

A spatial branch-and-cut method for nonconvex QCQP with bounded complex variables

Chen Chen¹ · Alper Atamtürk¹ · Shmuel S. Oren¹

Received: 15 August 2015 / Accepted: 17 November 2016 / Published online: 10 December 2016
© Springer-Verlag Berlin Heidelberg and Mathematical Optimization Society 2016

Abstract We develop a spatial branch-and-cut approach for nonconvex quadratically constrained quadratic programs with bounded complex variables (CQCQP). Linear valid inequalities are added at each node of the search tree to strengthen semidefinite programming relaxations of CQCQP. These valid inequalities are derived from the convex hull description of a nonconvex set of 2×2 positive semidefinite Hermitian matrices subject to a rank-one constraint. We propose branching rules based on an alternative to the rank-one constraint that allows for local measurement of constraint violation. Closed-form bound tightening procedures are used to reduce the domain of the problem. We apply the algorithm to solve the alternating current optimal power flow problem with complex variables as well as the box-constrained quadratic programming problem with real variables.

Mathematics Subject Classification 65K05 · 90C26

1 Introduction

The nonconvex quadratically-constrained quadratic programming problem with complex bounded variables (CQCQP) has numerous applications in signal processing [20,24,49] and control theory [11], among others. Our main motivation for develop-

✉ Alper Atamtürk
atamturk@berkeley.edu

Chen Chen
chenchen@berkeley.edu

Shmuel S. Oren
oren@berkeley.edu

¹ Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720-1777, USA

ing an algorithm for CQCQP is to solve power flow problems with alternating current [26,31]. We consider the following formulation of CQCQP:

$$\begin{aligned}
 & \min x^* Q_0 x + \operatorname{Re}(c_0^* x) + b_0 \\
 \text{(CQCQP)} \quad & \text{s.t. } x^* Q_i x + \operatorname{Re}(c_i^* x) + b_i \leq 0, \quad i = 1, \dots, m \\
 & \ell \leq x \leq u \\
 & x \in \mathbb{C}^n.
 \end{aligned}$$

We denote the conjugate transpose operator by $*$, and real components with $\operatorname{Re}(\cdot)$ and imaginary components with $\operatorname{Im}(\cdot)$. The decision vector $x \in \mathbb{C}^n$ has complex entries, and the remaining terms are data: Hermitian matrices $Q_i \in \mathbb{H}^{n \times n}$, real vector $b \in \mathbb{R}^n$, and complex vector $c_i \in \mathbb{C}^n$. Variable bounds $\ell \leq x \leq u$ are component-wise inequalities in the complex space—we assume these to be finite. Note that the Hermitian assumption on Q_i is without loss of generality since, otherwise, Q_i may be replaced with $(Q_i + Q_i^*)/2$.

Quadratically-constrained quadratic programs on real variables (RQCQP) can be considered a special case of CQCQP where all imaginary components are restricted to 0. Likewise, CQCQP can be solved using RQCQP by defining separate real decision vectors to represent $\operatorname{Re}(x)$ and $\operatorname{Im}(x)$. Despite this modeling equivalence, it is beneficial to exploit the structure of CQCQP to derive complex valid inequalities, bound tightening and spatial partitioning rules. Stronger results can be obtained in the complex space compared to their real counterparts. In particular, in Sect. 2.1 we show that the new valid inequalities derived in the complex space can be interpreted as a complex analogue of the RLT inequalities (e.g. [3]), and that the RLT inequalities applied to the RQCQP reformulation of CQCQP are dominated by these complex valid inequalities.

Observing the advantages of working with the complex formulations, a growing body of work considers CQCQP as a problem distinct from RQCQP. For instance, Jozs and Molzahn [29] show that moment-based semidefinite programming relaxations derived from CQCQP are not, in general, equivalent to those derived from a RQCQP transformation of the same problem and with computational experiments they demonstrate the power of retaining the CQCQP formulation. Stronger results are given regarding the SDP relaxation of the CQCQP associated with the S-lemma problem instead of the RQCQP equivalent [8,25]. Jiang et al. [28] demonstrate theoretically and empirically the benefits of working with CQCQP representation for generating strong relaxations. Kocuk et al. [30] demonstrate both theoretically and empirically for the optimal power flow problem that the spatial branch-and-bound approach can be improved by considering the complex formulation before converting it to a problem in the reals.

In this paper we give a spatial branch-and-cut (SBC) approach to solve CQCQP. For brevity, we assume familiarity with the general spatial branching framework; the reader is referred to Belotti et al. [9] for a thorough treatment on the subject. There are several spatial branching algorithms for RQCQP (e.g. [7,32,36,40,41]); however, to the best of our knowledge this paper presents the first spatial branching algorithm developed specifically for CQCQP. The proposed SBC algorithm has three distin-

guishing features. First, we use a complex SDP formulation strengthened with valid inequalities. These inequalities are derived from the convex hull description of a nonconvex set of 2×2 positive semidefinite Hermitian matrices subject to a rank-one constraint that arises from a lifted formulation of CQCQP. Their relationship with the RLT inequalities are discussed at the end of Sect. 2.1. Second, we propose spatial partitioning on the entries of the relaxation’s decision matrix, and we consider branching rules based on an alternative measure of constraint violation in lieu of the matrix rank constraint. This method does not rely on the CQCQP structure, and applies generally to any rank-constrained SDP formulation. Therefore, one can configure our algorithm to solve RQCQP using an SDP relaxation strengthened with RLT inequalities. Third, we develop bound tightening procedures based on closed-form solutions.

Computational experiments are conducted on the alternating current optimal power flow (ACOPF) problem and the box-constrained quadratic programming (BoxQP) problem. ACOPF is a generation dispatch problem that models alternating current (AC) using steady-state power flow equations, which are nonconvex quadratic constraints on complex variables relating power and voltage at buses and across transmission lines. ACOPF is commonly solved with heuristic iterative Newton-type solvers (e.g. [45]). ACOPF may be modeled as a QCQP problem, and there has been a recent interest in solving this formulation to optimality with SDP relaxations (e.g. [6,31]) and branch-and-bound methods (e.g. [22,30,40]) due to the potential for establishing global optimality. We show with numerical experiments that the formulation exploiting the complex structure leads to significantly faster solution times compared to the application of the algorithm on RQCQP reformulations on ACOPF instances. Since the proposed branching rules also apply to RQCQP, we test them on BoxQP—a well-studied nonconvex quadratic programming problem with nonhomogeneous quadratic objective and bounded real variables. BoxQP can be solved with a problem-specific finite branch-and-bound method [16]; therefore, it provides a useful benchmark for the general approach presented here. We demonstrate that the proposed method can also be used as a viable RLT and SDP-based RQCQP solver.

The rest of the paper is organized as follows: Sect. 2 details the spatial branch-and-cut algorithm with three major components: valid inequalities from the convex hull description of rank-one constrained relaxations, branching on matrix entries, and bound tightening procedures; Sect. 3 contains the results from computational experiments with ACOPF and BoxQP problems; Sect. 4 concludes the paper.

2 The spatial branch-and-cut method

The convex relaxation of QCQP considered comes from the rank-one constraint of a standard SDP reformulation [34,44], often called Shor’s relaxation:

$$\min \langle Q_0 X \rangle + \operatorname{Re}(c_0^* x) + b_0 \tag{1a}$$

$$\text{(CSDP) s.t. } \langle Q_i, X \rangle + \operatorname{Re}(c_i^* x) + b_i \leq 0, \quad i = 1, \dots, m \tag{1b}$$

$$\ell \leq x \leq u \tag{1c}$$

$$\begin{bmatrix} 1 & x^* \\ x & X \end{bmatrix} \succeq 0, \tag{1d}$$

where $X \in \mathbb{H}^{n \times n}$ is a Hermitian submatrix of decision variables. Imposing a rank-one constraint on the matrix $Y := \begin{bmatrix} 1 & x^* \\ x & X \end{bmatrix}$ gives an equivalent reformulation of CQCQP. Let $y := \begin{pmatrix} 1 \\ x \end{pmatrix}$ so that $Y = yy^*$. Furthermore, denote the real and imaginary components as $Y := W + iT$ and $y := w + it$. Since Y is Hermitian we have $W_{ij} = W_{ji}$, $T_{ij} = -T_{ji}$, $\forall i, j$ and, thus, $\text{diag}(T) = 0$.

This section is divided into three subsections. First, we derive valid inequalities to strengthen the SDP relaxation. Second, we propose a methodology for branching on the entries of matrix Y . Third, we present closed-form bound-tightening techniques.

2.1 Valid inequalities

Here we describe the convex hull of a nonconvex set that is applicable to any (i, j) entry of Y , where $1 \leq i \neq j \leq n + 1$. Let \mathcal{J}_C be the set of feasible solutions in $\mathbb{C}^{2 \times 2}$ to the following constraints:

$$L_{ii} \leq W_{ii} \leq U_{ii}, \tag{2a}$$

$$L_{jj} \leq W_{jj} \leq U_{jj}, \tag{2b}$$

$$L_{ij}W_{ij} \leq T_{ij} \leq U_{ij}W_{ij}, \tag{2c}$$

$$W_{ij} \geq 0, \tag{2d}$$

$$W_{ii}W_{jj} = W_{ij}^2 + T_{ij}^2, \tag{2e}$$

where L and U are real, symmetric bound matrices with $L \leq U$ and $L_{ii}, L_{jj} \geq 0$.

One can always find such bound values so that \mathcal{J}_C is valid for either CQCQP or an affine transformation of CQCQP. We will give a generic method to obtain such bounds at the end of this subsection. First we describe the convex hull of \mathcal{J}_C . The convex hull description will provide linear valid inequalities for \mathcal{J}_C , which can be applied to all 2×2 principal minors of Y in order to strengthen CSDP.

In order to describe the convex hull of \mathcal{J}_C it is convenient to use the following sigmoid function:

$$f(x) := \begin{cases} (\sqrt{1+x^2} - 1)/x, & x \neq 0 \\ 0, & x = 0 \end{cases}$$

Remark 1 $f(x)$ is increasing, strictly bounded above by $+1$ and strictly bounded below by -1 .

Consider the following linear inequalities:

$$\pi_0 + \pi_1 W_{ii} + \pi_2 W_{jj} + \pi_3 W_{ij} + \pi_4 T_{ij} \geq U_{jj}W_{ii} + U_{ii}W_{jj} - U_{ii}U_{jj}, \tag{3a}$$

$$\pi_0 + \pi_1 W_{ii} + \pi_2 W_{jj} + \pi_3 W_{ij} + \pi_4 T_{ij} \geq L_{jj}W_{ii} + L_{ii}W_{jj} - L_{ii}L_{jj}, \tag{3b}$$

where the coefficients π are defined as

$$\begin{aligned} \pi_0 &:= -\sqrt{L_{ii}L_{jj}U_{ii}U_{jj}}, \\ \pi_1 &:= -\sqrt{L_{jj}U_{jj}}, \\ \pi_2 &:= -\sqrt{L_{ii}U_{ii}}, \\ \pi_3 &:= (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{1 - f(L_{ij})f(U_{ij})}{1 + f(L_{ij})f(U_{ij})}, \\ \pi_4 &:= (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{f(L_{ij}) + f(U_{ij})}{1 + f(L_{ij})f(U_{ij})}. \end{aligned}$$

Lemma 1 For $\alpha_{ij} \in \{L_{ij}, U_{ij}\}$ we have

$$1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij})) = (1 + f(L_{ij})f(U_{ij}))\sqrt{1 + \alpha_{ij}^2}. \tag{4}$$

Proof If $\alpha_{ij} = 0$, then equality (4) follows immediately. Otherwise, suppose $\alpha_{ij} = L_{ij} \neq 0$. Then we have

$$\begin{aligned} &1 - f(L_{ij})f(U_{ij}) + L_{ij}(f(L_{ij}) + f(U_{ij})) - (1 + f(L_{ij})f(U_{ij}))\sqrt{1 + L_{ij}^2} \\ &= -f(L_{ij})f(U_{ij}) + L_{ij}f(U_{ij}) - f(L_{ij})f(U_{ij})\sqrt{1 + L_{ij}^2} \\ &= f(U_{ij})[L_{ij} - f(L_{ij})(1 + \sqrt{1 + L_{ij}^2})] \\ &= f(U_{ij})[L_{ij} - L_{ij}^2/L_{ij}] \\ &= 0. \end{aligned}$$

Therefore

$$1 - f(L_{ij})f(U_{ij}) + L_{ij}(f(L_{ij}) + f(U_{ij})) = (1 + f(L_{ij})f(U_{ij}))\sqrt{1 + L_{ij}^2}.$$

The case where $\alpha_{ij} = U_{ij} \neq 0$ follows by symmetry. □

Proposition 2 Inequalities (3a) and (3b) are valid for \mathcal{J}_C .

