal{PSD}^{d_3 d_4}.$$

Then defining

$$\textsc{Kron} := \left\{ X \in \mathbb{S}^n : (\mathrm{Arr}_{d_3} \boxtimes \mathrm{Arr}_{d_4})(L_4^T X L_3) \in \mathcal{PSD}^{d_3 d_4} \right\},$$

we have the strongest relaxation thus far:

$$\textsc{Shor} \cap \mathrm{RLT} \cap \mathrm{SOCRLT} \cap \textsc{Kron}.$$

In Sections 3–4, we will derive SDP relaxations using the building blocks just introduced, i.e., Shor combined with RLT, SOCRLT, and Kron. In a sense, each of RLT, SOCRLT, and Kron have the same goal—to combine information from pairs of constraints in (1)—but they differ due to the structure of the underlying cones, i.e., the nonnegative orthant, the second-order cone, and the PSD cone. Similar ideas can be derived for other cones, e.g., the rotated second-order cone, or for other linear operators, whose image is constrained to be positive semidefinite. Such variations will be introduced in Section 4.

2.2 Results from the literature

The relaxations Shor, RLT, SOCRLT, and Kron just introduced are known to be quite strong in a number of contexts closely related to (QP). Specifically, Shor is exact for the case of (QP) for $m = 1$ [21], and when a single linear constraint is added to the ball constraint

$\|x\| \le 1$, then $\textsc{Shor} \cap \text{SOCRLT}$ is exact [24]. In addition, for the more general case

$$\left\{ x : \begin{array}{l} \|x\| \le 1 \\ 0 \le g_i + h_i^T x \ \ \forall\, i = 2, \ldots, m \end{array} \right\}$$

in which multiple linear constraints are added, $\textsc{Shor} \cap \text{RLT} \cap \text{SOCRLT}$ is exact as long as none of the hyperplanes $0 = g_i + h_i^T x$ intersect inside the ball [11]. $\textsc{Kron}$ has also been used to strengthen relaxations of (QP) when there is a second ellipsoidal constraint [2], not necessarily a ball. As a footnote, we are unaware of any case in which enforcing $\textsc{Kron}$ specifically makes a relaxation exact for all objectives, but $\textsc{Kron}$ has proven to be a valuable tool for building strong relaxations.

We are aware of two papers [17, 16], which have studied techniques for further strengthening $\textsc{Shor} \cap \text{RLT} \cap \text{SOCRLT} \cap \textsc{Kron}$. Among these, [16] is more similar to the current paper. In [16], the authors introduce a class of linear cuts for SDP relaxations of

$$\{x : \|x\| \le 1,\ \|x - c_2\| \le g_2 + h_2^T x\},$$

and they show how to separate the cuts in polynomial time. Through a series of computational experiments, the authors also show that their cuts are effective particularly in lower dimensions, say, for $n \le 10$.

As mentioned in the Introduction, an exact convex relaxation of (QP) via a disjunctive formulation was recently given by Kelly et al. [18]. They showed that the feasible region of (QP) can be written as the union of two special sets:

$$\{x : \|x\| \le 1,\ 0 \le g_1 + h_1^T x\} \qquad \text{and} \qquad \{x : \|x - c_2\| \le \rho_2,\ 0 \le g_2 + h_2^T x\}.$$

Explicit formulas for g_i and h_i are given in their paper. Since optimizing over each of these can separately be accomplished with $\textsc{Shor} \cap \text{SOCRLT}$ as mentioned above, the authors then use a disjunctive formulation with two copies of $\textsc{Shor} \cap \text{SOCRLT}$ to derive an exact formulation of (QP); see proposition 5 in their paper. The authors also used similar ideas to derive an exact, disjunctive formulation for the case $\|x\| \le \min\{1, g_2 + h_2^T x\}$; we will derive an exact, *non-disjunctive* formulation for this case in Section 3.

Zhen et al. [27] have recently suggested another technique for combining information from two SOC constraints; see appendix B of [27]. We illustrate their idea using the two constraints $\|x\| \le 1$ and $\|x - c_2\| \le \rho_2$ of (QP). The fact that $\|uv^T\|_2 = \|u\|\|v\|$, where $\|\cdot\|_2$

denotes the matrix 2-norm, implies

$$\|xx^T - xc^T\|_2 = \|x(x-c)^T\|_2 = \|x\|\|x - c_2\| \le 1 \cdot \rho_2 = \rho_2,$$

which can be linearized

$$\|X - xc^T\|_2 \le \rho_2 \qquad \Longleftrightarrow \qquad \begin{pmatrix} \rho_2 I_n & X - xc^T \\ X - cx^T & I_n \end{pmatrix} \in \mathcal{PSD}^{2n}.$$

In our experiments in Sections 3–4, we added this constraint to Shor ∩ RLT ∩ SOCRLT ∩ Kron, but it did not provide added strength on our test instances. Although we do not consider this valid constraint further in this paper, investigating its precise relationship with existing constraints remains an interesting avenue for research.

2.3 Measuring the quality of a relaxation

Following the notation of Section 2.1, let $\mathcal{R}$ be a given semidefinite relaxation of $\mathcal{G}$, which is at least as strong as Shor, i.e., $\mathcal{R} \subseteq$ Shor. The SDP relaxation corresponding to $\mathcal{R}$ is

$$r^* := \min\left\{Q \bullet X : X \in \mathcal{R},\ X_{11} = 1\right\},$$

and we let X^* denote an optimal solution.

We can assess the quality of the relaxation by comparing r^* to any readily available feasible value $v = x^TQx$, where $x \in \mathcal{F}$ is some feasible point. In particular, as mentioned in Section 2.1, X^* has an embedded feasible solution, but there are often multiple methods for obtaining a good feasible value v in practice, e.g., using a rounding procedure from X^* or some other type of heuristic. Our primary measure of relaxation quality will be the *relative gap* between r^* and v defined as follows:

$$\text{relative gap} := \frac{v - r^*}{\max\{1, \frac{1}{2}|v + r^*|\}}. \tag{3}$$

A secondary measure of relaxation quality is the so-called *eigenvalue ratio* of X^*. By construction, if the rank of X^* is 1, then $X^* = x^*(x^*)^T$ for some optimal solution x^* of (1). Of course, in practice X^* will most likely not be exactly rank-1, but it may be numerically close to rank-1. The eigenvalue ratio tries to capture how close X^* is to being rank-1. Specifically,

$$\text{eigenvalue ratio} := \frac{\lambda_1[X^*]}{\lambda_2[X^*]},$$

where $\lambda_1[X^*]$ and $\lambda_2[X^*]$ are the largest and second-largest eigenvalues of X^*. Generally speaking, the higher the eigenvalue ratio, the closer X^* is to being rank-1.

In the computational results of Sections 3–4, we will say that an instance is *solved* by a relaxation if both of the following two conditions are satisfied:

- the relative gap between r^* and the feasible value v, which comes from the solution x embedded in X^*, is less than 10^{-4};
- the eigenvalue ratio of X^* is greater than 10^4.

Similar definitions of the term *solved* have been used in [9, 1, 16, 15]. Strictly speaking, a small relative gap is enough to verify approximate optimality, but we will also require a large eigenvalue ratio in order to bolster our confidence in the numerical results. It should also be noted that, for randomly generated instances such as those investigated in Sections 3–4, small relative gaps and large eigenvalue ratios are typically highly positively correlated.

