Noname manuscript No.
(will be inserted by the editor)

On Convex Quadratic Programs with Linear Complementarity Constraints

Lijie Bai · **John E.Mitchell** · **Jong-Shi Pang**

Received: date / Accepted: date

Abstract The paper shows that the global resolution of a general convex quadratic program with complementarity constraints (QPCC), possibly infeasible or unbounded, can be accomplished in finite time. The method constructs a minmax mixed integer formulation by introducing finitely many binary variables, one for each complementarity constraint. Based on the primal-dual relationship of a pair of convex quadratic programs and on a logical Benders scheme, an extreme ray/point generation procedure is developed, which relies on valid satisfiability constraints for the integer program. To improve this scheme, we propose a two-stage approach wherein the first stage solves the mixed integer quadratic program with pre-set upper bounds on the complementarity variables, and the second stage solves the program outside this bounded region by the Benders scheme. We report computational results with our method. We also investigate the addition of a penalty term $y^T Dw$ to the objective function, where y and w are the complementary variables and D is a nonnegative diagonal matrix. The matrix D can be chosen effectively by solving a semidefinite program, ensuring that the objective function remains convex. The addition of the penalty term can often reduce the overall runtime by at least 50%. We report preliminary computational testing on a QP relaxation method which can be used to obtain better lower bounds from infeasible points; this method could be incorporated into a branching scheme. By combining the penalty method and the QP relaxation method, more than 90% of the gap can be closed for some QPCC problems.

Keywords Convex quadratic programming · Logical Benders decomposition · Satisfiability constraints · Semidefinite programming.

1 Introduction

Mathematical programs with complementarity constraints (MPCCs) are constrained optimization problems subject to certain distinguished disjunctive constraints expressed by the complementarity relation between pairs of nonnegative variables. The origin of MPCCs can be traced back to the class of Stackelberg games in economics. Extensive applications of MPCCs can be found in hierarchical (particularly bi-level) decision making, inverse optimization, parameter identification, optimal design, and many other contexts; various examples of MPCCs are documented in [16, 22]. Linear and convex quadratic programs with complementarity constraints (LPCCs/QPCCs,

The work of Mitchell was supported by the National Science Foundation under grant DMS-0715446 and by the Air Force Office of Sponsored Research under grant FA9550-08-1-0081 and FA9550-11-1-0260. The work of Pang was supported by the U.S.A. National Science Foundation grant CMMI 0969600 and by the Air Force Office of Sponsored Research under grants FA9550-08-1-0061 and FA9550-11-1-0151.

Lijie Bai
Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
E-mail: bail@rpi.edu

John E.Mitchell
Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
E-mail: mitchj@rpi.edu

Jong-Shi Pang
Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
E-mail: jspang@illinois.edu

respectively) are two prominent subclasses of MPCCs that play the same fundamental role in this family of nonconvex problems as standard linear and convex quadratic programs play in the class of convex programs; thus it is very important to have an in-depth treatment of the L(Q)PCCs, in particular, their global resolution.

With its signature complementarity constraints, the MPCC is well known for its nonconvexity and disjunctive features, which invalidate all standard constraint qualifications in nonlinear programming (NLP) except under restrictive assumptions. In spite of such difficulties, many publicly available NLP solvers have built-in techniques to handle the complementarity constraints and are capable of computing stationary solutions of some kind fairly routinely. Yet, these solvers are incapable of ascertaining the quality of the computed iterates, let alone verifying their optimality, local or global. The lack of robust schemes for the global resolution of the MPCC provides a prime motivation to undertake a systematic investigation of this important topic, particularly for the LPCC and QPCC, which are closely tied to a linear/(convex) quadratic program.

The present paper is a continuation of our recent foray into the research on the global resolution of MPCCs, which has to date led to several publications [16,17,18,21] and two doctoral theses [15,24]. Extending the recent work [16] on the LPCC, our work here aims at developing a finite algorithm capable of ascertaining whether the QPCC is infeasible, feasible but with an unbounded objective, or solvable with a finite optimal solution. In each case, a certificate is produced with the respective conclusion. Inheriting all the challenges of the LPCC, the QPCC has the additional difficulty that an optimal solution of the problem, if it exists, in general cannot be found by optimizing the (convex) quadratic objective on the convex hull of the (nonconvex) feasible region. Thus, unlike the LPCC, knowing the latter hull is not sufficient to globally resolve the QPCC.

Like the previous work [16] for the global resolution of the LPCC, our approach is based on logical Benders decomposition for solving mixed integer programs with conditional constraints, which has its origin from [12,13, 14] for logic programming. By introducing finitely many binary variables and associated upper bounds for the complementary variables, the proposed approach is developed based on a minmax mixed integer formulation of the QPCC. The upper bounds are not explicitly specified and are employed only conceptually within the logical Benders framework. The approach exploits the primal-dual relationship of a pair of convex quadratic programs in the generation of valid satisfiability constraints [11] on the binary variables; the latter special constraints provide the needed guidance to search the polyhedral pieces of the disjunctive constraints. In addition to the generation of the satisfiability constraints, we present several new ideas to enhance the performance of the overall method. One idea is to embed the method in a two-stage procedure: in the first stage, we introduce upper bounds on the complementary variables and solve the resulting bounded-variable mixed integer quadratic programming by an existing solver such as CPLEX 12.1. From the optimal solution obtained, we add a constraint on the objective function and apply the logical Benders scheme to solve the QPCC outside the bounded region imposed in the first stage. This procedure turns out to be very effective for problems for which no known bounds on the complementary variables are readily available. The second idea that we introduce involves the addition of a penalty term $y^T Dw$ to the objective function, where y and w are complementary variables and D is a nonnegative diagonal matrix. The matrix D can be chosen efficiently by solving a semidefinite program, in order to ensure that the objective function remains convex. For our test problems, the addition of the penalty term often reduces the overall runtime by at least 50%. Lastly, we describe an equality-constrained quadratic programming pre-processing procedure to generate better lower bounds from infeasible points; this procedure could be incorporated into a branching scheme. By combining all these new ideas, more than 90% of the gap can be closed for some QPCC problems.

Notation: Several different classes of real matrices are represented as follows: $\mathscr{S}^k$ denotes $k \times k$ symmetric matrices, $\mathscr{S}^k_+$ denotes $k \times k$ symmetric positive semidefinite matrices, $\mathscr{D}^k_+$ denotes $k \times k$ positive semidefinite diagonal matrices, and $\mathscr{B}^m$ denotes $m-$binary vectors.

2 Problem Definition

Formally, the general QPCC is defined as follows:

$$\begin{array}{lll} \text{minimize}_{(x,y)} & q(x,y) \;:=\; c^T x + d^T y + \dfrac{1}{2}\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} & \\ \text{subject to} & Ax + By \geq f & \\ \text{and} & 0 \leq y \perp w := q + Nx + My \geq 0, & \end{array} \tag{1}$$

where $x \in \Re^n$, $y \in \Re^m$, $Q := \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \in \mathscr{S}_+^{n+m}$ is a positive semidefinite matrix, and $\perp$ is the notation for perpendicularity, which in this context is short hand for the complementarity condition: $y^T w = 0$, or equivalently, either $y_i = 0$ or $w_i = 0$ for every $i = 1, \cdots, m$.

The fact that the objective function is quadratic makes the QPCC problem quite different from a LPCC. Knowing the convex hull of the feasible region enables the resolution of a LPCC by solving a linear program; however, knowing the convex hull of the feasible region is not sufficient to globally resolve a QPCC. To see this, we consider the following simple example in the scalar variables y and w:

$$\begin{array}{ll} \text{minimize}_{(y,w)} & y^2 + w^2 \\ \text{subject to} & y + w = 1 \\ \text{and} & 0 \leq y \perp w \geq 0. \end{array} \tag{2}$$

It is obvious that $(1,0)$ and $(0,1)$ are the only feasible solutions to the above QPCC, thus the convex hull of the feasible set of the QPCC is

$$\text{conv} = \{ (y,w) \geq 0 \,|\, y + w = 1 \}.$$

The minimum of the quadratic objective function over the convex hull is achieved at $y = w = \frac{1}{2}$, which is infeasible in the QPCC.

