Journal of Global Optimization (2024) 90:293–322
https://doi.org/10.1007/s10898-024-01407-y

On exact and inexact RLT and SDP-RLT relaxations of quadratic programs with box constraints

Yuzhou Qiu[1] · **E. Alper Yıldırım**[1]

Received: 1 March 2023 / Accepted: 16 May 2024 / Published online: 30 May 2024

Abstract
Quadratic programs with box constraints involve minimizing a possibly nonconvex quadratic function subject to lower and upper bounds on each variable. This is a well-known NP-hard problem that frequently arises in various applications. We focus on two convex relaxations, namely the reformulation–linearization technique (RLT) relaxation and the SDP-RLT relaxation obtained by combining the Shor relaxation with the RLT relaxation. Both relaxations yield lower bounds on the optimal value of a quadratic program with box constraints. We show that each component of each vertex of the RLT relaxation lies in the set $\{0, \frac{1}{2}, 1\}$. We present complete algebraic descriptions of the set of instances that admit exact RLT relaxations as well as those that admit exact SDP-RLT relaxations. We show that our descriptions can be converted into algorithms for efficiently constructing instances with (1) exact RLT relaxations, (2) inexact RLT relaxations, (3) exact SDP-RLT relaxations, and (4) exact SDP-RLT but inexact RLT relaxations. Our preliminary computational experiments illustrate that our algorithms are capable of generating computationally challenging instances for state-of-the-art solvers.

Keywords Quadratic programming with box constraints · Reformulation–linearization technique · Semidefinite relaxation · Convex underestimator

1 Introduction

A quadratic program with box constraints is an optimization problem in which a possibly nonconvex quadratic function is minimized subject to lower and upper bounds on each variable:

$$\text{(BoxQP)} \quad \ell^* = \min_{x \in \mathbb{R}^n} \{q(x) : x \in F\},$$

where $q : \mathbb{R}^n \to \mathbb{R}$ and $F \subseteq \mathbb{R}^n$ are respectively given by

$$q(x) = \tfrac{1}{2} x^T Q x + c^T x, \quad F = \{x \in \mathbb{R}^n : 0 \le x \le e\}.$$

✉ E. Alper Yıldırım
E.A.Yildirim@ed.ac.uk

Yuzhou Qiu
y.qiu-16@sms.ed.ac.uk

1 School of Mathematics, The University of Edinburgh, Peter Guthrie Tait Road, Edinburgh EH9 3FD, UK

Here, $e \in \mathbb{R}^n$ denotes the vector of all ones. The parameters of the problem are given by the pair $(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n$, where $\mathcal{S}^n$ denotes the set of $n \times n$ real symmetric matrices. The optimal value is denoted by $\ell^* \in \mathbb{R}$. Note that any quadratic program with finite lower and upper bounds on each variable can be easily transformed into the above form.

(BoxQP) is regarded as a "fundamental problem" in global optimization that appears in a multitude of applications (see, e.g., [1, 2]). If Q is a positive semidefinite matrix, (BoxQP) can be solved in polynomial time [3]. However, if Q is an indefinite or negative semidefinite matrix, then (BoxQP) is an NP-hard problem [4, 5]. In fact, it is even NP-hard to approximate a local minimizer of (BoxQP) [6].

1.1 RLT and SDP-RLT relaxations

By using a simple "lifting" idea, (BoxQP) can be equivalently reformulated as

$$\text{(L-BoxQP)} \quad \ell^* = \min_{(x,X) \in \mathcal{F}} \tfrac{1}{2}\langle Q, X\rangle + c^T x,$$

where $\langle A, B\rangle = \mathrm{trace}(A^T B) = \sum_{i=1}^{p}\sum_{j=1}^{q} A_{ij}B_{ij}$ for any $A \in \mathbb{R}^{p\times q}$ and $B \in \mathbb{R}^{p\times q}$, and

$$\mathcal{F} = \left\{(x, X) \in \mathbb{R}^n \times \mathcal{S}^n : 0 \le x \le e, \quad X_{ij} = x_i x_j, \quad 1 \le i \le j \le n\right\}. \tag{1}$$

Since (L-BoxQP) is an optimization problem with a linear objective function over a nonconvex feasible region, one can replace $\mathcal{F}$ by $\mathrm{conv}(\mathcal{F})$, where $\mathrm{conv}(\cdot)$ denotes the convex hull, without affecting the optimal value. Many convex relaxations of (BoxQP) arise from this reformulation by employing outer approximations of $\mathrm{conv}(\mathcal{F})$ using tractable convex sets.

A well-known relaxation of $\mathrm{conv}(\mathcal{F})$ is obtained by replacing the nonlinear equalities $X_{ij} = x_i x_j$ by the so-called McCormick inequalities [7], which gives rise to the RLT (Reformulation–Linearization Technique) relaxation of (BoxQP) (see, e.g., [8]):

$$\text{(R)} \quad \ell^*_R = \min_{(x,X)\in\mathbb{R}^n\times\mathcal{S}^n} \left\{\frac{1}{2}\langle Q, X\rangle + c^T x : (x, X) \in \mathcal{F}_R\right\},$$

where

$$\mathcal{F}_R = \left\{(x, X) \in \mathbb{R}^n \times \mathcal{S}^n : \begin{array}{rcl} 0 \le & x & \le e \\ \max\{x_i + x_j - 1, 0\} \le & X_{ij} & \le \min\{x_i, x_j\}, \quad 1 \le i \le j \le n \end{array}\right\}. \tag{2}$$

The RLT relaxation (R) of (BoxQP) can be further strengthened by adding tighter semidefinite constraints [9], giving rise to the SDP-RLT relaxation:

$$\text{(RS)} \quad \ell^*_{RS} = \min_{(x,X)\in\mathbb{R}^n\times\mathcal{S}^n} \left\{\frac{1}{2}\langle Q, X\rangle + c^T x : (x, X) \in \mathcal{F}_{RS}\right\},$$

where

$$\mathcal{F}_{RS} = \left\{(x, X) \in \mathbb{R}^n \times \mathcal{S}^n : (x, X) \in \mathcal{F}_R, \quad X - xx^T \succeq 0\right\}. \tag{3}$$

The optimal value of each of the RLT and SDP-RLT relaxations, denoted by ℓ^*_R and ℓ^*_{RS}, respectively, yields a lower bound on the optimal value of (BoxQP). The SDP-RLT relaxation is clearly at least as tight as the RLT relaxation, i.e.,

$$\ell^*_R \le \ell^*_{RS} \le \ell^*. \tag{4}$$

1.2 Motivation and contributions

Convex relaxations play a fundamental role in the design of global solution methods for nonconvex optimization problems. In particular, one of the most prominent algorithmic approaches for globally solving nonconvex optimization problems is based on a branch-and-bound framework, in which the feasible region is systematically subdivided into smaller subregions and a sequence of subproblems is solved to obtain increasingly tighter lower and upper bounds on the optimal value in each subregion. The lower bounds in such a scheme are typically obtained by solving a convex relaxation. For instance, several well-known optimization solvers such as `ANTIGONE` [10], `BARON` [11], `CPLEX` [12], and `GUROBI` [13] utilize convex relaxations for globally solving nonconvex quadratic programs.

In this paper, our main goal is to describe the set of instances of (BoxQP) that admit exact RLT relaxations (i.e., $\ell_R^* = \ell^*$) as well as those that admit exact SDP-RLT relaxations (i.e., $\ell_{RS}^* = \ell^*$). Such descriptions shed light on easier subclasses of a difficult optimization problem. In addition, relying on these descriptions, we aim to develop efficient algorithms for constructing an instance of (BoxQP) that admits an exact or inexact relaxation. In particular, we propose efficient algorithms for constructing instances with (i) exact RLT relaxations, (ii) inexact RLT relaxations, (iii) exact SDP-RLT relaxations, and (iv) exact SDP-RLT but inexact RLT relaxations. Such algorithms can be quite useful in computational experiments for generating instances of (BoxQP) for which a particular relaxation will have a predetermined exactness or inexactness guarantee.

Our contributions are as follows.

1. By utilizing the recently proposed perspective on convex underestimators induced by convex relaxations [14], we establish several useful properties of each of the two convex underestimators associated with the RLT relaxation and the SDP-RLT relaxation.
2. We present two equivalent algebraic descriptions of the set of instances of (BoxQP) that admit exact RLT relaxations. The first description arises from the analysis of the convex underestimator induced by the RLT relaxation, whereas the second description is obtained by using linear programming duality. In particular, we show that each component of each vertex of $\mathcal{F}_R$ given by (2) lies in the set $\{0, \frac{1}{2}, 1\}$.
3. By relying on the second description of the set of instances with an exact RLT relaxation, we propose an algorithm for efficiently constructing an instance of (BoxQP) that admits an exact RLT relaxation and another algorithm for constructing an instance with an inexact RLT relaxation.
4. We establish that strong duality holds and that primal and dual optimal solutions are attained for the SDP-RLT relaxation and its dual. By relying on this relation, we give an algebraic description of the set of instances of (BoxQP) that admit an exact SDP-RLT relaxation.
5. By utilizing this algebraic description, we propose an algorithm for constructing an instance of (BoxQP) that admits an exact SDP-RLT relaxation and another one for constructing an instance that admits an exact SDP-RLT but an inexact RLT relaxation.

This paper is organized as follows. We briefly review the literature in Sect. 1.3 and define our notation in Sect. 1.4. We review the optimality conditions in Sect. 2. We present several properties of the convex underestimators arising from the RLT and SDP-RLT relaxations in Sect. 3. Section 4 focuses on the description of instances with exact RLT relaxations and presents two algorithms for constructing instances, one with exact RLT relaxations and another one with inexact RLT relaxations. SDP-RLT relaxations are treated in Sect. 5, which includes an algebraic description of instances with exact SDP-RLT relaxations and two algorithms for constructing instances, one with exact SDP-RLT relaxations and another one with exact SDP-RLT but inexact RLT relaxations. We present several numerical examples, preliminary computational results, and a brief discussion in Sect. 6. Finally, Sect. 7 concludes the paper.

1.3 Literature review

Quadratic programs with box constraints have been extensively studied in the literature. Since our focus is on convex relaxations in this paper, we will mainly restrict our review accordingly.

The set $\text{conv}(\mathcal{F})$, where $\mathcal{F}$ is given by (1), has been investigated in several papers (see, e.g., [15–20]). This is a nonpolyhedral convex set even for $n = 1$. However, it turns out that $\text{conv}(\mathcal{F})$ is closely related to the so-called Boolean quadric polytope [21] that arises in unconstrained binary quadratic programs, which can be formulated as an instance of (BoxQP) [22], and is given by $\text{conv}(\mathcal{F}^-)$, where

$$\mathcal{F}^- = \left\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^{\binom{n}{2}} : x_i \in \{0, 1\}, \quad z_{ij} = x_i x_j, \quad 1 \le i < j \le n\right\}. \tag{5}$$

The linear programming relaxation of $\text{conv}(\mathcal{F}^-)$, denoted by $\mathcal{F}_R^-$, is given by

$$\mathcal{F}_R^- = \left\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^{\binom{n}{2}} : \begin{array}{rcl} 0 \le & x & \le e \\ \max\{x_i + x_j - 1, 0\} \le & z_{ij} & \le \min\{x_i, x_j\}, \quad 1 \le i < j \le n \end{array}\right\}, \tag{6}$$

which is very similar to $\mathcal{F}_R$, except that McCormick inequalities are only applied to $1 \le i < j \le n$. In particular, $\mathcal{F}_R^- = \text{conv}(\mathcal{F}^-)$ for $n = 2$ [7]. Padberg [21] identifies several facets of $\text{conv}(\mathcal{F}^-)$ and shows that the components of each vertex of $\mathcal{F}_R^-$ are in the set $\{0, \frac{1}{2}, 1\}$. Yajima and Fujie [15] show how to extend the valid inequalities for $\mathcal{F}^-$ in [21] to $\mathcal{F}$. Burer and Letchford [16] extend this result further by observing that $\text{conv}(\mathcal{F}^-)$ is the projection of $\text{conv}(\mathcal{F})$ onto the "common variables." They also give a description of the set of extreme points of $\text{conv}(\mathcal{F})$. We refer the reader to [19] for further refinements and to [20] for a computational procedure based on such valid inequalities.

Anstreicher [17] reports computational results illustrating that the SDP-RLT relaxation significantly improves the RLT relaxation and gives a theoretical justification of the improvement by comparing $\mathcal{F}_R$ and $\mathcal{F}_{RS}$ for $n = 2$. Anstreicher and Burer [18] show that $\mathcal{F}_{RS} = \text{conv}(\mathcal{F})$ if and only if $n \le 2$. In particular, this implies that the SDP-RLT relaxation of (BoxQP) is always exact for $n \le 2$.

We next briefly review the literature on exact convex relaxations. Several papers have identified conditions under which a particular convex relaxation of a class of optimization problems is exact. For quadratically constrained quadratic programs, we refer the reader to [23–30] for various exactness conditions for second-order cone or semidefinite relaxations. Recently, a large family of convex relaxations of general quadratic programs was considered in a unified manner through induced convex underestimators and a general algorithmic procedure was proposed for constructing instances with inexact relaxations for various convex relaxations [14].

In this work, our focus is on algebraic descriptions and algorithmic constructions of instances of (BoxQP) that admit exact and inexact RLT relaxations as well as those that admit exact SDP-RLT relaxations and exact SDP-RLT but inexact RLT relaxations. Therefore, our focus is similar to [31, 32], which presented descriptions of such instances of standard quadratic programs for RLT and SDP-RLT relaxations, respectively.

1.4 Notation

We use $\mathbb{R}^n$, $\mathbb{R}^{m\times n}$, and $\mathcal{S}^n$ to denote the n-dimensional Euclidean space, the set of $m \times n$ real matrices, and the space of $n \times n$ real symmetric matrices, respectively. We use 0 to denote the real number 0, the vector of all zeroes, as well as the matrix of all zeroes, which should always be clear from the context. We reserve $e \in \mathbb{R}^n$ for the vector of all ones. All inequalities on vectors or matrices are componentwise. For $A \in \mathcal{S}^n$, we use $A \succeq 0$ (resp., $A \succ 0$) to denote that A is positive semidefinite (resp., positive definite). For $x \in \mathbb{R}^n$, $B \in \mathbb{R}^{m\times n}$, and index sets $\mathbb{J} \subseteq \{1,\ldots,m\}$ and $\mathbb{K} \subseteq \{1,\ldots,n\}$, we denote by $x_{\mathbb{K}} \in \mathbb{R}^{|\mathbb{K}|}$ the subvector of x restricted to the indices in $\mathbb{K}$ and by $B_{\mathbb{JK}} \in \mathbb{R}^{|\mathbb{J}|\times|\mathbb{K}|}$ the submatrix of B whose rows and columns are indexed by $\mathbb{J}$ and $\mathbb{K}$, respectively, where $|\cdot|$ denotes the cardinality of a finite set. We simply use x_j and Q_{ij} for singleton index sets. For any $U \in \mathbb{R}^{m\times n}$ and $V \in \mathbb{R}^{m\times n}$, the trace inner product is denoted by

$$\langle U, V\rangle = \mathrm{trace}(U^T V) = \sum_{i=1}^{m}\sum_{j=1}^{n} U_{ij}V_{ij}.$$

For an instance of (BoxQP) given by $(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n$, we define

$$q(x) = \tfrac{1}{2}x^T Qx + c^T x, \tag{7}$$
$$F = \{x \in \mathbb{R}^n : 0 \le x \le e\}, \tag{8}$$
$$\ell^* = \min_{x\in\mathbb{R}^n} \{q(x) : x \in F\}, \tag{9}$$
$$V = \{x \in F : x_j \in \{0,1\}, \quad j = 1,\ldots,n\}. \tag{10}$$

For a given instance of (BoxQP), note that $q(x)$, F, ℓ^*, and V denote the objective function, the feasible region, the optimal value, and the set of vertices, respectively. For $\hat{x} \in F$, we define the following index sets:

$$\mathbb{L} = \mathbb{L}(\hat{x}) = \{j \in \{1,\ldots,n\} : \hat{x}_j = 0\}, \tag{11}$$
$$\mathbb{B} = \mathbb{B}(\hat{x}) = \{j \in \{1,\ldots,n\} : 0 < \hat{x}_j < 1\}, \tag{12}$$
$$\mathbb{U} = \mathbb{U}(\hat{x}) = \{j \in \{1,\ldots,n\} : \hat{x}_j = 1\}. \tag{13}$$

2 Optimality conditions

In this section, we review first-order and second-order optimality conditions for (BoxQP).