3 The Case $\|x\| \le \min\{1, g_2 + h_2^T x\}$

In this section, we examine the case of nonconvex quadratic programming over the set

$$\left\{x \in \mathbb{R}^n : \|x\| \le g_i + h_i^T x \ \ \forall\, i = 1, 2\right\} \tag{4}$$

with $g_1 = 1$ and $h_1 = 0$ such that the first constraint is $\|x\| \le 1$. We assume (4) is nonempty, but otherwise no particular assumptions are made on g_2 and h_2. We have two goals in this section. First, we wish to develop intuition for building a strong relaxation for (QP) in Section 4. Second, this case is interesting in its own right as a substructure of the optimal power flow problem [14, 16]. Moreover, Kelly et al. [18] have also recently studied this case; see Section 2.2.

3.1 Semidefinite relaxations

As discussed in Section 2, there are several ways to build relaxations of nonconvex QPs over the set (4). We first homogenize (4) using an auxiliary variable $\alpha \in \mathbb{R}$:

$$\widetilde{\mathcal{F}} := \left\{ \begin{pmatrix} \alpha \\ x \end{pmatrix} \in \mathbb{R}^{n+1} : \ \|x\| \le g_i \alpha + h_i^T x \ \ \forall\, i = 1, 2 \ \right\},$$

Note that $\alpha \geq 0$ is implied by the first constraint and that (4) is recovered by projecting $\widetilde{\mathcal{F}} \cap \{\binom{\alpha}{x} : \alpha = 1\}$ back to $\mathbb{R}^n$. Then setting

$$\widetilde{w} := \begin{pmatrix} \alpha \\ x \end{pmatrix} \in \mathbb{R}^{n+1}, \tag{5}$$

we investigate relaxations of $\widetilde{\mathcal{G}} := \text{conic.hull}\{\widetilde{w}\widetilde{w}^T : \widetilde{w} \in \widetilde{\mathcal{F}}\}$. Defining

$$\widetilde{L}_i := \begin{pmatrix} g_i & 0 \\ h_i & I_n \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)} \quad \forall\, i = 1, 2,$$

we rewrite

$$\widetilde{\mathcal{F}} = \left\{ \widetilde{w} \in \mathbb{R}^{n+1} : \widetilde{L}_i^T w \in \mathcal{SOC}^{n+1} \quad \forall\, i = 1, 2 \right\}.$$

Then, in accordance with Section 2, we define the following relaxation of $\widetilde{\mathcal{G}}$:

$$\begin{aligned} \widetilde{\mathcal{R}} &:= \textsc{Shor} \cap \textsc{Kron} \\ &:= \left\{ \widetilde{W} \in \mathcal{PSD}^{n+1} : \begin{array}{l} J_{n+1} \bullet \widetilde{L}_i^T \widetilde{W} \widetilde{L}_i \geq 0 \quad \forall\, i = 1, 2 \\ (\text{Arr}_{n+1} \boxtimes \text{Arr}_{n+1})(\widetilde{L}_2^T \widetilde{W} \widetilde{L}_1) \in \mathcal{PSD}^{(n+1)^2} \end{array} \right\}. \end{aligned}$$

Note that RLT and SOCRLT are not applicable here because $\widetilde{\mathcal{F}}$ contains no explicit linear constraints.

In hopes of improving upon $\widetilde{\mathcal{R}}$, as discussed in the Introduction, we introduce the auxiliary variable $\beta \in \mathbb{R}$ into (4):

$$\left\{ \begin{pmatrix} x \\ \beta \end{pmatrix} \in \mathbb{R}^{n+1} : \begin{array}{l} \|x\| \leq \beta \\ \beta \leq g_i + h_i^T x \quad \forall\, i = 1, 2 \end{array} \right\}. \tag{6}$$

This swaps one SOC constraint in (4) for two linear constraints, hence simplifying the structure of the feasible set, but it does not affect optimization over (4) since β is an artificial variable not impacting the objective $x^T Q x + 2q^T x$. We then proceed as above by homogenizing:

$$\mathcal{F} := \left\{ w \in \mathbb{R}^{n+2} : \begin{array}{l} \|x\| \leq \beta \\ \beta \leq g_i \alpha + h_i^T x \quad \forall\, i = 1, 2 \end{array} \right\},$$

where

$$w := \begin{pmatrix} \alpha \\ x \\ \beta \end{pmatrix} \in \mathbb{R}^{n+2}, \tag{7}$$

Note that the first linear constraint, with $g_1 = 1$ and $h_1 = 0$, ensures $\alpha \geq 0$.

We next define

$$P := \begin{pmatrix} 0 & 0 \\ 0 & I_n \\ 1 & 0 \end{pmatrix} \in \mathbb{R}^{(n+2)\times(n+1)} \tag{8}$$

and

$$\ell_i := \begin{pmatrix} g_i \\ h_i \\ -1 \end{pmatrix} \in \mathbb{R}^{n+2} \quad \forall\, i = 1, 2$$

so that

$$\mathcal{F} = \left\{ w \in \mathbb{R}^{n+2} : P^T w \in \mathcal{SOC}^{n+1},\ \ell_i^T w \geq 0\ \forall\, i = 1, 2 \right\}.$$

Then Section 2 provides the following relaxation of $\mathcal{G} := \text{conic.hull}\{ww^T \in \mathbb{S}^{n+1} : w \in \mathcal{F}\}$:

$$\begin{aligned} \mathcal{R} &:= \textsc{Shor} \cap \text{RLT} \cap \text{SOCRLT} \\ &:= \left\{ W \in \mathcal{PSD}^{n+2} : \begin{array}{l} J_{n+1} \bullet P^T W P \geq 0 \\ \ell_1^T W \ell_2 \geq 0 \\ P^T W \ell_i \in \mathcal{SOC}^{n+1} \quad \forall\, i = 1, 2 \end{array} \right\}. \end{aligned}$$

In this case, Kron is not relevant because $\mathcal{F}$ only contains one SOC constraint. Compared to $\widetilde{\mathcal{R}}$, the relaxation $\mathcal{R}$ has only one PSD constraint but contains one extra linear constraint and two SOC constraints.

3.2 Computational results, an exactness result, and a conjecture

From the previous subsection, to approximate the problem

$$\min \left\{ x^T Q x + 2q^T x : \|x\| \leq 1,\ \|x\| \leq g_2 + h_2^T x \right\}, \tag{9}$$

we have the semidefinite relaxations

$$\min \left\{ \widetilde{Q} \bullet \widetilde{W} : \widetilde{W} \in \widetilde{\mathcal{R}},\ \widetilde{W}_{11} = 1 \right\} \tag{Kron}$$

$$\min \left\{ \widehat{Q} \bullet W : W \in \mathcal{R},\ W_{11} = 1 \right\} \tag{Beta}$$

where

$$\widetilde{Q} := \begin{pmatrix} 0 & q^T \\ q & Q \end{pmatrix} \qquad \text{and} \qquad \widehat{Q} := \begin{pmatrix} 0 & q^T & 0 \\ q & Q & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{10}$$

n	m	# Instances	# Solved		Total Time (s)		
			KRON	BETA	SHOR	KRON	BETA
2	2	1,000	979	1,000	0.4	1.9	0.8
4	2	1,000	987	1,000	0.7	11.7	1.1
6	2	1,000	990	1,000	0.9	151.9	1.5

Table 1: Number of instances of (9) solved by KRON and BETA over 3,000 randomly generated instances, which were not already solved by SHOR. Instances are grouped by dimensions (n, m). Also shown are the total time (in seconds) to solve all instances for each method in each grouping. Times for SHOR are also shown for reference.

We use the name "KRON" to remind the reader that $\widetilde{\mathcal{R}}$ is equivalent to SHOR $\cap$ KRON and "BETA" to denote our relaxation, which is based on the artificial variable β.