Proof For any convex function f , the secant line connecting $(a, f(a))$ and $(b, f(b))$ lies above the graph of f . Thus for any variable q we have $\ell_q \leq q \leq u_q \implies (\ell_q + u_q)q - \ell_q u_q \geq q^2$. Now for $k \in \{i, j\}$, since $\sqrt{L_{kk}} \leq \sqrt{W_{kk}} \leq \sqrt{U_{kk}}$, applying this secant principle yields

$$(\sqrt{L_{kk}} + \sqrt{U_{kk}})\sqrt{W_{kk}} \geq \sqrt{L_{kk}U_{kk}} + W_{kk}. \tag{5a}$$

Multiplying inequalities from (5a) for $k = 1$ and $k = 2$ gives

$$\sqrt{W_{ii}W_{jj}}(\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \geq (\sqrt{L_{ii}U_{ii}} + W_{ii})(\sqrt{L_{jj}U_{jj}} + W_{jj}).$$

Rearranging terms, we have

$$\sqrt{W_{ii}W_{jj}}(\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) + \pi_0 + \pi_1W_{ii} + \pi_2W_{jj} \geq W_{ii}W_{jj}. \tag{5b}$$

The right-hand-sides of constraints (3a) and (3b) are RLT inequalities (discussed in the next subsection) and thus are valid for the bilinear term $W_{ii}W_{jj}$, which is the right-hand-side of the valid inequality (5b). It remains to show that the left-hand-side of (5b) is overestimated by the left-hand sides of (3a) and (3b), i.e.:

$$\pi_3W_{ij} + \pi_4T_{ij} \geq \sqrt{W_{ii}W_{jj}}(\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}). \tag{5c}$$

Let $T_{ij} = \alpha W_{ij}$ for some $L_{ij} \leq \alpha_{ij} \leq U_{ij}$. Note that constraint (2d) restricts W_{ij} to be nonnegative, so from constraint (2e) we have:

$$\sqrt{W_{ii}W_{jj}} = \sqrt{1 + \alpha_{ij}^2}W_{ij}. \tag{5d}$$

Substituting equality (5d) into inequality (5c), we want to show

$$(\pi_3 + \alpha_{ij}\pi_4)W_{ij} \geq \sqrt{1 + \alpha_{ij}^2}(\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}})W_{ij} \tag{5e}$$

for any $L_{ij} \leq \alpha_{ij} \leq U_{ij}$. Replacing the coefficients with their definitions and simplifying, we get the following equivalent condition:

$$1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij})) - (1 + f(L_{ij})f(U_{ij}))\sqrt{1 + \alpha_{ij}^2} \geq 0. \tag{5f}$$

To show that inequality (5f) is valid, first observe that the second derivative of the left-hand side with respect to α_{ij} is $-(1 + f(L_{ij})f(U_{ij}))/ (1 + \alpha^2)^{3/2}$. Thus the left-hand side is concave w.r.t α (as noted in Remark 1 we have that $|f(L_{ij})|, |f(U_{ij})| < 1$). Therefore, we only need to check that the left-hand side is nonnegative for $\alpha_{ij} \in \{L_{ij}, U_{ij}\}$. From Lemma 1 we have that the left-hand side is exactly zero, and so inequality (5f) and consequently inequalities (3a) and (3b) are valid for \mathcal{J}_C . \square

Let \mathcal{J}_S be the set of feasible solutions to the natural SDP relaxation of \mathcal{J}_C :

$$L_{ii} \leq W_{ii} \leq U_{ii}, \tag{6a}$$

$$L_{jj} \leq W_{jj} \leq U_{jj}, \tag{6b}$$

$$L_{ij}W_{ij} \leq T_{ij} \leq U_{ij}W_{ij} \tag{6c}$$

$$W_{ij} \geq 0, \tag{6d}$$

$$W_{ii}W_{jj} \geq W_{ij}^2 + T_{ij}^2. \tag{6e}$$

Remark 2 For $n = 2$, $W + \iota T \geq 0$ is equivalent to the principal minor constraints: $W_{ii} \geq 0$, $W_{jj} \geq 0$, $W_{ii}W_{jj} \geq W_{ij}^2 + T_{ij}^2$. Since by construction $L_{ii}, L_{jj} \geq 0$, then constraint (6e) is equivalent to the positive semidefinite constraint for \mathcal{J}_S .

Let \mathcal{J}_V be the set of X satisfying inequalities (3a)–(3b). We shall prove that the convex hull of \mathcal{J}_C can be obtained by adding the valid inequalities (3a)–(3b) to the SDP relaxation.

Proposition 3 $\text{conv}(\mathcal{J}_C) = \mathcal{J}_S \cap \mathcal{J}_V$.

Proof From Proposition 2 we have that $\text{conv}(\mathcal{J}_C) \subseteq \mathcal{J}_S \cap \mathcal{J}_V$. From constraints (2a)–(2b) and (2e) we can observe that $\mathcal{J}_S \cap \mathcal{J}_V$ is bounded. Thus to prove that $\mathcal{J}_S \cap \mathcal{J}_V \subseteq \text{conv}(\mathcal{J}_C)$, it is sufficient to ensure that all the extreme points of $\mathcal{J}_S \cap \mathcal{J}_V$ are in \mathcal{J}_C . First, let us invoke the following claim:

Claim 4 *If $W_{ij} = 0$, then either $X \notin \mathcal{J}_S \cap \mathcal{J}_V$ or $X \in \text{conv}(\mathcal{J}_C)$.*

Proof By way of contradiction, suppose there exists $\bar{X} \in \mathcal{J}_S \cap \mathcal{J}_V \setminus \text{conv}(\mathcal{J}_C)$ such that $\bar{W}_{ij} = 0$. From constraint (2c) we have that $\bar{T}_{ij} = 0$. Consequently, if $\bar{W}_{ii} = 0$ or $\bar{W}_{jj} = 0$, then constraint (2e) is satisfied, so either $\bar{X} \notin \mathcal{J}_S \cap \mathcal{J}_V$ or $\bar{X} \in \text{conv}(\mathcal{J}_C)$.

From Proposition 2 we only need to consider $\bar{W}_{ii}\bar{W}_{jj} > 0$, which $\bar{W}_{ii}\bar{W}_{jj} > 0$ implies that the right-hand side of constraint (3b) is nonnegative since $0 \leq L_{kk} \leq \bar{W}_{kk}$, $k \in \{i, j\}$, and so $L_{jj}\bar{W}_{ii} + L_{ii}\bar{W}_{jj} - L_{ii}L_{jj} \geq L_{ii}\bar{W}_{jj} \geq 0$. On the left-hand side, all terms are nonpositive, so we require all terms to be zero in order to ensure $\bar{X} \in \mathcal{J}_S \cap \mathcal{J}_V$. Since we have that $\bar{W}_{ii}\bar{W}_{jj} > 0$, then $\pi_1\bar{W}_{ii} = \pi_2\bar{W}_{jj} = 0$ only if $L_{ii} = L_{jj} = 0$.

Now let us define the following two matrices:

$$U^1 := \begin{bmatrix} U_{ii} & 0 \\ 0 & 0 \end{bmatrix}, \quad U^2 := \begin{bmatrix} 0 & 0 \\ 0 & U_{jj} \end{bmatrix}.$$

Since $L_{ii} = L_{jj} = 0$, then $U^1, U^2 \in \mathcal{J}_C$, and since $\bar{W}_{ij} = \bar{T}_{ij} = 0$. Checking constraint (3a) we have $0 \geq U_{jj}W_{ii} + U_{ii}W_{jj} - U_{ii}U_{jj}$, so $\bar{X} \in \mathcal{J}_S \cap \mathcal{J}_V$ provided $W_{ii} \leq U_{ii}$, $W_{jj} \leq U_{jj}$. Observe that \bar{X} can be expressed as the convex combination of U^1, U^2 , and the zeros matrix. $L_{ii} = L_{jj} = 0$ implies the zeros matrix is a member of \mathcal{J}_C , which in turn implies $\bar{X} \in \text{conv}(\mathcal{J}_C)$, which contradicts our initial assumption. □

Now we will prove that $\text{ext}(\mathcal{J}_S \cap \mathcal{J}_V) \in \mathcal{J}_C$. Observe that if the constraint (6e) is binding at an extreme point of $\mathcal{J}_S \cap \mathcal{J}_V$, then it is a member of \mathcal{J}_C . Moreover, by Claim (4), if a point \bar{X} with $\bar{W}_{ij} = 0$ is in $\text{ext}(\mathcal{J}_S \cap \mathcal{J}_V)$, then $\bar{X} \in \text{conv}(\mathcal{J}_C)$. It follows that \bar{X} is a member of \mathcal{J}_C since $\mathcal{J}_S \cap \mathcal{J}_V \supseteq \text{conv}(\mathcal{J}_C) \supseteq \mathcal{J}_C$. Therefore, we shall check by cases for any extreme point where constraints (6e) and (6d) are not binding.

Case 1 Constraints (3a) and (3b) are not binding:

If constraint (6e) is not binding, then we require at least four linearly independent linear constraints to be binding. To obtain four such constraints, we require the two

variable bounds (2a) and (2b) to be binding. Moreover, we require that constraint (2c) is binding on both sides with $L_{ij} \neq U_{ij}$, which implies $W_{ij} = T_{ij} = 0$. By Claim (4), this point can be disregarded.