Let $\hat{x} \in F$ be a local minimizer of (BoxQP). By the first-order optimality conditions, there exists $(\hat{r}, \hat{s}) \in \mathbb{R}^n \times \mathbb{R}^n$ such that

$$Q\hat{x} + c + \hat{r} - \hat{s} = 0, \tag{14}$$
$$\hat{r}_{\mathbb{L}\cup\mathbb{B}} = 0, \tag{15}$$
$$\hat{s}_{\mathbb{B}\cup\mathbb{U}} = 0, \tag{16}$$
$$\hat{r} \ge 0, \tag{17}$$
$$\hat{s} \ge 0. \tag{18}$$

Note that $\hat{r} \in \mathbb{R}^n$ and $\hat{s} \in \mathbb{R}^n$ are the Lagrange multipliers corresponding to the constraints $x \le e$ and $x \ge 0$ in (BoxQP), respectively.

For a local minimizer $\hat{x} \in F$ of (BoxQP), the second-order optimality conditions are given by

$$d^T Qd \geq 0, \quad \forall\, d \in D(\hat{x}), \tag{19}$$

where $D(\hat{x})$ is the set of feasible directions at $\hat{x}$ at which the directional derivative of the objective function vanishes, i.e.,

$$D(\hat{x}) := \{d \in \mathbb{R}^n : (Q\hat{x} + c)^T d = 0, \quad d_{\mathbb{L}} \geq 0, \quad d_{\mathbb{U}} \leq 0\}. \tag{20}$$

Note, in particular, that

$$\hat{x} \in F \text{ is a local minimizer} \Rightarrow Q_{\mathbb{BB}} \succeq 0. \tag{21}$$

In fact, $\hat{x} \in F$ is a local minimizer of (BoxQP) if and only if the first-order and second-order optimality conditions given by (14)–(18) and (19), respectively, are satisfied (see, e.g., [33, 34]).

3 Properties of RLT and SDP-RLT relaxations

Given an instance of (BoxQP), recall that the RLT relaxation is given by

$$\text{(R)} \quad \ell_R^* = \min_{(x,X) \in \mathbb{R}^n \times \mathcal{S}^n} \left\{ \frac{1}{2} \langle Q, X \rangle + c^T x : (x, X) \in \mathcal{F}_R \right\},$$

where $\mathcal{F}_R$ is given by (2), and the SDP-RLT relaxation by

$$\text{(RS)} \quad \ell_{RS}^* = \min_{(x,X) \in \mathbb{R}^n \times \mathcal{S}^n} \left\{ \frac{1}{2} \langle Q, X \rangle + c^T x : (x, X) \in \mathcal{F}_{RS} \right\},$$

where $\mathcal{F}_{RS}$ is given by (3).

Every convex relaxation of a nonconvex quadratic program obtained through lifting induces a convex underestimator of the objective function over the feasible region [14]. In this section, we introduce the convex underestimators induced by RLT and SDP-RLT relaxations and establish several properties of these underestimators.

3.1 Convex underestimators

In this section, we introduce the convex underestimators induced by the RLT and SDP-RLT relaxations. Let us first define the following sets parametrized by $\hat{x} \in F$:

$$\mathcal{F}_R(\hat{x}) = \{(x, X) \in \mathcal{F}_R : x = \hat{x}\}, \quad \hat{x} \in F, \tag{22}$$

$$\mathcal{F}_{RS}(\hat{x}) = \{(x, X) \in \mathcal{F}_{RS} : x = \hat{x}\}, \quad \hat{x} \in F. \tag{23}$$

For each $\hat{x} \in F$, we clearly have $\{(\hat{x}, \hat{x}\hat{x}^T)\} \subseteq \mathcal{F}_{RS}(\hat{x}) \subseteq \mathcal{F}_R(\hat{x})$ and

$$\bigcup_{\hat{x} \in F} \mathcal{F}_R(\hat{x}) = \mathcal{F}_R, \quad \bigcup_{\hat{x} \in F} \mathcal{F}_{RS}(\hat{x}) = \mathcal{F}_{RS}.$$

Next, we define the following functions:

$$\ell_R(\hat{x}) = \min_{x \in \mathbb{R}^n, X \in \mathcal{S}^n} \left\{ \tfrac{1}{2} \langle Q, X \rangle + c^T x : (x, X) \in \mathcal{F}_R(\hat{x}) \right\}, \quad \hat{x} \in F, \tag{24}$$

$$\ell_{RS}(\hat{x}) = \min_{x \in \mathbb{R}^n, X \in \mathcal{S}^n} \left\{ \tfrac{1}{2} \langle Q, X \rangle + c^T x : (x, X) \in \mathcal{F}_{RS}(\hat{x}) \right\}, \quad \hat{x} \in F. \tag{25}$$

Note that the functions $\ell_R(\cdot)$ and $\ell_{RS}(\cdot)$ return the best objective function value of the corresponding relaxation subject to the additional constraint that $x = \hat{x}$. By [14], each of $\ell_R(\cdot)$ and $\ell_{RS}(\cdot)$ is a convex function over F satisfying the relations

$$\ell_R(\hat{x}) \leq \ell_{RS}(\hat{x}) \leq q(\hat{x}), \quad \hat{x} \in F, \tag{26}$$

and

$$\text{(R1)} \quad \ell_R^* = \min_{x \in F} \ell_R(x), \tag{27}$$

$$\text{(RS1)} \quad \ell_{RS}^* = \min_{x \in F} \ell_{RS}(x). \tag{28}$$

The convex underestimators $\ell_R(\cdot)$ and $\ell_{RS}(\cdot)$ allow us to view the RLT and SDP-RLT relaxations in the original space $\mathbb{R}^n$ of (BoxQP) by appropriately projecting out the lifted variables $X \in \mathcal{S}^n$ that appear in each of (R) and (RS). As such, (R1) and (RS1) can be viewed as "reduced" formulations of the RLT relaxation and the SDP-RLT relaxation, respectively. In the remainder of this manuscript, we will alternate between the two equivalent formulations (R) and (R1) for the RLT relaxation as well as (RS) and (RS1) for the SDP-RLT relaxation.

3.2 Properties of convex underestimators

In this section, we present several properties of the convex underestimators $\ell_R(\cdot)$ and $\ell_{RS}(\cdot)$ given by (24) and (25), respectively.

First, we start with the observation that $\ell_R(\cdot)$ has a very specific structure with a simple closed-form expression.

Lemma 1 *$\ell_R(\cdot)$ is a piecewise linear convex function on F given by*

$$\ell_R(\hat{x}) = \frac{1}{2}\left(\sum_{(i,j):Q_{ij}>0} Q_{ij}\max\{0, \hat{x}_i + \hat{x}_j - 1\} + \sum_{(i,j):Q_{ij}<0} Q_{ij}\min\{\hat{x}_i, \hat{x}_j\}\right) + c^T\hat{x} \tag{29}$$

for each $\hat{x} \in F$.

Proof For each $\hat{x} \in F$, the relation (29) follows from (24) and (2). It follows that $\ell_R(\cdot)$ is a piecewise linear convex function on F since it is given by the sum of a finite number of piecewise linear convex functions. □

In contrast with $\ell_R(\cdot)$ given by the optimal value of a simple linear programming problem with bound constraints, $\ell_{RS}(\cdot)$ does not, in general, have a simple closed-form expression as it is given by the optimal value of a semidefinite programming problem.

The next result states a useful decomposition property regarding the sets $\mathcal{F}_R(\hat{x})$ and $\mathcal{F}_{RS}(\hat{x})$.

Lemma 2 *For any $\hat{x} \in F$, $(\hat{x}, \hat{X}) \in \mathcal{F}_R(\hat{x})$ if and only if there exists $\hat{M} \in \mathcal{M}_R(\hat{x})$ such that $\hat{X} = \hat{x}\hat{x}^T + \hat{M}$, where*

$$\mathcal{M}_R(\hat{x}) = \left\{ M \in \mathcal{S}^n : \begin{array}{ll} M_{ij} \leq \min\{\hat{x}_i - \hat{x}_i\hat{x}_j, \hat{x}_j - \hat{x}_i\hat{x}_j\}, & i \in \mathbb{B}, j \in \mathbb{B}, \\ M_{ij} \geq \max\{-\hat{x}_i\hat{x}_j, \hat{x}_i + \hat{x}_j - 1 - \hat{x}_i\hat{x}_j\}, & i \in \mathbb{B}, j \in \mathbb{B}, \\ M_{ij} = 0, \quad \textit{otherwise}. & \end{array} \right\}, \tag{30}$$

where $\mathbb{B}$ is given by (12). Furthermore, $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}(\hat{x})$ if and only if $\hat{M} \in \mathcal{M}_{RS}(\hat{x})$, where

$$\mathcal{M}_{RS}(\hat{x}) = \left\{ M \in \mathcal{S}^n : M \in \mathcal{M}_R(\hat{x}), \quad M \succeq 0 \right\}. \tag{31}$$

Proof Both assertions follow from (22), (2), (23), (3), and the decomposition $\hat{X} = \hat{x}\hat{x}^T + \hat{M}$. □

By Lemma 2, we remark that M_{ij} has a negative lower bound and a positive upper bound in (30) if and only if $i \in \mathbb{B}$ and $j \in \mathbb{B}$. Therefore, for any $\hat{x} \in \mathcal{F}$ and any $(\hat{x}, \hat{X}) \in \mathcal{F}_R$ (and hence any $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$), we obtain

$$\hat{X}_{ij} = \hat{x}_i \hat{x}_j, \quad i \notin \mathbb{B}, \text{ or } j \notin \mathbb{B}. \tag{32}$$

This observation yields the following result.

Corollary 3 *For any vertex $v \in F$, $\mathcal{F}_R(v) = \mathcal{F}_{RS}(v) = \{(v, vv^T)\}$.*

Proof The claim directly follows from (32) since $\mathbb{B} = \emptyset$. □

The decomposition in Lemma 2 can be translated into the functions $\ell_R(\cdot)$ and $\ell_{RS}(\cdot)$.

Lemma 4 *For each $\hat{x} \in F$,*

$$\ell_R(\hat{x}) = q(\hat{x}) + \tfrac{1}{2} \min_{M \in \mathcal{M}_R(\hat{x})} \langle Q, M \rangle, \tag{33}$$

$$\ell_{RS}(\hat{x}) = q(\hat{x}) + \tfrac{1}{2} \min_{M \in \mathcal{M}_{RS}(\hat{x})} \langle Q, M \rangle, \tag{34}$$

where $\mathcal{M}_R(\hat{x})$ and $\mathcal{M}_{RS}(\hat{x})$ are given by (30) and (31), respectively.

Proof The assertions directly follow from (24), (25), and Lemma 2. □

By Lemma 4, we can easily establish the following properties.

Lemma 5 *Let $\hat{x} \in F$ and let $\mathbb{B} = \mathbb{B}(\hat{x})$, where $\mathbb{B}(\hat{x})$ is given by (12).*

(i) $\ell_R(\hat{x}) = q(\hat{x})$ if and only if $\hat{x}$ is a vertex of F or $Q_{\mathbb{B}\mathbb{B}} = 0$.
(ii) $\ell_{RS}(\hat{x}) = q(\hat{x})$ if and only if $\hat{x}$ is a vertex of F or $Q_{\mathbb{B}\mathbb{B}} \succeq 0$.

Proof By Lemma 4, $\ell_R(\hat{x}) = q(\hat{x})$ (resp., $\ell_{RS}(\hat{x}) = q(\hat{x})$) if and only if $\min_{M \in \mathcal{M}_R(\hat{x})} \langle Q, M \rangle = 0$ (resp., $\min_{M \in \mathcal{M}_{RS}(\hat{x})} \langle Q, M \rangle = 0$). The assertions now follow from Lemma 2. □

Lemma 5 immediately gives rise to the following results about the underestimator $\ell_{RS}(\cdot)$.

Corollary 6 *(i) If $Q \succeq 0$, then $\ell_{RS}(\hat{x}) = q(\hat{x})$ for each $\hat{x} \in F$.*
(ii) For any local or global minimizer $\hat{x} \in F$ of (BoxQP), we have $\ell_{RS}(\hat{x}) = q(\hat{x})$.

Proof If $Q \succeq 0$, then $Q_{\mathbb{B}\mathbb{B}} \succeq 0$ for each $\mathbb{B} \subseteq \{1, \ldots, n\}$. Therefore, both assertions follow from Lemma 5(ii) since $Q_{\mathbb{B}\mathbb{B}} \succeq 0$ at any local or global minimizer of (BoxQP) by (21). □

Corollary 6(i) in fact holds for SDP relaxations of general quadratic programs and a result similar to Corollary 6(ii) was established for general quadratic programs with a bounded feasible region [14]. We remark that Corollary 6(ii) presents a desirable property of the SDP-RLT relaxation, which is a necessary condition for its exactness by (26) and (28). However, this necessary condition, in general, is not sufficient for exactness.

4 Exact and inexact RLT relaxations

In this section, we focus on instances of (BoxQP) that admit exact and inexact RLT relaxations. We first establish a useful property of the set of optimal solutions of RLT relaxations. Using this property, we present two equivalent but different algebraic descriptions of instances with exact RLT relaxations. By utilizing one of these descriptions, we present an algorithm for constructing instances of (BoxQP) with an exact RLT relaxation and another algorithm for constructing instances with an inexact RLT relaxation.

4.1 Optimal solutions of RLT relaxations

In this section, we present useful properties of the set of optimal solutions of RLT relaxations. Our first result establishes the existence of a minimizer of (R1) with a very specific structure.

Proposition 7 *For the RLT relaxation of any instance of (BoxQP), there exists an optimal solution $\hat{x} \in F$ of (R1), where (R1) is given by (27), such that $\hat{x}_j \in \{0, \frac{1}{2}, 1\}$ for each $j = 1, \ldots, n$.*

Proof Let $\hat{x} \in F$ be an optimal solution of (R1), i.e., $\ell_R^* = \ell_R(\hat{x})$. Suppose that there exists $k \in \{1, \ldots, n\}$ such that $\hat{x}_k \notin \{0, \frac{1}{2}, 1\}$. By appropriately perturbing $\hat{x}$, we will show that one can construct another optimal solution $\tilde{x} \in F$ such that $\ell_R(\tilde{x}) = \ell_R(\hat{x}) = \ell_R^*$ and $\tilde{x}_j \in \{0, \frac{1}{2}, 1\}$ for each $j = 1, \ldots, n$.