We first provide a small example showing that BETA does indeed impove upon KRON. Recall from Section 2.3, that we say a relaxation *solves* an instance if both the relative gap is less than 10^{-4} and the eigenvalue ratio is more than 10^4.

Example 1. *Consider an instance of (9) with $n = 2$ and*

$$Q = \begin{pmatrix} -0.67 & 0.95 \\ 0.95 & -1.59 \end{pmatrix}, \quad q = \begin{pmatrix} -0.89 \\ -0.89 \end{pmatrix}, \quad g_2 = 1.52, \quad h_2 = \begin{pmatrix} 0.19 \\ -0.91 \end{pmatrix}.$$

KRON *returns the lower bound -2.6363, the method of [16] returns -2.5543, and* BETA *solves the instance with an optimal value of -2.4672 and optimal solution*

$$x^* \approx \begin{pmatrix} 0.978358 \\ -0.206920 \end{pmatrix}.$$

Next we generated 1,000 feasible instances of (9) for each of the dimensions $n \in \{2, 4, 6\}$, and we specifically did *not* include instances that were already solved by SHOR in this test set. For example, this guaranteed that no test instance was convex due to Q being positive semidefinite. Note also that the number of constraints here is always $m = 2$. In particular, to generate each instance, we first sampled a random point x^0 in the unit ball, and then generated h_2 with entries i.i.d. according to the standard normal distribution $N(0, 1)$. Then g_2 was generated according to the formula $U(0,1) + h_2^T x^0 - \|x^0\|$, where $U(0, 1)$ is the uniform distribution between 0 and 1, so that x^0 was interior feasible. Then Q and q were generated with all entries i.i.d. in $N(0, 1)$. Finally, SHOR was solved, and if SHOR did *not* solve the instance, it was included in our test set.

Table 1 shows the results of our experiments. We see clearly that, while KRON solves many instances, BETA solves all instances in much less time. In fact, BETA takes about the same time as SHOR.

Upon further examination of the details of the results depicted in Table 1, we noticed that the RLT constraint $\ell_1^T W \ell_2 \geq 0$ in $\mathcal{R}$ was active at all optimal solutions. This led us to investigate the condition $\ell_1^T W \ell_2 = 0$ more carefully. Recall that β was inserted between $\|x\|$ and $\min\{1, g_2 + h_2^T x\}$. When $\|x\|$ is strictly less than $\min\{1, g_2 + h_2^T x\}$, there are multiple values of β, which are feasible for the same value of x. To remove this ambiguity, we can actually force $\beta = \min\{1, g_2 + h_2^T x\}$ by adding the quadratic complementarity equation

$$(1 - \beta)(g_2 + h_2^T x - \beta) = (\ell_1^T w)(\ell_2^T w) = 0$$

to the definition of $\mathcal{F}$. Formally, we define

$$\begin{aligned}
\mathcal{F}^0 &:= \left\{ w \in \mathcal{F} : (\ell_1^T w)(\ell_2^T w) = 0 \right\}, \\
\mathcal{G}^0 &:= \text{conic. hull}\{W = ww^T : w \in \mathcal{F}^0\}, \\
\mathcal{R}^0 &:= \{W \in \mathcal{R} : \ell_1^T W \ell_2 = 0\}.
\end{aligned}$$

Note that optimizing $x^T Q x + 2q^T x$ over $\mathcal{F}^0 \cap \{x : \alpha = 1\}$ is equivalent to (9) because β is artificial. In Section 5, we will prove the following theorem:

Theorem 1. $\mathcal{G}^0 = \mathcal{R}^0$.

This ensures that (9) is equivalent to

$$\min \left\{ \widehat{Q} \bullet W : W \in \mathcal{R},\ W_{11} = 1,\ \ell_1^T W \ell_2 = 0 \right\} = \min \left\{ \widehat{Q} \bullet W : W \in \mathcal{R}^0,\ W_{11} = 1 \right\},$$

that is, equivalent to BETA with the strengthened constraint $\ell_1^T W \ell_2 = 0$. Hence, the fact that $\ell_1^T W \ell_2 \geq 0$ is active at optimality for all instances in Table 1 is actually another verification that BETA solves these instances.

This begs the question: why should the constraint $\ell_1^T W \ell_2 \geq 0$ be active at optimality for the instances tested? Indeed, after extensive experimentation, this appears to always be the case, and accordingly, we offer the following conjecture:

Conjecture 1. *There exists an optimal solution W^* of* BETA *with* $\ell_1^T W^* \ell_2 = 0$.

If this conjecture is true, then it must rely on the fact that the the variable β does not appear in the objective of (9). Indeed, in Section 5, we will give an example inside the proof of Proposition 1 showing that, when β is involved in the objective, $\ell_1^T W \ell_2 = 0$ may not hold at optimality.

4 The Case of (QP)

Following the path laid in Sections 2–3, we now return to our primary case of interest, problem (QP). We first construct two semidefinite relaxations, one which already exists in the literature and our new construction based on the variable β discussed in the Introduction. Then we conduct computational experiments showing the strength of our relaxation.

As in previous sections, we will overload some of our notation. For example, $\widetilde{\mathcal{R}}$ and $\mathcal{R}$ will be used similarly in this section as in Section 3, but their technical definitions will be different.

4.1 Semidefinite relaxations

We first homogenize the feasible set of (QP) by introducing $\alpha \in \mathbb{R}$:

$$\widetilde{\mathcal{F}} := \left\{ \begin{pmatrix} \alpha \\ x \end{pmatrix} \in \mathbb{R}^{n+1} : \ \|x - \alpha c_i\| \le g_i \alpha + h_i^T x \ \ \forall\, i = 1, \ldots, m \right\}, \tag{11}$$

Then, defining $\widetilde{w}$ by (5) and

$$\widetilde{L}_i := \begin{pmatrix} g_i & -c_i^T \\ h_i & I_n \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)} \ \ \forall\, i = 1, \ldots, m$$

we rewrite

$$\widetilde{\mathcal{F}} = \left\{ \widetilde{w} \in \mathbb{R}^{n+1} : \widetilde{L}_i^T w \in \mathcal{SOC}^{n+1} \quad \forall\, i = 1, \ldots, m \right\}.$$

Then, in accordance with Section 2, we define the following relaxation of $\widetilde{\mathcal{G}} := \text{conic.hull}\{\widetilde{w}\widetilde{w}^T \in \mathbb{S}^{n+1} : \widetilde{w} \in \widetilde{\mathcal{F}}\}$:

$$\begin{aligned} \widetilde{\mathcal{R}} &:= \textsc{Shor} \cap \textsc{Kron} \\ &:= \left\{ \widetilde{W} \in \mathcal{PSD}^{n+1} : \begin{array}{ll} J_{n+1} \bullet \widetilde{L}_i^T \widetilde{W} \widetilde{L}_i \ge 0 & \forall\, i = 1, \ldots, m \\ (\mathrm{Arr}_{n+1} \boxtimes \mathrm{Arr}_{n+1})(\widetilde{L}_k^T \widetilde{W} \widetilde{L}_i) \in \mathcal{PSD}^{(n+1)^2} & \forall\, 1 \le i < k \le m \end{array} \right\}, \end{aligned}$$

Note that RLT and SOCRLT are not applicable here because $\widetilde{\mathcal{F}}$ contains no explicit linear constraints.