Case 2 Constraints (3a) and (3b) are both binding:

Since constraints (3a) and (3b) share the same coefficients π_3, π_4 for W_{ij}, T_{ij} , then for an extreme point we require that constraint (2c) is binding on at least one side. Due to Claim (4) we need only consider $W_{ij} \neq 0$, so constraint (2c) can count for at most one linearly independent constraint; thus, let $T_{ij} = \alpha_{ij} W_{ij}$, where $\alpha_{ij} \in \{L_{ij}, U_{ij}\}$. This gives at most three linearly independent constraints, (2c), (3a) and (3b), so at least one of the variable bounds (2a) and (2b) must be binding. Define α_{kk} so that $W_{kk} = (U_{kk} - L_{kk})\alpha_{kk} + L_{kk}$ for $k \in \{i, j\}$. Since at least one of W_{ii}, W_{jj} is at a variable bound, then $\alpha_{kk} \in \{0, 1\}$ for either $k = i$ or $k = j$. Moreover, the right-hand sides of constraints (3a) and (3b) must be equal since the left-hand sides are the same and both constraints are binding in this case. Therefore, we can write:

$$\begin{aligned} U_{jj}W_{ii} + U_{ii}W_{jj} - U_{ii}U_{jj} &= L_{jj}W_{ii} + L_{ii}W_{jj} - L_{ii}L_{jj} \\ \iff (\alpha_{ii} + \alpha_{jj})[U_{ii}U_{jj} - L_{ii}U_{jj} - L_{jj}U_{ii} + L_{ii}L_{jj}] \\ &= U_{ii}U_{jj} - L_{ii}U_{jj} - L_{jj}U_{ii} + L_{ii}L_{jj} \\ \iff \alpha_{ii} + \alpha_{jj} &= 1. \end{aligned}$$

Thus we have either $\alpha_{ii} = 1$, in which case $W_{ii} = U_{ii}, W_{jj} = L_{jj}$ or $\alpha_{jj} = 1$, in which case $W_{ii} = L_{ii}, W_{jj} = U_{jj}$. First suppose $W_{ii} = U_{ii}, W_{jj} = L_{jj}$. Define the following matrix:

$$X^A := \begin{bmatrix} U_{ii} & \sqrt{\frac{U_{ii}L_{jj}}{1+\alpha_{ij}^2}} \\ \sqrt{\frac{U_{ii}L_{jj}}{1+\alpha_{ij}^2}} & L_{jj} \end{bmatrix} + \iota \begin{bmatrix} 0 & \alpha_{ij}\sqrt{\frac{U_{ii}L_{jj}}{1+\alpha_{ij}^2}} \\ -\alpha_{ij}\sqrt{\frac{U_{ii}L_{jj}}{1+\alpha_{ij}^2}} & 0 \end{bmatrix},$$

and as usual, denote the components as $X^A := W^A + \iota T^A$. By construction, we have that $(W_{ij}^A)^2 + (T_{ij}^A)^2 = W_{ii}^A W_{jj}^A$, so $X^A \in \mathcal{J}_C$. We want to show that, with X^A , constraints (2a)–(2c), (3a), and (3b) are binding. Constraints (2a)–(2c) can be confirmed by observation. Recall that constraints (3a) and (3b) share the same right-hand side:

$$U_{jj}W_{ii}^A + U_{ii}W_{jj}^A - U_{ii}U_{jj} = U_{ii}L_{jj}.$$

The left-hand side of constraint (3a) or (3b) is:

$$\begin{aligned} &\pi_0 + \pi_1 W_{ii}^A + \pi_2 W_{jj}^A + \pi_3 W_{ij}^A + \pi_4 T_{ij}^A \\ &= \pi_0 + \pi_1 W_{ii}^A + \pi_2 W_{jj}^A + (\pi_3 + \alpha_{ij}\pi_4)W_{ij}^A \\ &= -\sqrt{L_{ii}L_{jj}U_{ii}U_{jj}} - \sqrt{L_{jj}U_{jj}U_{ii}} - \sqrt{L_{ii}U_{ii}L_{jj}} \end{aligned}$$

$$\begin{aligned}
 & + (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij}))}{1 + f(L_{ij})f(U_{ij})} \sqrt{\frac{U_{ii}L_{jj}}{1 + \alpha_{ij}^2}} \\
 & = \sqrt{U_{ii}L_{jj}} \left(-\sqrt{L_{ii}U_{jj}} - \sqrt{U_{ii}L_{jj}} - \sqrt{L_{ii}L_{jj}} \right. \\
 & \quad \left. + (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij}))}{(1 + f(L_{ij})f(U_{ij}))\sqrt{1 + \alpha_{ij}^2}} \right) \\
 & = U_{ii}L_{jj}.
 \end{aligned}$$

The last equality follows from Lemma 1. The argument for the case $W_{ii} = L_{ii}$, $W_{jj} = U_{jj}$ follows by symmetry.

Case 3 Exactly one of the constraints (3a) and (3b) is binding:

From Claim (4) we need only consider $W_{ij} \neq 0$, so constraint (2c) can count for at most one linearly independent constraint; thus, let $T_{ij} = \alpha_{ij}W_{ij}$, where $\alpha_{ij} \in \{L_{ij}, U_{ij}\}$. This gives us at most two linearly independent linear constraints: (2c) and either (3a) or (3b). Thus both variable bounds (2a) and (2b) must be binding. In Case 2 we have already considered the possibilities that $W_{ii} = U_{ii}$, $W_{jj} = L_{jj}$ or $W_{ii} = U_{ii}$, $W_{jj} = L_{jj}$. Suppose, then, that $W_{ii} = U_{ii}$, $W_{jj} = U_{jj}$ and define the corresponding matrix:

$$X^B := \begin{bmatrix} U_{ii} & \sqrt{\frac{U_{ii}U_{jj}}{1 + \alpha_{ij}^2}} \\ \sqrt{\frac{U_{ii}U_{jj}}{1 + \alpha_{ij}^2}} & U_{jj} \end{bmatrix} + t \begin{bmatrix} 0 & \alpha_{ij}\sqrt{\frac{U_{ii}U_{jj}}{1 + \alpha_{ij}^2}} \\ -\alpha_{ij}\sqrt{\frac{U_{ii}U_{jj}}{1 + \alpha_{ij}^2}} & 0 \end{bmatrix},$$

and as usual, denote the components as $X^B := W^B + tT^B$. By construction, we have that $(W_{ij}^B)^2 + (T_{ij}^B)^2 = (W_{ii}^B W_{jj}^B)$, so $X^B \in \mathcal{J}_C$. We want to show that, with X^B , constraints (2a)–(2c), and (3a) are binding. This can be done as in Case 2, by invoking Lemma 1:

$$\begin{aligned}
 & \pi_0 + \pi_1 W_{ii}^B + \pi_2 W_{jj}^B + \pi_3 W_{ij}^B + \pi_4 T_{ij}^B \\
 & = \pi_0 + \pi_1 W_{ii}^B + \pi_2 W_{jj}^B + (\pi_3 + \alpha_{ij}\pi_4)W_{ij}^B \\
 & = -\sqrt{L_{ii}L_{jj}U_{ii}U_{jj}} - \sqrt{L_{jj}U_{jj}U_{ii}} - \sqrt{L_{ii}U_{ii}U_{jj}} \\
 & \quad + (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij}))}{1 + f(L_{ij})f(U_{ij})} \sqrt{\frac{U_{ii}U_{jj}}{1 + \alpha_{ij}^2}} \\
 & = \sqrt{U_{ii}U_{jj}} \left(-\sqrt{L_{ii}U_{jj}} - \sqrt{U_{ii}L_{jj}} - \sqrt{L_{ii}L_{jj}} \right. \\
 & \quad \left. + (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}}) \frac{1 - f(L_{ij})f(U_{ij}) + \alpha_{ij}(f(L_{ij}) + f(U_{ij}))}{(1 + f(L_{ij})f(U_{ij}))\sqrt{1 + \alpha_{ij}^2}} \right) \\
 & = U_{ii}U_{jj} \\
 & = U_{jj}W_{ii}^B + U_{ii}W_{jj}^B - U_{ii}U_{jj}.
 \end{aligned}$$

Finally, suppose that $W_{ii} = L_{ii}$, $W_{jj} = L_{jj}$ and define the corresponding matrix:

$$X^C := \begin{bmatrix} L_{ii} & \sqrt{\frac{L_{ii}L_{jj}}{1+\alpha_{ij}^2}} \\ \sqrt{\frac{L_{ii}L_{jj}}{1+\alpha_{ij}^2}} & L_{jj} \end{bmatrix} + \iota \begin{bmatrix} 0 & \alpha_{ij}\sqrt{\frac{L_{ii}L_{jj}}{1+\alpha_{ij}^2}} \\ -\alpha_{ij}\sqrt{\frac{L_{ii}L_{jj}}{1+\alpha_{ij}^2}} & 0 \end{bmatrix}.$$

By the same argument as used with X^B , which we omit to avoid repetition, X^C belongs to \mathcal{J}_C where constraints (2a)–(2c), and (3b) are binding. Thus, in all cases every extreme point belongs to \mathcal{J}_C . □

Numerical example of \mathcal{J}_C Consider the following instance of \mathcal{J}_C with $i = 1, j = 2$:

$$0 \leq W_{11} \leq 1, \tag{7a}$$

$$1 \leq W_{22} \leq 4, \tag{7b}$$

$$0 \leq T_{12} \leq \frac{4}{3}W_{12}, \tag{7c}$$

$$W_{12} \geq 0, \tag{7d}$$

$$W_{11}W_{22} = W_{12}^2 + T_{12}^2. \tag{7e}$$

The coefficients π of the valid inequalities (3a)–(3b) are

$$\begin{aligned} f(L_{12}) = 0, f(U_{12}) = 0.5, \\ \pi_0 = 0, \pi_1 = -2, \pi_2 = 0, \pi_3 = 3, \pi_4 = 1.5. \end{aligned}$$

Valid inequality (3a) is

$$-6W_{11} - W_{22} + 3W_{12} + 1.5T_{12} + 4 \geq 0,$$

and valid inequality (3b) is

$$-3W_{11} + 3W_{12} + 1.5T_{12} \geq 0.$$

The real case: \mathcal{J}_R We now consider the special case of the real variables. Let \mathcal{J}_R be the set of symmetric matrices W that satisfy the following constraints:

$$L_{ii} \leq W_{ii} \leq U_{ii}, \tag{8a}$$

$$L_{jj} \leq W_{jj} \leq U_{jj}, \tag{8b}$$

$$W_{ij} \geq 0, \tag{8c}$$

$$W_{ii}W_{jj} = W_{ij}^2. \tag{8d}$$

\mathcal{J}_R can be seen as a special case of \mathcal{J}_C where $L_{ij} = U_{ij} = 0$, and, therefore, we have the following corollary.

Corollary 5 *The convex hull of \mathcal{J}_R can be described with the following constraints:*

$$L_{ii} \leq W_{ii} \leq U_{ii} \tag{9a}$$

$$L_{jj} \leq W_{jj} \leq U_{jj} \tag{9b}$$

$$\begin{aligned} (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}})W_{ij} &\geq (U_{jj} + \sqrt{L_{jj}U_{jj}})W_{ii} \\ &+ (U_{ii} + \sqrt{L_{ii}U_{ii}})W_{jj} + \sqrt{L_{ii}L_{jj}U_{ii}U_{jj}} - U_{ii}U_{jj} \end{aligned} \tag{9c}$$

$$\begin{aligned} (\sqrt{L_{ii}} + \sqrt{U_{ii}})(\sqrt{L_{jj}} + \sqrt{U_{jj}})W_{ij} &\geq (L_{jj} + \sqrt{L_{jj}U_{jj}})W_{ii} \\ &+ (L_{ii} + \sqrt{L_{ii}U_{ii}})W_{jj} + \sqrt{L_{ii}L_{jj}U_{ii}U_{jj}} - L_{ii}L_{jj} \end{aligned} \tag{9d}$$

$$W \geq 0 \tag{9e}$$

Proof This is a special case of Proposition 3 with $L_{ij} = U_{ij} = 0$, which due to constraint (2c) is equivalent to setting $T_{ij} = 0$, resulting in a matrix with only real entries. Dropping T_{ij} gives the desired result. \square

Numerical example of \mathcal{J}_R We shall use the numerical example of \mathcal{J}_C but setting $L_{12} = U_{12} = 0$ and dropping T_{12} accordingly. For the real case, valid inequality (3a) is

$$-6W_{11} - W_{22} + 3W_{12} + 4 \geq 0,$$

and valid inequality (3b) is

$$-3W_{11} + 3W_{12} \geq 0.$$

The inequalities for the real case are shown in Fig. 1. The feasible region of \mathcal{J}_R is depicted in the upper-left quadrant. Black spheres are centered around the intersection points of variable bounds on the cone. The upper-right quadrant depicts the intersection of $W_{12}^2 = W_{11}W_{22}$ with the variable bounds at $W_{11} = 1$ and $W_{22} = 1$. The lower-right quadrant depicts the cone with valid inequality (3a). The intersection is ellipsoidal and three of the highlighted points lie on the boundary of the valid inequality ($\{W_{11} = 0, W_{22} = 4, W_{12} = 0, \}, \{W_{11} = 1, W_{22} = 1, W_{12} = 1\}, \{W_{11} = 1, W_{22} = 4, W_{12} = 2\}$). The lower-left quadrant depicts the intersection of the cone with valid inequality (3b). The intersection is a hyperbola, and three highlighted points lie on the boundary of the valid inequality ($\{W_{11} = 0, W_{22} = 1, W_{12} = 0, \}, \{W_{11} = 0, W_{22} = 4, W_{12} = 0\}, \{W_{11} = 1, W_{22} = 1, W_{12} = 1\}$).