In view of the above example, attention is needed to consider a proper subset of the convex hull of the feasible region of the QPCC. Following the approach for the LPCC [16], we introduce a binary vector $z \in \mathbb{B}^m$ to describe the complementarity constraints, and define the value function of a convex quadratic program:

$$\begin{array}{rlll} \Re \cup \{\pm\infty\} \ni \theta(z) := & \text{minimum}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} & \\ & \text{subject to} & Ax + By \geq f & (\lambda) \\ & & y_j = 0, \quad j \in J & (v_J^+) \\ & & y \geq 0 & (v^-) \\ & & w_i = (q + Nx + My)_i = 0, \quad i \in I & (u_I^+) \\ & \text{and} & w = q + Nx + My \geq 0 & (u^-); \end{array} \tag{3}$$

where $I := \{i | z_i = 1\}$ and $J := \{j | z_j = 0\}$. In (3), $(\lambda, u_I^+, u^-, v_J^+, v^-)$ are the Lagrangian multipliers associated with the corresponding constraints. The feasible region of (3) is a polyhedral *piece* of the QPCC (1) defined by the binary vector z. Defining $u_j^+ = 0, j \in J$ and $v_i^+ = 0, i \in I$, the dual problem of $\theta(z)$ can be written as:

$$\begin{array}{rll} \Re \cup \{\pm\infty\} \ni \varphi(z) := & \text{maximum}_{(x,y,\lambda,u^\pm,v^\pm)} & -\frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + f^T \lambda + q^T (u^+ - u^-) \\ & \text{subject to} & c + Q_1 x + R y - A^T \lambda - N^T u^- + N^T u^+ = 0 \\ & & d + Q_2 y + R^T x - B^T \lambda - M^T u^- + M^T u^+ - v^- + v^+ = 0 \\ & \text{and} & (1 - z)^T u^+ + z^T v^+ = 0 \\ & & \lambda, u^\pm, v^\pm \geq 0. \end{array} \tag{4}$$

By duality, it follows that if either $\theta(z)$ or $\varphi(z)$ is finite, then both are finite and equal. Consider the following minmax problem of minimizing the value function $\varphi(z)$:

$$\begin{array}{lll} \textit{minimize}_{z\in\{0,1\}^m} & \text{maximize}_{(x,y,\lambda,u^\pm,v^\pm)} & -\frac{1}{2}\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + f^T\lambda + q^T(u^+ - u^-) \\ & \text{subject to} & c + Q_1x + Ry - A^T\lambda - N^Tu^- + N^Tu^+ = 0 \\ & & d + Q_2y + R^Tx - B^T\lambda - M^Tu^- + M^Tu^+ - v^- + v^+ = 0 \\ & & (1-z)^Tu^+ + z^Tv^+ = 0 \\ & \text{and} & \lambda, u^\pm, v^\pm \geq 0. \end{array} \tag{5}$$

We discuss the relationship of (5) to the QPCC (1) below. Care is needed because it is possible that (3) and (4) may both be infeasible. It is useful to recall [8] that a feasible quadratic program is unbounded below if and only if there exists a feasible ray on which the objective function tends to $-\infty$. This basic fact of quadratic programming motivates the introduction of several homogenized linear/quadratic programs as described next.

2.1 Homogenization

To detect the infeasibility or unboundedness of the QPCC (1), we introduce two kinds of homogeneous problems. The first homogeneous problem deals with infeasibility. Specifically, letting both c and d be zero we get a QPCC with a homogeneous objective function:

$$\begin{array}{ll} \text{minimize}_{(x,y)} & \frac{1}{2}\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} & Ax + By \geq f \\ \text{and} & 0 \leq y \perp q + Nx + My \geq 0. \end{array} \tag{6}$$

Correspondingly, we define the following value function obtained by setting c and d equal to 0 in (4):

$$\begin{array}{ll} \Re \cup \{+\infty\} \ni \varphi_0(z) := \text{maximum}_{(x,y,\lambda,u^\pm,v^\pm)} & -\frac{1}{2}\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + f^T\lambda + q^T(u^+ - u^-) \\ \text{subject to} & Q_1x + Ry - A^T\lambda - N^Tu^- + N^Tu^+ = 0 \\ & Q_2y + R^Tx - B^T\lambda - M^Tu^- + M^Tu^+ - v^- + v^+ = 0 \\ & (1-z)^Tu^+ + z^Tv^+ = 0 \\ \text{and} & \lambda, u^\pm, v^\pm \geq 0. \end{array} \tag{7}$$

Clearly, $\varphi_0(z) \geq 0$; it turns out that $\varphi_0(z) = 0$ is equivalent to the following homogeneous linear program having a finite optimal value, namely zero:

$$\begin{array}{ll} \{0,\infty\} \ni \vartheta_0(z) := \text{maximum}_{(\lambda,u^\pm,v^+)} & f^T\lambda + q^T(u^+ - u^-) \\ \text{subject to} & -A^T\lambda - N^Tu^- + N^Tu^+ = 0 \\ & -B^T\lambda - M^Tu^- + M^Tu^+ + v^+ \geq 0 \\ & (1-z)^Tu^+ + z^Tv^+ = 0 \\ \text{and} & \lambda, u^\pm, v^+ \geq 0. \end{array} \tag{8}$$

More can be said about the above homogenized problems.

Proposition 1 *The following 4 statements hold:*

(a) *both $\varphi_0(z)$ and $\vartheta_0(z)$ are either* 0 *or* ∞*;*

(b) $\varphi_0(z) = 0$ *if and only if* $\vartheta_0(z) = 0$*;*
(c) *if (4) is feasible, then* $\varphi(z)$ *is unbounded if and only if* $\varphi_0(z)$ *is unbounded;*
(d) *(3) is infeasible if and only if* $\vartheta_0(z)$*, or equivalently* $\varphi_0(z)$*, is unbounded.*

Proof Neither (a) nor (c) requires a proof. To prove (b), suppose $\varphi_0(z)$ is unbounded. There exists a feasible ray $(x, y, \lambda, u^{\pm}, v^{\pm})$ of (7) such that $f^T\lambda + q^T(u^+ - u^-) > 0$ and $\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix} = 0$. Clearly, $(\lambda, u^{\pm}, v^{\pm})$ is feasible to (8) and the linear function ϑ_0 tends to ∞ on this ray. Conversely if $\vartheta_0(z)$ is unbounded above along the ray $(\lambda, u^{\pm}, v^+)$, then $\varphi_0(z)$ is unbounded above along its ray $(0, 0, \lambda, u^{\pm}, v^+)$. This establishes (b). To prove (d) note that (3) is infeasible if and only if the linear program:

$$\begin{array}{lll} \text{minimize}_{(x,y)} & 0^T x + 0^T y & \\ \text{subject to} & Ax + By \geq f & (\lambda) \\ & y_j = 0, \, j \in J & (v_J^+) \\ & y \geq 0 & (v^-) \\ & (q + Nx + My)_i = 0, \, i \in I & (u_I^+) \\ \text{and} & q + Nx + My \geq 0 & (u^-). \end{array} \tag{9}$$

is infeasible, whose dual is precisely (8). Thus (d) holds readily.

Remark 1 It is interesting to note that (3) is infeasible if and only if the homogeneous dual problem $\varphi_0(z)$ is unbounded, regardless of whether (4) is feasible or not. □

Having dealt with the infeasibility issue of a piece (3) of the QPCC, we next turn to the unboundedness issue. For this purpose, we set the constant vectors f and q in the constraints of the QPCC to zero and consider the following homogeneously constrained QPCC:

$$\begin{array}{ll} \text{minimize}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} & Ax + By \geq 0 \\ \text{and} & 0 \leq y \perp Nx + My \geq 0. \end{array} \tag{10}$$

Similar to the value function $\theta(z)$, we define, for a binary vector $z \in \mathbb{B}^m$, the following (primal) quadratic program corresponding to the polyhedral piece of (10) defined by z:

$$\begin{array}{lll} \Re \cup \{-\infty\} \ni \eta_0(z) := \text{minimum}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} & \\ \text{subject to} & Ax + By \geq 0 & (\lambda) \\ & y_j = 0, \quad j \in J & (v_J^+) \\ & y \geq 0 & (v^-) \\ & (Nx + My)_i = 0, \quad i \in I & (u_I^+) \\ \text{and} & Nx + My \geq 0 & (u^-), \end{array} \tag{11}$$

where $I = \{i | z_i = 1\}$ and $J = \{j | z_j = 0\}$. It follows easily that $\eta_0(z) = -\infty$ if and only $\gamma_0(z) = -\infty$, where

$$\begin{array}{ll} \{0, -\infty\} \ni \gamma_0(z) := \text{minimum}_{(x,y)} & c^T x + d^T y \\ \text{subject to} & Ax + By \geq 0 \\ & y_j = 0, \, j \in J \\ & y \geq 0 \\ & (Nx + My)_i = 0, \, i \in I \\ & Nx + My \geq 0 \\ \text{and} & \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 0. \end{array} \tag{12}$$

Table 1 Exclusive statements of $\theta(z)$

$\vartheta_0(z) = +\infty$	$\vartheta_0(z) = 0$	
$\theta(z)$ is infeasible; $\varphi(z)$ is infeasible or unbounded.	$\theta(z)$ is feasible.	
	$\gamma_0(z) = 0$:	Both $\varphi(z)$ and $\theta(z)$ have finite optimum. $\theta(z) = \varphi(z)$
	$\gamma_0(z) = -\infty$:	$\theta(z)$ is unbounded. $\varphi(z)$ is infeasible.

The following proposition requires no proof.

Proposition 2 *Provided $\theta(z)$ is feasible, (3) is unbounded if and only if (11), or equivalently (12) is unbounded.*
□

We summarize the above results in Table 1 that presents the mutually exclusive conclusions about a given piece of the QPCC. The conclusions in the table are made precise in the following proposition.