In hopes of improving upon $\widetilde{\mathcal{R}}$, we introduce the auxiliary variable $\beta \in \mathbb{R}$ into (QP):

$$\left\{ \begin{pmatrix} x \\ \beta \end{pmatrix} \in \mathbb{R}^{n+1} : \begin{array}{l} \|x\| \le \beta, \ \beta \le 1 \\ \beta \le \sqrt{g_i + h_i^T x} \ \ \forall\, i = 2, \ldots, m \end{array} \right\},$$

where $g_i := \rho_i^2 - c_i^T c_i$ and $h_i := 2c_i$ for all $i = 2, \ldots, m$. This swaps $m-1$ SOC constraints in (QP) for one linear constraint and $m-1$ low-dimensional rotated SOC constraints, hence simplifying the structure of the feasible set. In particular, defining

$$\mathcal{RSOC}^3 := \{v \in \mathbb{R}^3 : v_1, v_2 \geq 0,\ v_3^2 \leq v_1 v_2\}$$

to be the 3-dimensional rotated SOC, we have

$$\beta \leq \sqrt{g_i + h_i^T x} \quad \iff \quad \begin{pmatrix} 1 \\ g_i + h_i^T x \\ \beta \end{pmatrix} \in \mathcal{RSOC}^3.$$

Then we homogenize:

$$\mathcal{F} := \left\{ w \in \mathbb{R}^{n+2} : \|x\| \leq \beta,\ \beta \leq \alpha,\ \begin{pmatrix} \alpha \\ g_i\alpha + h_i^T x \\ \beta \end{pmatrix} \in \mathcal{RSOC}^3 \quad \forall\, i = 2, \ldots, m \right\},$$

where w is defined by (7). Note that this feasible set ensures $\alpha \geq 0$. We next define P by (8) and

$$\ell_1 := \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \in \mathbb{R}^{n+2}, \quad L_i := \begin{pmatrix} 1 & g_i & 0 \\ 0 & h_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \in \mathbb{R}^{(n+2)\times 3} \ \ \forall\, i = 2, \ldots, m$$

so that we may rewrite

$$\mathcal{F} = \left\{ w \in \mathbb{R}^{n+2} : \begin{array}{l} P^T w \in \mathcal{SOC}^{n+1},\ \ell_1^T w \geq 0, \\ L_i^T w \in \mathcal{RSOC}^3 \ \forall\, i = 2, \ldots, m \end{array} \right\}$$

To build a relaxation of $\mathcal{G} := \operatorname{conic.hull}\left\{ww^T \in \mathbb{S}^{n+1} : w \in \mathcal{F}\right\}$, we define the matrix

$$K := \begin{pmatrix} 0 & 1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \in \mathbb{S}^3$$

and the linear operator

$$\operatorname{sMat}_2 : \mathbb{R}^3 \to \mathbb{S}^2 \qquad \text{by} \qquad \operatorname{sMat}_2(v) := \begin{pmatrix} v_1 & v_3 \\ v_3 & v_2 \end{pmatrix}$$

so that

$$v \in \mathcal{RSOC}^3 \quad \iff \quad v_1, v_2 \geq 0,\ v^T K v \geq 0 \quad \iff \quad \mathrm{sMat}_2(v) \in \mathcal{PSD}^2.$$

Then our relaxation is

$$\begin{aligned}
\mathcal{R} &:= \textsc{Shor} \cap \text{SOCRLT} \cap \textsc{Kron} \\
&:= \left\{ W \in \mathcal{PSD}^{n+2} : \begin{array}{ll} J_{n+1} \bullet P^T W P \geq 0 & \\ P^T W \ell_1 \in \mathcal{SOC}^{n+1} & \\ K \bullet L_i^T W L_i \geq 0 & \forall\, i = 2, \ldots, m \\ L_i^T W \ell_1 \in \mathcal{RSOC}^3 & \forall\, i = 2, \ldots, m \\ (\mathrm{Arr}_{n+1} \boxtimes \mathrm{sMat}_2)(L_i^T W P) \in \mathcal{PSD}^{2(n+1)} & \forall\, i = 2, \ldots, m \\ (\mathrm{sMat}_2 \boxtimes \mathrm{sMat}_2)(L_k^T W L_i) \in \mathcal{PSD}^4 & \forall\, 2 \leq i < k \leq m \end{array} \right\}.
\end{aligned}$$

Note that, compared to $\widetilde{\mathcal{R}}$, the relaxation $\mathcal{R}$ utilizes more, but smaller, cones. Specifically, $\widetilde{\mathcal{R}}$ has two PSD conditions, one of size roughly $n \times n$ and one of size $n^2 \times n^2$. In contrast, $\mathcal{R}$ has roughly m PSD conditions of size $n \times n$, m linear constraints, m 3-dimensional rotated SOC constraints, and m^2 PSD conditions of size 4×4.

4.2 Computational results for $m = 2$

From the previous subsection, to approximate (QP), we have the semidefinite relaxations

$$\min \left\{ \widetilde{Q} \bullet \widetilde{W} : \widetilde{W} \in \widetilde{\mathcal{R}},\ \widetilde{W}_{11} = 1 \right\} \tag{\textsc{Kron}}$$

$$\min \left\{ \widehat{Q} \bullet W : W \in \mathcal{R},\ W_{11} = 1 \right\} \tag{\textsc{Beta}}$$

where $\widetilde{Q}$ and $\widehat{Q}$ are given by (10). We first provide a small example showing that Beta can significantly impove Kron.

Example 2. *Consider an instance of (QP) with $(n, m) = (2, 2)$ and*

$$Q = \begin{pmatrix} -0.12 & 0.66 \\ 0.66 & -1.58 \end{pmatrix}, \quad q = \begin{pmatrix} 1.04 \\ 0.10 \end{pmatrix}, \quad c_2 = \begin{pmatrix} 0.09 \\ -0.34 \end{pmatrix}, \quad \rho_2 = 0.98.$$

Kron *returns the lower bound* -1.9206*, the method of [16] returns* -1.9076*, and* Beta *solves*

n	m	# Instances	# Solved		Total Time (s)		
			Kron	Beta	Shor	Kron	Beta
2	2	1,000	949	997	0.4	3.7	1.9
4	2	1,000	885	997	0.7	25.5	3.5
6	2	1,000	886	1,000	1.0	353.4	5.5

Table 2: Number of instances of (QP) solved by Kron and Beta over 3,000 randomly generated Martinez instances, which were not already solved by Shor. Instances are grouped by dimensions (n, m). Also shown are the total time (in seconds) to solve all instances for each method in each grouping. Times for Shor are also shown for reference.

the instance with an optimal value of -1.8856 *and optimal solution*

$$x^* \approx \begin{pmatrix} -0.303464 \\ -0.952843 \end{pmatrix}.$$

Next we generated 1,000 feasible instances of (QP) for each of the dimension pairs $(n, m) \in \{(2, 2), (4, 2), (8, 2)\}$, and we specifically excluded instances that were already solved by Shor. Note that, for this set of experiments, the number of constraints is kept constant at $m = 2$. In particular, to generate each instance, we followed the method of [9]; see section 5.3 of that paper and the discussion therein. The idea is to first generate and solve globally an instance of (QP) with $m = 1$ and then to add a second ball constraint, which cuts off the optimal solution just calculated. The resulting problem is likely to possess multiple good candidates for optimal solutions, thus ostensibly making it a more challenging instance of (QP). We call these the *Martinez instances* following the terminology introduced in [9].

Table 2 shows the results of our experiments. We see that, while Kron solves many instances, Beta solves nearly all instances in much less time. In fact, Beta takes just a bit more time than Shor.

We also examined the so-called *gap closure* for each relaxation. For a given instance, let s^* be the optimal value of Shor, and let v be the best (i.e., minimum) feasible value from the solutions embedded within the three relaxations Shor, Kron, and Beta at optimality. Then

$$\text{gap closure for Kron} := \left(\frac{[\text{optimal value for Kron}] - s^*}{v - s^*} \right) \times 100\%$$

$$\text{gap closure for Beta} := \left(\frac{[\text{optimal value for Beta}] - s^*}{v - s^*} \right) \times 100\%$$

A gap closure of 0% means that the relaxation did not improve upon Shor, and a gap closure of 100% means that the relaxation was exact.