Let

$$\alpha = \min\{\hat{x}_k, 1 - \hat{x}_k\} \in (0, \tfrac{1}{2}), \tag{35}$$

and let

$$\alpha_l = \max\left\{\left(\max_{j:\min\{\hat{x}_j, 1-\hat{x}_j\} < \alpha} \min\{\hat{x}_j, 1 - \hat{x}_j\}\right), 0\right\}, \tag{36}$$

$$\alpha_u = \min\left\{\left(\min_{j:\min\{\hat{x}_j, 1-\hat{x}_j\} > \alpha} \min\{\hat{x}_j, 1 - \hat{x}_j\}\right), \tfrac{1}{2}\right\}, \tag{37}$$

with the usual conventions that the minimum and the maximum over the empty set are defined to be $+\infty$ and $-\infty$, respectively. Note that $0 \leq \alpha_l < \alpha < \alpha_u \leq \frac{1}{2}$ and

$$\min\{\hat{x}_j, 1 - \hat{x}_j\} \in [0, \alpha_l] \cup \{\alpha\} \cup [\alpha_u, \tfrac{1}{2}], \quad j = 1, \ldots, n. \tag{38}$$

Let us define the following index sets:

$$\begin{aligned}
\mathbb{I}_1 &= \{j \in \{1, \ldots, n\} : \hat{x}_j = \alpha\}, \\
\mathbb{I}_2 &= \{j \in \{1, \ldots, n\} : \hat{x}_j = 1 - \alpha\}, \\
\mathbb{I}_3 &= \{j \in \{1, \ldots, n\} : \hat{x}_j \in [0, \alpha_l] \cup [\alpha_u, 1 - \alpha_u] \cup [1 - \alpha_l, 1]\}.
\end{aligned}$$

Note that $\mathbb{I}_1, \mathbb{I}_2, \mathbb{I}_3$ is a partition of the index set by (38), and we have $k \in \mathbb{I}_1 \cup \mathbb{I}_2$.

Let us define a direction $\hat{d} \in \mathbb{R}^n$ by

$$\hat{d}_j = \begin{cases} 1, & j \in \mathbb{I}_1, \\ -1, & j \in \mathbb{I}_2, \\ 0, & j \in \mathbb{I}_3. \end{cases}$$

Consider $x^\beta = \hat{x} + \beta\hat{d}$. It is easy to verify that $x^\beta \in F$ for each $\beta \in [\alpha_l - \alpha, \alpha_u - \alpha]$. We claim that $\ell_R(x^\beta)$ is a linear function of β on this interval. By (29), it suffices to show that each term is a linear function.

First, let us focus on the term given by $\max\{0, x_i^\beta + x_j^\beta - 1\} = \max\{0, \hat{x}_i + \hat{x}_j - 1 + \beta\hat{d}_i + \beta\hat{d}_j\}$, where $i = 1, \ldots, n;\ \ j = 1, \ldots, n$. It suffices to show that the sign of $\hat{x}_i + \hat{x}_j - 1 + \beta\hat{d}_i + \beta\hat{d}_j$ does not change for each $\beta \in [\alpha_l - \alpha, \alpha_u - \alpha]$ and for each $i = 1, \ldots, n;\ \ j = 1, \ldots, n$. Clearly, $\hat{x}_i + \hat{x}_j - 1 + \beta\hat{d}_i + \beta\hat{d}_j = \hat{x}_i + \hat{x}_j - 1$ if $\{i, j\} \subseteq \mathbb{I}_3$; or $i \in \mathbb{I}_1$, $j \in \mathbb{I}_2$; or $i \in \mathbb{I}_2$, $j \in \mathbb{I}_1$. For the remaining cases, it follows from the definitions of $\mathbb{I}_1$, $\mathbb{I}_2$, and $\mathbb{I}_3$ that

$$\hat{x}_i + \hat{x}_j - 1 + \beta\hat{d}_i + \beta\hat{d}_j \in \begin{cases} [2\alpha_l - 1, 2\alpha_u - 1], & \{i, j\} \subseteq \mathbb{I}_1, \\ [1 - 2\alpha_u, 1 - 2\alpha_l], & \{i, j\} \subseteq \mathbb{I}_2, \\ [\alpha_l - 1, \alpha_l + \alpha_u - 1] \cup [\alpha_l + \alpha_u - 1, 0] \cup [0, \alpha_u], & i \in \mathbb{I}_1, j \in \mathbb{I}_3; \\ & \text{or } i \in \mathbb{I}_3, j \in \mathbb{I}_1, \\ [-\alpha_u, 0] \cup [0, 1 - \alpha_l - \alpha_u] \cup [1 - \alpha_l - \alpha_u, 1 - \alpha_l], & i \in \mathbb{I}_2, j \in \mathbb{I}_3; \\ & \text{or } i \in \mathbb{I}_3, j \in \mathbb{I}_2. \end{cases}$$

Our claim now follows from $0 \le \alpha_l < \alpha_u \le \frac{1}{2}$. Therefore, $\max\{0, \hat{x}_i + \hat{x}_j - 1 + \beta\hat{d}_i + \beta\hat{d}_j\}$ is a linear function of β on the interval $[\alpha_l - \alpha, \alpha_u - \alpha]$ for each $i = 1, \ldots, n;\ \ j = 1, \ldots, n$.

Let us now consider the term $\min\{x_i^\beta, x_j^\beta\} = \min\{\hat{x}_i + \beta\hat{d}_i, \hat{x}_j + \beta\hat{d}_j\}$. By the choice of $\hat{d}$, it is easy to see that the order of the components of x^β remains unchanged for each $\beta \in [\alpha_l - \alpha, \alpha_u - \alpha]$, i.e., if $\hat{x}_i \le \hat{x}_j$, then $\hat{x}_i + \beta\hat{d}_i \le \hat{x}_j + \beta\hat{d}_j$ for each $i = 1, \ldots, n;\ \ j = 1, \ldots, n$. It follows that $\min\{\hat{x}_i + \beta\hat{d}_i, \hat{x}_j + \beta\hat{d}_j\}$ is a linear function of β on the interval $[\alpha_l - \alpha, \alpha_u - \alpha]$.

Since the third term in (29) is also a linear function of β on the interval $[\alpha_l - \alpha, \alpha_u - \alpha]$, it follows that $\ell_R(x^\beta)$ is a linear function on $[\alpha_l - \alpha, \alpha_u - \alpha]$. Therefore, by the optimality of $\hat{x}$ in (R1), $\ell_R(x^\beta)$ is a constant function on this interval. If $\alpha_l = 0$ and $\alpha_u = \frac{1}{2}$, then the alternate optimal solution $x^\beta = \hat{x} + \beta\hat{d}$ obtained by setting $\beta = \alpha_l - \alpha = -\alpha$ or by setting $\beta = \alpha_u - \alpha = \frac{1}{2} - \alpha$ satisfies the desired property by (38). Otherwise, we set (i) $\beta = \alpha_l - \alpha$ if $\alpha_l > 0$, or (ii) $\beta = \alpha_u - \alpha$ if $\alpha_u < \frac{1}{2}$. In each case, we have $\min\{x_j^\beta, 1 - x_j^\beta\} \in [0, \alpha_l] \cup [\alpha_u, \frac{1}{2}]$ for each $j = 1, \ldots, n$ (cf. (38)). Starting now with the new optimal solution x^β, one can repeat the same procedure in an iterative manner to arrive at an alternate optimal solution with the desired property. Note that this procedure is finite since, at each iteration, either α_l given by (36) strictly decreases in case (i), or α_u given by (37) strictly increases in case (ii). □

We can utilize Proposition 7 to obtain the following result about the set of optimal solutions of (R).

Corollary 8 *There exists an optimal solution $(\hat{x}, \hat{X}) \in \mathbb{R}^n \times \mathcal{S}^n$ of the RLT relaxation (R) such that $\hat{x}_j \in \{0, \frac{1}{2}, 1\}$ for each $j = 1, \ldots, n$ and $\hat{X}_{ij} \in \{0, \frac{1}{2}, 1\}$ for each $i = 1, \ldots, n;\ \ j = 1, \ldots, n$ such that*

$$\begin{bmatrix} \hat{X}_{\mathbb{LL}} & \hat{X}_{\mathbb{LB}} & \hat{X}_{\mathbb{LU}} \\ \hat{X}_{\mathbb{BL}} & \hat{X}_{\mathbb{BB}} & \hat{X}_{\mathbb{BU}} \\ \hat{X}_{\mathbb{UL}} & \hat{X}_{\mathbb{UB}} & \hat{X}_{\mathbb{UU}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \hat{X}_{\mathbb{BB}} & \frac{1}{2} e_{\mathbb{B}} e_{\mathbb{U}}^T \\ 0 & \frac{1}{2} e_{\mathbb{U}} e_{\mathbb{B}}^T & e_{\mathbb{U}} e_{\mathbb{U}}^T \end{bmatrix}, \quad \hat{X}_{ij} \in \{0, \tfrac{1}{2}\}, \quad i \in \mathbb{B},\ j \in \mathbb{B}.$$

Proof By Proposition 7, there exists $\hat{x} \in F$ such that $\ell_R^* = \ell_R(\hat{x})$ and $\hat{x}_j \in \{0, \frac{1}{2}, 1\}$ for each $j = 1, \ldots, n$. Define $\hat{X} \in \mathcal{S}^n$ such that

$$\hat{X}_{ij} = \begin{cases} \max\{0, \hat{x}_i + \hat{x}_j - 1\}, & \text{if } Q_{ij} > 0, \\ \min\{\hat{x}_i, \hat{x}_j\}, & \text{if } Q_{ij} < 0, \\ 0, & \text{otherwise,} \end{cases} \quad i = 1, \ldots, n; \; j = 1, \ldots, n.$$

Note that $(\hat{x}, \hat{X}) \in \mathcal{F}_R$ by (2) and $\ell_R(\hat{x}) = \frac{1}{2}\langle Q, \hat{X}\rangle + c^T\hat{x}$ by Lemma 1. Therefore, $(\hat{x}, \hat{X})$ is an optimal solution of (R) with the desired property. □

The next result follows from Corollary 8.

Corollary 9 *For each vertex $(\hat{x}, \hat{X}) \in \mathcal{F}_R$, $\hat{x}_j \in \{0, \frac{1}{2}, 1\}$ for each $j = 1, \ldots, n$ and $\hat{X}_{ij} \in \{0, \frac{1}{2}, 1\}$ for each $i = 1, \ldots, n; \; j = 1, \ldots, n$.*

Proof Since $\mathcal{F}_R$ is a polytope, $(\hat{x}, \hat{X}) \in \mathcal{F}_R$ is a vertex if and only if there exists a $(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n$ such that $(\hat{x}, \hat{X}) \in \mathcal{F}_R$ is the unique optimal solution of (R). The assertion follows from Corollary 8. □

We remark that Padberg [21] established a similar result for the set $\mathcal{F}_R^-$ given by (6), which is the projection of $\mathcal{F}_R$ on the onto the "common variables." In fact, Padberg's result is implied by Corollary 9 since each vertex of $\mathcal{F}_R^-$ is a projection of some vertex in $\mathcal{F}_R$. In contrast with the proof of [21], which relies on linearly independent active constraints, our proof uses a specific property of the set of optimal solutions of the reduced formulation (R1).

4.2 First description of instances with exact RLT relaxations

In this section, we present our first description of the set of instances of (BoxQP) with an exact RLT relaxation. We start with a useful property of such instances.

Proposition 10 *For any instance of (BoxQP), the RLT relaxation is exact, i.e., $\ell_R^* = \ell^*$, if and only if there exists a vertex $v \in F$ such that v is an optimal solution of (R1), where (R1) is given by (27).*

Proof Suppose that $\ell_R^* = \ell^*$. Then, by (26) and (27), for any optimal solution $x^* \in F$ of (BoxQP), we have $q(x^*) = \ell^* = \ell_R^* \leq \ell_R(x^*) \leq q(x^*) = \ell^*$, which implies that $\ell_R^* = \ell_R(x^*) = q(x^*)$. By Lemma 5(i), either x^* is a vertex of F, in which case, we are done, or $Q_{\mathbb{B}\mathbb{B}} = 0$, where $\mathbb{B}$ is given by (12). In the latter case, since $\hat{x}_{\mathbb{L}}^* = 0$ and $\hat{x}_{\mathbb{U}}^* = e_{\mathbb{U}}$ by (11) and (13), respectively, we obtain

$$\ell_R^* = \ell^* = q(x^*) = \frac{1}{2}e_{\mathbb{U}}^T Q_{\mathbb{U}\mathbb{U}} e_{\mathbb{U}} + (x_{\mathbb{B}}^*)^T Q_{\mathbb{B}\mathbb{U}} e_{\mathbb{U}} + c_{\mathbb{U}}^T e_{\mathbb{U}} + c_{\mathbb{B}}^T x_{\mathbb{B}}^* = \frac{1}{2}e_{\mathbb{U}}^T Q_{\mathbb{U}\mathbb{U}} e_{\mathbb{U}} + c_{\mathbb{U}}^T e_{\mathbb{U}},$$

where the last equality follows from the identity $Q_{\mathbb{B}\mathbb{U}} e_{\mathbb{U}} + c_{\mathbb{B}} = 0$ by (14)–(16). Therefore, for the vertex $v \in F$ given by $v_j = 1$ for each $j \in \mathbb{U}$ and $v_j = 0$ for each $j \in \mathbb{L} \cup \mathbb{B} = 0$, we obtain $q(v) = \ell^* = \ell_R(v)$ by Lemma 5(i).

Conversely, if there exists a vertex $v \in F$ such that $\ell_R(v) = \ell_R^*$, then we have $\ell^* \leq q(v) = \ell_R(v) = \ell_R^*$ by Lemma 5(i). The assertion follows from (4). □

Proposition 10 presents an important property of the set of instances of (BoxQP) with exact RLT relaxations in terms of the set of optimal solutions and gives rise to the following corollary.

Corollary 11 *For any instance of (BoxQP), the RLT relaxation is exact if and only if there exists a vertex $v \in F$ such that (v, vv^T) is an optimal solution of (R). Furthermore, in this case, v is an optimal solution of (BoxQP).*

Proof The assertion directly follows from Proposition 10, Lemma 5(i), and Corollary 3. □

By Corollary 11, if the set of optimal solutions of (BoxQP) does not contain a vertex, then the RLT relaxation is inexact. Note that if $q(\cdot)$ is a concave function, then the set of optimal solutions of (BoxQP) contains at least one vertex. However, this is an NP-hard problem (see, e.g., [4]), which implies that the RLT relaxation can be inexact even if the set of optimal solutions of (BoxQP) contains at least one vertex. The next example illustrates that the RLT relaxation can be inexact even if every optimal solution of (BoxQP) is a vertex.

Example 1 Consider an instance of (BoxQP) with

$$Q = \begin{bmatrix} -1 & -2 \\ -2 & 1 \end{bmatrix}, \quad c = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Note that

$$q(x) = \frac{1}{2}\left(-x_1^2 + x_2^2 - 4x_1x_2\right) + x_1 + x_2 = \frac{1}{2}(x_2 - x_1)^2 + (x_1 + x_2)(1 - x_1),$$

which implies that $q(x) \geq 0$ for each $x \in F$ and $q(x) = 0$ if and only if $x \in \{0, e\}$. Therefore, $\ell^* = 0$ and the set of optimal solution is given by $\{0, e\}$, which consists of two vertices of F. However, for $\hat{x} = \frac{1}{2}e \in \mathbb{R}^2$, it is easy to verify that $\ell_R(\hat{x}) = -1/4 < \ell^*$. In fact, $\ell_R^* = \ell_R(\hat{x}) = -1/4$ and $\ell_R(x) > \ell_R(\hat{x})$ for each $x \in F \backslash \{\hat{x}\}$. Figure 1 illustrates the functions $q(\cdot)$ and $\ell_R(\cdot)$.