Proposition 3 *The following 3 statements hold:*

(a) *$\theta(z)$ is infeasible if and only if $\vartheta_0(z) = \infty$;*
(b) *$\theta(z)$ is finitely solvable if and only $\vartheta_0(z) = 0$ and $\gamma_0(z) = 0$; in this case, $\theta(z) = \varphi(z)$;*
(c) *$\theta(z)$ is unbounded below if and only if $\vartheta_0(z) = 0$ and $\gamma_0(z) = -\infty$.* □

3 Cuts and Algorithms for Mixed Integer QPs

We established the minmax framework in (5). A ray/point cut generation scheme will be set up; valid inequalities in the form of satisfiability constraints on the binary vectors z are to be obtained. In this way, some of the QPCC pieces will be cut off so that we do not need to visit all 2^m QPCC pieces in order to solve the QPCC.

3.1 Ray Cuts and Point Cuts

To start, we first define the following cone:

$$\Xi := \left\{ \begin{array}{ll} (\lambda, u^{\pm}, v^{\pm}): & A^T\lambda + N^T u^- - N^T u^+ = 0 \\ & B^T\lambda + M^T u^- - M^T u^+ - v^+ \leq 0 \\ & (\lambda, u^{\pm}, v^+) \geq 0. \end{array} \right\} \tag{13}$$

Assume there is a ray $(\lambda, u^{\pm}, v^+) \in \Xi$ such that

$$f^T\lambda + q^T(u^+ - u^-) > 0.$$

If a binary vector z satisfies

$$(1-z)^T u^+ + z^T v^+ = 0$$

for such a ray $(\lambda, u^{\pm}, v^+)$, then $\vartheta_0(z) = +\infty$ and it follows from Proposition 1 that $\theta(z)$ is infeasible. We define the following ray cut as a satisfiability constraint:

Proposition 4 *Given that $(\lambda, u^{\pm}, v^+)$ is a ray of the cone Ξ such that $f^T\lambda + q^T(u^+ - u^-) > 0$, then any m-binary vector z such that the QPCC piece $\theta(z)$ is feasible must satisfy the inequality*

$$\sum_{i:u_i^+>0} (1-z_i) + \sum_{j:v_j^+>0} z_j \geq 1.$$

Proof Given an m-binary vector z such that

$$\sum_{i:u_i^+>0} (1-z_i) + \sum_{j:v_j^+>0} z_j = 0,$$

then $(\lambda, u^{\pm}, v^+)$ is a feasible ray of $\vartheta_0(z)$, so $\theta(z)$ is infeasible from Proposition 1.

We define some other sets that illustrate the strength of the ray cuts:

$$\Lambda := \left\{ \begin{array}{ll} (\lambda, u^{\pm}, v^{+}): & A^T\lambda + N^T u^- - N^T u^+ = 0 \\ & B^T\lambda + M^T u^- - M^T u^+ - v^+ \leq 0 \\ & f^T\lambda + q^T(u^+ - u^-) = 1 \\ & (\lambda, u^{\pm}, v^{+}) \geq 0. \end{array} \right\} \tag{14}$$

$$\Gamma := \{z \in \{0,1\}^m : \textstyle\sum_{i:u_i^+>0}(1-z_i) + \sum_{j:v_j^+>0} z_j \geq 1, \ \forall (\lambda, u^{\pm}, v^{+}) \in \Lambda\}$$
$$\Omega := \{z \in \{0,1\}^m : \text{QPCC piece } \theta(z) \text{ is feasible}\}.$$

Here Λ is the set of all the (normalized) rays from which we can define ray cuts, Γ is the set of binary vectors z that satisfy all the possible ray cuts and Ω is the set of binary vectors z such that the QPCC piece $\theta(z)$ is feasible. Obviously, Ω is a subset of Γ. We prove that the converse also holds.

Proposition 5 *The two sets Γ and Ω are equal.*

Proof Let $\bar{z} \in \Gamma$. Then for every feasible solution $(\lambda, u^{\pm}, v^{+})$ for (8), we have that $f^T\lambda + q^T(u^+ - u^-) \leq 0$. It follows that $\vartheta_0(\bar{z}) = 0$, and so $\bar{z} \in \Omega$, by Proposition 3.

Below we define another type of satisfiability constraint — point cuts. Assume $\hat{z} \in \mathbb{B}^m$ is such that the QPCC piece $\theta(\hat{z})$ is feasible and finite. Let $(\hat{x}, \hat{y}, \hat{\lambda}, \hat{u}^{\pm}, \hat{v}^{\pm})$ be the optimal solution of the dual problem $\varphi(\hat{z})$. We can define a point cut as:

$$\sum_{i:\hat{u}_i^+>0} (1-z_i) + \sum_{j:\hat{v}_j^+>0} z_j \geq 1. \tag{15}$$

In the following proposition we conclude that it is enough to visit the remaining QPCC pieces $\theta(z)$ with binary vectors z statisfying point cut (15).

Proposition 6 *Given that $\hat{z} \in \mathbb{B}^m$ is such that the corresponding QPCC piece $\theta(\hat{z})$ is feasible and finite, with $(\hat{x}, \hat{y}, \hat{\lambda}, \hat{u}^{\pm}, \hat{v}^{\pm})$ being its optimal dual solution, then any binary vector z must satisfy point cut (15) in order to improve the objective value of the QPCC.*

Proof Assume that z is a binary vector such that $\sum_{i:\hat{u}_i^+>0}(1-z_i) + \sum_{j:\hat{v}_j^+>0} z_j = 0$, then $(\hat{x}, \hat{y}, \hat{\lambda}, \hat{u}^{\pm}, \hat{v}^{\pm})$ is a feasible solution to $\varphi(z)$, so $\varphi(z) \geq \varphi(\hat{z})$. From Proposition 1, $\theta(z) \geq \theta(\hat{z})$.

Generally, we can write the two types of cuts as:

$$\sum_{i \in I}(1-z_i) + \sum_{j \in J} z_j \geq 1, \tag{16}$$

where $I = \{i \,|\, u_i^+ > 0\}$ and $J = \{j \,|\, v_j^+ > 0\}$. Here I and J are disjoint.

3.2 The Algorithms

The general framework of our logical Benders decomposition algorithm can be summarized as follows:

1. Initialize the Master Problem with all binary vectors z. Set upper bound to $QP_{ub} = \infty$.
2. Choose a $\hat{z}$ that is feasible in the Master Problem.
3. Determine the value of $\theta(\hat{z})$.
4. If $\theta(\hat{z}) = -\infty$, STOP: the QPCC is unbounded.
5. If $\theta(\hat{z}) < QP_{ub}$, update $QP_{ub} \leftarrow \theta(\hat{z})$.
6. Use the solution of $\theta(\hat{z})$ to determine valid cuts of the form (16) that can be added to the Master Problem.
7. While the Master Problem is still feasible, return to Step 2.
8. Otherwise QP_{ub} is the optimal value, and any $\hat{z}$ achieving this value is optimal.

A satisfiability constraint can be written as (16), so the Master Problem is a satisfiability problem. The results in §2 can be used in the determination of $\theta(\hat{z})$ in Step 3. In particular, if the subproblem is infeasible then it is only necessary to solve linear programs. Much of the work of the algorithm is in the determination of cuts in Step 6. The solution to the appropriate dual problem gives an initial constraint of the form (16), but this constraint should be tightened in order to speed up the overall solution process. We discuss constraint tightening methods in detail, before giving the complete algorithm later as Algorithm 1.

The smaller the index sets I and J are, the more binary vectors z will be cut off by this satisfiability constraint. We thus want to sparsify the satisfiability constraints. Sparsification requires the selection of $I_1 \subseteq I$, $J_1 \subseteq J$. If the binary vectors z that don't satisfy

$$\sum_{i \in I_1} (1 - z_i) + \sum_{j \in J_1} z_j \geq 1 \tag{17}$$

either lead to infeasible pieces or do not do better than what we have, we can then replace constraint (16) by a stronger constraint (17). This is the basic idea of sparsification.

The complement of this constraint is

$$\sum_{i \in I_1} (1 - z_i) + \sum_{j \in J_1} z_j = 0 \tag{18}$$

which implies that $z_{I_1} = 1$, $z_{J_1} = 0$, and further, $w_{I_1} = 0$, $y_{J_1} = 0$ in the QPCC piece $\theta(z)$. We then want to consider the following program,

$$\begin{array}{rlll}
\Re \cup \{\pm\infty\} \ni \theta(I_1, J_1) := \text{minimum}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} & \\
\text{subject to} & Ax + By \geq f & (\lambda) \\
& y \geq 0 & (v^-) \\
& w = q + Nx + My \geq 0 & (u^-) \\
& y_j \leq 0, j \in J_1 & (v^+_{J_1}) \\
& w_i \leq 0, i \in I_1 & (u^+_{I_1}).
\end{array} \tag{19}$$

We call (19) the QP relaxation of the QPCC with additional constraint (18). Three cases may happen with $\theta(I_1, J_1)$:

(1) $\theta(I_1, J_1)$ is infeasible:

As in Proposition 1, the infeasibility of $\theta(I_1, J_1)$ is equivalent to the unboundedness of the following homogeneous linear program:

$$\begin{array}{rll}
\{0, \infty\} \ni \vartheta(I_1, J_1) := \text{maximum}_{(\lambda, u^\pm, v^\pm)} & f^T \lambda + q^T (u^+ - u^-) & \\
\text{subject to} & -A^T \lambda - N^T u^- + N^T u^+ = 0 & \\
& -B^T \lambda - M^T u^- + M^T u^+ + v^+ \geq 0 & \\
& \sum_{i \in \bar{I}_1} u^+_i + \sum_{j \in \bar{J}_1} v^+_j = 0 & \\
& \lambda, u^\pm, v^+ \geq 0. &
\end{array} \tag{20}$$

Here $\bar{I}_1$, $\bar{J}_1$ are the complementary sets of I_1, J_1. So we can solve for a ray $(\lambda, u^\pm, v^+)$ of $\vartheta(I_1, J_1)$ and define a new ray cut

$$\sum_{i \in I_{new}} (1 - z_i) + \sum_{j \in J_{new}} z_j \geq 1,$$

with $I_{new} = \{i \,|\, u^+_i > 0\}$, $J_{new} = \{j \,|\, v^+_j > 0\}$.