Comparison with RLT inequalities Consider the RLT inequalities:

$$W_{ij} \leq u_j w_i + \ell_i w_j - \ell_i u_j, \tag{10a}$$

$$\text{(RLT)} \quad W_{ij} \leq \ell_j w_i + u_i w_j - \ell_j u_i, \tag{10b}$$

$$W_{ij} \geq \ell_j w_i + \ell_i w_j - \ell_i \ell_j, \tag{10c}$$

$$W_{ij} \geq u_j w_i + u_i w_j - u_i u_j. \tag{10d}$$

Note that for RLT we allow the possibility that $i = j$. Inequalities (10a)–(10d) are derived from the RLT procedure of Sherali and Tuncbilek [43]. They are also known as

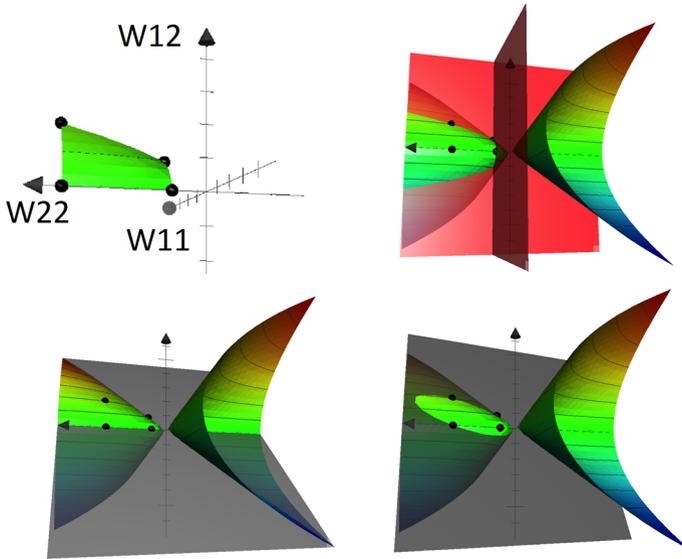


Fig. 1 Clockwise, starting from *top-left* \mathcal{J}_R ; cone of (7e), upper bound of (7a), and lower bound of (7b); cone of (7e) and valid inequality (3a); cone of (7e) and valid inequality (3b)

McCormick estimators as they can be derived from earlier work on convex envelopes by McCormick [35]. For $w \in \mathbb{R}^2$, Anstreicher and Burer [4] prove that the convex hull of the real bounded set $\{(W, w) : W = ww^T, \ell \leq w \leq u\}$ is given by the RLT inequalities on W_{ii} , W_{jj} and W_{ij} together with the SDP constraint $W \succeq ww^T$. As discussed, valid bounds for \mathcal{J}_C can be derived from the complex bounded set $l \leq x \leq u$ for $x \in \mathbb{C}^2$. We will do a thorough comparison at the end of this section, but let us first proceed to analyzing the convex hull of \mathcal{J}_C .

\mathcal{J}_R is implied by the following constraints:

$$\sqrt{L_{ii}} \leq w_i \leq \sqrt{U_{ii}}, \tag{11a}$$

$$\sqrt{L_{jj}} \leq w_j \leq \sqrt{U_{jj}}, \tag{11b}$$

$$W = ww^T. \tag{11c}$$

From Anstreicher and Burer [4] we know the convex hull of constraints (11a)–(11c) can be described by the RLT inequalities on W_{ii} , W_{jj} and W_{ij} together with the convex constraint $W \succeq ww^T$. Constraints (11a)–(11c) can be rewritten as:

$$L_{ii} \leq W_{ii} \leq U_{ii}, \tag{12a}$$

$$L_{jj} \leq W_{jj} \leq U_{jj}, \tag{12b}$$

$$W = ww^T. \tag{12c}$$

Finally, $W = ww^T$ for a vector of free variables w holds iff W is positive semi-definite and of rank one, which gives equivalence in the space of W between \mathcal{J}_R and the set of feasible solutions to constraints (11a)–(11c). Hence the RLT inequalities on W_{ii} , W_{jj} , W_{ij} dominate valid inequalities (3a) and (3b) in the special case of the reals

as they describe the convex hull of the same set in the lifted space that includes w variables.

We note that the inequalities coincide on diagonal entries of W . The RLT inequalities for diagonal matrix entries are

$$W_{ii} \leq u_i w_i + \ell_i w_i - \ell_i u_i, \tag{13a}$$

$$W_{ii} \geq 2\ell_i w_i - \ell_i^2, \tag{13b}$$

$$W_{ii} \geq 2u_i w_i - u_i^2. \tag{13c}$$

For $i > 1$ ($i = 1$ merely gives $W_{11} = 1$) inequality (13a) can also be obtained by applying either (9c) or (9d) to $\begin{bmatrix} 1 & w_i \\ w_i & W_{ii} \end{bmatrix}$, which is the 2×2 principal submatrix of Y with indices 1 and i . The remaining inequalities (13b)–(13c) are implied by the SDP constraint since $(w_i - \ell_i)^2 \geq 0$, $(w_i - u_i)^2 \geq 0 \implies w_i^2 \geq \max\{2\ell_i w_i - \ell_i^2, 2u_i w_i - u_i^2\}$ and $W_{ii} \geq w_i^2$.

In the complex case \mathcal{J}_C , we can show that domination runs the other direction: a natural application of RLT does not capture the convex hull of \mathcal{J}_C . One possible transformation to RQCQP involves a matrix of the form

$$\begin{pmatrix} 1 \\ w \\ t \end{pmatrix} \begin{pmatrix} 1 \\ w \\ t \end{pmatrix}'$$

where the components of the complex vector $x := w + it$ are treated as separate decision variables. In this case, the RLT inequalities together with the SDP constraint (6e) do not give the convex hull of \mathcal{J}_C as shown in the example below.

We continue with the numerical example of \mathcal{J}_C , and we will show that there is a point satisfying the positive semidefinite condition and the RLT inequalities, but is outside $\text{conv}(\mathcal{J}_C)$. Thus an application of RLT inequalities to the RQCQP reformulation is not sufficient to describe $\text{conv}(\mathcal{J}_C)$. Let us add the complex variables x_1, x_2 with the following bounds on magnitude:

$$\begin{aligned} 0 &\leq |x_1| \leq 1, \\ 1 &\leq |x_2| \leq 2. \end{aligned}$$

Considering the real and imaginary components of x as separate decision variables, let us transform the problem into RQCQP. Moreover, we require bounds on $\text{Re}(x)$, $\text{Im}(x)$. We shall use the magnitude-implied bounds:

$$\begin{aligned} |w_1| &\leq 1, \quad |t_1| \leq 1, \\ |w_2| &\leq 2, \quad |t_2| \leq 2. \end{aligned}$$

Now $W_{12} = \text{Re}(x_1 x_2^*) = w_1 w_2 + t_1 t_2$, and $T_{12} = \text{Im}(x_1 x_2^*) = t_1 w_2 - w_1 t_2$. Applying RLT inequalities to each bilinear term, we obtain the following inequalities:

$$W_{12} \leq -|2w_1 - w_2| - |2t_1 - t_2| + 4 \tag{14a}$$

$$W_{12} \geq |2w_1 + w_2| + |2t_1 + t_2| - 4 \tag{14b}$$

$$T_{12} \leq -|2t_1 - w_2| - |2w_1 + t_2| + 4 \tag{14c}$$

$$T_{12} \geq |2t_1 + w_2| + |2w_1 - t_2| - 4 \tag{14d}$$

$$W_{11} \leq 2, \tag{14e}$$

$$W_{11} \geq 2|w_1| + 2|t_1| - 2, \tag{14f}$$

$$W_{22} \leq 8, \tag{14g}$$

$$W_{22} \geq 4|w_1| + 4|t_1| - 8. \tag{14h}$$

RLT inequalities (14a)–(14h) admit the solution $x = 0, T_{12} = 0, W_{12} = 0, W_{11} = 1, W_{22} = 4$, which satisfies constraints (7a)–(7c) and the positive semidefinite constraint. However, this solution violates valid inequality (3b), where $W_{12} = T_{12} = 0$ implying $W_{11} = 0$.

Constructing bound matrices L and U We provide a procedure to derive valid bounds so that \mathcal{J}_C is applicable to any CQCQP. For certain problems the procedure may be unnecessary: values might be obtained from the problem directly or via bound tightening procedures. For instance, in ACOPF, constraints (2a) and (2b) are given as voltage magnitude bounds and constraint (2c) models nodal phase angle difference bounds.

Constraints (2a) and (2b) $W_{ii} = |x_{i-1}|$ for $i \geq 2$, representing constraints on CQCQP variable magnitudes, and likewise for j . As magnitudes are nonnegative, a valid lower bound can always be obtained by setting $L_{ii} = L_{jj} = 0$. Furthermore, upper bounds can be obtained directly from the original variable bounds (1c). For W_{11} we can set $L_{11} = U_{11} = 1$.

Constraints (2c) and (2d) A sufficient condition to derive valid bounds is that either w or t (or both) have strictly positive entries. This can be done using an affine shift. Observe that for any ℓ we have:

$$x^* Q x + c^* x + b = (x - \ell)^* Q (x - \ell) + (c^* + 2\ell^* Q)(x - \ell) + b + \ell^* Q \ell + c^* \ell.$$

Therefore with substitution of variables $q := x - \ell + e + \iota e$, where e is the ones vector, any (bounded) complex QCQP may be rewritten with a decision vector with only positive components.

Since $Y = yy^*$ in the lifted formulation of CQCQP we have $W_{ij} = w_i w_j + t_i t_j$ and $T_{ij} = t_i w_j - w_i t_j$. Thus we can assume $0 < W^- \leq W_{ij}$, where $W^- := w_i^L w_j^L + t_i^L t_j^L$. From $\text{rank}(Y) = 1$ we have that $W_{ij}^2 + T_{ij}^2 = W_{ii} W_{jj}$, and so:

$$\sqrt{\frac{U_{ii} U_{jj}}{(W^-)^2} - 1} \geq \sqrt{\frac{W_{ii} W_{jj}}{W_{ij}^2} - 1} = \frac{|T_{ij}|}{W_{ij}}.$$

Hence we have valid bounds for constraint (2c): $-L_{ij} = U_{ij} = \sqrt{U_{ii}U_{jj}/(W^-)^2 - 1}$. Moreover, by construction $W_{ij} \geq 0$ which implies constraint (2d).