We next present our first description of the set of instances that admit an exact RLT relaxation. To that end, let us define

$$\mathcal{E}_R = \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell^* = \ell_R^*\}, \tag{39}$$

i.e., $\mathcal{E}_R$ denotes the set of all instances of (BoxQP) that admit an exact RLT relaxation. For a given $\hat{x} \in F$, let us define

$$\begin{aligned} \mathcal{O}_R(\hat{x}) &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R(\hat{x}) = \ell_R^*\} \\ &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R(\hat{x}) \leq \ell_R(x), \quad x \in F\}, \end{aligned} \tag{40}$$

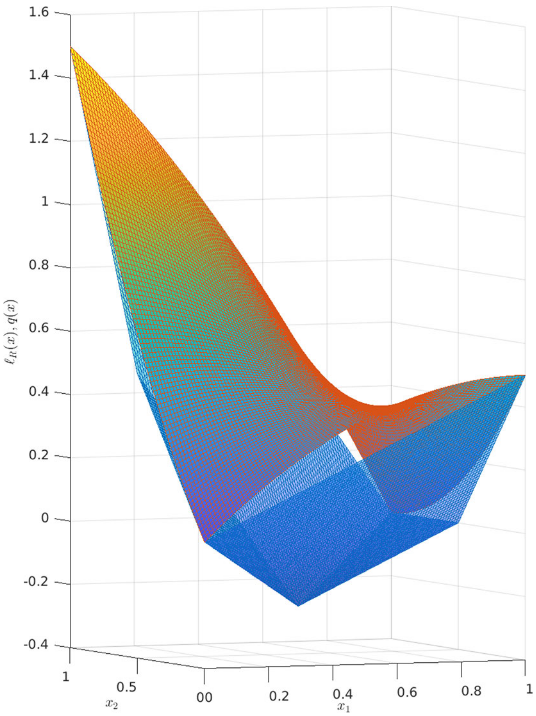
i.e., $\mathcal{O}_R(\hat{x})$ denotes the set of instances of (BoxQP) such that $\hat{x}$ is a minimizer of (R1), where (R1) is given by (27). By (29), it is easy to see that $\mathcal{O}_R(\hat{x})$ is a convex cone for each $\hat{x} \in F$. Our next result provides an algebraic description of the set $\mathcal{E}_R$.

Proposition 12 *Let $V \subset F$ denote the set of vertices of F given by (10) and let $V^+ = \{x \in F : x_j \in \{0, \frac{1}{2}, 1\}, \quad j = 1, \ldots, n\}$. Then, $\mathcal{E}_R$ defined as in (39) is given by the union of a finite number of polyhedral cones and admits the following description:*

$$\mathcal{E}_R = \bigcup_{v \in V} \mathcal{O}_R(v) = \bigcup_{v \in V} \left(\bigcap_{\hat{x} \in V^+} \left\{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R(v) \leq \ell_R(\hat{x})\right\} \right). \tag{41}$$

Proof By Proposition 10, the RLT relaxation is exact if and only if there exists a vertex $v \in F$ such that $\ell_R(v) = \ell_R^*$, which, together with (40), leads to the first equality in (41). By Proposition 7, the set V^+ contains at least one minimizer of $\ell_R(\cdot)$, which implies the second equality in (41). $\mathcal{E}_R$ is the union of a finite number of polyhedral cones since $\ell_R(x)$ is a linear function of (Q, c) for each fixed $x \in F$ by Lemma 1 and V^+ is a finite set. □

Fig. 1 Graphs of $q(\cdot)$ (above) and $\ell_R(\cdot)$ (below) for Example 1



For each $v \in V$, we remark that Proposition 12 gives a description of $\mathcal{O}_R(v)$ using 3^n linear inequalities since $|V^+| = 3^n$. In fact, for each $v \in V$, the convexity of $\ell_R(\cdot)$ on F implies that it suffices to consider only those $\hat{x} \in V_+$ such that $\hat{x}_j \in \{0, \frac{1}{2}\}$ for $j \in \mathbb{L}(v)$ and $\hat{x}_j \in \{\frac{1}{2}, 1\}$ for $j \in \mathbb{U}(v)$, where $\mathbb{L}(v)$ and $\mathbb{U}(v)$ are given by (11) and (13), respectively, which implies a simpler description of $\mathcal{E}_R$ with 2^n inequalities. Due to the exponential number of such inequalities, this description is not very useful for efficiently checking if a particular instance of (BoxQP) admits an exact RLT relaxation. Similarly, this description cannot be used easily for constructing such an instance of (BoxQP). In the next section, we present an alternative description of $\mathcal{E}_R$ using linear programming duality, which gives rise to algorithms for efficiently constructing instances of (BoxQP) with exact or inexact RLT relaxations.

4.3 An alternative description of instances with exact RLT relaxations

In this section, our main goal is to present an alternative description of the set $\mathcal{E}_R$ defined as in (39) using duality.

Recall that the RLT relaxation is given by

$$\text{(R)} \quad \ell_R^* = \min_{(x,X)\in\mathbb{R}^n\times\mathcal{S}^n} \left\{ \frac{1}{2}\langle Q, X\rangle + c^T x : (x, X) \in \mathcal{F}_R \right\},$$

where, $\mathcal{F}_R$ given by (2), can be expressed in the following form:

$$\mathcal{F}_R = \left\{ (x, X) \in \mathbb{R}^n \times \mathcal{S}^n : \begin{array}{r} x \leq e \\ x \geq 0 \\ X - xe^T - ex^T + ee^T \geq 0 \\ -X + ex^T \geq 0 \\ X \geq 0 \end{array} \right\}. \tag{42}$$

By defining dual variables $(r, s, W, Y, Z) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n$ corresponding to each of the five constraints in (42), respectively, the dual problem of (R) is given by

$$\begin{array}{llr} \text{(R-D)} & \max\limits_{(r,s,W,Y,Z)\in\mathbb{R}^n\times\mathbb{R}^n\times\mathcal{S}^n\times\mathbb{R}^{n\times n}\times\mathcal{S}^n} & -e^T r - \frac{1}{2}e^T We \\ & \text{s.t.} & \\ & & -r + s - We + Y^T e = c \\ & & W - Y - Y^T + Z = Q \\ & & r \geq 0 \\ & & s \geq 0 \\ & & W \geq 0 \\ & & Y \geq 0 \\ & & Z \geq 0. \end{array}$$

Note that the variables $(W, Y, Z) \in \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n$ in (R-D) are scaled by a factor of $\frac{1}{2}$. First, we start with optimality conditions for (R) and (R-D).

Lemma 13 *$(\hat{x}, \hat{X}) \in \mathcal{F}_R$ is an optimal solution of (R) if and only if there exists $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n$ such that*

$$Q = \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z} \tag{43}$$
$$c = -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e \tag{44}$$
$$\hat{r}^T(e - \hat{x}) = 0 \tag{45}$$
$$\hat{s}^T \hat{x} = 0 \tag{46}$$
$$\langle \hat{W}, \hat{X} - \hat{x}e^T - e\hat{x}^T + ee^T \rangle = 0 \tag{47}$$
$$\langle \hat{Y}, e\hat{x}^T - \hat{X} \rangle = 0 \tag{48}$$
$$\langle \hat{Z}, \hat{X} \rangle = 0 \tag{49}$$
$$\hat{r} \geq 0 \tag{50}$$
$$\hat{s} \geq 0 \tag{51}$$
$$\hat{W} \geq 0 \tag{52}$$
$$\hat{Y} \geq 0 \tag{53}$$
$$\hat{Z} \geq 0. \tag{54}$$

Proof The assertion follows from strong duality since each of (R) and (R-D) is a linear programming problem. □

Lemma 13 gives rise to an alternative description of the set of instances of (BoxQP) that admit an exact RLT relaxation.

Corollary 14 *$(Q, c) \in \mathcal{E}_R$, where $\mathcal{E}_R$ is defined as in (39), if and only if there exists a vertex $v \in F$ and there exists $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n$ such that the relations (43)–(54) hold, where $(\hat{x}, \hat{X}) = (v, vv^T)$.*

Proof Note that $(Q, c) \in \mathcal{E}_R$ if and only if there exists a vertex $v \in F$ such that (v, vv^T) is an optimal solution of (R) by Corollary 11. The assertion now follows from Lemma 13. $\square$

In the next section, we discuss how Corollary 14 can be utilized to construct instances of (BoxQP) with exact and inexact RLT relaxations.

4.4 Construction of instances with exact RLT relaxations

In this section, we describe an algorithm for constructing instances of (BoxQP) with an exact RLT relaxation. Algorithm 1 is based on designating a vertex $v \in F$ and constructing an appropriate dual feasible solution that satisfies optimality conditions together with $(v, vv^T) \in \mathcal{F}_R$.

Algorithm 1 (BoxQP) Instance with an Exact RLT Relaxation

Require: $n; \mathbb{L} \subseteq \{1, \dots, n\}$
Ensure: $(Q, c) \in \mathcal{E}_R$
1: $\mathbb{U} \leftarrow \{1, \dots, n\} \backslash \mathbb{L}$
2: Choose an arbitrary $\hat{r} \in \mathbb{R}^n$ such that $\hat{r} \geq 0$ and $\hat{r}_{\mathbb{L}} = 0$.
3: Choose an arbitrary $\hat{s} \in \mathbb{R}^n$ such that $\hat{s} \geq 0$ and $\hat{s}_{\mathbb{U}} = 0$.
4: Choose an arbitrary $\hat{W} \in \mathcal{S}^n$ such that $\hat{W} \geq 0$ and $\hat{W}_{\mathbb{L}\mathbb{L}} = 0$.
5: Choose an arbitrary $\hat{Y} \in \mathbb{R}^{n \times n}$ such that $\hat{Y} \geq 0$ and $\hat{Y}_{\mathbb{L}\mathbb{U}} = 0$.
6: Choose an arbitrary $\hat{Z} \in \mathcal{S}^n$ such that $\hat{Z} \geq 0$ and $\hat{Z}_{\mathbb{U}\mathbb{U}} = 0$.
7: $Q \leftarrow \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z}$
8: $c \leftarrow -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e$

The following result establishes the correctness of Algorithm 1.

Proposition 15 *Algorithm 1 returns $(Q, c) \in \mathcal{E}_R$, where $\mathcal{E}_R$ is defined as in (39). Conversely, any $(Q, c) \in \mathcal{E}_R$ can be generated by Algorithm 1 with appropriate choices of $\mathbb{L} \subseteq \{1, \dots, n\}$ and $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n \times n} \times \mathcal{S}^n$.*

Proof Let $\mathbb{L} \subseteq \{1, \dots, n\}$ and define $\mathbb{U} = \{1, \dots, n\} \backslash \mathbb{L}$. Let $v \in F$ be the vertex given by $v_j = 0,\ j \in \mathbb{L}$ and $v_j = 1,\ j \in \mathbb{U}$. It is easy to verify that $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n \times n} \times \mathcal{S}^n$ and $(\hat{x}, \hat{X}) = (v, vv^T)$ satisfy the hypotheses of Corollary 14, which establishes the first assertion. The second assertion also follows from Corollary 14. $\square$

By considering all possible subsets $\mathbb{L} \subseteq \{1, \dots, n\}$, Proposition 15 yields an alternative characterization of $\mathcal{E}_R$ given by the union of 2^n polyhedral cones (cf. Proposition 12). In contrast with the description in Proposition 12, this alternative description enables us to easily construct an instance of (BoxQP) with a known optimal vertex and an exact RLT relaxation (cf. Corollary 11). Note, however, that even the alternative description is not very useful for effectively checking if $(Q, c) \in \mathcal{E}_R$ due to the exponential number of polyhedral cones.

4.5 Construction of instances with inexact RLT relaxations

In this section, we propose an algorithm for constructing instances of (BoxQP) with an inexact RLT relaxation. Algorithm 2 is based on constructing a dual optimal solution of (R-D) in

Algorithm 2 (BoxQP) Instance with an Inexact RLT Relaxation

Require: n; $\mathbb{B} \subseteq \{1, \ldots, n\}$; $\mathbb{B} \neq \emptyset$
Ensure: $(Q, c) \notin \mathcal{E}_R$
1: Choose an arbitrary $\mathbb{L} \subseteq \{1, \ldots, n\} \backslash \mathbb{B}$
2: $\mathbb{U} \leftarrow \{1, \ldots, n\} \backslash (\mathbb{B} \cup \mathbb{L})$
3: Choose an arbitrary $k \in \mathbb{B}$.
4: Choose an arbitrary $\hat{r} \in \mathbb{R}^n$ such that $\hat{r}_{\mathbb{U}} \geq 0$ and $\hat{r}_{\mathbb{L} \cup \mathbb{B}} = 0$.
5: Choose an arbitrary $\hat{s} \in \mathbb{R}^n$ such that $\hat{s}_{\mathbb{L}} \geq 0$ and $\hat{s}_{\mathbb{B} \cup \mathbb{U}} = 0$.
6: Choose an arbitrary $\hat{W} \in \mathcal{S}^n$ such that $\hat{W}_{\mathbb{L}\mathbb{L}} = 0$, $\hat{W}_{\mathbb{L}\mathbb{B}} = 0$, $\hat{W}_{\mathbb{B}\mathbb{L}} = 0$, $\hat{W}_{kk} > 0$, and $\hat{W}_{ij} \geq 0$ otherwise.
7: Choose an arbitrary $\hat{Y} \in \mathbb{R}^{n \times n}$ such that $\hat{Y}_{\mathbb{L}\mathbb{B}} = 0$, $\hat{Y}_{\mathbb{L}\mathbb{U}} = 0$, $\hat{Y}_{\mathbb{B}\mathbb{B}} = 0$, $\hat{Y}_{\mathbb{B}\mathbb{U}} = 0$, and $\hat{Y}_{ij} \geq 0$ otherwise.
8: Choose an arbitrary $\hat{Z} \in \mathcal{S}^n$ such that $\hat{Z}_{\mathbb{B}\mathbb{U}} = 0$, $\hat{Z}_{\mathbb{U}\mathbb{B}} = 0$, $\hat{Z}_{\mathbb{U}\mathbb{U}} = 0$, $\hat{Z}_{kk} > 0$, and $\hat{Z}_{ij} \geq 0$ otherwise.
9: $Q \leftarrow \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z}$
10: $c \leftarrow -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e$

such a way that no feasible solution of the form $(v, vv^T) \in \mathcal{F}_R$ can be an optimal solution of (R), where $v \in F$ is a vertex.

The next result establishes that the output from Algorithm 2 is an instance of (BoxQP) with an inexact RLT relaxation.

Proposition 16 *Algorithm 2 returns $(Q, c) \notin \mathcal{E}_R$, where $\mathcal{E}_R$ is defined as in (39).*

Proof Consider the partition $(\mathbb{L}, \mathbb{B}, \mathbb{U})$ of the index set $\{1, \ldots, n\}$ as defined in Algorithm 2, where $\mathbb{B} \neq \emptyset$. Clearly, $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n \times n} \times \mathcal{S}^n$ is a feasible solution of (R-D). We will construct a feasible solution $(\hat{x}, \hat{X}) \in \mathbb{R}^n \times \mathcal{S}^n$ of (R) that satisfies the optimality conditions of Lemma 13.