(2) $\theta(I_1, J_1)$ is finitely feasible:

If $\theta(I_1,J_1) \geq QP_{ub}$, all QPCC pieces $\theta(z)$ such that z satisfies (18) will not do better than we already have, using a similar argument to Proposition 6. Thus a new point cut can be defined:

$$\sum_{i\in I_{new}} (1-z_i) + \sum_{j\in J_{new}} z_j \geq 1,$$

with $I_{new} = \{i \,|\, u_i^+ > 0, i \in I_1\}$, $J_{new} = \{j \,|\, v_j^+ > 0, j \in J_1\}$. Here u^+ and v^+ are the Lagrangian multipliers. If $\theta(I_1,J_1) < QP_{ub}$, some QPCC piece $\theta(\hat{z})$ satisfying (18) may do better than QP_{ub}. We thus try to recover $\hat{z}$ such that $\hat{z}_i = 0$ when $\bar{y}_i = 0$ and $\hat{z}_i = 1$ otherwise. If QPCC piece $\theta(\hat{z})$ is infeasible, we can then define a new ray cut. Otherwise, if $\theta(\hat{z})$ does better than QP_{ub}, we update QP_{ub}. If $\theta(\hat{z})$ does not improve QP_{ub}, we can define a new point cut.

(3) $\theta(I_1,J_1)$ is unbounded below:

In this case, the attempted sparsification does not provide anything useful.

What we described above is the general process of sparsification. Additionally, whenever we get a new satisfiability constraint (cut), we immediately sparsify it. When we are able to update QP_{ub}, we sparsify all the satisfiability constraints from the previous sparsification processes. The general algorithm is given in Algorithm 1. A candidate binary vector z is chosen. This point is used in Algorithm 2 to generate index sets I and J that give a cut that is violated by z. The index sets are sparsified using the subroutine found in Algorithm 3. When the sparsification routine terminates, a candidate solution $\hat{z}$ may be generated, and this point leads to index sets which are then sparsified.

Initialization: $SAT \longleftarrow \emptyset$, $Z_{left} \longleftarrow$ all m-binary vectors, $QP_{ub} \longleftarrow \infty$;
/* SAT is the set of satisfiability constraints on binary vectors z. Z_{left} is the set of vectors satisfying the constraints in SAT */
while $Z_{left} \neq \emptyset$ **do**
  pick $z \in Z_{left}$;
  call *Find I, J* ;
  call *Sparsification* to sparsify cut (I,J) and try to obtain $\hat{z}$;
  SAT $\longleftarrow$ SAT $\cup \sum_{i\in I}(1-z_i) + \sum_{j\in J} z_j \geq 1$;
  if *get 'new_cand'* **then**
    call *Find I, J* with candidate point $z = \hat{z}$;
    call *Sparsification* to sparsify cut (I,J) ;
    SAT $\longleftarrow$ SAT $\cup \sum_{i\in I}(1-z_i) + \sum_{j\in J} z_j \geq 1$;
  end
  if *get 'new uB'* **then** call *Sparsification* to sparsify all cuts $\in$ SAT; update SAT;
end
case $QP_{ub} = +\infty$: 'Infeasible QPCC';
case $QP_{ub} < +\infty$: 'Finitely feasible QPCC';

Algorithm 1: The general algorithm

4 Preprocessing

In this section, we want to investigate the addition of a penalty term $y^T Dw$ to the quadratic objective function of the QPCC, where y and w are the complementary variables and D is a nonnegative diagonal matrix.

We investigated the following example problem in the introduction section:

$$\begin{aligned} &\text{minimize}_{(y,w)} \ y^2 + w^2 \\ &\text{subject to} \quad y + w = 1 \\ &\qquad\qquad\quad 0 \leq y \perp w \geq 0. \end{aligned} \tag{21}$$

Input : z, QP_{ub}
Output: I and J, QP_{ub}, flag

solve $\vartheta_0(z)$;
if $\vartheta_0(z)$ *is unbounded* **then**
  solve for a ray $(\lambda, u^{\pm}, v^{\pm})$ so that $\vartheta_0(z)$ is unbounded on this ray;
  set $I = \{i|u_i^+ > 0\}$, $J = \{i|v_i^+ > 0\}$;
else
  find $\theta(z)$;
  if $\theta(z) = -\infty$ **then**
    terminate, 'Unbounded QPCC';
  else
    set $I = \{i|u_i^+ > 0\}$, $J = \{i|v_i^+ > 0\}$;
    if $\theta(z) < QP_{ub}$ **then**
      $QP_{ub} \longleftarrow \theta(z)$;
      flag:'new uB';
    end
  end
end

Algorithm 2: Find I, J

Input : cut $(I,J) \longleftarrow \sum_{i\in I}(1-z_i)+\sum_{j\in J} z_j \geq 1$, QP_{ub} ; /* I and J are disjoint index sets */
Output: updated I and J, candidate $\hat{z}$, flag

repeat
  choose $I_1 \subseteq I$, $J_1 \subseteq J$;
  find $\vartheta(I_1, J_1)$;
  if $\vartheta(I_1, J_1) = \infty$ **then** /* $\vartheta(I_1,J_1)$ is bounded if (I,J) is a point cut */
    solve for a ray $(\lambda, u^{\pm}, v^{\pm})$ so that $\vartheta(I_1, J_1)$ is unbounded on this ray;
    update $I = \{i|u_i^+ > 0\}$, $J = \{i|v_i^+ > 0\}$;
    flag:'ray';
  else /* solve $\theta(I_1,J_1)$ as it is feasible */
    find $\theta(I_1, J_1)$;
    case $\theta(I_1, J_1) \geq QP_{ub}$
      update $I = \{i|u_i^+ > 0\}$, $J = \{i|v_i^+ > 0\}$;
      flag:'point';
    case $-\infty < \theta(I_1, J_1) < QP_{ub}$:
      $\hat{z}_i = 1$ if $\bar{y}_i > \bar{w}_i$, $\hat{z}_i = 0$ otherwise, where $(\bar{x}, \bar{y}, \bar{w})$ optimal for $\theta(I_1, J_1)$;
      flag:'new_cand';
    endsw
  end
until *don't get 'ray' or 'point'*;

Algorithm 3: Sparsification

QPCC (21) has an optimal value of 1 whereas its QP relaxation has an optimal value of 0.5. By adding a penalty term $2yw$ to the objective function and solving the QP relaxation of the new QPCC problem

$$\begin{aligned} \text{minimize}_{(y,w)} \quad & (y+w)^2 \\ \text{subject to} \quad & y + w = 1 \\ & 0 \leq y \perp w \geq 0, \end{aligned} \tag{22}$$

we have an optimal value of 1. It is straightforward that (22) has the same resolution with the (21) due to the complementarity of y and w. In addition, the objective function remains convex after we add the penalty term $2yw$.

For a general QPCC, we would like to add a penalty term $w^T Dy$ to the objective function while maintaining its convexity. We define a linear function $G : \mathscr{D}_+^m \rightarrow \mathscr{S}^{n+m}$ by

$$G(D) = \begin{bmatrix} 0 & N^T D \\ DN & M^T D + DM \end{bmatrix} = \sum_{i=1}^m d_i \begin{bmatrix} 0 & n_i^T e_i^T \\ e_i n_i & e_i m_i + m_i^T e_i^T \end{bmatrix} =: \sum_i d_i K_i, \tag{23}$$

where d_i denotes the ith diagonal entry of D. Here $\{e_i, i = 1, 2, \cdots, m\}$ is the standard basis for $\mathscr{R}^m$, and n_i, m_i are the ith rows of matrices N and M respectively. Note that

$$w^T Dy = q^T Dy + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T G(D) \begin{pmatrix} x \\ y \end{pmatrix}.$$

If the rank of the quadratic term in the objective can be reduced then the optimal solution to the quadratic programming relaxation may sit in a lower dimensional face, as formalized in the following lemma.