Solution Status		# Instances	Avg Gap Closed	
Kron	Beta		Kron	Beta
unsolved	unsolved	6	57%	66%
unsolved	solved	274	87%	100%
solved	unsolved	0	-	-
solved	solved	2720	100%	100%

Table 3: Average gap closures for Table 2. Instances are grouped by solution status for Kron and Beta.

Table 3 shows that the average gap closures for Kron were around 10% less than for Beta on those instances unsolved by Kron. We also see that there were no instances that were solved by Kron and yet unsolved by Beta. This suggests empirically that Beta is always at least as strong as Kron, although we do not have a formal proof establishing this.

Eltved and Burer [16] also solved instances of (QP) with $m = 2$, and in their paper, they reported that their method, which is at least as strong as Kron, was unable to solve 267 randomly generated instances with n ranging from 2 to 10; see table 6 therein. We ran Beta on these instances and solved 262 of them. We also ran the method of [16] on the six instances in Table 2, which were unsolved by Beta, and their method generated worse lower bounds in all six cases.

4.3 Computational results for $m > 2$

Continuing the prevous subsection, we now investigate the use of Kron and Beta to approximate instances of (QP) with $m > 2$. To this end, we generated random instances of the following geometric problem, which we call *max-norm*:

> In $\mathbb{R}^n$, given a point p and m balls containing the origin, find a point in the intersection of the balls with maximum distance to p.

This is an instance of (QP) with $Q = -I$ and $q = p$. Specifically, the first ball is the unit ball, and then remaining $m - 1$ balls are generated with random centers c_i in the unit ball and random radii ρ_i, each in $\|c_i\| + U(0, 1.5)$, where $U(0, 1.5)$ is continuous uniform between 0 and 1.5. In particular, this guarantees that $x = 0$ is feasible. Finally, p is generated uniformly in the ball of radius 4 centered at the origin. As in previous experiments, we exclude instances that are solved by Shor.

Similar to Tables 2–3 in the previous subsection, Tables 4 and 5 show the results of Kron and Beta on 1,000 max-norm instances for four pairs of dimensions (n, m). These results show clearly that Beta significantly outperforms Kron in terms of both number of

n	m	# Instances	# Solved		Total Time (s)		
			Kron	Beta	Shor	Kron	Beta
2	5	1,000	91	951	0.3	6.3	7.3
2	9	1,000	103	948	0.4	11.6	28.3
4	9	1,000	0	861	0.7	115.0	61.8
4	17	1,000	1	830	0.9	432.5	632.7

Table 4: Number of max-norm instances solved by Kron and Beta on 4,000 instances unsolved by Shor. Instances are grouped by dimensions (n, m). Also included are the total time (in seconds) to solve all instances for each method in each grouping.

Solution Status		# Instances	Avg Gap Closed	
Kron	Beta		Kron	Beta
unsolved	unsolved	410	2%	30%
unsolved	solved	3395	11%	100%
solved	unsolved	0	-	-
solved	solved	195	100%	100%

Table 5: Average gap closures for Table 4. Instances are grouped by solution status for Kron and Beta.

instances solved and the average gap closed. In terms of timings, we see that Beta can take significantly longer than Kron for larger values of m.

5 Proof of Theorem 1

In this section, we prove Theorem 1 from Section 3, which states that

$$\mathcal{G}^0 := \text{conic. hull} \left\{ ww^T \in \mathbb{S}^{n+2} : \begin{array}{l} P^T w \in \mathcal{SOC}^{n+1} \\ \ell_i^T w \geq 0 \ \ \forall\, i = 1, 2 \\ (\ell_1^T w)(\ell_2^T w) = 0 \end{array} \right\}$$

equals

$$\mathcal{R}^0 := \left\{ W \in \mathcal{PSD}^{n+2} : \begin{array}{l} J_{n+1} \bullet P^T W P \geq 0 \\ P^T W \ell_i \in \mathcal{SOC}^{n+1} \ \ \forall\, i = 1, 2 \\ \ell_1^T W \ell_2 = 0 \end{array} \right\}.$$

A closely related result is theorem 2.7 in [26]. Theorem 1 is slightly different, however, because the second-order cone is expressed in a smaller dimension $(n + 1)$ than the ambient dimension $(n + 2)$. So we prove Theorem 1 rigorously here, and in particular, we adapt a proof technique from [8].

We first prove some lemmas. To simplify notation a bit, let $J := J_{n+1}$ throughout the rest of this section. Our first lemma states some straightforward properties of J.

Lemma 1. *Regarding J, it holds that:*

(i) If $v \in \mathcal{SOC}^{n+1}$, then $Jv \in \mathcal{SOC}^{n+1}$, and hence $w^T J v \geq 0$ for all $w \in \mathcal{SOC}^{n+1}$.

(ii) If $v \in \operatorname{bd}(\mathcal{SOC}^{n+1})$, then $v^T J v = 0$.

The next two lemmas are analagous to the exactness results of Shor and Shor$\cap$SOCRLT discussed in Section 2.2.

Lemma 2. $\operatorname{conic.hull}\{ww^T : P^T w \in \mathcal{SOC}^{n+1}\} = \{W \in \mathcal{PSD}^{n+2} : J \bullet P^T W P \geq 0\}$.

Proof. This can be proven, for example, by showing that all extreme rays of the right-hand-side set have rank-1, which can in turn be shown via Pataki's bound on the rank of extreme matrices of SDP-representable sets [20]. □

Lemma 3. *Fix $i = 1$ or $i = 2$. It holds that*

$$\operatorname{conic.hull}\left\{ ww^T : \begin{array}{l} P^T w \in \mathcal{SOC}^{n+1} \\ \ell_i^T w \geq 0 \end{array} \right\} = \left\{ W \in \mathcal{PSD}^{n+2} : \begin{array}{l} J \bullet P^T W P \geq 0 \\ P^T W \ell_i \in \mathcal{SOC}^{n+1} \end{array} \right\}.$$

Proof. In the statement of the proposition, let $\mathcal{G}[i]$ be the left-hand-side set, and let $\mathcal{R}[i]$ be the right-hand side. Clearly $\mathcal{G}[i] \subseteq \mathcal{R}[i]$. We show the reverse inclusion by proving $\mathcal{R}[i]$ has rank-1 extreme rays. Indeed, let $W \neq 0$ be an arbitrary extreme ray in $\mathcal{R}[i]$. We will show $\operatorname{rank}(W) = 1$ by considering three cases

First consider when $P^T W \ell_i \in \operatorname{int}(\mathcal{SOC}^{n+1})$, in which case W must also be extreme in $\{W \in \mathcal{PSD}^{n+2} : J \bullet P^T W P \geq 0\}$. Then $\operatorname{rank}(W) = 1$ by Lemma 2.

Next consider when $P^T W \ell_i \in \operatorname{bd}(\mathcal{SOC}^{n+1})$ with $W \ell_i = 0$. Using Lemma 2, we write $W = \sum_k w^k (w^k)^T$ for w^k satisfying $P^T w^k \in \mathcal{SOC}^{n+1}$. We then have

$$W\ell_i = 0 \quad \Rightarrow \quad \ell_i^T W \ell_i = 0 \quad \Rightarrow \quad \sum_k (\ell_i^T w^k)^2 = 0 \quad \Rightarrow \quad \ell_i^T w^k = 0 \;\; \forall\, k,$$

which proves $W \in \mathcal{G}[i]$ and hence W is extreme in $\mathcal{G}[i]$. Thus, $\operatorname{rank}(W) = 1$, as desired.