Consider the following solution $(\hat{x}, \hat{X}) \in \mathbb{R}^n \times \mathcal{S}^n$:

$$\hat{x}_{\mathbb{L}} = 0, \quad \hat{x}_{\mathbb{B}} = \tfrac{1}{2} e_{\mathbb{B}} \quad \hat{x}_{\mathbb{U}} = e_{\mathbb{U}},$$

and

$$\begin{bmatrix} \hat{X}_{\mathbb{L}\mathbb{L}} & \hat{X}_{\mathbb{L}\mathbb{B}} & \hat{X}_{\mathbb{L}\mathbb{U}} \\ \hat{X}_{\mathbb{B}\mathbb{L}} & \hat{X}_{\mathbb{B}\mathbb{B}} & \hat{X}_{\mathbb{B}\mathbb{U}} \\ \hat{X}_{\mathbb{U}\mathbb{L}} & \hat{X}_{\mathbb{U}\mathbb{B}} & \hat{X}_{\mathbb{U}\mathbb{U}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} e_{\mathbb{B}} e_{\mathbb{U}}^T \\ 0 & \frac{1}{2} e_{\mathbb{U}} e_{\mathbb{B}}^T & e_{\mathbb{U}} e_{\mathbb{U}}^T \end{bmatrix}.$$

By Lemma 2, $(\hat{x}, \hat{X}) \in \mathcal{F}_R$. By Steps 4, 5, 6, 7, and 8 of Algorithm 2, it is easy to verify that (45), (46), (47), (48), and (49) are respectively satisfied. Therefore, by Lemma 13, we conclude that $(\hat{x}, \hat{X})$ is an optimal solution of (R) and $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z})$ is an optimal solution of (R-D).

We next argue that the RLT relaxation is inexact. Let $(\tilde{x}, \tilde{X}) \in \mathcal{F}_R$ be an arbitrary optimal solution of (R). By Lemma 2, $(\tilde{x}, \tilde{X})$ and $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z})$ satisfy the conditions (45), (46), (47), (48), and (49). By (49) and Step 8 of Algorithm 2, we obtain $\tilde{X}_{kk} = 0$ since $\hat{Z}_{kk} > 0$. Since $\hat{W}_{kk} > 0$ by Step 6 of Algorithm 2, the relation (47) implies that $\tilde{X}_{kk} - 2\tilde{x}_k + 1 = 0$, i.e., $\tilde{x}_k = \frac{1}{2}$ since $\tilde{X}_{kk} = 0$. By Lemma 13, we conclude that $\tilde{x}_k = \frac{1}{2}$ for each optimal solution $(\tilde{x}, \tilde{X}) \in \mathcal{F}_R$ of (R). By Corollary 11, we conclude that $(Q, c) \notin \mathcal{E}_R$. □

Algorithm 2 can be used to generate an instance of (BoxQP) with an inexact RLT relaxation. Note that Algorithm 2 constructs an instance (Q, c) with the property that there is an index k such that $\hat{x}_k = \frac{1}{2}$ at *every* optimal solution $(\hat{x}, \hat{X})$ of (R), which is sufficient for having an inexact RLT relaxation by Corollary 11 since each optimal solution $(\hat{x}, \hat{X})$ of (R) of an instance $(Q, c) \notin \mathcal{E}_R$ should have a fractional component $\hat{x}_k$ by Corollary 3.

However, an instance with an inexact RLT relaxation may not necessarily have the property that *every* optimal solution $(\hat{x}, \hat{X})$ of (R) has *the same* fractional component $\hat{x}_k$. In particular, note that an instance generated by Algorithm 2 cannot have a concave objective function since $Q_{kk} = \hat{W}_{kk} + \hat{Z}_{kk} > 0$. On the other hand, for the specific instance $(Q, c) \in \mathcal{S}^3 \times \mathbb{R}^3$ in [18] given by $Q = \frac{1}{3}ee^T - I$, where $I \in \mathcal{S}^3$ denotes the identity matrix, and $c = 0$, the objective function is concave and the optimal value is given by $\ell^* = -\frac{1}{3}$, which is attained at any vertex that has exactly one component equal to 1. For $\hat{x} = \frac{1}{2}e \in \mathbb{R}^3$, we have $\ell_R(\hat{x}) = -\frac{1}{2} < -\frac{1}{3} = \ell^*$ by Lemma 1, which implies that the RLT relaxation is inexact on this instance. Therefore, in contrast with Algorithm 1, we conclude that Algorithm 2 may not necessarily generate all possible instances $(Q, c) \notin \mathcal{E}_R$.

5 Exact and inexact SDP-RLT relaxations

In this section, we focus on the set of instances of (BoxQP) that admit exact and inexact SDP-RLT relaxations. We give a complete algebraic description of the set of instances of (BoxQP) that admit an exact SDP-RLT relaxation. In addition, we develop an algorithm for constructing such an instance of (BoxQP) as well as for constructing an instance of (BoxQP) with an exact SDP-RLT relaxation but an inexact RLT relaxation.

Similar to the RLT relaxation, let us define

$$\mathcal{E}_{RS} = \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell^* = \ell_{RS}\}, \tag{55}$$

i.e., $\mathcal{E}_{RS}$ denotes the set of all instances of (BoxQP) that admit an exact SDP-RLT relaxation. By (4), the SDP-RLT relaxation of any instance of (BoxQP) is at least as tight as the RLT relaxation. It follows that

$$\mathcal{E}_R \subseteq \mathcal{E}_{RS}, \tag{56}$$

where $\mathcal{E}_R$ is given by (39).

By Corollary 6 and (28), we clearly have $(Q, c) \in \mathcal{E}_{RS}$ whenever $Q \succeq 0$. Furthermore, the SDP-RLT relaxation is always exact (i.e., $\mathcal{E}_{RS} = \mathcal{S}^n \times \mathbb{R}^n$) if and only if $n \leq 2$ [18].

For the RLT relaxation, Proposition 7 established the existence of an optimal solution of the RLT relaxation with a particularly simple structure. This observation enabled us to characterize the set of instances of (BoxQP) with an exact RLT relaxation as the union of a finite number polyhedral cones (see Proposition 12). In contrast, the next result shows that the set of optimal solutions of the SDP-RLT relaxation cannot have such a simple structure.

Lemma 17 *For any $\hat{x} \in F$, there exists an instance (Q, c) of (BoxQP) such that $\hat{x}$ is the unique optimal solution of (RS1), where (RS1) is given by (28), and $(Q, c) \in \mathcal{E}_{RS}$.*

Proof For any $\hat{x} \in F$, consider an instance of (BoxQP) with $(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n$, where $Q \succ 0$ and $c = -Q\hat{x}$. We obtain $q(x) = \frac{1}{2}\left((x - \hat{x})^T Q(x - \hat{x}) - \hat{x}^T Q\hat{x}\right)$. Since $Q \succ 0$, $\hat{x}$ is the unique unconstrained minimizer of $q(x)$. By Lemma 5, since $Q \succ 0$, we have $\ell_{RS}(x) = q(x)$ for each $x \in F$, which implies that $\ell^* = q(\hat{x}) = \ell_{RS}(\hat{x}) = \ell^*_{RS}$. It follows that $(Q, c) \in \mathcal{E}_{RS}$. The uniqueness follows from the strict convexity of $\ell_{RS}(\cdot)$ since $\ell_{RS}(x) = q(x)$ for each $x \in F$. □

In the next section, we rely on duality theory to obtain a description of the set $\mathcal{E}_{RS}$.

5.1 The dual problem

In this section, we present the dual of the SDP-RLT relaxation given by (RS) and establish several useful properties.

Recall that the SDP-RLT relaxation is given by

$$\text{(RS)} \quad \ell^*_{RS} = \min_{(x,X)\in\mathbb{R}^n\times\mathcal{S}^n} \left\{ \frac{1}{2}\langle Q, X\rangle + c^T x : (x, X) \in \mathcal{F}_{RS} \right\},$$

where $\mathcal{F}_{RS}$ is given by (3).

By the Schur complement property, we have

$$X - xx^T \succeq 0 \Longleftrightarrow \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0, \tag{57}$$

which implies that (RS) can be formulated as a linear semidefinite programming problem.

By using the same set of dual variables $(r, s, W, Y, Z) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n$ as in (R-D) corresponding to the common constraints in $\mathcal{F}_R$ and $\mathcal{F}_{RS}$ (see (42)), and defining the dual variable

$$\begin{bmatrix} \beta & h^T \\ h & H \end{bmatrix} \in \mathcal{S}^{n+1},$$

where $\beta \in \mathbb{R}$, $h \in \mathbb{R}^n$, and $H \in \mathcal{S}^n$, corresponding to the additional semidefinite constraint in (57), the dual problem of (RS) is given by

$$\begin{array}{lll} \text{(RS-D)} & \displaystyle\max_{\substack{(r,s,W,Y,Z,\beta,h,H) \\ \in\mathbb{R}^n\times\mathbb{R}^n\times\mathcal{S}^n\times\mathbb{R}^{n\times n}\times\mathcal{S}^n\times\mathbb{R}\times\mathbb{R}^n\times\mathcal{S}^n}} & -e^T r - \frac{1}{2}e^T W e - \frac{1}{2}\beta \\ & \text{s.t.} & \\ & & -r + s - We + Y^T e + h = c \\ & & W - Y - Y^T + Z + H = Q \\ & & r \geq 0 \\ & & s \geq 0 \\ & & W \geq 0 \\ & & Y \geq 0 \\ & & Z \geq 0 \\ & & \begin{bmatrix} \beta & h^T \\ h & H \end{bmatrix} \succeq 0. \end{array}$$

In contrast with linear programming, strong duality and attainment of optimal solutions may fail in semidefinite programming (see, e.g., [35]). We first establish that (RS) and (RS-D) satisfy well-known sufficient conditions for strong duality and attainment.

Lemma 18 *Strong duality holds between (RS) and (RS-D), and optimal solutions are attained in both (RS) and (RS-D).*

Proof Note that $\mathcal{F}_{RS}$ is a nonempty and bounded set since $0 \leq x_j \leq 1$ and $X_{jj} \leq 1$ for each $j = 1, \ldots, n$. Therefore, the set of optimal solutions of (RS) is nonempty. Let $\hat{x} = \frac{1}{2}e \in \mathbb{R}^n$ and let $\hat{X} = \hat{x}\hat{x}^T + \epsilon I \in \mathcal{S}^n$, where $\epsilon \in (0, \frac{1}{4})$. By Lemma 2, $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$. Furthermore, it is a strictly feasible solution of (RS) since $(\hat{x}, \hat{X})$ satisfies all the constraints strictly. Strong duality and attainment in (RS-D) follow from conic duality (see, e.g., [36]). □

Lemma 18 allows us to give a complete characterization of optimality conditions for the pair (RS) and (RS-D).

Lemma 19 *$(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$ is an optimal solution of (RS) if and only if there exists*

$$(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}, \hat{\beta}, \hat{h}, \hat{H}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n \times \mathbb{R} \times \mathbb{R}^n \times \mathcal{S}^n$$

such that

$$Q = \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z} + \hat{H} \tag{58}$$

$$c = -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e + \hat{h} \tag{59}$$

$$\begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} \succeq 0 \tag{60}$$

$$\left\langle \begin{bmatrix} 1 & \hat{x}^T \\ \hat{x} & \hat{X} \end{bmatrix}, \begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} \right\rangle = 0, \tag{61}$$

and (45)–(54) *are satisfied.*

Proof The claim follows from strong duality between (RS) and (RS-D), which holds by Lemma 18. □

Using Lemma 19, we obtain the following description of the set of instances of (BoxQP) with an exact SDP-RLT relaxation.

Proposition 20 *$(Q, c) \in \mathcal{E}_{RS}$, where $\mathcal{E}_{RS}$ is defined as in (55), if and only if there exists $\hat{x} \in F$ and there exists $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}, \hat{\beta}, \hat{h}, \hat{H}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n \times \mathbb{R} \times \mathbb{R}^n \times \mathcal{S}^n$ such that the conditions of Lemma 19 are satisfied, where $(\hat{x}, \hat{X}) = (\hat{x}, \hat{x}\hat{x}^T)$. Furthermore, in this case, $\hat{x}$ is an optimal solution of (BoxQP).*

Proof Suppose that $(Q, c) \in \mathcal{E}_{RS}$. Let $\hat{x} \in F$ be an optimal solution of (BoxQP). By Corollary 6(ii), we obtain $\ell^*_{RS} = \ell^* = q(\hat{x}) = \ell_{RS}(\hat{x})$. Therefore, $\hat{x}$ is an optimal solution of (RS1) given by (28). Let $(\hat{x}, \hat{X}) = (\hat{x}, \hat{x}\hat{x}^T) \in \mathcal{F}_{RS}$. We obtain $\frac{1}{2}\langle Q, \hat{X}\rangle + c^T\hat{x} = q(\hat{x}) = \ell^*_{RS}$, which implies that $(\hat{x}, \hat{x}\hat{x}^T)$ is an optimal solution of (RS). The claim follows from Lemma 19.

For the reverse implication, note that $(\hat{x}, \hat{X}) = (\hat{x}, \hat{x}\hat{x}^T)$ is an optimal solution of (RS) by Lemma 19. By a similar argument and using (4), we obtain $\ell^* \le q(\hat{x}) = \ell^*_{RS} \le \ell^*$, which implies that $\ell^*_{RS} = \ell^*$, or equivalently, that $(Q, c) \in \mathcal{E}_{RS}$.

The second assertion follows directly from the previous arguments. □

In the next section, by relying on Proposition 20, we propose two algorithms to construct instances of (BoxQP) with different exactness guarantees.

5.2 Construction of instances with exact SDP-RLT relaxations

In this section, we present an algorithm for constructing instances of (BoxQP) with an exact SDP-RLT relaxation. Similar to Algorithm 1, Algorithm 3 is based on designating $\hat{x} \in F$ and constructing an appropriate dual feasible solution that satisfies optimality conditions together with $(\hat{x}, \hat{x}\hat{x}^T) \in \mathcal{F}_{RS}$.

The next proposition establishes the correctness of Algorithm 3.