Note that constraints describing \mathcal{J}_C shifted back to the original variables will include the original variables of CQCQP. For instance, suppose we set $q = x-l+e+ie$. Moreover, define the components $q := w^q + it^q$. Then we have

$$\begin{aligned} W_{ij}^q &= (w_i - w_i^L + 1)(w_j - w_j^L + 1) + (t_i - t_i^L + 1)(t_j - t_j^L + 1) \\ &= w_i w_j + t_i t_j + (1 - w_j^L)w_i + (1 - w_i^L)w_j + (1 - w_i^L)(1 - w_j^L) \\ &\quad + (1 - t_j^L)t_i + (1 - t_i^L)t_j + (1 - t_i^L)(1 - t_j^L) \\ &= W_{ij} + (1 - w_j^L)w_i + (1 - w_i^L)w_j + (1 - t_j^L)t_i + (1 - t_i^L)t_j \\ &\quad + (1 - w_i^L)(1 - w_j^L) + (1 - t_i^L)(1 - t_j^L). \end{aligned}$$

Thus a valid inequality derived for W_{ij}^q gives a valid inequality for W_{ij}, x_i, x_j .

In the special case of real QCQP, we can instead set $L_{ij} = U_{ij} = 0$, which enforces the matrix of imaginary entries T to be the zeros matrix. An affine transformation to a problem with nonnegative variables is sufficient to ensure that constraint (2d) is valid.

Constraint (2e) This constraint is implied by $Y = yy^*$, which requires Y to have rank one.

2.2 Branching on a complex matrix entry

In this section we consider branching on upper and lower bounds of Y as specified in \mathcal{J}_C , i.e. bounds on every complex matrix entry. We examine branching rules for selecting a single (i, j) entry of Y to branch on. One way to form a branching rule is to use a scoring function, where the branching option with the highest score is selected. A score can be based on the violation of relaxed constraints, or an estimate of the impact of branching on the children nodes' optimal relaxation objective value. In the standard development of the SDP relaxation the only relaxed constraint is the rank-one constraint on Y . However, the rank function is discrete and applies globally to all variables of the decision matrix, which seems problematic for use in variable branching. Therefore, we consider an alternative to the rank-one condition:

Proposition 6 *For $n > 1$ a nonzero Hermitian positive semidefinite $n \times n$ matrix Y has rank one iff all of its 2×2 principal minors are zero.*

Proof Suppose Y has rank $r > 1$. Since Y is Hermitian it has an $r \times r$ nonzero principal minor. Since Y is positive semidefinite this principal minor corresponds to a positive definite $r \times r$ submatrix. As $r \geq 2$, this implies there exists a 2×2 strictly positive principal minor. Now suppose instead that Y has a strictly positive 2×2 principal minor. Then Y contains a rank-two principal submatrix and thus $r > 1$. □

We will use the equivalent condition that the minimum eigenvalue of each 2×2 principal submatrix of Y be zero. Recall that by the Hermitian property of Y we have

for any i, j that $W_{ij} = W_{ji}, T_{ij} = -T_{ji}$ and thus $T_{ii} = 0$. Algebraically the minimum eigenvalue condition can be expressed as

$$\lambda_{\min} = \frac{1}{2} \left(W_{ii} + W_{jj} - \|(W_{ii} - W_{jj}, 2W_{ij}, 2T_{ij})\| \right).$$

We branch by partitioning the range $[L_{ij}, U_{ij}]$ for some (i, j) via updating the bounds as:

$$\begin{aligned} L'_{ij} &\leftarrow \alpha L_{ij} + (1 - \alpha)U_{ij} \\ U'_{ij} &\leftarrow \alpha L_{ij} + (1 - \alpha)U_{ij} \end{aligned}$$

where $\alpha \in (0, 1)$ is a parameter. In our implementation we use the bisection rule, i.e. set $\alpha = 0.5$. We will refer to the assignment $L'_{ij} \leftarrow \frac{L_{ij}+U_{ij}}{2}$ as the up branch with respect to a matrix entry, and similarly down branch refers to the assignment on the upper bound. Now let us consider rules for selecting an entry to branch on.

Most violated with strong branching (MVSb) Let c be a pair of indices (i, j) for $i \neq j$. From Proposition 6 it follows that for CSDP $\text{rank}(Y) \in \{0, 1\}$ iff $\lambda_{\min}(Y_c) = 0$ for all pairs. Select the 2×2 principal submatrix of Y with the greatest minimum eigenvalue, and let c^* be the indices of this submatrix. Given c^* , there are three possible candidate entries. These are evaluated by strong branching: for each entry we will solve the up branch problem and obtain the solution matrix Y^+ , and likewise Y^- from the down branch. The following score function is used:

$$\mu \max\{-\lambda_{\min}(Y_{c^*}^+), -\lambda_{\min}(Y_{c^*}^-)\} + (1 - \mu) \min\{-\lambda_{\min}(Y_{c^*}^+), -\lambda_{\min}(Y_{c^*}^-)\},$$

where $\mu \in [0, 1]$ is a tuning parameter; we follow the example of COUENNE [9] and use a value of 0.15. The entry with the highest score is selected for branching.

Most violated with worst-case bounds (MVWB) Since strong branching is computationally expensive, we consider solving a simpler subproblem to produce a score. Consider the Worst-Case Eigenvalue (WEV) problem of finding the greatest minimum eigenvalue that can be obtained within $\text{conv}(\mathcal{J}_C)$:

$$\begin{aligned} \text{(WEV)} \quad \max \lambda \text{ s.t.: } &\|(W_{ii} - W_{jj}, 2W_{ij}, 2T_{ij})\| \\ &\leq W_{ii} + W_{jj} - 2\lambda, \quad (2a) - (2c), \quad (3a) - (3b). \end{aligned}$$

Note that we dropped the positive semidefinite condition on X since the objective maximizes the minimum eigenvalue. We solve WEV in lieu of solving the children problems Y^+ and Y^- . Thus, overestimates of $\lambda_{\min}(Y_{c^*}^+)$ and $\lambda_{\min}(Y_{c^*}^-)$ are used in the score function. MVWB is otherwise the same as MVSb.

Reliability branching with entry bounds (RBEB) Since MVSb and MVWB rely on a particular violation metric, for benchmarking purposes we consider a method that is agnostic to the measure of violation. RBEB is an application of the *rb-int-br* rule of Belotti et al. [9].

Reliability branching uses pseudocosts, which capture information about previous branching decisions. Let Φ^+ and Φ^- be matrices containing pseudocosts, estimating the improvement in objective value by branching up or down, respectively. If, at search tree node k , L_{ij}^k is selected for branching up and the objective improves by δ^k in the child node's relaxation, then let $D^k := \delta^k / (\frac{U_{ij}^k - L_{ij}^k}{2} - L_{ij}^k)$ be the per-unit improvement. Φ^+ is the running average of all D^k for up branches, and Φ^- is the running average for down branches.

For a given candidate (i, j) , the following score function is used:

$$\left(\mu \max\{\Phi_{ij}^+, \Phi_{ij}^-\} + (1 - \mu) \min\{\Phi_{ij}^+, \Phi_{ij}^-\} \right) \left(\frac{U_{ij} - L_{ij}}{2} \right).$$

As with MVSB and MVWB, we use a value of $\mu = 0.15$. The candidate with the highest score is selected for branching. Note that we can restrict the set of candidate entries to those with violation, i.e. corresponding to members of 2×2 principal submatrices with strictly positive minimum eigenvalue.

In reliability branching, strong branching is used in lieu of pseudocosts until η evaluations have been performed on a given up or down branch; η is called the reliability parameter [1]. In computational experiments we report results for $\eta = 1$; $\eta = 4$ gave similar but poorer performance due to the computational burden of strong branching on large instances.

2.3 Bound tightening

It is well recognized that reducing the domain of a problem by tightening the bounds of the variables can improve the performance of branching methods substantially. In this subsection we describe two fast procedures to tighten the bounds L_{ij}, U_{ij} . In a companion applications paper [17], we apply these procedures along with others that exploit the special structure of the ACOFP problem.

Tightening with a quadratic inequality Let us consider the following two variable problem:

$$\begin{aligned} aq^2 + qy + c &\leq 0, \\ \ell_y &\leq y \leq u_y. \end{aligned}$$

Here a and c are parameters, and q and y are real-valued variables. Given variable bounds on y , we would like to infer variable bounds on q . With appropriate transformations, a variety of quadratic constraints of QCQP can be put in the simple form above. For instance, one can convert a complex quadratic inequality into a real quadratic inequality in terms of the real and imaginary variable components. Now any real quadratic constraint $q^T Dq + b^T q + c \leq 0$ may be written in the form $d_{ii}q_i^2 + q_i \sum_{j \neq i} d_{ij}q_j + \sum_{j \neq i} d_{jj}q_j^2 + b^T q + c \leq 0$. If all variables are bounded, then $\sum_{j \neq i} d_{ij}q_j$ may be treated as a single bounded variable and $b^T q + c$ can be

fixed to its minimum value, which allows us to apply the proposed bound tightening structure. One can also apply the same principles to derive bounds on the magnitude of x_i , and thereby derive bounds for W_{ii} (see [17] for details).

By adding a slack variable s , let us write the two variable quadratic constraint as an equality:

$$\begin{aligned} aq^2 + qy + c + s &= 0, \\ s &\geq 0, \\ \ell_y &\leq y \leq u_y. \end{aligned}$$

Solving the quadratic equation for q we have:

$$q = \frac{-y \pm \sqrt{y^2 - 4ac - 4as}}{2a}. \tag{15}$$

From here, it is possible to find the maximum and minimum values of the right-hand side of equality (15) with respect to y, s , given their bounds. We can thus infer lower and upper bounds on q ; and if the lower bound is greater than the upper bound, then infeasibility can be inferred.

Tightening on cycles We can tighten the off-diagonal entry bounds, $L_{ij}, U_{ij}, i \neq j$ based on the simple principle that the sum of differences around a cycle must equal to zero. Denote the difference of some variables of $x, \delta_{ij} := x_i - x_j$. Given some cycle of indices, say $\{1, 2, 3\}$, we have $\delta_{12} + \delta_{23} + \delta_{31} = 0$.

To interpret the off-diagonal terms as difference of variables, it will be convenient to reformulate CQCQP in polar coordinates. For any complex variable x_i , we may replace the real and imaginary components with the complex angle θ_i and the magnitude $|x_i|$, where $\text{Re}(x_i) = |x_i| \cos(\theta_i), \text{Im}(x_i) = |x_i| \sin(\theta_i)$. Then we have:

$$\begin{aligned} W_{ij} &= \text{Re}(x_i x_j) = |x_i| |x_j| \cos(\theta_i - \theta_j), \\ T_{ij} &= \text{Im}(x_i x_j) = |x_i| |x_j| \sin(\theta_i - \theta_j). \end{aligned}$$

Therefore, constraint (2c) implies the following:

$$\begin{aligned} L_{ij} W_{ij} &\leq T_{ij} \leq U_{ij} W_{ij} \\ \implies L_{ij} |x_i| |x_j| \cos(\theta_i - \theta_j) &\leq |x_i| |x_j| \sin(\theta_i - \theta_j) \leq U_{ij} |x_i| |x_j| \cos(\theta_i - \theta_j) \\ \implies L_{ij} &\leq \tan(\theta_i - \theta_j) \leq U_{ij}. \end{aligned}$$

The last implication is due to the implicit nonnegativity of W_{ij} per constraint (2c). Using the fact that the arctangent is an increasing function, we can now apply the cycle rule to infer new bounds by fixing all but one variable at variable bounds. For instance, for the cycle $\{1, 2, 3\}$ we can see if L_{12} can be tightened:

$$\theta_1 - \theta_2 + \theta_2 - \theta_3 + \theta_3 - \theta_1 = 0$$

$$\implies \arctan(L_{12}) + \arctan(U_{23}) + \arctan(U_{31}) \geq 0.$$

For instance, if $U_{23} = 0.25$, $U_{31} = 0.5$, then $L_{12} \geq -\tan(\arctan(0.5) + \arctan(0.25)) = -\frac{6}{7}$.