Lemma 1 *Let $\mathscr{P} \subseteq \mathscr{R}^n$ be a pointed polyhedron, let $c \in \mathscr{R}^n$, and let $Q \in \mathscr{R}^{n \times n}$ be symmetric and positive semidefinite with rank k. If there exists an optimal solution to the convex quadratic program* $\min\{c^T x + \frac{1}{2} x^T Q x : x \in \mathscr{P}\}$ *then there exists an optimal solution in a face of $\mathscr{P}$ of dimension no larger than k.*

Proof Let $\bar{x}$ be an optimal solution to the quadratic program that is in the interior of a face of dimension greater than k. There exists a direction $d \neq 0$ that lies in the face and in the nullspace of Q. Since $\bar{x}$ is optimal, we must also have $(c + Q\bar{x})^T d = 0$, so any feasible point of the form $\bar{x} + \alpha d$ is also optimal, so there exists an optimal solution of this form in a lower dimensional face.

A nonnegative diagonal matrix D can be picked effectively by solving a semidefinite program:

$$\begin{array}{lll} \text{maximize}_{(d_1, \cdots, d_m)} & \sum_{i=1}^m d_i p_i & \\ \text{subject to} & -G(D) \preceq \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} & \\ & d_i \geq 0, \forall i. & \end{array} \tag{24}$$

The semidefinite program can be solved effectively by using SDPT3 version 4.0 [23], which is a MATLAB software for semidefinite-quadratic-linear programming. Possible choices of p_i include $-\text{trace}(K_i)$, 1, or $\bar{w}_i \bar{y}_i$ with $(\bar{x}, \bar{y}, \bar{w})$ being an initial infeasible point in the QPCC.

In matrix rank minimization, it is desired to find a low rank matrix that meets various constraints. It was shown by Fazel et al. [9, 10] that the tightest convex underestimator of this nonconvex problem is to minimize the nuclear norm of the desired matrix. Candes and Recht [5] showed that minimizing the nuclear norm can find the optimal solution to the matrix completion problem with high probability, under certain assumptions.

In our problems, when minimizing the rank of the matrix

$$Q(D) := G(D) + \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix},$$

the choice of $p_i = -\text{trace}(K_i)$ corresponds to minimizing the nuclear norm $\text{trace}(Q(D))$. However, the structure of each K_i is such that this approach is typically unsuccessful, often resulting in $D = 0$ in our test problems. We thus used alternative choices of p_i in (24) that emphasize larger values of D, and we found that $\bar{w}_i \bar{y}_i$ works the best, with $(\bar{x}, \bar{y}, \bar{w})$ typically being the optimal solution to the QP relaxation.

By the addition of a penalty term $w^T Dy$ to the quadratic objective function, we construct an equivalent QPCC problem:

$$\begin{array}{ll} \text{minimize}_{(x,y)} & c^T x + (d + Dq)^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \left(\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} + G(D) \right) \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} & Ax + By \geq f \\ & 0 \leq y \perp w = q + Nx + My \geq 0. \end{array} \tag{25}$$

We then solve the modified QPCC for the global resolution. Some computational results will be reported later in §5. In the case of problem (21), the solution to (24) leads to the new QPCC (22).

5 Computational Experience

We have implemented the above algorithms for solving QPCCs in MATLAB. We solve the LPs and QPs using CPLEX 12.1 by calling it in a MATLAB environment. The experiments were run on a Dell laptop with an Intel® Core 2 Duo Processor, 1.99 GB of RAM. To implement the algorithms we have generated several classes of QPCCs: feasible and infeasible, bounded and unbounded QPCCs. In these problems, to generate a positive semidefinite matrix $Q \in \mathscr{S}_+^{n+m}$ with rank not exceeding r, we first generate a $(m+n) \times r$ matrix, namely P, whose entries are randomly generated numbers following the standard normal distribution. The matrix Q is then set equal to the matrix product PP^T. To generate a feasible QPCC, we use the same techniques as in [19] except that we generate Q differently as we described. Optimality conditions are considered in [19] and a stationary point is first generated in each QPCC, which in most cases turned out to be the optimal solution. To generate an unbounded QPCC, a feasible ray $(\hat{x}, \hat{y})$ is first generated. This can be done by first generating matrices A, B, M, N, and then flipping the signs or zeroing out some of the rows, so that

$$A\hat{x} + B\hat{y} \geq 0$$

and

$$0 \leq \hat{y} \perp N\hat{x} + M\hat{y} \geq 0.$$

We then generate $Q \in \mathscr{S}^{n+m}$ such that

$$Q \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = 0.$$

This can be done through matrix decompositions and multiplications. We pick vectors c and d such that

$$c^T \hat{x} + d^T \hat{y} < 0.$$

In this way, the QPCC is unbounded along the ray $(\hat{x}, \hat{y})$ as long as it is feasible. Additionally, a feasible solution can be generated by picking the right vectors f and q. In all problems, our constraints $Ax + By \geq f$ imply bounds on the x variables.

In implementation, to make the computation more efficient, we decompose a general QPCC into an inner box constrained QPCC where the complementary variables all have a bound T, and an outer QPCC to examine the feasible region outside the box. We define the inner box constrained QPCC problem and the outer QPCC problem to be

$$\begin{array}{ll} \text{minimize}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} & Ax + By \geq f \\ & 0 \leq y \perp w = q + Nx + My \geq 0 \\ & y_i + w_i \leq T, \quad i = 1, \ldots, m \end{array} \tag{26}$$

and

$$\begin{array}{ll} \text{minimize}_{(x,y)} & c^T x + d^T y + \frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to} & Ax + By \geq f \\ & 0 \leq y \perp w = q + Nx + My \geq 0 \\ & \sum_{i=1}^m y_i + \sum_{i=1}^m w_i \geq T, \end{array} \tag{27}$$

respectively. Note that if y_i and w_i are both constrained to be no larger than T then their sum is also no larger than T by complementarity. A tight upper bound for the QPCC can often be achieved by solving the box constrained QPCC (26) in a mixed integer quadratic program formulation. We experimented with different choices for T.

As we described above, we can solve the inner box constrained QPCC by CPLEX 12.1, so that the optimal value of the inner QPCC gives a tight upper bound QP_{ub} to the outer QPCC. Then we solve the outer QPCC using the general algorithms described in \$3. Let $(\bar{x}, \bar{y})$ be the optimal solution of the QP relaxation $\theta(\emptyset, \emptyset)$, if it exists. The quadratic objective function $q(x,y)$ is convex, so we exploit its gradient to add a linear constraint

$$q(\bar{x}, \bar{y}) + \nabla q(\bar{x}, \bar{y})^T [(x,y) - (\bar{x}, \bar{y}))] \leq QP_{ub}$$

to the outer QPCC. This constraint is useful when solving $\vartheta(z)$.

Table 2 Computational time of feasible QPCCs with bounded QP relaxations (with 90 complementary variables).

$T=100$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	number of LPs	number of QPs
Max	72.46	5.47	77.50	16	4
Min	10.041	4.77	14.81	16	4
Deviation	19.82	0.25	19.88	0	0
Average	34.08	4.99	39.07	16	4

Table 3 Computational time of feasible QPCCs with bounded QP relaxations (with 90 complementary variables).

$T=10^3$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	number of LPs	number of QPs
Max	73.88	3.16	76.81	11	2
Min	14.21	2.91	17.15	11	2
Deviation	20.83	0.08	20.93	0	0
Average	42.24	2.98	44.92	11	2

5.1 Some Numerical Results without the Preprocessing

It is straightforward that the larger the T is, the easier the outer QPCC will be. However, when T gets larger, the inner box QPCC will take a longer time to solve. Tables 2 and 3 contain the computational results for feasible QPCCs with bounded QP relaxations when using different bounds T. Here Q is a 95×95 positive semidefinite matrix of rank 90 and the dimension of the complementary variables y is $m=90$.

In Table 2, $T=100$ is used as the upper bound on the complementary variables in the box constrained QPCCs to solve a collection of 10 randomly generated feasible QPCCs. This collection of QPCCs all have bounded QP relaxations. On average, the box constrained QPCCs take about 34 seconds to solve, and the outer QPCCs take about 5 seconds to solve. In addition, more LPs than QPs are solved. In Table 3, $T=10^3$ is used as the upper bound on the complementary variables in the box constrained QPCCs to solve the same group of 10 QPCCs as in Table 2. As a larger T is used this time, the outer QPCCs take about 3 seconds to solve and the box constrained QPCCs take about 42 seconds to solve on average. Although the outer QPCCs are easier to solve in this case, the total computational times are longer. Figure 1 is the performance profile of the algorithms with $T=100$ and with $T=10^3$. The plot confirms that using $T=100$ in the box constrained QPCCs is more efficient with this collection of feasible QPCCs, which have bounded QP relaxations and 90 complementary variables.

In our experiments, we found that finitely solvable QPCCs with unbounded QP relaxations are more time consuming to solve than QPCCs with bounded QP relaxations. This is because when $\theta(I,J)$ becomes unbounded, the satisfiability constraint cannot be sparsified any further.