Proposition 21 *Algorithm 3 returns $(Q, c) \in \mathcal{E}_{RS}$, where $\mathcal{E}_{RS}$ is defined as in (55). Conversely, any $(Q, c) \in \mathcal{E}_{RS}$ can be generated by Algorithm 3 with appropriate choices of $\hat{x} \in F$ and $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}, \hat{\beta}, \hat{h}, \hat{H}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n \times \mathbb{R} \times \mathbb{R}^n \times \mathcal{S}^n$.*

Algorithm 3 (BoxQP) Instance with an Exact SDP-RLT Relaxation

Require: $n, \hat{x} \in F$
Ensure: $(Q, c) \in \mathcal{E}_{RS}$
1: $\mathbb{L} \leftarrow \mathbb{L}(\hat{x}), \mathbb{B} \leftarrow \mathbb{B}(\hat{x}), \mathbb{U} \leftarrow \mathbb{U}(\hat{x})$
2: Choose an arbitrary $\hat{r} \in \mathbb{R}^n$ such that $\hat{r}_{\mathbb{U}} \geq 0$ and $\hat{r}_{\mathbb{L}\cup\mathbb{B}} = 0$.
3: Choose an arbitrary $\hat{s} \in \mathbb{R}^n$ such that $\hat{s}_{\mathbb{L}} \geq 0$ and $\hat{s}_{\mathbb{B}\cup\mathbb{U}} = 0$.
4: Choose an arbitrary $\hat{W} \in \mathcal{S}^n$ such that $\hat{W}_{\mathbb{B}\cup\mathbb{L},\mathbb{B}\cup\mathbb{L}} = 0$ and $\hat{W}_{ij} \geq 0$ otherwise.
5: Choose an arbitrary $\hat{Y} \in \mathbb{R}^{n\times n}$ such that $\hat{Y}_{\mathbb{L}\mathbb{B}} = 0,\ \hat{Y}_{\mathbb{L}\mathbb{U}} = 0,\ \hat{Y}_{\mathbb{B}\mathbb{B}} = 0,\ \hat{Y}_{\mathbb{B}\mathbb{U}} = 0$ and $\hat{Y}_{ij} \geq 0$ otherwise.
6: Choose an arbitrary $\hat{Z} \in \mathcal{S}^n$ such that $\hat{Z}_{\mathbb{B}\cup\mathbb{U},\mathbb{B}\cup\mathbb{U}} = 0$ and $\hat{Z}_{ij} \geq 0$ otherwise.
7: Choose an arbitrary $\hat{H} \in \mathcal{S}^n$ such that $\hat{H} \succeq 0$.
8: $\hat{h} \leftarrow -\hat{H}\hat{x}, \hat{\beta} \leftarrow -\hat{h}^T\hat{x}$
9: $Q \leftarrow \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z} + \hat{H}$
10: $c \leftarrow -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e + \hat{h}$

Proof Since $\hat{H} \succeq 0$, it follows from Steps 8 and 9 of Algorithm 3 that

$$\begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} = \begin{bmatrix} \hat{x}^T \\ -I \end{bmatrix} \hat{H} \begin{bmatrix} \hat{x}^T \\ -I \end{bmatrix}^T \succeq 0. \tag{62}$$

Therefore, $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}, \hat{\beta}, \hat{h}, \hat{H}) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathcal{S}^n \times \mathbb{R}^{n\times n} \times \mathcal{S}^n \times \mathbb{R} \times \mathbb{R}^n \times \mathcal{S}^n$ is a feasible solution of (RS-D). Furthermore, the identity in (62) also implies that

$$\left\langle \begin{bmatrix} 1 & \hat{x}^T \\ \hat{x} & \hat{x}\hat{x}^T \end{bmatrix}, \begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} \right\rangle = \begin{bmatrix} 1 \\ \hat{x} \end{bmatrix}^T \begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} \begin{bmatrix} 1 \\ \hat{x} \end{bmatrix} = 0.$$

It is easy to verify that the conditions of Lemma 19 are satisfied with $(\hat{x}, \hat{X}) = (\hat{x}, \hat{x}\hat{x}^T) \in \mathcal{F}_{RS}$. Both assertions follow from Proposition 20. □

By Proposition 21, we conclude that $\mathcal{E}_{RS}$ is given by the union of infinitely many convex cones each of which can be represented by semidefinite and linear constraints.

Similar to Algorithm 1, we remark that Algorithm 3 can be utilized to generate an instance of (BoxQP) with an exact SDP-RLT relaxation such that any designated feasible solution $\hat{x} \in F$ is an optimal solution of (BoxQP).

5.3 Construction of instances with exact SDP-RLT and inexact RLT relaxations

Recall that the SDP-RLT relaxation of any instance of (BoxQP) is at least as tight as the RLT relaxation. In this section, we present another algorithm for constructing instances of (BoxQP) that admit an exact SDP-RLT relaxation but an inexact RLT relaxation, i.e., an instance in $\mathcal{E}_{RS}\backslash\mathcal{E}_R$ (cf. (56)). In particular, this algorithm can be used to construct instances of (BoxQP) such that the SDP-RLT relaxation not only strengthens the RLT relaxation, but also yields an exact lower bound.

Note that Algorithm 3 is capable of constructing all instances of (BoxQP) in the set $\mathcal{E}_{RS}$. On the other hand, if one chooses $\hat{x} \in V$ and $\hat{H} = 0$ in Algorithm 3, which, in turn, would imply that $\hat{h} = 0$ and $\hat{\beta} = 0$, it is easy to verify that the choices of the remaining parameters satisfy the conditions of Algorithm 1, which implies that the resulting instance would already have an exact RLT relaxation, i.e., $(Q, c) \in \mathcal{E}_R$.

In this section, we present Algorithm 4, where we use a similar idea as in Algorithm 2, i.e., we aim to construct an instance of (BoxQP) such that $(\hat{x}, \hat{x}\hat{x}^T)$ is the unique optimal solution of (RS), where $\hat{x} \in F\backslash V$.

Algorithm 4 (BoxQP) Instance with an Exact SDP-RLT Relaxation and an Inexact RLT Relaxation

Require: $n, \hat{x} \in F \backslash V$
Ensure: $(Q, c) \in \mathcal{E}_{RS} \backslash \mathcal{E}_R$
1: $\mathbb{L} \leftarrow \mathbb{L}(\hat{x}), \mathbb{B} \leftarrow \mathbb{B}(\hat{x}), \mathbb{U} \leftarrow \mathbb{U}(\hat{x})$
2: Choose an arbitrary $\hat{r} \in \mathbb{R}^n$ such that $\hat{r}_{\mathbb{U}} \geq 0$ and $\hat{r}_{\mathbb{L} \cup \mathbb{B}} = 0$.
3: Choose an arbitrary $\hat{s} \in \mathbb{R}^n$ such that $\hat{s}_{\mathbb{L}} \geq 0$ and $\hat{s}_{\mathbb{B} \cup \mathbb{U}} = 0$.
4: Choose an arbitrary $\hat{W} \in \mathcal{S}^n$ such that $\hat{W}_{\mathbb{B} \cup \mathbb{L}, \mathbb{B} \cup \mathbb{L}} = 0$ and $\hat{W}_{ij} \geq 0$ otherwise.
5: Choose an arbitrary $\hat{Y} \in \mathbb{R}^{n \times n}$ such that $\hat{Y}_{\mathbb{L}\mathbb{B}} = 0,\ \hat{Y}_{\mathbb{L}\mathbb{U}} = 0,\ \hat{Y}_{\mathbb{B}\mathbb{B}} = 0,\ \hat{Y}_{\mathbb{B}\mathbb{U}} = 0$ and $\hat{Y}_{ij} \geq 0$ otherwise.
6: Choose an arbitrary $\hat{Z} \in \mathcal{S}^n$ such that $\hat{Z}_{\mathbb{B} \cup \mathbb{U}, \mathbb{B} \cup \mathbb{U}} = 0$ and $\hat{Z}_{ij} \geq 0$ otherwise.
7: Choose an arbitrary $\hat{H} \in \mathcal{S}^n$ such that $\hat{H} \succ 0$.
8: $\hat{h} \leftarrow -\hat{H}\hat{x}, \hat{\beta} \leftarrow -\hat{h}^T \hat{x}$
9: $Q \leftarrow \hat{W} - \hat{Y} - \hat{Y}^T + \hat{Z} + \hat{H}$
10: $c \leftarrow -\hat{r} + \hat{s} - \hat{W}e + \hat{Y}^T e + \hat{h}$

Note that Algorithm 3 and Algorithm 4 are almost identical, except that, in Step 7, we require that $\hat{H} \succ 0$ in Algorithm 4 as opposed to $\hat{H} \succeq 0$ in Algorithm 3. The next result establishes that the output from Algorithm 4 is an instance of (BoxQP) with an exact SDP-RLT but inexact RLT relaxation.

Proposition 22 *Algorithm 4 returns $(Q, c) \in \mathcal{E}_{RS} \backslash \mathcal{E}_R$, where $\mathcal{E}_R$ and $\mathcal{E}_{RS}$ are defined as in (39) and (55), respectively.*

Proof By the observation preceding the statement, it follows from Propositions 20 and 21 that $(Q, c) \in \mathcal{E}_{RS}$ and that $(\hat{x}, \hat{x}\hat{x}^T)$ is an optimal solution of (RS). First, we show that this is the unique optimal solution of (RS). Suppose, for a contradiction, that there exists another optimal solution $(\tilde{x}, \tilde{X}) \in \mathcal{F}_{RS}$. Note that, for any $A \succeq 0$ and $B \succeq 0$, $\langle A, B \rangle = 0$ holds if and only if $AB = 0$. Therefore, it follows from (61) that $\hat{h} - \hat{H}\tilde{x} = 0$. Since $\hat{H} \succ 0$, we obtain $\tilde{x} = \hat{x}$ by Step 8. By (62), we obtain

$$\left\langle \begin{bmatrix} 1 & \hat{x}^T \\ \hat{x} & \tilde{X} \end{bmatrix}, \begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} \right\rangle = \left\langle \begin{bmatrix} 1 & \hat{x}^T \\ \hat{x} & \tilde{X} \end{bmatrix}, \begin{bmatrix} \hat{x}^T \\ -I \end{bmatrix} \hat{H} \begin{bmatrix} \hat{x}^T \\ -I \end{bmatrix}^T \right\rangle = \langle \hat{H}, \tilde{X} - \hat{x}\hat{x}^T \rangle = 0.$$

Since $\hat{H} \succ 0$ by Step 7 and $\tilde{X} - \hat{x}\hat{x}^T \succeq 0$, it follows that $\tilde{X} = \hat{x}\hat{x}^T$, which contradicts our assumption. It follows that $(\hat{x}, \hat{x}\hat{x}^T)$ is the unique optimal solution of (RS), or equivalently, that $\hat{x}$ is the unique optimal solution of (RS1) given by (28). By Proposition 20 and (26), we conclude that $(Q, c) \in \mathcal{E}_{RS}$ and that $\hat{x} \in F \backslash V$ is the unique optimal solution of (BoxQP). By Corollary 11, $(Q, c) \notin \mathcal{E}_R$, which completes the proof. □

Algorithm 4 can be used to construct an instance in the set $\mathcal{E}_{RS} \backslash \mathcal{E}_R$. In particular, we remark that the family of instances used in the proof of Lemma 17 can be constructed by Algorithm 4 by simply choosing $(\hat{r}, \hat{s}, \hat{W}, \hat{Y}, \hat{Z}) = (0, 0, 0, 0, 0)$ and any $\hat{x} \in F$. In particular, similar to Algorithm 2, it is worth noting that any instance constructed by Algorithm 4 necessarily satisfies $Q_{kk} > 0$ for each $k \in \mathbb{B}$. On the other hand, recall that the SDP-RLT relaxation is always exact for $n \leq 2$. It follows that the instance in Example 1 belongs to $\mathcal{E}_{RS} \backslash \mathcal{E}_R$. However, it cannot be constructed by Algorithm 4 since (RS) does not have a unique optimal solution. Therefore, similar to our discussion about Algorithm 2, we conclude that the set of instances that can be constructed by Algorithm 4 may not necessarily encompass all instances in $\mathcal{E}_{RS} \backslash \mathcal{E}_R$.

5.4 A stronger relaxation

In this section, we present a stronger version of the SDP-RLT relaxation and briefly discuss the implications of our results on this relaxation.

The SDP-RLT relaxation (RS) can be further strengthened by adding the so-called triangle inequalities (see [15, 16, 21]):

$$\mathcal{F}_{RST} = \left\{ (x, X) \in \mathcal{F}_{RS} : \begin{array}{rl} x_i + x_j + x_k - X_{ij} - X_{jk} - X_{ik} \leq 1, & 1 \leq i < j < k \leq n \\ X_{ij} + X_{ik} - x_i - X_{jk} \leq 0, & 1 \leq i < j < k \leq n \\ X_{ij} + X_{jk} - x_j - X_{ik} \leq 0, & 1 \leq i < j < k \leq n \\ X_{ik} + X_{jk} - x_k - X_{ij} \leq 0, & 1 \leq i < j < k \leq n \end{array} \right\}. \tag{63}$$

We can now define the following relaxation, referred to as the SDP-RLT-TRI relaxation:

$$\text{(RST)} \quad \ell^*_{RST} = \min_{(x,X) \in \mathbb{R}^n \times \mathcal{S}^n} \left\{ \frac{1}{2} \langle Q, X \rangle + c^T x : (x, X) \in \mathcal{F}_{RST} \right\}.$$

Clearly, $\ell^*_{RS} \leq \ell^*_{RST} \leq \ell^*$ since $\mathcal{F}_{RST} \subseteq \mathcal{F}_{RS}$. Let us similarly define

$$\mathcal{E}_{RST} = \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell^* = \ell_{RST}\}. \tag{64}$$

We readily obtain $\mathcal{E}_R \subseteq \mathcal{E}_{RS} \subseteq \mathcal{E}_{RST}$. In fact, computational results indicate that this stronger relaxation is usually exact on small- to medium-scale instances (see, e.g., [20, 37, 38]).

Denoting the Lagrangian dual of (RST) by (RST-D), it is easy to show that Lemma 18 also holds for this primal-dual pair. Therefore, by relying on the corresponding versions of Lemma 19 and Proposition 20 for this relaxation, one can easily extend Algorithm 3 and Algorithm 4 to construct an instance of (BoxQP) with an exact SDP-RLT-TRI relaxation and an instance with an exact SDP-RLT-TRI but inexact RLT relaxation.

6 Examples and discussion

In this section, we present numerical examples generated by each of the four algorithms given by Algorithms 1–4. We then report computational results on randomly generated instances using Algorithms 1–4 and close the section with a brief discussion.

6.1 Examples

In this section, we present instances of (BoxQP) generated by each of the four algorithms given by Algorithms 1–4. Our main goal is to demonstrate that our algorithms are capable of generating nontrivial instances of (BoxQP) with predetermined exactness or inexactness guarantees.

Example 2 Let $n = 2$, $\mathbb{L} = \{1\}$, and $\mathbb{U} = \{2\}$ in Algorithm 1. Then, by Steps 2–6, we have

$$\hat{r} = \begin{bmatrix} 0 \\ \hat{r}_2 \end{bmatrix}, \quad \hat{s} = \begin{bmatrix} \hat{s}_1 \\ 0 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} 0 & \hat{W}_{12} \\ \hat{W}_{12} & \hat{W}_{22} \end{bmatrix}, \quad \hat{Y} = \begin{bmatrix} \hat{Y}_{11} & 0 \\ \hat{Y}_{21} & \hat{Y}_{22} \end{bmatrix}, \quad \hat{Z} = \begin{bmatrix} \hat{Z}_{11} & \hat{Z}_{12} \\ \hat{Z}_{12} & 0 \end{bmatrix},$$

where each of $\hat{r}_2, \hat{s}_1, \hat{W}_{12}, \hat{W}_{22}, \hat{Y}_{11}, \hat{Y}_{21}, \hat{Y}_{22}, \hat{Z}_{11}, \hat{Z}_{12}$ is a nonnegative real number. By Steps 7 and 8, we obtain

$$Q = \begin{bmatrix} \hat{Z}_{11} - 2\hat{Y}_{11} & \hat{W}_{12} + \hat{Z}_{12} - \hat{Y}_{21} \\ \hat{W}_{12} + \hat{Z}_{12} - \hat{Y}_{21} & \hat{W}_{22} - 2\hat{Y}_{22} \end{bmatrix}, \quad c = \begin{bmatrix} \hat{s}_1 + \hat{Y}_{11} + \hat{Y}_{21} - \hat{W}_{12} \\ \hat{Y}_{22} - \hat{r}_2 - \hat{W}_{12} - \hat{W}_{22} \end{bmatrix}.$$

For instance, if we choose $\hat{Y}_{11} = \hat{Y}_{22} = \hat{W}_{12} = \hat{Z}_{12} = \hat{Y}_{21} = 0$, $\hat{Z}_{11} + \hat{W}_{22} > 0$, $\hat{r}_2 \geq 0$, and $\hat{s}_1 \geq 0$, then $Q \succeq 0$, which implies that $q(x)$ is a convex function. If we choose $\hat{Z}_{11} = \hat{W}_{22} = \hat{W}_{12} = \hat{Z}_{12} = \hat{Y}_{21} = 0$, $\hat{Y}_{11} + \hat{Y}_{22} > 0$, $\hat{r}_2 \geq 0$, and $\hat{s}_1 \geq 0$, then $-Q \succeq 0$, which implies that $q(x)$ is a concave function. Finally, if we choose $\hat{Y}_{11} = \hat{W}_{22} = \hat{W}_{12} = \hat{Z}_{12} = \hat{Y}_{21} = 0$, $\hat{Z}_{11} > 0$, $\hat{Y}_{22} > 0$, $\hat{r}_2 \geq 0$, and $\hat{s}_1 \geq 0$, then Q is indefinite, which implies that $q(x)$ is an indefinite quadratic function. For each of the three choices, the RLT relaxation is exact and $\hat{x} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$ is an optimal solution of the resulting instance of (BoxQP). Note that by choosing $\hat{x} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$ and setting $\hat{H} = 0$ and $\hat{h} = 0$ in Algorithm 3, the same observations carry over.