Finally, consider when $P^T W \ell_i \in \operatorname{bd}(\mathcal{SOC}^{n+1})$ with $W \ell_i \neq 0$. Define $v := W \ell_i \neq 0$ so that $P^T v \in \operatorname{bd}(\mathcal{SOC}^{n+1})$. In addition, $W \in \mathcal{PSD}^{n+2}$ implies $\ell_i^T v = \ell_i^T W \ell_i \geq 0$. Hence, we conclude that vv^T is a nonzero member of $\mathcal{G}[i] \subseteq \mathcal{R}[i]$. Next, for small $\epsilon > 0$, define $W_\epsilon := W - \epsilon\, vv^T$; we claim $W_\epsilon \in \mathcal{R}[i]$. Indeed, $W_\epsilon \in \mathcal{PSD}^{n+2}$ because it is a rank-1 perturbation of $W \in \mathcal{PSD}^{n+2}$ with $v \in \operatorname{Range}(W)$ [10, lemma 2]. Moreover, $P^T v \in \operatorname{bd}(\mathcal{SOC}^{n+1})$ implies by Lemma 1(ii) that

$$J \bullet P^T W_\epsilon P = J \bullet P^T W P - \epsilon (P^T v)^T J (P^T v) = J \bullet P^T W P - \epsilon \cdot 0 \geq 0.$$

We also have

$$P^T W_\epsilon \ell_i = P^T W \ell_i - \epsilon(\ell_i^T v) P^T v = (1 - \epsilon \cdot \ell_i^T v) P^T v \in \mathrm{bd}(\mathcal{SOC}^{n+1}).$$

Thus, when $\epsilon > 0$ is small, $W_\epsilon \in \mathcal{R}[i]$ as claimed. Then the equation $W = W_\epsilon + \epsilon\, vv^T$ and the fact that W is extreme in $\mathcal{R}[i]$ imply W must be a positive multiple of vv^T, i.e., $\mathrm{rank}(W) = 1$, as desired. □

Our next lemma is a technical result about extreme rays in the intersection of two closed convex cones.

Lemma 4. *Let $\mathcal{P}$ be a closed convex cone, and let $\mathcal{Q}$ be a half-space containing the origin. Every extreme ray of $\mathcal{P} \cap \mathcal{Q}$ is either an extreme ray of $\mathcal{P}$ or can be expressed as the sum of two extreme rays of $\mathcal{P}$.*

Proof. See [8, Lemma 5]. □

We are now ready to prove Theorem 1.

Proof. Since $\mathcal{G}^0 \subseteq \mathcal{R}^0$ by construction, we show the reverse inclusion by proving that every extreme ray W of $\mathcal{R}^0$ has rank 1 and hence is an element of $\mathcal{G}^0$. We define $v_i := W\ell_i$ for $i = 1, 2$. Note that $v_1 v_1^T \in \mathcal{G}^0$ because $\ell_1^T v_1 = \ell_1^T W \ell_1 \geq 0$, $\ell_2^T v_1 = \ell_2^T W \ell_1 = 0$, and $P^T v_1 = P^T W \ell_1 \in \mathcal{SOC}^{n+1}$. A similar argument shows $v_2 v_2^T \in \mathcal{G}^0$.

We first consider the case when $v_1 = 0$. Applying Lemma 3 for the case $i = 2$, we express W as

$$W = \sum_k w^k (w^k)^T, \quad P^T w^k \in \mathcal{SOC}^{n+1} \text{ and } \ell_2^T w^k \geq 0 \quad \forall\, k.$$

As in the proof of Lemma 3, $v_1 = 0$ then implies $\ell_1^T w^k = 0$ for all k. So $W \in \mathcal{G}^0$, and because W is extreme in $\mathcal{R}^0 \supseteq \mathcal{G}^0$, it must have rank 1. A similar argument shows $\mathrm{rank}(W) = 1$ for the case $v_2 = 0$.

So we assume from this point forward that $v_i := W\ell_i \neq 0$ for both $i = 1, 2$. Since $W \in \mathcal{PSD}^{n+2}$ ensures $W\ell_i = 0 \Leftrightarrow \ell_i^T W \ell_i = 0$, we have $\ell_i^T v_i = \ell_i^T W \ell_i > 0$ for both i.

The second case we consider assumes $J \bullet P^T W P = 0$ and $P^T v_i = P^T W \ell_i \in \mathrm{int}(\mathcal{SOC}^{n+1})$ for both $i = 1, 2$. Then W is extreme for the equality-constrained cone $\{W \in \mathcal{PSD}^{n+2} : J \bullet P^T W P = 0, \ell_1^T W \ell_2 = 0\}$, which in turn implies that W is extreme for the inequality-constrained cone $\mathcal{P} \cap \mathcal{Q}$, where

$$\mathcal{P} := \left\{W \in \mathcal{PSD}^{n+2} : J \bullet P^T W P \geq 0\right\}, \quad \mathcal{Q} := \{W \in \mathbb{S}^{n+2} : \ell_1^T W \ell_2 \geq 0\},$$

Applying Lemma 2 with $\mathcal{P}$ and Lemma 4 with $\mathcal{P} \cap \mathcal{Q}$, we conclude that $\text{rank}(W) \le 2$. If its rank equals 1, we are done. So assume $\text{rank}(Y) = 2$. We derive a contradiction to the assumption that W is extreme in $\mathcal{R}^0$. Consider the equation

$$U := \begin{pmatrix} \ell_1^T \\ \ell_2^T \\ I \end{pmatrix} W \begin{pmatrix} \ell_1 & \ell_2 & I \end{pmatrix} = \begin{pmatrix} \ell_1^T W \ell_1 & \ell_1^T W \ell_2 & \ell_1^T W \\ \ell_2^T W \ell_1 & \ell_2 W \ell_2 & \ell_2^T W \\ W \ell_1 & W \ell_2 & W \end{pmatrix} = \begin{pmatrix} \ell_1^T v_1 & 0 & v_1^T \\ 0 & \ell_2^T v_2 & v_2^T \\ v_1 & v_2 & W \end{pmatrix},$$

and recall that $\ell_1^T v_1 > 0$ and $\ell_2^T v_2 > 0$. It holds that U is PSD with $\text{rank}(U) \le \text{rank}(W) = 2$, and the Schur complement theorem implies

$$M := W - (\ell_1^T v_1)^{-1} v_1 v_1^T - (\ell_2^T v_2)^{-1} v_2 v_2^T \in \mathcal{PSD}^{n+2},$$

with $\text{rank}(M) = \text{rank}(W) - 2$. So $\text{rank}(M) \le 0$ and hence $M = 0$, that is,

$$W = (\ell_1^T v_1)^{-1} v_1 v_1^T + (\ell_2^T v_2)^{-1} v_2 v_2^T,$$

contradicting the assumption that W is extreme in $\mathcal{R}^0$ due to the fact that both $v_1 v_1^T$ and $v_2 v_2^T$ are elements of $\mathcal{G}^0 \subseteq \mathcal{R}^0$.

For our third and final case, we assume $J \bullet P^T W P > 0$, $P^T v_1 \in \text{bd}(\mathcal{SOC}^{n+1})$, or $P^T v_2 \in \text{bd}(\mathcal{SOC}^{n+1})$. Let us consider two perturbations of W:

$$W_{\epsilon_i} := W - \epsilon_i \, v_i v_i^T \quad \forall \; i = 1, 2$$

for two parameters $\epsilon_i > 0$. We claim that $W_{\epsilon_i} \in \mathcal{R}^0$ for at least one i, in which case W must be rank-1 as argued in the proof of Lemma 3.