As proved in [8], a convex quadratic function either attains its optimum on a polyhedral convex set or becomes unbounded below along a ray. Therefore, if $\theta(I,J)$ is unbounded, then the value $\gamma_0(I,J)$ is also unbounded,

$$\begin{aligned}\gamma_0(I,J) := \text{minimize}_{(x,y)}\ & c^Tx+d^Ty \\ \text{subject to}\quad & Ax+By\geq 0 \\ & y\geq 0 \\ & w=Nx+My\geq 0 \\ & \begin{bmatrix} Q_1 & R \\ R^T & Q_2\end{bmatrix}\begin{pmatrix} x \\ y\end{pmatrix}=0 \\ & y_j\leq 0,\, j\in J \\ & w_i\leq 0,\, i\in I.\end{aligned}\tag{28}$$

When $\theta(I,J)$ becomes unbounded, we solve for a ray on which $\gamma_0(I,J)$ is unbounded. We then branch the outer QPCC into two QPCCs:

Given that $\gamma_0(I,J)$ is unbounded on a ray $\left\{\beta\begin{pmatrix} x^r \\ y^r\end{pmatrix}:\beta\geq 0\right\}$, we have

$$y^r\geq 0,\, y^r_j\leq 0,\, j\in J$$

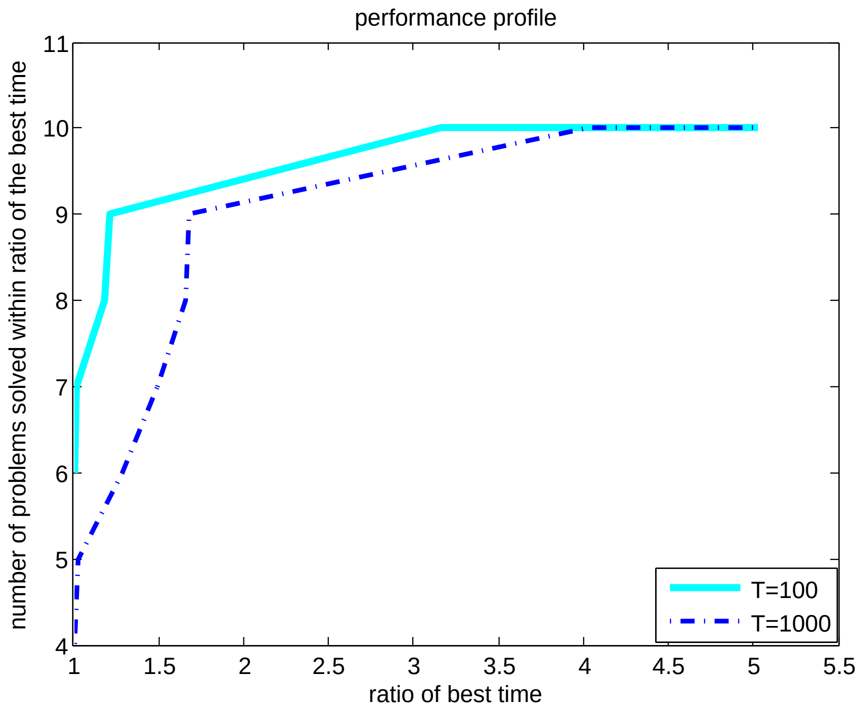

Fig. 1 Performance profile with feasible QPCCs of bounded QP relaxations when using different T's (of $m = 90$ complementary variables)

Table 4 Computational time of QPCCs with unbounded QP relaxation(of 30 complementary variables).

$T = 10^3$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	number of LPs	number of QPs
Max	2.00	87.10	88.07	84	204
Min	0.97	4.40	5.46	13	5
Deviation	0.32	28.81	28.84	23.7	68.0
Average	1.26	46.70	47.96	47.4	105.7

and

$$w^r = Nx^r + My^r \geq 0, w_i^r \leq 0, i \in I.$$

We then pick an index i such that both

$$y_i^r > 0$$

and

$$w_i^r > 0.$$

By fixing y_i to be 0 or fixing w_i to be 0 in the outer QPCC, we can branch the outer QPCC into two QPCCs. Then we solve the two QPCCs using the ray/point generation scheme respectively. We do so to avoid rays where $\theta(I,J)$ become unbounded.

Table 4 and Table 5 contain the computational results with a collection of 10 finitely solvable QPCCs with unbounded QP relaxations, with $T = 10^3$ and $T = 10^5$ as the bounds on the complementary variables in the box constrained QPCCs respectively. In contrast to the class of feasible QPCCs with bounded QP relaxations, here the computational times of the box constrained QPCCs are as short as several seconds, and the outer QPCCs take longer to solve.

As T increases from 10^3 to 10^5, the computational time of the box constrained QPCCs did not increase too much, while the outer QPCCs' time decreased a lot. When using $T = 10^5$ in the box constrained QPCCs, the collection of QPCCs can be solved as fast as within 15 seconds on average. Figure 2 is the performance profile of both algorithms using $T = 10^3$ and $T = 10^5$ in the box constrained QPCCs.

For QPCCs with a moderate number of complementary variables, we have more choices for the bounds T. Even if T is quite large, CPLEX MIQP solver can still solve the box constrained QPCCs accurately. As the number of complementary variables gets large, the possible choices for T become limited. If T is too large, the CPLEX

Table 5 Computational time of QPCCs of unbounded QP relaxation (of 30 complementary variables).

$T = 10^5$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	number of LPs	number of QPs
Max	5.92	41.47	47.39	51	92
Min	1.13	4.39	5.73	12	5
Deviation	1.42	12.90	13.90	14.0	29.5
Average	2.06	12.09	14.15	22.2	21.6

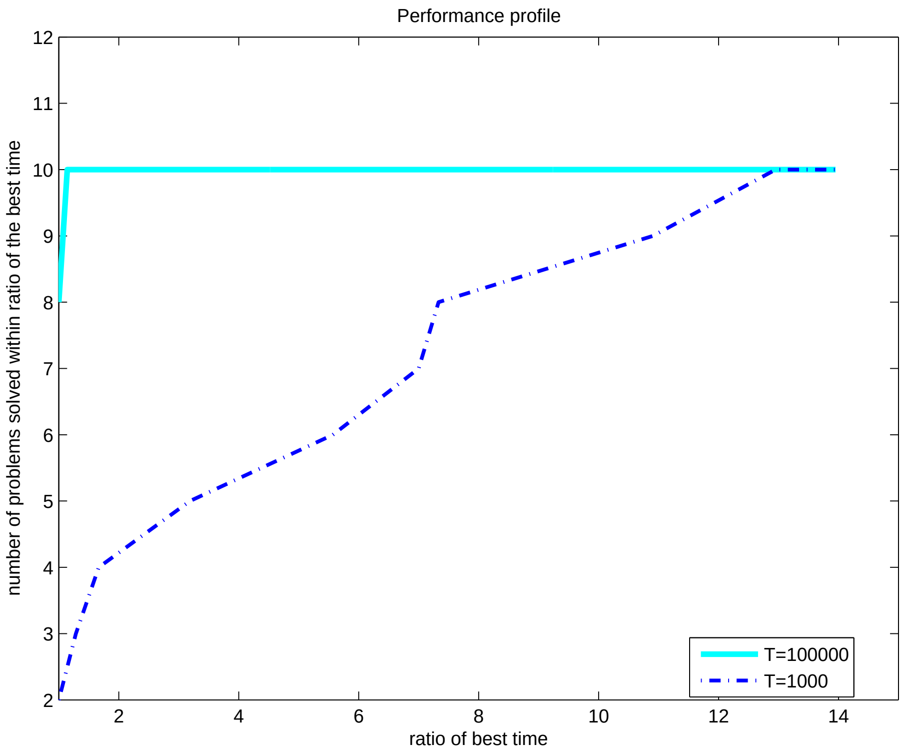

Fig. 2 Performance profile with bounded QPCCs with unbounded QP relaxations when using different T's (of $m = 30$)

Table 6 Computational time of larger sized QPCCs of bounded QP relaxation

$(m,n,k,rank(Q))$	T	box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
$(100,10,8,90)$	100	449.05	3.12	452.17	15	4
$(120,4,8,90)$	100	849.98	2.53	852.51	13	4
$(150,4,8,150)$	100	62.11	3.78	65.89	17	4
$(150,4,4,150)$	80	102.42	2.82	105.24	14	4
$(180,2,4,170)$	80	1126.83	1123.07	2249.90	3323	3115
$(200,5,6,200)$	50	261.67	675.46	937,13	817	1224
$(200,5,6,200)$	80	>2426.99				
$(220,2,4,200(A))$	50	67.68	2239.41	2307.09	6926	3992
$(250,2,3,250(A))$	50	>1441.48				
$(280,2,3,280)$	50	579.76	>657.20	>1236.96		
$(300,2,3,300)$	50	327.46	>5716.48	> 6043.94		

MIQP solver has difficulties solving the box constrained QPCCs, let alone solving it accurately. For one thing, the MIQP solver may encounter "out of memory" errors; for another, the default tolerance for integer variables may need to be adjusted in order to ensure accuracy of the solution. Thus a stronger MIQP solver is needed.

Table 6 contains the computational results for a collection of large-sized QPCCs. In the table, $(m,n,k,rank(Q))$ represents the number of complementary variables y, the number of first level variables x, the number of first level constraints and the rank of the positive semidefinite matrix Q respectively. Also, the letter "A" in the first column denotes that the first level constraints $Ax + By \geq f$ in the QPCC do not involve y.

Table 7 Computational time of QPCCs with unbounded QP relaxation(of 50 complementary variables).