3 Computational results and analysis

In this section we present the results of experiments on solving ACOPF and BoxQP problems using the spacial branch-and-cut approach described in the previous section. The experiments test the effect of using different relaxations—standard SDP, SDP+RLT inequalities (10a)–(10d), SDP+complex valid inequalities (3a)–(3b)—as well as the proposed branching rules MVSB, MVWB and the benchmark reliability rule RBEB on both RQCQP and CQCQP instances. For ACOPF RLT inequalities were generated with the same complex-to-reals transformation as used for inequalities (14a)–(14h). Results are summarized in this paper, and instance-specific data is provided at <https://sites.google.com/site/cchenresearch/>.

All experiments herein are conducted with a 3.2 GHz quad-core Intel i5-4460 CPU processor and 8 GB main memory. Algorithms are implemented using MATLAB [47] with model processing performed by YALMIP [33]. Conic programs are solved with MOSEK version 7.1 [2]. IPOPT version 3.11.1 [48] is used as a local solver to obtain primal feasible solutions to CQCQP at each search tree node.

All spatial branch-and-cut (SBC) configurations are implemented with a depth-first search node selection rule. The search termination criteria are: an explored nodes limit of 10,000, a time limit of 1.5 h, and a relative optimality gap limit. A search tree depth of 100 is applied, pruning all children nodes past this limit. The optimality gap is calculated using the global upper bound (gub) and global lower bound (glb): $gap = 1 - (gub - glb)/|gub|$.

3.1 Problem instances

ACOPF The ACOPF problem is a power generation scheduling problem that can be formulated as CQCQP. The problem formulation can be found in “Appendix 2” for a thorough treatment on optimization issues related to optimal power flow we refer the reader to Bienstock [12].

Our experiments include the test cases of Gopalakrishnan et al. [22]. Small duality gaps were reported for these cases, so the root relaxation is known to provide a good lower bound. These instances are named $g9$, $g14$, $g30$, and $g57$, where the number indicates the number of buses in the problem. We include the modified IEEE test cases from Chen et al. [17], which are named 9Na, 9Nb, 14S, 14P, and 118IN. We also use the NESTA instances [19] listed in Table 1. Here SDP_{gap} indicates the percentage gap between the standard SDP relaxation and the best known upper bound. NESTA instances with trivial gap ($< 0.1\%$) or with more than 1000 buses are excluded. The latter criterion excludes two large instances that are too challenging for the SDP solver to handle even at the root node.

Table 1 NESTA instances

name	SDP gap (%)
case3_lmbd	0.39
case3_lmbd_api	1.26
case3_lmbd_sad	2.06
case5_pjm	5.22
case24_ieee_rts_api	1.45
case24_ieee_rts_sad	6.05
case29_edin_sad	28.44
case73_ieee_rts_api	4.1
case73_ieee_rts_sad	4.29
case30_as_sad	0.47
case30_fsr_api	11.06
case89_pegase_api	18.11
case118_ieee_api	31.5
case118_ieee_sad	7.57
case162_ieee_dtc	1.08
case162_ieee_dtc_api	0.85
case162_ieee_dtc_sad	3.65
case189_edin_sad	1.2

Since the standard IEEE test cases do not include phase angle differences, we have applied a 30° bound across all connected buses if not otherwise specified. We also use a sparse formulation of CSDP that replaces the PSD constraint with multiple positive semidefinite constraints on submatrices and linear equality constraints. Fukuda et al. [21] developed the methodology for sparse formulation of a generic SDP, and several authors [5, 13, 27, 38] have studied its effects in improving solution times for ACOPF. We follow this methodology, using a minimum-degree ordering heuristic on the admittance matrix and finding a corresponding symbolic Cholesky decomposition in order to determine a suitable clique decomposition. In “Appendix 1” we show that enforcing the rank constraint on each submatrix is sufficient to ensure equivalence between CQCQP and the sparse version of CSDP. Hence valid inequalities and branching rules are applied only to 2×2 principal minors that have been kept after sparse decomposition.

All instances were solved using an optimality criterion of 0.1%. Bound tightening is activated on all instances (see [17] for its effects).

BoxQP The BoxQP problem is formulated as $\min \frac{1}{2}x'Qx + f'x : 0 \leq x \leq 1$, where $x \in \mathbb{R}^n$ is decision vector, and $f \in \mathbb{R}^n$, $Q \in \mathbb{R}^{n \times n}$ are data. We use the 90 BoxQP instances of Burer and Vandembussche [16]. The instances are named *sparAAA-BBB-C*, where *AAA* is the dimension of x , *BBB* is the density of Q , and *C* is the random seed number. We set an optimality gap limit of 0.01% for these instances. The relaxation used for these instances is the SDP relaxation strengthened with RLT inequalities.

3.2 Results

Table 2 shows average the performance of different convex relaxations and branching rules for the ACOPF; the averages are taken over instances solved by a given configuration. The first column `relax` shows the relaxation used to solve the problem: CVI refers to CSDP together with complex valid inequalities (3a)–(3b); RLT refers to the SDP relaxation of the real QCQP formulation together with RLT inequalities; and SDP refers the standard SDP relaxation without additional valid inequalities. The column `rule` shows the branching rule used, `nodes` the number of search tree nodes explored before termination, `depth` the maximum search tree depth, `lbttime` the time in seconds spent solving relaxations, `ubtime` the time in seconds spent obtaining primal solutions, `time` the total time spent in seconds by the SBC algorithm, and finally `solved` the total proportion of solved instances.

The formulation CVI with the complex valid inequalities (3a)–(3b) leads to the best performance. The new branching rules MVWB and MVSB designed for rank-one constraints perform significantly better compared to the benchmark reliability rule RBEB. Without using the complex valid inequalities only two of the 26 instances are solved with the RLT formulation and MVWB branching. The plain SDP formulation without using any cuts do not converge for any of the instances. Only the best rule MVWB is presented for relaxations RLT and SDP for brevity; similar or worse results are obtained with MVSB and RBEB.

For a more detailed view, the computational results are also summarized with performance profiles [39]; we use the standard \log_2 base on the x axis. The graph represents the proportion of instances for which a given configuration was within x times the best configuration. For instance, for a profile of time to solve, a point $(x, y) = (2, 0.5)$ indicates that for 50% of cases (the y axis) a spatial branch-and-cut configuration achieves at most two times the best time (the x axis) among all configurations tested.

For ACOPF we present performance profiles for time to solve in Fig. 2 and total search tree nodes in Fig. 3. Using CVI, MVWB shows consistently better results over the other branching rules, and both MVWB and MVSB compare favourably to RBEB. MVSB used smaller search tree when it converged, although MVWB resulted in more convergent instances. Both violation strategies tend to produce smaller search trees compared to the reliability rule. The complex relaxation CVI produces substantially better results than the RQCQP approach represented by RLT. The addition of the valid inequalities led to faster convergence, whereas with a plain SDP relaxation no practical

Table 2 ACOPF performance summary

relax	rule	nodes	depth	lbttime	ubtime	time	solved
CVI	MVWB	1459.5	33.5	107.4	169.5	382.1	15/26
CVI	MVSB	804.6	21.3	128	149	734.3	12/26
CVI	RBEB	4450.8	62.4	555.9	1140.7	2077.5	7/26
RLT	MVWB	94	17.5	2	6	9	2/26
SDP	MVWB	–	–	–	–	–	0/26

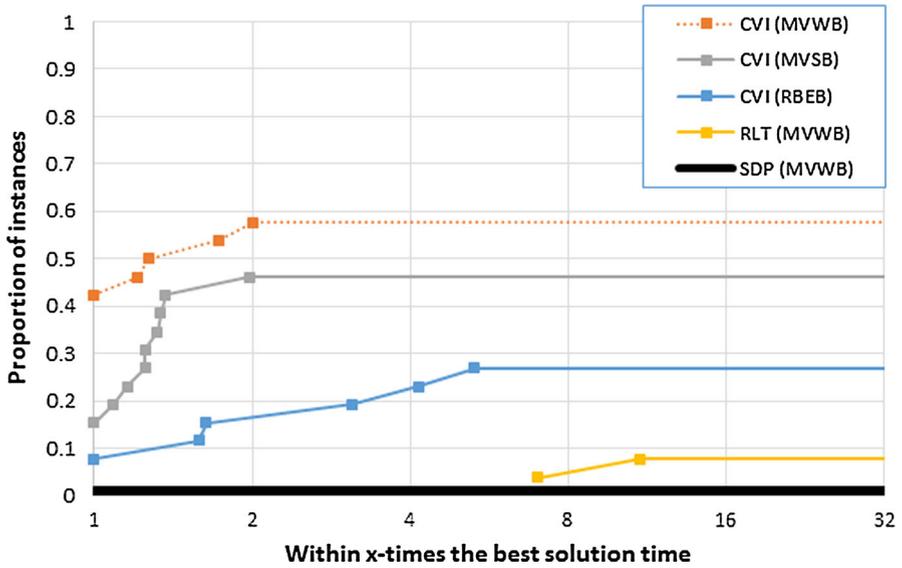


Fig. 2 Performance profile of solve times using different relaxations and branching rules for ACOFP

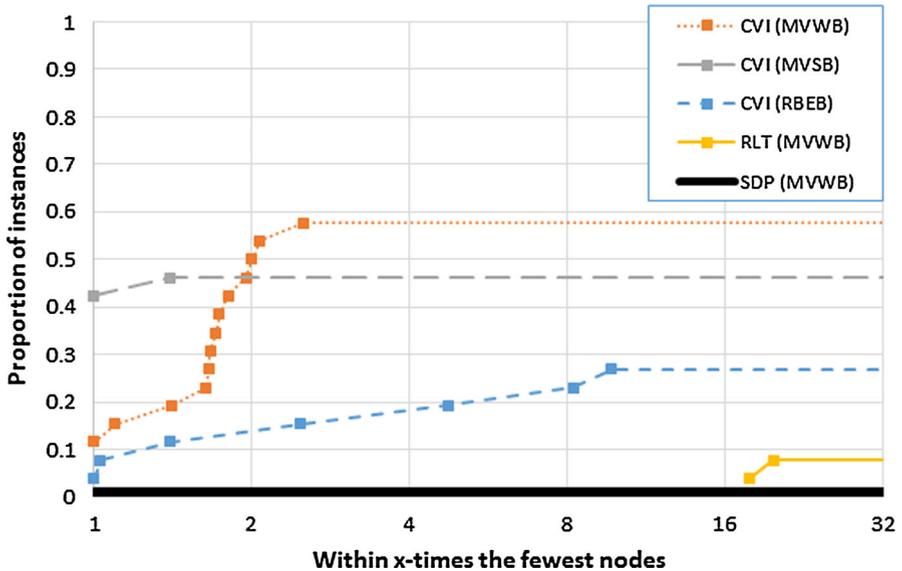


Fig. 3 Performance profile of search tree nodes using different relaxations and branching rules for ACOFP

convergence is observed. These results clearly demonstrate the benefits of exploiting the structure in the complex formulation. The stronger complex inequalities and the new branching rules based on the violation of the rank-one constraints lead to a faster convergence.