Using $W \in \mathcal{R}^0$, we first argue that each W_{ϵ_i} satisfies all constraints of $\mathcal{R}^0$, except possibly $J \bullet P^T W_{\epsilon_i} P \ge 0$. Fix $i = 1$; the proof for $i = 2$ is similar. We know $W_{\epsilon_1} \in \mathcal{PSD}^{n+2}$ since $v_1 \in \text{Range}(W)$ [10, Lemma 2]. We also have

$$P^T W_{\epsilon_1} \ell_1 = P^T v_1 - \epsilon_1 P^T v_1 (v_1^T \ell_1) = (1 - \epsilon_1 (v_1^T \ell_1)) P^T v_1 \in \mathcal{SOC}^{n+1}$$

for small $\epsilon_1 > 0$. Furthermore, noting that $v_1^T \ell_2 = \ell_1^T W \ell_2 = 0$, we see

$$P^T W_{\epsilon_1} \ell_2 = P^T v_2 - \epsilon_1 P^T v_1 (\ell_1^T W \ell_2) = P^T v_2 - \epsilon_1 P^T v_1 \cdot 0 = P^T v_2 \in \mathcal{SOC}^{n+1}.$$

Finally,

$$\ell_1^T W_{\epsilon_1} \ell_2 = \ell_1^T W \ell_2 - \epsilon_1 (\ell_1^T v_1)(v_1^T \ell_2) = \ell_1^T W \ell_2 - \epsilon_1 (\ell_1^T v_1)(\ell_1^T W \ell_2) = 0 - \epsilon_1 \cdot \ell_1^T v_1 \cdot 0 = 0,$$

as desired.

We now claim that at least W_{ϵ_i} satisfies the remaining constraint

$$J \bullet P^T W_{\epsilon_i} P = J \bullet P^T W P - \epsilon_i J \bullet (P^T v_i)(P^T v_i)^T \geq 0$$

for small $\epsilon_i > 0$, thus completing the proof as discussed above. If $J \bullet P^T W P > 0$, then both W_{ϵ_i} satisfy the inequality. If $P^T v_1 \in \text{bd}(\mathcal{SOC}^{n+1})$, then W_{ϵ_1} satisfies the inequality because $J \bullet (P^T v_i)(P^T v_i)^T = 0$ by Lemma 1(ii). Similarly if $P^T v_2 \in \text{bd}(\mathcal{SOC}^{n+1})$, then W_{ϵ_2} satisfies the inequality. $\square$

As just proven, Theorem 1 establishes $\mathcal{G}^0 = \mathcal{R}^0$. A natural question is whether the analagous sets $\mathcal{G}$ and $\mathcal{R}$ as defined in Section 3 are also equal, i.e., without the complementarity conditions $(\ell_1^T w)(\ell_2^T w) = 0$ and $\ell_1^T W \ell_2 = 0$. We show in the following proposition that this is *not* true.

Proposition 1. *Let $\mathcal{F}$, $\mathcal{G}$, and $\mathcal{R}$ be defined as in Section 2 with respect to the set (4). Then $\mathcal{G} \subsetneq \mathcal{R}$.*

Proof. For $n = 3$, $g_1 = 1$, $h_1 = 0$, $g_2 = 0$, and $h_2 = e$, where e is the vector of all ones, define

$$\hat{U} := \frac{1}{3} \begin{pmatrix} 2 & \sqrt{2} & \sqrt{3} \\ 1 & 1 & -1 \\ 1 & 1 & -1 \\ 1 & 0 & -1 \\ \sqrt{3} & \sqrt{2} & \sqrt{3} \end{pmatrix}$$

and

$$\hat{W} := \hat{U}\hat{U}^T.$$

It is straightforward to verify $\hat{W} \in \{W \in \mathcal{R} : W_{11} = 1\}$ with $P^T W \ell_i \in \text{bd}(\mathcal{SOC}^{n+1})$ for both

$i = 1, 2$. As $\operatorname{rank}(\hat{W}) = 3$, let

$$\hat{N} = \begin{pmatrix} 0 & -2\sqrt{3} \\ -1 & \sqrt{2}(\sqrt{3}-2) \\ 1 & 0 \\ 0 & 2(\sqrt{2}+\sqrt{3}) - \sqrt{6} - 3 \\ 0 & \sqrt{3}+2 \end{pmatrix} \approx \begin{pmatrix} 0 & -3.4641 \\ -1 & -0.3789 \\ 1 & 0.0000 \\ 0 & 0.8430 \\ 0 & 3.7321 \end{pmatrix}$$

be a matrix whose columns span $\operatorname{Null}(\hat{W})$. Also define

$$\begin{aligned} \hat{M} &:= e_1 e_1^T + \operatorname{symm}\left((PJP^T\hat{W})(\ell_1\ell_1^T + \ell_2\ell_2^T)\right) + \hat{N}\hat{N}^T \\ &\approx \begin{pmatrix} 13.0000 & 1.2682 & -0.0445 & -2.9649 & -12.8511 \\ 1.2682 & -0.4830 & -1.6266 & -0.8488 & -0.5634 \\ -0.0445 & -1.6266 & -0.6266 & -0.5293 & 0.8508 \\ -2.9649 & -0.8488 & -0.5293 & -0.7213 & 3.8998 \\ -12.8511 & -0.5634 & 0.8508 & 3.8998 & 13.7881 \end{pmatrix}, \end{aligned}$$

where e_1 is the first unit vector and , for any square matrix H, we define $\operatorname{symm}(H) := \frac{1}{2}(H + H^T)$.

Next, we consider the two optimization problems

$$\begin{aligned} v^* &:= \min\{w^T\hat{M}w : w \in \mathcal{F}, \alpha = 1\} = \min\{\hat{M} \bullet W : W \in \mathcal{G}, W_{11} = 1\}, \\ r^* &:= \min\{\hat{M} \bullet W : W \in \mathcal{R}, W_{11} = 1\}. \end{aligned}$$

By construction, the first problem is relaxed by the second via the inclusion $\mathcal{G} \subseteq \mathcal{R}$. In particular, $v^* \geq r^*$. In fact, we claim $v^* > 1 = r^*$, which will demonstrate that $\mathcal{G} \subsetneq \mathcal{R}$.

We first argue $r^* = 1$. Let $W \in \mathcal{R}$ with $W_{11} = 1$ be any feasible point. We have

$$\begin{aligned} \hat{M} \bullet W &= \left(e_1e_1^T + \operatorname{symm}\left((PJP^T\hat{W})(\ell_1\ell_1^T + \ell_2\ell_2^T)\right) + \hat{N}\hat{N}^T\right) \bullet W \\ &= e_1e_1^T \bullet W + (PJP^T\hat{W})(\ell_1\ell_1^T + \ell_2\ell_2^T) \bullet W + \hat{N}\hat{N}^T \bullet W \\ &= W_{11} + (P^T\hat{W}\ell_1)^T J(P^TW\ell_1) + (P^T\hat{W}\ell_2)^T J(P^TW\ell_2) + \hat{N}\hat{N}^T \bullet W \\ &= 1 + (P^T\hat{W}\ell_1)^T J(P^TW\ell_1) + (P^T\hat{W}\ell_2)^T J(P^TW\ell_2) + 0 \\ &\geq 1 + 0 + 0 + 0 \\ &= 1, \end{aligned}$$

where the inequality follows from Lemma 1(i). Moreover, by evaluating $\hat{M} \bullet W$ at $\hat{W}$, we have

$$\begin{aligned}\hat{M} \bullet \hat{W} &= 1 + (P^T \hat{W} \ell_1)^T J (P^T \hat{W} \ell_1) + (P^T \hat{W} \ell_2)^T J (P^T \hat{W} \ell_2) + 0 \\ &= 1 + 0 + 0 + 0 \\ &= 1,\end{aligned}$$

where the second equality comes from Lemma 1(ii) and the fact that $P^T W \ell_i \in \operatorname{bd}(\mathcal{SOC}^{n+1})$ for both i. It follows that $r^* = 1$.