$T = 10^4$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
Max	175.93	191.82	259.43	176	439
Min	1.59	6.87	10.26	20	7
Deviation	56.08	61.62	99.48	49.0	146.8
Average	39.64	52.12	91.76	56.7	113.7

Table 8 Computational time of unbounded QPCCs (with 50 complementary variables).

$T = 10^4$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
Max	2.72	0.55	3.28	1	1
Min	0.96	0.39	1.41	1	1
Deviation	0.65	0.04	0.68	0	0
Average	1.68	0.46	2.14	1	1

Table 9 Computational time of infeasible QPCCs (with 50 complementary variables).

$T = 10^3$	inner box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
Max	0.68	8.15	8.81	28	6
Min	0.53	0.25	0.79	1	1
Deviation	0.06	2.61	2.66	9.0	1.6
Average	0.56	1.39	1.96	4.9	1.7

Table 10 Computational time of some MacMPEC QPCCs

	T	box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
$bilevel2.mod$	100	2.02	14.44	16.50	34	15
$bileve2m.mod$	100	0.80	12.69	13.49	29	13
$flp2.mod$	100	0.76	1.40	2.15	3	2
$flp4-1.dat$	100	0.70	3.73	4.43	10	4
$flp4-2.dat$	100	0.74	4.57	5.31	12	4
$flp4-3.dat$	1000	0.81	4.89	5.70	13	4
$flp4-4.dat$	1000	1.04	7.82	8.86	15	4

We also considered larger-sized finitely solvable QPCCs with unbounded QP relaxations. Table 7 contains the computational results of QPCCs with 50 complementary variables and with unbounded QP relaxations. For this class of QPCCs, we pick T as 10^4 in the inner box constrained QPCCs.

Table 8 displays the computational results of a collection of 10 unbounded QPCCs using $T = 10^4$.

Unbounded QPCCs and infeasible QPCCs are actually easier to solve using our method. They can be solved as efficiently as within a couple of seconds, as we showed in Table 8 and Table 9. In general, only a few LPs and QPs are solved when solving the outer QPCCs.

Finally, we also solved some of the MacMPEC QPCCs [25]. Table 10 displays the computational results for these problems. We confirmed by our approach that the solutions given on the MacMPEC web site were the real global resolutions to the QPCCs.

5.2 Some Numerical Results after Applying the Preprocessing Procedure

As shown in Table 6, when the sizes of the QPCCs are large, we have to pick smaller scalars T as the upper bounds on the complementary variables, otherwise the MIQP solver may collapse due to “out of memory” errors. Thus we would like to combine the MIQP solver with some other techniques, so that at least to some extent, the computational difficulty could be reduced. The preprocessing procedure based on the addition of the penalty term $y^T Dw$ to the objective function turns out to be very helpful. In this subsection, we report some computational results when applying this preprocessing procedure to QPCCs.

We first tested this procedure on the QPCCs in Table 2, ten finitely solvable QPCCs with bounded QP relaxations. The results are given in Table 11, with columns 2–4 giving more details for the computational results in Table 2, and the last 4 columns giving the computational results of using the preprocessing procedure. Using the

Table 11 Comparison between the computational times before and after the preprocessing (as in Table 2)

time(s):	box QPCC	outer QPCC	total	SDPT3	box QPCC	outer QPCC	total
1	12.80	4.79	17.59	3.34	7.95	4.86	16.15
2	10.04	4.77	14.81	3.05	7.42	4.82	15.28
3	30.47	5.47	35.94	3.09	8.01	4.81	15.91
4	23.29	4.79	28.08	3.29	11.09	4.81	19.19
5	27.76	5.01	32.77	3.12	23.93	4.67	31.72
6	72.46	5.04	77.50	3.34	11.63	4.94	19.91
7	38.45	5.40	43.84	3.84	21.49	5.45	30.78
8	21.10	4.93	26.03	3.69	7.98	3.51	15.19
9	55.09	4.89	59.97	3.26	20.97	4.76	28.99
10	49.33	4.82	54.14	3.15	8.43	4.82	16.40
mean	34.08	4.99	39.07	3.32	12.89	4.74	20.95
min	10.04	4.77	14.81	3.05	7.42	3.51	15.19
max	72.46	5.47	77.50	3.84	23.93	5.45	31.72
stdev	19.82	0.25	19.87	0.26	6.57	0.48	6.80

Table 12 Comparison between the initial lower bounds with and without the preprocessing (as in Table 2)

	QPCC optimal value	QP_{rlx} optimal value	QP_{rlx} optimal value (preprocessed)	gap closed(%)
1	-898.0942	-898.4921	-898.1876	76.53
2	-806.4929	-807.0095	-806.6029	78.71
3	-904.9591	-905.3146	-905.0182	83.38
4	-1053.8	-1054.3	-1054.1	60.00
5	-898.6491	-899.0082	-898.781	63.27
6	-842.7464	-843.1964	-842.8756	71.29
7	-986.3559	-986.7223	-986.442	76.50
8	-1329.3	-1329.7	-1329.5	75.00
9	-1282.7	-1283	-1282.8	66.67
10	-744.679	-745.1682	-744.7599	83.46
mean				73.48

Table 13 Computational times of larger sized QPCCs of bounded QP relaxation after the preprocessing (as in Table 6)

$(m,n,k,rank(Q))$	T	SDPT3	box QPCC time(s)	outer QPCC time(s)	total time(s)	LPs	QPs
$(100,10,8,90)$	100	3.77	444.44	3.17	451.37	15	4
$(120,4,8,90)$	100	5.04	499.06	2.57	506.67	13	4
$(150,4,8,150)$	100	7.25	24.01	3.05	34.31	17	4
$(150,4,4,150)$	80	11.44	99.99	2.52	113.95	14	4
$(180,2,4,170)$	50	19.19	1002.22	792.24	1813.65	2329	2216
$(200,5,6,200)$	50	16.59	196.38	3146.30	3359.28	6662	4266
$(200,5,6,200)$	80	16.61	31.43	1756.58	1804.62	6637	3329
$(220,2,4,200(A))$	50	24.72	75.97	438.45	539.14	1406	602
$(250,2,3,250(A))$	50	32.80	58.54	599.37	690.71	1733	627
$(280,2,3,280)$	50	43.68	65.79	694.19	803.67	1873	724
$(300,2,3,300)$	50	54.40	376.09	1655.44	2085.93	2429	1396

preprocessing routine reduces the average computational time of the box constrained QPCCs from 34 seconds to 13 seconds. Further, the average of the total computational times is reduced by nearly a half – from 39 seconds to 21 seconds, even after taking the SDPT3 computational times into account.

In Table 12, we also report the comparison between the initial lower bounds with and without the preprocessing procedure. The initial lower bounds are obtained by solving the QP relaxations. The "gap closed" column is to show how strong the new initial lower bounds (after the addition of the penalty term) are in comparison with the initial lower bounds before the addition of the penalty term.

Finally, we tested the preprocessing procedure on the QPCCs in Table 6. The total runtime is reduced for most of the instances. For some of the QPCCs, the total run times can be reduced by hundreds of seconds. A plot of

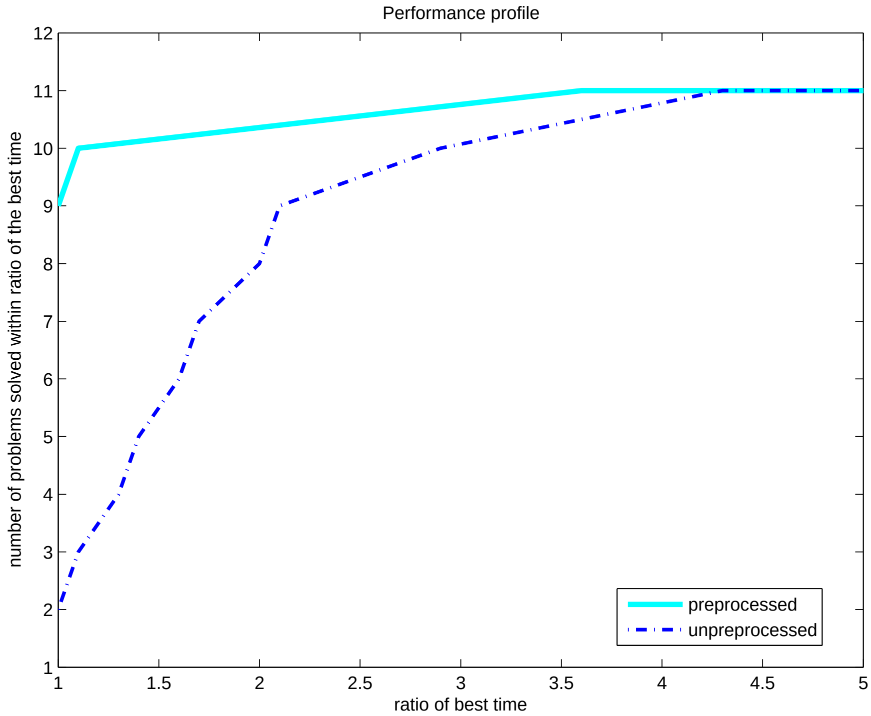

Fig. 3 Performance profile with large-sized QPCCs before and after the addition of the penalty term $y^T Dw$