Table 3 BoxQP performance summary

rule	nodes	depth	lbttime	ubtime	time	solved
MVWB	24.4	3.7	382.0	4.1	386.3	81/90
MVSB	9.9	2.7	125.1	1.4	467.6	80/90
RBEB	26.8	10.1	173.1	4.3	1121.1	72/90

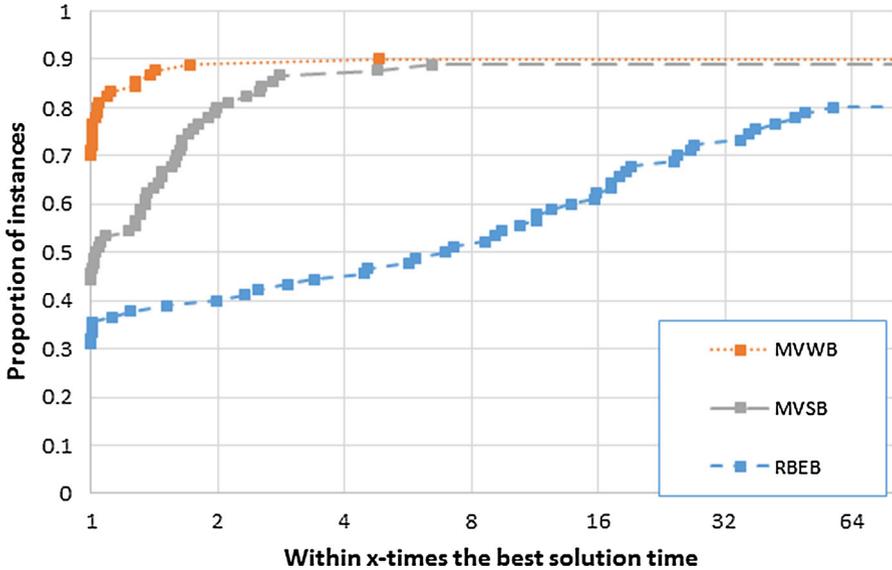


Fig. 4 Performance profile of solve times for different branching rules on BoxQP

Table 3 shows a comparison of the branching rules for BoxQP. For BoxQP the lower bound times are substantially higher than the upper bound times due to the quadratic increase in variables in the lifted relaxation. The proportion of lower and upper bound solve times to total time indicates the overhead cost of strong branching: RBEB (16%) MVSB (27%), MVWB (>99%). With the new branching rules over 80 of the 90 instances are solved; whereas with the reliability rule RBEB 72 instances are solved.

The performance profiles provide a more detailed view on the BoxQP results. Figure 4 shows that MVWB was the best method in terms of solution times, and both MVWB and MVSB performed substantially better than the reliability rule RBEB. Figure 5 indicates that MVSB tends to produce the smallest search trees, demonstrating the power of the strong branching. Both MVWB and MVSB lead to significantly smaller search trees compared to the reliability rule RBEB. The results on BoxQP instances demonstrate that even for real QCQPs the branching rules MVWB and MVSB exploiting the rank-one constraint can be very effective. The results are competitive compared to published results of Misener et al. [37], which report results from global solvers GloMIQO [36], BARON [42], and Couenne [10]. Burer and Vandembussche [16]

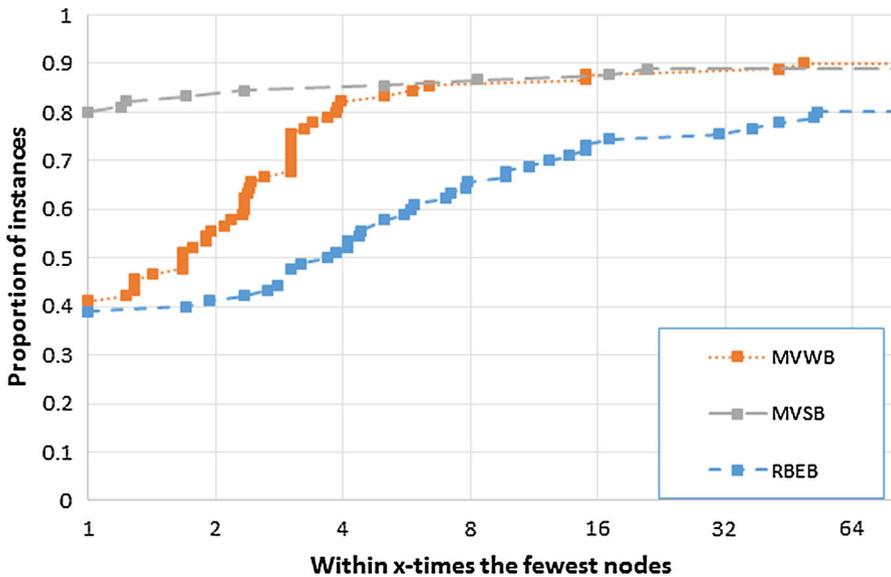


Fig. 5 Performance profile of search tree nodes with respect to branching rules on BoxQP

developed an exact SDP-based branch-and-bound method for nonconvex quadratic programming problems based on KKT conditions. Substantial improvements in performance based on completely positive programming are reported by Burer [15] and Chen and Burer [18].

4 Conclusion

We developed a spatial branch-and-cut method for generic Quadratically-Constrained Quadratic Programs with bounded complex variables. We derived valid inequalities from the convex hull description of nonconvex rank-one restricted sets to strengthen the SDP relaxations. We gave a new branching method based on an alternative characterization of a rank-one constraint. Experiments on alternating current optimal power flow problems show the valid inequalities derived from the complex formulation are critical for improving the performance of the algorithm. The proposed branching methods based on the rank-one constraint resulted in better performance compared to the benchmark reliability branching method. Tests on box-constrained nonconvex Quadratic Programming instances suggest that the violation-based branching methods may also be effective for problems with real variables.

Acknowledgements The authors would like to thank Dr. Richard P. O'Neill of the Federal Energy Regulatory Commission for the initial impetus to study conic relaxations of ACOPF, and for helpful comments in early drafts of the paper. This research has been supported, in part, by Federal Energy Regulatory Commission and by Grant FA9550-10-1-0168 from the Office of Assistant Secretary Defense for Research and Engineering. Chen Chen was supported, in part, by a NSERC PGS-D fellowship.

Appendix 1: Sufficiency of sparse valid inequalities

Consider a Hermitian matrix X with spectral decomposition $X = \sum_{k=1}^N \lambda_k d_k d_k^*$, where the eigenvalues are ordered so that $\lambda_k \geq \lambda_{k+1}$. Unlike in the real symmetric case, if λ_k has multiplicity 1, then the eigenvector d_k is only unique up to rotation by a complex phase $e^{j\theta_k}$ [46, pp. 41]. In polar coordinates (as used commonly in ACOFP) we have that the eigenvector is unique up to scaling of all phase angles by the same degree (preserving angle differences). That is, if $d_{ki} = |d_{ki}|(\cos(\theta_{ki}) + j \sin(\theta_{ki}))$, then we can add $\delta \in \mathbb{R}$ to all angles and replace d_k in the eigenbasis. In terms of rectangular coordinates (i.e. real and imaginary components), we can state that d_k can be replaced in the eigenbasis with

$$\text{Re}(d_k) + \delta \mathbf{1} + j(\delta^I - \text{Im}(d_k)),$$

for any $\delta \in \mathbb{R}$ such that $(\text{Re}(d_{ki}) + \delta)^2 \leq |d_{ki}|^2 \forall i$ and $\delta^I \in \mathbb{R}^N$ with entries $\delta_i^I = \sqrt{|d_{ki}|^2 - (\delta + \text{Re}(d_{ki}))^2}$.

Sparse positive semidefinite decomposition of a Hermitian matrix $X \in \mathbb{H}^{N \times N}$ yields a set of index sets C , where $X_c \geq 0 \forall c \in C \iff X \succeq 0$. Note that $\bigcap_{c \in C} c = \{1, \dots, N\}$. A property of sparse decomposition is that C can be represented with an acyclic graph where each node is an element c of C and an edge between two nodes indicates that at least one index is shared between the corresponding index sets; this is known as a clique tree [21, 23]. The clique tree of a chordal graph can be constructed in time and space linear with respect to the number of edges [14]. A matrix can be completed for a certain property if there exist values for the entries not specified by the clique tree such that the fully specified matrix can attain the property.

Proposition 7 $X_c \geq 0, \text{rank}(X_c) \leq 1 \forall c \in C$ iff X can be completed so that $X = xx^*$ for some $x \in \mathbb{C}^N$.

Proof If $X = xx^*$, then $\text{rank}(X) \leq 1, X \succeq 0$ and so one direction is obvious. Now consider the other direction: suppose that $X_c \geq 0, \text{rank}(X_c) \leq 1 \forall c \in C$. We will use a constructive proof, i.e. we shall construct an x so that $(xx^*)_c = X_c \forall c \in C$.

Let us consider the clique tree corresponding to the chordal graph formed by C . Label a terminal node c_1 . Since X_{c_1} has rank one and is positive semidefinite, we have that $X_{c_1} = \lambda_1^{c_1} (d_1^{c_1})^* d_1^{c_1}$, and so we can set $x_{c_1} = \sqrt{\lambda_1^{c_1}} d_1^{c_1}$ for any normalized principal eigenvector $d_1^{c_1}$. Now denote a neighbouring node c_2 and consider its corresponding index set with some normalized principal eigenvector, $d_2^{c_2}$. By clique tree property, X_{c_1} and X_{c_2} share at least one entry, say X_{mm} , so $|(d_1^{c_1})_m| = |(d_2^{c_2})_m|$. Since eigenvectors of the eigenbasis are only unique up to rotation by complex phase, d_2 can be rotated to form \hat{d}_2 so that $(d_1^{c_1})_m = (\hat{d}_2^{c_2})_m$, i.e. rotating the eigenvector so that one entry attains a specific angle. Then we can set $x_{c_2} = \sqrt{\lambda_1^{c_2}} \hat{d}_2$, where x_{mm} , the shared entry of x_{c_1}, x_{c_2} retains the same value. The remaining elements of x can be found by proceeding through neighbours in the same manner, with the acyclic property ensuring that each element of x is set once. \square

This relies on a generalization of the fact that in ACOPF and in load flow the bus angles of any solution can be scaled up or down by constants. From the proposition it immediately follows that in the alternative rank condition only the 2×2 principal minors related to the submatrices X_c need to be considered, and so valid inequalities (3a) and (3b) can be applied in a sparse fashion.

Appendix 2: ACOPF formulation

ACOPF can be written in the form of CQCQP [31]; we state a lifted formulation of the ACOPF problem as follows:

$$\begin{aligned}
 \min \quad & c'_2[P + D_P]^2 + c'_1(P + D_P) + c_0 & (16a) \\
 \text{s.t.} \quad & P = \text{diag}(GW - BT) & (16b) \\
 & Q = \text{diag}(-BW - GT) & (16c) \\
 & P^{\min} \leq P + D_P \leq P^{\max} & (16d) \\
 & Q^{\min} \leq Q + D_Q \leq P^{\max} & (16e) \\
 & [V^{\min}]^2 \leq \text{diag}(W) \leq [V^{\max}]^2 & (16f) \\
 \text{(LACOPF)} \quad & \tan(\theta^{\min})W_{ij} \leq T_{ij} \leq \tan(\theta^{\max})W_{ij} & (16g) \\
 & P_f = \text{diag}(C_f(G_fW - B_fT)) & (16h) \\
 & Q_f = \text{diag}(C_f(-B_fW - G_fT)) & (16i) \\
 & P_t = \text{diag}(C_t(G_tW - B_tT)) & (16j) \\
 & Q_t = \text{diag}(C_t(-B_tW - G_tT)) & (16k) \\
 & [P_f]^2 + [Q_f]^2 \leq [S^{\max}]^2 & (16l) \\
 & [P_t]^2 + [Q_t]^2 \leq [S^{\max}]^2 & (16m) \\
 & W + \iota T \geq 0 & (16n) \\
 & \text{rank}(W + \iota T) = 1 & (16o)
 \end{aligned}$$

Let n be the number of nodes in the graph of the problem, with nodes representing either buses or transformers, and let k be the number of edges (aka branches).