To establish $v^* > 1$, we used the global nonconvex quadratic programming solver of Gurobi 9.5 to calculate v^*. Gurobi returned the optimal solution

$$w^* = \begin{pmatrix} \alpha^* \\ x^* \\ \beta^* \end{pmatrix} \approx \begin{pmatrix} 1.0000 \\ 0.5690 \\ 0.5689 \\ 0.3957 \\ 0.8966 \end{pmatrix},$$

satisfying

$$\begin{aligned}\ell_1^T w^* = \beta^* - \alpha^* g_0 + h_0^T x^* &\approx 0.1034 > 0, \\ \ell_2^T w^* = \beta^* - \alpha^* g_1 + h_1^T x^* &\approx 0.6369 > 0, \\ \beta^* - \|x^*\| &\approx 0.0000, \\ v^* = (w^*)^T \hat{M} w^* &\approx 1.0002 > 1.\end{aligned}$$

Moreover, Gurobi reported a best lower bound of approximately 1.0002 with an optimality gap of 1.2×10^{-8}. □

Related to Conjecture 1, we note that this proof is based on the relaxation $\min\{\hat{M} \bullet W : W \in \mathcal{R}, W_{11} = 1\}$ for which the RLT constraint $\ell_1^T W \ell_2 \geq 0$ is not active at optimality. However, note that this specific objective matrix $\hat{M}$ involves the artificial variable β, whereas the conjecture applies only to objectives not involving β. See also the discussion after Conjecture 1 in Section 3.

6 Conclusions

In this paper, we have constructed stronger relaxations for (QP) by transforming its feasible set, lifting to one higher dimension, and employing standard relaxation techniques from the literature. Our computational results demonstrate the strength of our relaxation. In addition, the time required for solving our relaxation is modest for small values of m. We have also examined an important variant of (QP) based on the feasible set $\|x\| \leq \min\{1, g_2 + h_2^T x\}$ and shown that it has an exact, non-disjunctive semidefinite representation.

There are many open questions related to this research, e.g., Conjecture 1. In addition, is it possible to prove that our relaxation Beta is always at least as tight as Kron, which is supported by the computational evidence? Furthermore, is Beta provably as strong as the method of [16]? Is it possible to extend Theorem 1 to the case of more constraints of the form $\|x\| \leq g_i + h_i^T x$? Finally, could our relaxation be used as the basis for an effective global optimization algorithm of (QP)?

References

[1] T. A. Almaadeed, S. Ansary Karbasy, M. Salahi, and A. Hamdi. On indefinite quadratic optimization over the intersection of balls and linear constraints. *J. Optim. Theory Appl.*, 194(1):246–264, 2022.

[2] K. M. Anstreicher. Kronecker product constraints with an application to the two-trust-region subproblem. *SIAM Journal on Optimization*, 27(1):368–378, 2017.

[3] M. ApS. *The MOSEK Fusion API for Python Manual. Version 10.0.37.*, 2023.

[4] G. Azuma, M. Fukuda, S. Kim, and M. Yamashita. Exact sdp relaxations for quadratic programs with bipartite graph structures, 2022.

[5] A. Beck and D. Pan. A branch and bound algorithm for nonconvex quadratic optimization with ball and linear constraints. *J. Global Optim.*, 69(2):309–342, 2017.

[6] D. Bienstock. A note on polynomial solvability of the CDT problem. *SIAM Journal on Optimization*, 26(1):488–498, 2016.

[7] D. Bienstock and A. Michalka. Polynomial solvability of variants of the trust-region subproblem. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 380–390. ACM, New York, 2014.

[8] S. Burer. A gentle, geometric introduction to copositive optimization. *Math. Program.*, 151(1, Ser. B):89–116, 2015.

[9] S. Burer and K. M. Anstreicher. Second-order-cone constraints for extended trust-region subproblems. *SIAM Journal on Optimization*, 23(1):432–451, 2013.

[10] S. Burer, K. M. Anstreicher, and M. Dür. The difference between 5×5 doubly non-negative and completely positive matrices. *Linear Algebra Appl.*, 431(9):1539–1552, 2009.

[11] S. Burer and B. Yang. The trust region subproblem with non-intersecting linear constraints. *Mathematical Programming*, 149(1-2, Ser. A):253–264, 2015.

[12] S. Burer and Y. Ye. Exact semidefinite formulations for a class of (random and non-random) nonconvex quadratic programs. *Math. Program.*, 181(1, Ser. A):1–17, 2020.

[13] S. Burer and Y. Ye. Correction to: Exact semidefinite formulations for a class of (random and non-random) nonconvex quadratic programs. *Math. Program.*, 190(1-2, Ser. A):845–848, 2021.

[14] C. Chen, A. Atamtürk, and S. S. Oren. A spatial branch-and-cut method for nonconvex QCQP with bounded complex variables. *Mathematical Programming*, 165(2, Ser. A):549–577, 2017.

[15] L. Consolini and M. Locatelli. Sharp and fast bounds for the celis-dennis-tapia problem, 2021.

[16] A. Eltved and S. Burer. Strengthened SDP relaxation for an extended trust region subproblem with an application to optimal power flow. *Math. Program.*, 197(1, Ser. A):281–306, 2023.

[17] R. Jiang and D. Li. Second order cone constrained convex relaxations for nonconvex quadratically constrained quadratic programming. *Journal of Global Optimization*, 75(2):461–494, June 2019.

[18] S. Kelly, Y. Ouyang, and B. Yang. A note on semidefinite representable reformulations for two variants of the trust-region subproblem. Manuscript, School of Mathematical and Statistical Sciences, Clemson University, Clemson, South Carolina, USA, 2022.

[19] M. Locatelli. Kkt-based primal-dual exactness conditions for the shor relaxation. *Journal of Global Optimization*, 2022. Published online: 29 November 2022.

[20] G. Pataki. On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. *Math. Oper. Res.*, 23:339–358, 1998.

[21] F. Rendl and H. Wolkowicz. A semidefinite framework for trust region subproblems with applications to large scale minimization. *Math. Programming*, 77(2, Ser. B):273–299, 1997.

[22] H. D. Sherali and W. P. Adams. *A reformulation-linearization technique for solving discrete and continuous nonconvex problems*, volume 31 of *Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, Dordrecht, 1999.

[23] S. Sojoudi and J. Lavaei. Exactness of semidefinite relaxations for nonlinear optimization problems with underlying graph structure. *SIAM J. Optim.*, 24(4):1746–1778, 2014.

[24] J. F. Sturm and S. Zhang. On cones of nonnegative quadratic functions. *Math. Oper. Res.*, 28(2):246–267, 2003.

[25] A. L. Wang and F. Kılı nç Karzan. On the tightness of SDP relaxations of QCQPs. *Math. Program.*, 193(1, Ser. A):33–73, 2022.

[26] Y. Ye and S. Zhang. New results on quadratic minimization. *SIAM J. Optim.*, 14(1):245–267 (electronic), 2003.

[27] J. Zhen, D. de Moor, and D. den Hertog. An extension of the reformulation-linearization technique to nonlinear optimization. Manuscript, ETH Zürich, , Zürich, Switzerland, 2022.