Example 3 Let $n = 3$, $\mathbb{L} = \{1\}$, $\mathbb{B} = \{2\}$, and $\mathbb{U} = \{3\}$ in Algorithm 2. Then, by Steps 3–8, we have $k = 2$ and

$$\hat{r} = \begin{bmatrix} 0 \\ 0 \\ \hat{r}_3 \end{bmatrix}, \quad \hat{s} = \begin{bmatrix} \hat{s}_1 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} 0 & 0 & \hat{W}_{13} \\ 0 & \hat{W}_{22} & \hat{W}_{23} \\ \hat{W}_{13} & \hat{W}_{23} & \hat{W}_{33} \end{bmatrix}, \quad \hat{Y} = \begin{bmatrix} \hat{Y}_{11} & 0 & 0 \\ \hat{Y}_{21} & 0 & 0 \\ \hat{Y}_{31} & \hat{Y}_{32} & \hat{Y}_{33} \end{bmatrix}, \quad \hat{Z} = \begin{bmatrix} \hat{Z}_{11} & \hat{Z}_{12} & \hat{Z}_{13} \\ \hat{Z}_{12} & \hat{Z}_{22} & 0 \\ \hat{Z}_{13} & 0 & 0 \end{bmatrix},$$

where each of $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{11}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Y}_{33}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ is a nonnegative real number, $\hat{W}_{22} > 0$ and $\hat{Z}_{22} > 0$. By Steps 9 and 10, we obtain

$$Q = \begin{bmatrix} \hat{Z}_{11} - 2\hat{Y}_{11} & \hat{Z}_{12} - \hat{Y}_{21} & \hat{W}_{13} + \hat{Z}_{13} - \hat{Y}_{31} \\ \hat{Z}_{12} - \hat{Y}_{21} & \hat{W}_{22} + \hat{Z}_{22} & \hat{W}_{23} - \hat{Y}_{32} \\ \hat{W}_{13} + \hat{Z}_{13} - \hat{Y}_{31} & \hat{W}_{23} - \hat{Y}_{32} & \hat{W}_{33} - 2\hat{Y}_{33} \end{bmatrix}, \quad c = \begin{bmatrix} \hat{s}_1 + \hat{Y}_{11} + \hat{Y}_{21} + \hat{Y}_{31} - \hat{W}_{13} \\ \hat{Y}_{32} - \hat{W}_{22} - \hat{W}_{23} \\ \hat{Y}_{33} - \hat{W}_{13} - \hat{W}_{23} - \hat{W}_{33} - \hat{r}_3 \end{bmatrix}.$$

If we set each of the parameters $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{11}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Y}_{33}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ to zero, and choose any $\hat{W}_{22} > 0$ and $\hat{Z}_{22} > 0$, then $Q \succeq 0$, which implies that $q(x)$ is a convex function. On the other hand, if we set each of the parameters $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ to zero, and choose any $\hat{W}_{22} > 0$, $\hat{Z}_{22} > 0$ and $\hat{Y}_{11} + \hat{Y}_{33} > 0$, then Q is indefinite, which implies that $q(x)$ is an indefinite quadratic function. For each of the two choices, the RLT relaxation is inexact. Recall that an instance generated by Algorithm 2 cannot have a concave objective function since $Q_{kk} = \hat{W}_{kk} + \hat{Z}_{kk} > 0$.

Example 4 Let $n = 3$ and let $\hat{x} \in V \backslash F$ be such that $\mathbb{L} = \{1\}$, $\mathbb{B} = \{2\}$, and $\mathbb{U} = \{3\}$ in Algorithm 4. Then, by Steps 2–6, we have $k = 2$ and

$$\hat{r} = \begin{bmatrix} 0 \\ 0 \\ \hat{r}_3 \end{bmatrix}, \quad \hat{s} = \begin{bmatrix} \hat{s}_1 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{W} = \begin{bmatrix} 0 & 0 & \hat{W}_{13} \\ 0 & 0 & \hat{W}_{23} \\ \hat{W}_{13} & \hat{W}_{23} & \hat{W}_{33} \end{bmatrix}, \quad \hat{Y} = \begin{bmatrix} \hat{Y}_{11} & 0 & 0 \\ \hat{Y}_{21} & 0 & 0 \\ \hat{Y}_{31} & \hat{Y}_{32} & \hat{Y}_{33} \end{bmatrix}, \quad \hat{Z} = \begin{bmatrix} \hat{Z}_{11} & \hat{Z}_{12} & \hat{Z}_{13} \\ \hat{Z}_{12} & 0 & 0 \\ \hat{Z}_{13} & 0 & 0 \end{bmatrix},$$

where each of $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{11}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Y}_{33}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ is a nonnegative real number. By Step 7, $\hat{H} \succ 0$ is arbitrarily chosen. By Step 8, we have $\hat{h} = -\hat{H}\hat{x}$ and $\hat{\beta} = -\hat{h}^T \hat{x}$. By Steps 9 and 10, we therefore obtain

$$Q = \begin{bmatrix} \hat{Z}_{11} - 2\hat{Y}_{11} & \hat{Z}_{12} - \hat{Y}_{21} & \hat{W}_{13} + \hat{Z}_{13} - \hat{Y}_{31} \\ \hat{Z}_{12} - \hat{Y}_{21} & 0 & \hat{W}_{23} - \hat{Y}_{32} \\ \hat{W}_{13} + \hat{Z}_{13} - \hat{Y}_{31} & \hat{W}_{23} - \hat{Y}_{32} & \hat{W}_{33} - 2\hat{Y}_{33} \end{bmatrix} + \hat{H},$$

$$c = \begin{bmatrix} \hat{s}_1 + \hat{Y}_{11} + \hat{Y}_{21} + \hat{Y}_{31} - \hat{W}_{13} \\ \hat{Y}_{32} - \hat{W}_{23} \\ \hat{Y}_{33} - \hat{W}_{13} - \hat{W}_{23} - \hat{W}_{33} - \hat{r}_3 \end{bmatrix} + \hat{h}.$$

If we set each of the parameters $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{11}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Y}_{33}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ to zero, then $Q = \hat{H} \succ 0$, which implies that $q(x)$ is a strictly convex function. On the other hand, if we set each of the parameters $\hat{r}_3, \hat{s}_1, \hat{W}_{13}, \hat{W}_{23}, \hat{W}_{33}, \hat{Y}_{21}, \hat{Y}_{31}, \hat{Y}_{32}, \hat{Z}_{11}, \hat{Z}_{12}, \hat{Z}_{13}$ to zero, and choose $\hat{Y}_{11} + \hat{Y}_{33} = \frac{1}{2}(\hat{H}_{11} + \hat{H}_{22} + \hat{H}_{33}) > 0$, then Q is indefinite since $\text{trace}(Q) = 0$ and $Q \neq 0$ which implies that $q(x)$ is an indefinite quadratic function. For each of the two choices, $\hat{x}$ is the unique optimal solution of the resulting instance of (BoxQP) and the SDP-RLT relaxation is exact whereas the RLT relaxation is inexact. Recall that an instance generated by Algorithm 4 cannot have a concave objective function since $Q_{kk} = \hat{H}_{kk} > 0$. Indeed, such an instance of (BoxQP) necessarily has an optimal solution at a vertex whereas Algorithm 4 ensures that the resulting instance of (BoxQP) has a unique solution $\hat{x} \in F \backslash V$.

6.2 Computational experiments

In this section, in an attempt to shed light on the computational cost of globally solving instances of (BoxQP) generated by Algorithms 1–4 using a state-of-the-art solver, we report preliminary results.

In our experiments, we chose $n \in \{25, 50, 75, 100\}$. For each choice of n and each of the four algorithms given by Algorithms 1–4, we generated 100 random instances of (BoxQP), giving rise to a total of 1600 instances. For each algorithm, all nonnegative (resp., positive) parameters were generated uniformly from the set of integers $\{0, 1, \ldots, 10\}$ (resp., $\{1, 2, \ldots, 10\}$). For Algorithm 1 (resp., Algorithms 2–4), each component was assigned to index sets $\mathbb{L}$ and $\mathbb{U}$ (resp, $\mathbb{L}$, $\mathbb{B}$, and $\mathbb{U}$) with equal probabilities while ensuring that $\mathbb{B} \neq \emptyset$ in Algorithms 2–4. For Algorithms 3 and 4, each $\hat{x}_j,\ j \in \mathbb{B}$ was chosen uniformly from the finite set $\{0.01, 0.02, \ldots, 0.99\}$. Regarding the matrices $\hat{H} \succeq 0$ in Algorithm 3 and $\hat{H} \succ 0$ in Algorithm 4, we generated a random matrix $\hat{A} \in \mathbb{R}^{n \times n}$ whose entries are uniformly chosen from the set $\{-5, -4, \ldots, 4, 5\}$ and computed its QR-factorization $\hat{A} = \hat{Q}\hat{R}$. We then generated a diagonal matrix $\hat{\Lambda} \in \mathcal{S}^n$ with entries uniformly chosen from the sets $\{0, 1, \ldots, 10\}$ and $\{1, 2, \ldots, 10\}$ for Algorithm 3 and Algorithm 4, respectively, and set $\hat{H} = \hat{Q}\hat{\Lambda}\hat{Q}^T$.

We implemented our algorithms in Julia version 1.8.5 [39] and solved the instances of (BoxQP) calling CPLEX version 22.1.1.0 [12] via the modeling language JuMP [40]. We imposed a time limit of 600 s on each instance. Our computational experiments were carried out on a 64-bit HP workstation with 24 threads (2 sockets, 6 cores per socket, 2 threads per core) running Ubuntu Linux with 96 GB of RAM and Intel Xeon CPU E5-2667 processors with a clock speed of 2.90 GHz. Note, however, that we set the number of threads to one by setting `CPX_PARAM_THREADS = 1`. The default settings were used for all of the other parameters of CPLEX.[1]

We present a summary of our results in Table 1, which is organized as follows. The first column denotes the dimension n. The second set of columns reports our results corresponding to the instances that were solved to global optimality by CPLEX within the time limit whereas the third set is devoted to the instances that were terminated due to the time limit. Each of the second and third sets of columns is further subdivided into two sets of columns, the first of which reports the number of instances in each category. The second sets of columns report the average solution time (in seconds) and the average gap (in percentages) reported by

[1] All of the instances, our detailed results, and our codes for generating and solving the instances are publicly available at https://github.com/eayildirim/BoxQPInstanceGeneration

Table 1 Summary statistics

n	Optimal								Time Limit							
	Number of Instances				Average Time				Number of Instances				Average Gap (%)			
	Alg1	Alg2	Alg3	Alg4	Alg1	Alg2	Alg3	Alg4	Alg1	Alg2	Alg3	Alg4	Alg1	Alg2	Alg3	Alg4
25	100	100	100	100	0.00	0.22	0.87	0.30	0	0	0	0	–	–	–	–
50	100	91	82	90	0.00	51.22	65.22	48.80	0	9	18	10	–	0.88	0.05	0.05
75	100	10	14	29	0.00	180.58	220.43	237.01	0	90	86	71	–	0.89	0.05	0.04
100	100	1	2	2	0.00	14.01	115.90	52.91	0	99	98	98	–	1.40	0.06	0.05

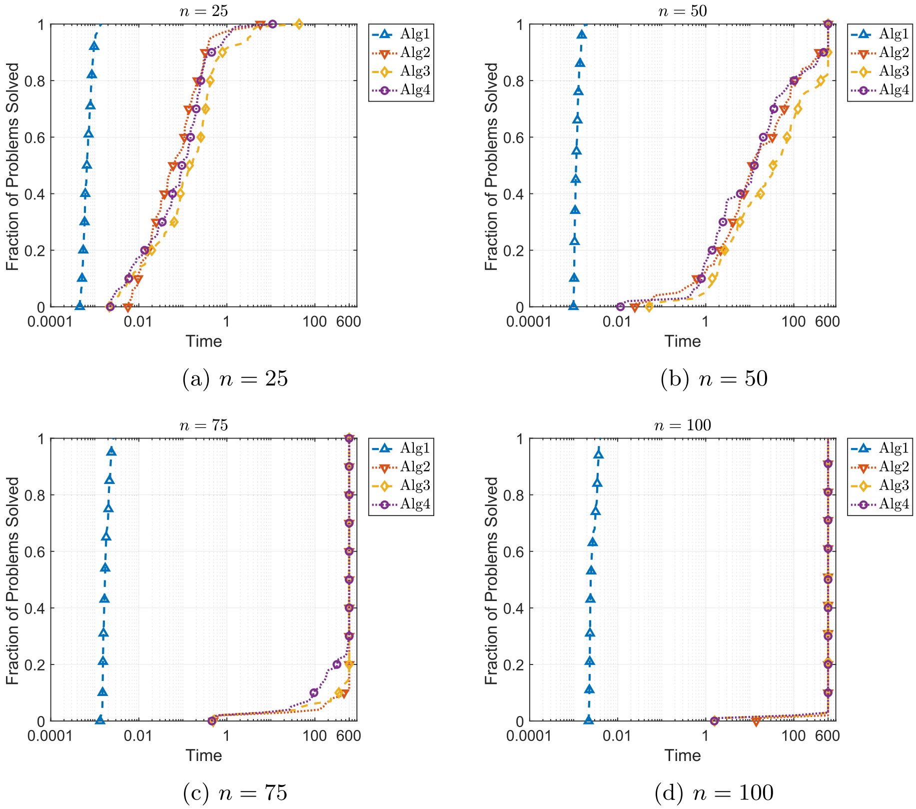


Fig. 2 Cumulative distribution functions of solution times

the solver, respectively. Finally, the columns Alg1, Alg2, Alg3, and Alg4 report the results corresponding to each of Algorithms 1–4, respectively.

Table 1 illustrates that all of the instances generated by Algorithm 1 can be solved very quickly regardless of the value of n. Recall that each of these instances admits an exact RLT relaxation and the solver seems to be capable of exploiting this property. On the other hand, the instances generated by Algorithms 2–4 seem to be computationally more demanding as n increases. In fact, while all instances with $n = 25$ are solved to global optimality fairly quickly, almost all instances hit the time limit for $n = 100$. Concerning the average gap, while Algorithms 3 and 4 seem to be somewhat less sensitive to n, we notice a stronger correlation for Algorithm 2.