Variables and data The decision variables used in LACOPF are: nodal powers $P + jQ \in \mathbb{C}^n$; a Hermitian decision matrix representing the outer product of nodal voltages, $W + \iota T \in \mathbb{H}^{n \times n}$; and power to and from buses (respectively) across branches, $P_f + jQ_f, P_t + jQ_t \in \mathbb{C}^k$. All other parameters are fixed data: convex costs $c_0 \in \mathbb{R}, c_1 \in \mathbb{R}^N, c_2 \in \mathbb{R}_+^N$; load, $D_P + jD_Q \in \mathbb{C}^n$; admittance matrices, $Y \in \mathbb{C}^{n \times n}, Y_f, Y_t \in \mathbb{C}^{k \times n}$; voltage magnitude limits, $V^{\min}, V^{\max} \in \mathbb{R}^n$; phase angle limits $\theta^{\min}, \theta^{\max} \in \mathbb{R}^k$; generator limits, $P^{\min} + jQ^{\min}, P^{\max} + jQ^{\max} \in \mathbb{C}^n$; and line limits, $S^{\max} \in \mathbb{R}_{++}^n$. $Y := G + jB$ is the bus admittance matrix, and $Y_f := G_f + jB_f, Y_t := G_t + jB_t$ are branch admittances corresponding to ‘from’ and ‘to’ nodes, respectively. Admittance is composed of conductance G and susceptance

B. For a branch r from m to n , the (r, m) entry of $C_f \in \mathbb{R}^{k \times n}$ and the (r, n) entry of $C_l \in \mathbb{R}^{k \times n}$ are 1; all unconnected entries in those matrices are 0 [50].

Objective and constraints The objective is to minimize the cost of real power generation, where $P + P^D$ is the net generation of real nodal power. Constraints (16b) and (16c) are the power flow equations, relating nodal power to nodal voltage. Constraints (16d) and (16e) model demand and generation limits. Constraint (16f) bounds voltage magnitudes. Constraint (16g) bounds phase angle differences between buses. Constraints (16h) to (16k) are the branch power flow equations. Constraints (16l) and (16m) constrain apparent power across lines. We will also consider other types of line limits, but omit them here for brevity. Constraints (16n) and (16o) ensure that $W + iT$ can represent the outer product of nodal voltages [31].

SDP relaxation In LACOPF only the rank constraint (16o) is nonconvex, and dropping it gives a convex relaxation (RACOPF). This primal relaxation approach was first applied to ACOPF by Bai et al. [6] and the conic dual formulation was first considered by Lavaei and Low [31]. Note that in ACOPF the bounds $L_{ij}, U_{ij}, i \neq j$ are specified as angle bounds in polar coordinates, i.e. $L_{ij} = \tan(\theta_{ij}^{\min}), U_{ij} = \tan(\theta_{ij}^{\max})$.

References

1. Achterberg, T., Koch, T., Martin, A.: Branching rules revisited. *Oper. Res. Lett.* **33**, 42–54 (2005)
2. Andersen, E.D., Andersen, K.D.: The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. *High Perform. Optim.* **33**, 197–232 (2000)
3. Anstreicher, K.M.: Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. *J. Global Optim.* **43**, 471–484 (2009)
4. Anstreicher, K.M., Burer, S.: Computable representations for convex hulls of low-dimensional quadratic forms. *Math. Program.* **124**, 33–43 (2010)
5. Bai, X., Wei, H.: A semidefinite programming method with graph partitioning technique for optimal power flow problems. *Int. J. Electr. Power Energy Syst.* **33**, 1309–1314 (2011)
6. Bai, X., Wei, H., Fujisawa, K., Wang, Y.: Semidefinite programming for optimal power flow problems. *Int. J. Electr. Power Energy Syst.* **30**, 383–392 (2008)
7. Bao, X., Sahinidis, N.V., Tawarmalani, M.: Multiterm polyhedral relaxations for nonconvex, quadratically constrained quadratic programs. *Optim. Methods Softw.* **24**, 485–504 (2009)
8. Beck, A., Eldar, Y.C.: Strong duality in nonconvex quadratic optimization with two quadratic constraints. *SIAM J. Optim.* **17**, 844–860 (2006)
9. Belotti, P., Kirches, C., Leyffer, S., Linderoth, J., Luedtke, J., Mahajan, A.: Mixed-integer nonlinear optimization. *Acta Numer.* **22**, 1–131 (2013)
10. Belotti, P., Lee, J., Liberti, L., Margot, F., Wächter, A.: Branching and bounds tightening techniques for non-convex MINLP. *Optim. Methods Softw.* **24**, 597–634 (2009)
11. Ben-Tal, A., Nemirovski, A., Roos, C.: Extended matrix cube theorems with applications to μ -theory in control. *Math. Oper. Res.* **28**, 497–523 (2003)
12. Bienstock, D.: *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*, vol. 22. SIAM, Philadelphia (2016)
13. Bienstock, D., Munoz, G.: LP Approximations to Mixed-Integer Polynomial Optimization Problems. [arXiv:1501.00288](https://arxiv.org/abs/1501.00288) (2015)
14. Blair, J.R.S., Peyton, B.: An introduction to chordal graphs and clique trees. In: George, A., Gilbert, J.R., Liu, J.W.H. (eds.) *Graph Theory and Sparse Matrix Computation*, pp. 1–29. Springer, New York (1993)
15. Burer, S.: Optimizing a polyhedral-semidefinite relaxation of completely positive programs. *Math. Program. Comput.* **2**, 1–19 (2010)
16. Burer, S., Vandenbussche, D.: Globally solving box-constrained nonconvex quadratic programs with semidefinite-based finite branch-and-bound. *Comput. Optim. Appl.* **43**, 181–195 (2009)

17. Chen, C., Atamtürk, A., Oren, S.S.: Bound tightening for the alternating current optimal power flow problem. *IEEE Trans. Power Syst.* (2015). doi:[10.1109/TPWRS.2015.2497160](https://doi.org/10.1109/TPWRS.2015.2497160)
18. Chen, J., Burer, S.: Globally solving nonconvex quadratic programming problems via completely positive programming. *Math. Program. Comput.* **4**, 33–52 (2012)
19. Coffrin, C., Gordon, D., Scott, P.: NESTA, The NICTA Energy System Test Case Archive. [arXiv:1411.0359](https://arxiv.org/abs/1411.0359) (2014)
20. De Maio, A., Huang, Y., Piezzo, M., Zhang, S., Farina, A.: Design of optimized radar codes with a peak to average power ratio constraint. *IEEE Trans. Signal Process.* **59**, 2683–2697 (2011)
21. Fukuda, M., Kojima, M., Murota, K., Nakata, K.: Exploiting sparsity in semidefinite programming via matrix completion I: general framework. *SIAM J. Optim.* **11**, 647–674 (2001)
22. Gopalakrishnan, A., Raghunathan, A.U., Nikovski, D., Biegler, L.T.: Global optimization of optimal power flow using a branch & bound algorithm. In: *IEEE 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 609–616 (2012)
23. Grone, R., Johnson, C.R., Sá, E.M., Wolkowicz, H.: Positive definite completions of partial Hermitian matrices. *Linear Algebra Appl.* **58**, 109–124 (1984)
24. Huang, Y., Palomar, D.P.: Randomized algorithms for optimal solutions of double-sided QCQP with applications in signal processing. *IEEE Trans. Signal Process.* **62**, 1093–1108 (2014)
25. Huang, Y., Zhang, S.: Complex matrix decomposition and quadratic programming. *Math. Oper. Res.* **32**, 758–768 (2007)
26. Jabr, R.A.: Optimal power flow using an extended conic quadratic formulation. *IEEE Trans. Power Syst.* **23**, 1000–1008 (2008)
27. Jabr, R.A.: Exploiting sparsity in SDP relaxations of the OPF problem. *IEEE Trans. Power Syst.* **27**, 1138–1139 (2012)
28. Jiang, B., Li, Z., Zhang, S.: Approximation methods for complex polynomial optimization. *Comput. Optim. Appl.* **59**, 219–248 (2014)
29. Josz, C., Molzahn, D.K.: Moment/Sum-of-Squares Hierarchy for Complex Polynomial Optimization. [arXiv:1508.02068](https://arxiv.org/abs/1508.02068) (2015)
30. Kocuk, B., Dey, S.S., Sun, X.A.: Inexactness of SDP relaxation and valid inequalities for optimal power flow. *IEEE Trans. Power Syst.* **31**, 642–651 (2016)
31. Lavaei, J., Low, S.H.: Zero duality gap in optimal power flow problem. *IEEE Trans. Power Syst.* **27**, 92–107 (2012)
32. Linderoth, J.: A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Math. Program.* **103**, 251–282 (2005)
33. Lofberg, J.: YALMIP: a toolbox for modeling and optimization in MATLAB. In: *IEEE International Symposium on Computer Aided Control Systems Design*, pp. 284–289 (2004)
34. Lovász, L., Schrijver, A.: Cones of matrices and set-functions and 0–1 optimization. *SIAM J. Optim.* **1**, 166–190 (1991)
35. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: part I convex underestimating problems. *Math. Program.* **10**, 147–175 (1976)
36. Misener, R., Floudas, C.A.: GloMIQO: global mixed-integer quadratic optimizer. *J. Global Optim.* **57**, 3–50 (2013)
37. Misener, R., Smadbeck, J.B., Floudas, C.A.: Dynamically generated cutting planes for mixed-integer quadratically constrained quadratic programs and their incorporation into GloMIQO 2. *Optim. Methods Softw.* **30**, 215–249 (2015)
38. Molzahn, D.K., Holzer, J.T., Lesieutre, B.C., DeMarco, C.L.: Implementation of a large-scale optimal power flow solver based on semidefinite programming. *IEEE Trans. Power Syst.* **28**, 1–12 (2013)
39. Petriu, D.C., Shen, H.: Applying the UML performance profile: graph grammar-based derivation of LQN models from UML specifications. In: *Field, T., Harrison, P.G., Bradley, J., Harder, U. (eds.) Computer Performance Evaluation: Modelling Techniques and Tools*, vol. 2324, pp. 159–177. Springer, Berlin (2002)
40. Phan, D.T.: Lagrangian duality-based branch and bound algorithms for optimal power flow. *Oper. Res.* **60**, 275–285 (2012)
41. Raber, U.: A simplicial branch-and-bound method for solving nonconvex all-quadratic programs. *J. Global Optim.* **13**, 417–432 (1998)
42. Sahinidis, N.V.: BARON: a general purpose global optimization software package. *J. Global Optim.* **8**, 201–205 (1996)

43. Sherali, H.D., Tuncbilek, C.H.: A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. *J. Global Optim.* **2**, 101–112 (1992)
44. Shor, N.Z.: Quadratic optimization problems. *Soviet J. Circuits Syst. Sci.* **25**, 6 (1987)
45. Sun, D.I., Ashley, B., Brewer, B., Hughes, A., Tinney, W.F.: Optimal power flow by Newton approach. *IEEE Trans. Power Appar. Syst.* **10**, 2864–2880 (1984)
46. Tao, T.: *Topics in Random Matrix Theory*, vol. 132. American Mathematical Society, Providence (2012)
47. The MathWorks. *MATLAB User's Guide* (1998)
48. Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* **106**, 25–57 (2006)
49. Waldspurger, I., D'Aspremont, A., Mallat, S.: Phase recovery, maxcut and complex semidefinite programming. *Math. Program.* **149**, 47–81 (2015)
50. Zimmerman, R.D., Murillo-Sánchez, C.E., Thomas, R.J.: MATPOWER: steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Trans. Power Syst.* **26**, 12–19 (2011)