In an attempt to provide more insight into the distribution of the solution times of instances, we report empirical cumulative distribution functions in Fig. 2. Each of the four plots in Fig. 2 corresponds to each choice of n. The horizontal axis denotes the solution time in logarithmic scale and the vertical axis represents the fraction of instances solved. Markers indicate the solution time of every 10th instance. We used identical axis limits in all figures to facilitate an easier comparison. Note that we include the solution times of all instances, including the ones that were terminated due to the time limit.

Figure 2 clearly illustrates that the instances generated by Algorithm 1 can be solved much faster than the instances generated by each of the remaining three algorithms for each choice of

n. The instances generated by each of Algorithms 2–4 exhibit a somewhat similar distribution in terms of solution times. Interestingly, the sets of instances generated by Algorithm 3 seem to be slightly more challenging than those generated by Algorithm 4. As n increases, a larger fraction of instances hit the time limit.

We therefore conclude that Algorithms 2–4 are capable of generating computationally more demanding instances, especially as n increases. It is particularly worth noticing that Algorithms 3 and 4 can generate computationally challenging instances even though they admit an exact SDP-RLT relaxation. Therefore, such instances could be useful for testing the performance of algorithms in the future since a globally optimal solution of (BoxQP) is already known a priori.

6.3 Discussion

We close this section with a discussion of the four algorithms given by Algorithms 1–4. Note that all instances of (BoxQP) can be divided into the following four sets:

$$\begin{aligned}\mathcal{E}_1 &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R^* = \ell_{RS}^* = \ell^*\},\\ \mathcal{E}_2 &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R^* < \ell_{RS}^* = \ell^*\},\\ \mathcal{E}_3 &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R^* = \ell_{RS}^* < \ell^*\},\\ \mathcal{E}_4 &= \{(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n : \ell_R^* < \ell_{RS}^* < \ell^*\}.\end{aligned}$$

We clearly have $\mathcal{E}_1 = \mathcal{E}_R$, and any such instance can be constructed by Algorithm 1. On the other hand, Algorithm 2 returns an instance in $\mathcal{E}_2 \cup \mathcal{E}_3 \cup \mathcal{E}_4$. Any instance in $\mathcal{E}_1 \cup \mathcal{E}_2$ can be constructed by Algorithm 3. Finally, Algorithm 4 outputs an instance in the set $\mathcal{E}_2 = \mathcal{E}_{RS} \backslash \mathcal{E}_R$.

Note that one can generate a specific instance of (BoxQP) with an inexact SDP-RLT relaxation by extending the example in Sect. 4.5 [18]. Let $n = 2k + 1 \geq 3$ and consider the instance $(Q, c) \in \mathcal{S}^n \times \mathbb{R}^n$ given by $Q = \frac{1}{n} ee^T - I$, where $I \in \mathcal{S}^n$ denotes the identity matrix, and $c = 0$. Since Q is negative semidefinite, the optimal solution of (BoxQP) is attained at one of the vertices. It is easy to verify that any vertex $v \in F$ with k (or $k + 1$) components equal to 1 and the remaining ones equal to zero is an optimal solution, which implies that $\ell^* = \frac{1}{2}\left(\frac{k^2}{n} - k\right)$. Let $\hat{x} = \frac{1}{2} e$ and

$$\begin{aligned}\hat{X} &= \hat{x}\hat{x}^T + \hat{M}\\ &= \frac{1}{4} ee^T + \frac{1}{4(n-1)}\left(nI - ee^T\right)\\ &= \frac{1}{4}\left(1 + \frac{1}{n-1}\right) I + \frac{1}{4}\left(1 - \frac{1}{n-1}\right) ee^T.\end{aligned}$$

It is easy to verify that $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$. Therefore,

$$\ell_{RS}^* \leq \frac{1}{2}\langle Q, \hat{X}\rangle + c^T \hat{x} = -\frac{n}{8}.$$

Using $n = 2k+1$, we conclude that $\ell_{RS}^* < \ell^*$, i.e., the SDP-RLT relaxation is inexact. Finally, this example can be extended to an even dimension $n = 2k \geq 4$ by simply constructing the same example corresponding to $n = 2k - 1$ and then adding a component of zero to each of $\hat{x}$ and c, and adding a column and row of zeros to each of Q and $\hat{X}$.

An interesting question is whether an algorithm can be developed for generating more general instances with inexact SDP-RLT relaxations, i.e., the set of instances given by $\mathcal{E}_3 \cup \mathcal{E}_4$.

One possible approach is to use a similar idea as in Algorithms 2 and 4, i.e., designate an optimal solution $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$, which is not in the form of (v, vv^T) for any vertex $v \in F$, and identify the conditions on the other parameters so as to guarantee that $(\hat{x}, \hat{X})$ is the unique optimal solution of the SDP-RLT relaxation (RS). Note that Lemma 19 can be used to easily construct an instance of (BoxQP) such that any feasible solution $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$ is an optimal solution of (RS). In particular, the condition (61) can be satisfied by simply choosing an arbitrary matrix $B \in \mathcal{S}^k$ such that $B \succeq 0$, and by defining

$$\begin{bmatrix} \hat{\beta} & \hat{h}^T \\ \hat{h} & \hat{H} \end{bmatrix} = PBP^T,$$

where $P \in \mathbb{R}^{(n+1)\times k}$ is a matrix whose columns form a basis for the nullspace of the matrix

$$\begin{bmatrix} 1 & \hat{x}^T \\ \hat{x} & \hat{X} \end{bmatrix}.$$

For instance, the columns of P can be chosen to be the set of eigenvectors corresponding to zero eigenvalues. However, this procedure does not necessarily guarantee that $(\hat{x}, \hat{X}) \in \mathcal{F}_{RS}$ is the *unique* optimal solution of (RS). Therefore, a characterization of the extreme points and the facial structure of $\mathcal{F}_{RS}$ may shed light on the algorithmic construction of such instances. We intend to investigate this direction in the near future for the SDP-RLT as well as the stronger SDP-RLT-TRI relaxation.

7 Concluding remarks

In this paper, we considered RLT and SDP-RLT relaxations of quadratic programs with box constraints. We presented algebraic descriptions of instances of (BoxQP) that admit exact RLT relaxations as well as those that admit exact SDP-RLT relaxations. Using these descriptions, we proposed four algorithms for efficiently constructing an instance of (BoxQP) with predetermined exactness or inexactness guarantees. Our preliminary computational experiments revealed that Algorithms 2–4 are capable of generating computationally challenging instances. In particular, we remark that Algorithms 1, 3, and 4 can be used to construct an instance of (BoxQP) with a known optimal solution, which may be of independent interest for computational purposes.

In the near future, we intend to investigate the facial structure of the feasible region of the SDP-RLT relaxation and exploit it to develop algorithms for generating instances of (BoxQP) with an inexact SDP-RLT relaxation.

Another interesting direction is the computational complexity of determining whether, for a given instance of (BoxQP), the RLT or the SDP-RLT relaxation is exact. Our algebraic descriptions do not yield an efficient procedure for this problem. An efficient recognition algorithm may have significant implications for extending the reach of global solvers for (BoxQP).

Acknowledgements The authors gratefully acknowledge the insightful comments and suggestions of two anonymous reviewers and the editor, which considerably improved the manuscript.

Data availability All of the instances, our detailed results, and our codes for generating and solving the instances are publicly available at https://github.com/eayildirim/BoxQPInstanceGeneration.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Angelis, P.L.D., Pardalos, P.M., Toraldo, G.: Quadratic programming with box constraints. In: Developments in Global Optimization, pp. 73–93. Springer, New York, NY (1997). https://doi.org/10.1007/978-1-4757-2600-8_5
2. Horst, R., Pardalos, P.M., Thoai, N.V.: Introduction to Global Optimization. Nonconvex Optimization and Its Applications, vol. 48. Springer, New York (2000)
3. Kozlov, M.K., Tarasov, S.P., Khachiyan, L.G.: The polynomial solvability of convex quadratic programming. USSR Comput. Math. Math. Phys. **20**(5), 223–228 (1980). https://doi.org/10.1016/0041-5553(80)90098-1
4. Sahni, S.: Computationally related problems. SIAM J. Comput. **3**(4), 262–279 (1974). https://doi.org/10.1137/0203021
5. Pardalos, P.M., Vavasis, S.A.: Quadratic programming with one negative eigenvalue is NP-hard. J. Glob. Optim. **1**, 15–22 (1991). https://doi.org/10.1007/BF00120662
6. Ahmadi, A.A., Zhang, J.: On the complexity of finding a local minimizer of a quadratic function over a polytope. Math. Program. **195**(1), 783–792 (2022). https://doi.org/10.1007/s10107-021-01714-2
7. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs. Part I. Convex underestimating problems. Math. Program. **10**(1), 147–175 (1976). https://doi.org/10.1007/BF01580665
8. Sherali, H.D., Tunçbilek, C.H.: A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. J. Glob. Optim. **2**(1), 101–112 (1992). https://doi.org/10.1007/BF00121304
9. Shor, N.Z.: Quadratic optimization problems. Sov. J. Comput. Syst. Sci. **25**(6), 1–11 (1987)
10. Misener, R., Floudas, C.A.: ANTIGONE: algorithms for continuous/integer global optimization of nonlinear equations. J. Glob. Optim. **59**(2–3), 503–526 (2014). https://doi.org/10.1007/s10898-014-0166-2
11. Tawarmalani, M., Sahinidis, N.V.: A polyhedral branch-and-cut approach to global optimization. Math. Program. **103**, 225–249 (2005). https://doi.org/10.1007/s10107-005-0581-8
12. IBM ILOG CPLEX Optimization Studio: User's Manual for CPLEX (2023). https://www.ibm.com/docs/en/icos
13. Gurobi Optimization, LLC: Gurobi Optimizer Reference Manual (2023). https://www.gurobi.com
14. Yıldırım, E.A.: An alternative perspective on copositive and convex relaxations of nonconvex quadratic programs. J. Glob. Optim. **82**(1), 1–20 (2022). https://doi.org/10.1007/s10898-021-01066-3
15. Yajima, Y., Fujie, T.: A polyhedral approach for nonconvex quadratic programming problems with box constraints. J. Glob. Optim. **13**, 151–170 (1998). https://doi.org/10.1023/A:1008293029350
16. Burer, S., Letchford, A.N.: On nonconvex quadratic programming with box constraints. SIAM J. Optim. **20**(2), 1073–1089 (2009). https://doi.org/10.1137/080729529
17. Anstreicher, K.M.: Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. J. Glob. Optim. **43**, 471–484 (2009). https://doi.org/10.1007/s10898-008-9372-0
18. Anstreicher, K.M., Burer, S.: Computable representations for convex hulls of low-dimensional quadratic forms. Math. Program. **124**(1–2), 33–43 (2010). https://doi.org/10.1007/s10107-010-0355-9
19. Dong, H., Linderoth, J.T.: On valid inequalities for quadratic programming with continuous variables and binary indicators. In: Goemans, M.X., Correa, J.R. (eds.) Integer Programming and Combinatorial Optimization—16th International Conference, IPCO 2013, Valparaíso, Chile, March 18–20, 2013. Pro-

ceedings. Lecture Notes in Computer Science, vol. 7801, pp. 169–180. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36694-9_15
20. Bonami, P., Günlük, O., Linderoth, J.T.: Globally solving nonconvex quadratic programming problems with box constraints via integer programming methods. Math. Program. Comput. **10**(3), 333–382 (2018). https://doi.org/10.1007/s12532-018-0133-x
21. Padberg, M.: The Boolean quadric polytope: some characteristics, facets and relatives. Math. Program. **45**(1), 139–172 (1989). https://doi.org/10.1007/BF01589101
22. Raghavachari, M.: On connections between zero-one integer programming and concave programming under linear constraints. Oper. Res. **17**(4), 680–684 (1969). https://doi.org/10.1287/opre.17.4.680
23. Kim, S., Kojima, M.: Exact solutions of some nonconvex quadratic optimization problems via SDP and SOCP relaxations. Comput. Optim. Appl. **26**(2), 143–154 (2003). https://doi.org/10.1023/A:1025794313696
24. Sojoudi, S., Lavaei, J.: Exactness of semidefinite relaxations for nonlinear optimization problems with underlying graph structure. SIAM J. Optim. **24**(4), 1746–1778 (2014). https://doi.org/10.1137/130915261
25. Jeyakumar, V., Li, G.: Trust-region problems with linear inequality constraints: exact SDP relaxation, global optimality and robust optimization. Math. Program. **147**(1–2), 171–206 (2014). https://doi.org/10.1007/s10107-013-0716-2
26. Locatelli, M.: Exactness conditions for an SDP relaxation of the extended trust region problem. Optim. Lett. **10**(6), 1141–1151 (2016). https://doi.org/10.1007/s11590-016-1001-0
27. Ho-Nguyen, N., Kilinç-Karzan, F.: A second-order cone based approach for solving the trust-region subproblem and its variants. SIAM J. Optim. **27**(3), 1485–1512 (2017). https://doi.org/10.1137/16M1065197
28. Burer, S., Ye, Y.: Exact semidefinite formulations for a class of (random and non-random) nonconvex quadratic programs. Math. Program. **181**(1), 1–17 (2020). https://doi.org/10.1007/s10107-019-01367-2
29. Wang, A.L., Kilinç-Karzan, F.: On the tightness of SDP relaxations of QCQPs. Math. Program. **193**(1), 33–73 (2022). https://doi.org/10.1007/s10107-020-01589-9
30. Azuma, G., Fukuda, M., Kim, S., Yamashita, M.: Exact SDP relaxations of quadratically constrained quadratic programs with forest structures. J. Glob. Optim. **82**, 243–262 (2022). https://doi.org/10.1007/s10898-021-01071-6
31. Sağol, G., Yıldırım, E.A.: Analysis of copositive optimization based linear programming bounds on standard quadratic optimization. J. Glob. Optim. **63**(1), 37–59 (2015). https://doi.org/10.1007/s10898-015-0269-4
32. Gökmen, Y.G., Yıldırım, E.A.: On standard quadratic programs with exact and inexact doubly nonnegative relaxations. Math. Program. **193**(1), 365–403 (2022). https://doi.org/10.1007/s10107-020-01611-0
33. Majthay, A.: Optimality conditions for quadratic programming. Math. Program. **1**(1), 359–365 (1971). https://doi.org/10.1007/BF01584097
34. Jiaquan, L., Tiantai, S., Dingzhu, D.: On the necessary and sufficient condition of the local optimal solution of quadratic programming. Chin. Ann. Math. Ser. B **3**(5), 625–630 (1982)
35. Vandenberghe, L., Boyd, S.P.: Semidefinite programming. SIAM Rev. **38**(1), 49–95 (1996). https://doi.org/10.1137/1038003
36. Rockafellar, R.T.: Convex Analysis. Princeton Landmarks in Mathematics and Physics. Princeton University Press, Princeton (1970)
37. Anstreicher, K.M.: On convex relaxations for quadratically constrained quadratic programming. Math. Program. **136**(2), 233–251 (2012). https://doi.org/10.1007/S10107-012-0602-3
38. Locatelli, M., Piccialli, V., Sudoso, A.M.: Fix and bound: an efficient approach for solving large-scale quadratic programming problems with box constraints (2022). arXiv:2211.08911
39. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: a fresh approach to numerical computing. SIAM Rev. **59**(1), 65–98 (2017). https://doi.org/10.1137/141000671
40. Lubin, M., Dowson, O., Dias Garcia, J., Huchette, J., Legat, B., Vielma, J.P.: JuMP 1.0: recent improvements to a modeling language for mathematical optimization. Math. Program. Comput. (2023). https://doi.org/10.1007/s12532-023-00239-3