Table 14 KNITRO computational results of QPCCs with bounded QP relaxations as in Table 6

$(m,n,k,rank(Q))$	true optima	solution by KNITRO	feasibility error	exit flags
$(100,10,8,90)$	-1017.3	-1017.30415	7.96E-13	$flag$ 1
$(120,4,8,90)$	-121.248	-121.247512	4.41E-12	$flag$ 1
$(150,4,8,150)$	-1765.1	-1765.07846	8.71E-08	$flag$ 2
$(150,4,4,150)$	-260.463	-260.462886	3.26E-13	$flag$ 1
$(180,2,4,170)$	-2614.2	-2614.21733	3.11E-13	$flag$ 1
$(200,5,6,200)$	-5596.1	-5523.322465	1.12e-09	$flag$ 2
$(200,5,6,200)$	-815.4117	-667.9118236	1.07e-12	$flag$ 2
$(220,2,4,200(A))$	-2371.2	-2096.131103	0.000112	$flag$ 1
$(250,2,3,250(A))$	-953.9291	-893.0508607	2.12e-10	$flag$ 2
$(280,2,3,280)$	-90.9804	69.19056883	0.000507	$flag$ 1
$(300,2,3,300)$	-6659.1	-6397.381507	1.69e-06	$flag$ 2

performance profile is included in Figure 3. It confirms that the computational time is dramatically reduced by adding the penalty term to the objective function of the QPCC.

5.3 Comparison with KNITRO

In this section, we present the computational results of our test problems solved by KNITRO and then make a comparison with results obtained by our approach. We tested all four types of QPCCs on KNITRO: feasible QPCCs with bounded QP relaxations, finitely solvable QPCCs with unbounded QP relaxations, unbounded QPCCs and infeasible QPCCs. The computational results are presented in Tables 14-17. The problems tested may have one of the following possible exit flags:

1. $flag$ 1: "Primal feasible solution; terminate because the relative change in solution estimate $< x_{tol}$".
2. $flag$ 2: "Locally optimal solution".
3. $flag$ 3: "Iteration limit reached".
4. $flag$ 4: "Presolve finds no feasible solution possible".

In Table 14, KNITRO is mostly incapable of evaluating the quality of its computational iterates for the class of large-sized feasible QPCCs with bounded QP relaxations. KNITRO either terminated as the relative change in the

Table 15 KNITRO computational results of QPCCs with unbounded QP relaxations as in Table 4

	true optima	solution by KNITRO	feasibility error	exit flags
1	-444.0181	-444.0317911	3.45E-08	*flag* 2
2	-1703.3	-1703.207792	4.5E-10	*flag* 2
3	-639.6325	-469.4700176	5.88E-09	*flag* 2
4	-158.9225	-158.922	4.88E-11	*flag* 2
5	-822.0306	-178.3615839	5.34E-09	*flag* 2
6	-2506.9	-1392.066308	1.37E-11	*flag* 2
7	-120.5437	-120.5452211	1.32E-09	*flag* 2
8	-743.4272	-743.4658513	7.00E-08	*flag* 2
9	-232.8212	-64.35878218	8.65E-10	*flag* 2
10	-171.3706	-171.3696801	1.23E-05	*flag* 2

Table 16 KNITRO computational results of unbounded QPCCs as in Table 8

	solution by KNITRO	feasibility error	exit flags
1	-19971367.53	2.58E-11	*flag* 2
2	-5382.016321	1.38E-09	*flag* 2
3	-3.95E+11	0.00905	*flag* 3
4	-508.1576072	5.69E-10	*flag* 2
5	-48903.2844	1.44E-11	*flag* 2
6	-7.65E+15	3.16E-08	*flag* 1
7	-8.07E+14	4.76E-06	*flag* 3
8	-7.23E+15	0.000607	*flag* 3
9	-4.08E+12	2.82E-05	*flag* 3
10	-46002.16193	0.100984	*flag* 2

Table 17 KNITRO computational results of infeasible QPCCs as in Table 9

	solution by KNITRO	feasibility error	exit flags
1	∞		*flag* 4
2	∞		*flag* 4
3	∞		*flag* 4
4	5551995318	0.00274	*flag* 3
5	∞		*flag* 4
6	∞		*flag* 4
7	∞		*flag* 4
8	∞		*flag* 4
9	121919866.5	0.00173	*flag* 3

solution became smaller than the default tolerance, in which case a smaller tolerance x_{tol} may be needed in order to get possibly better results, or could only find "local optima" but not the global optima.

In Table 15, KNITRO can successfully terminate on the class of finitely solvable QPCCs with unbounded QP relaxations, but the obtained solutions are not guaranteed to be globally optimal. The KNITRO solutions of Problems $3, 5, 6, 9$ are not globally optimal. In contrast, due to the incorporation of some mechanisms for verifying global optimality, our approach can achieve the global resolution of the QPCCs.

Table 16 and Table 17 show that KNITRO is not very successful in confirming either the unboundedness or the infeasibility of the QPCCs. In Table 16, KNITRO either terminated unsuccessfully or wrongly ascertained the status of the unbounded QPCCs. In Table 17, KNITRO could not determine the status of certain infeasible QPCCs as we showed. In fact, our approach can easily ascertain the correct statuses of both unbounded and infeasible QPCCs.

6 A Method to Obtain Better Lower Bounds from Infeasibility

In this section, we are going to introduce a method to get better lower bounds of QPCCs from infeasibility; this method could be incorporated into a branching scheme. A lower bound can be obtained by choosing a subset I_0 of the complementarities and solving all $2^{|I_0|}$ quadratic programs corresponding to a particular assignment of the complementarities in I_0. The minimum value of these quadratic programs gives a valid lower bound. The standard quadratic program relaxation of the QPCC corresponds to taking $I_0 = \emptyset$. Inspired by the work of Bienstock [4] on certain types of mixed integer quadratic programs, the lower bounding procedure is modified in two ways. First, solving $2^{|I_0|}$ quadratic programs is too expensive, so we drop the inequality constraints from the quadratic programs, thus the relaxed quadratic programs have closed form solutions. Secondly, in order to improve the bound, we include an additional linear constraint based upon the gradient of the objective function. This constraint tightens up the relaxation because it restricts the feasible region of each of the quadratic programs to lie in the halfspace of points that are no better than the solution to the standard QP relaxation of the QPCC. The KKT conditions provide another motivation: the gradient of the objective is a nonpositive combination of the gradients of the active inequality constraints, so the gradient constraint is a proxy for the set of discarded inequality constraints.

Let $I_0 \subseteq \{1,\dots,m\}$. Any feasible point of the QPCC should at least have

$$0 \le y_{I_0} \perp w_{I_0} \ge 0.$$

An underestimation $\tilde{\theta}_0(I_0)$ of the QPCC could be obtained by minimizing (29) over all possible disjoint partitions (I,J) of I_0 and relaxing all the inequality constraints:

$$\tilde{\theta}_0(I_0) := \min_{I,J\subseteq I_0, I\cup J = I_0, I\cap J=\emptyset} \left\{ \tilde{\theta}(I,J) := \begin{array}{ll} \text{minimize}_{(x,y)} & q(x,y) \\ \text{subject to} & w_I = (q+Nx+My)_I = 0 \\ & y_J = 0. \end{array} \right\} \tag{29}$$

No feasible solution to QPCC can have objective function value better than $(\bar{x},\bar{y})$ (the optimal solution to the standard QP relaxation), so we can strengthen the lower bound by picking g equal to $\nabla q(\bar{x},\bar{y})$ and adding a linear constraint to (29), giving the quadratic program:

$$\begin{aligned} \hat{\theta}(I,J) := \text{minimize}_{(x,y)} \;& q(x,y) \\ \text{subject to} \quad & w_I = (q+Nx+My)_I = 0 \\ & y_J = 0 \\ & g^T \begin{pmatrix} x \\ y \end{pmatrix} \ge g^T \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}. \end{aligned} \tag{30}$$

Every feasible solution to the QPCC is feasible in (30) for at least one partition (I,J) of I_0, so we have the valid lower bound

$$\hat{\theta}_0(I_0) := \min_{I,J\subseteq I_0, I\cup J = I_0, I\cap J=\emptyset} \hat{\theta}(I,J). \tag{31}$$

Define programs $\hat{\theta}_1(I,J)$ and $\hat{\theta}_2(I,J)$,

$$\begin{aligned} \hat{\theta}_1(I,J) := \text{minimize}_{(x,y)} \;& q(x,y) \\ \text{subject to} \quad & w_I = (q+Nx+My)_I = 0 \\ & y_J = 0 \\ & g^T \begin{pmatrix} x \\ y \end{pmatrix} = g^T \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}, \end{aligned} \tag{32}$$

$$\begin{aligned} \hat{\theta}_2(I,J) := \text{minimize}_{(x,y)} \;& q(x,y) \\ \text{subject to} \quad & w_I = (q+Nx+My)_I = 0 \\ & y_J = 0. \end{aligned} \tag{33}$$

Note that $\hat{\theta}_1(I,J) \ge \hat{\theta}(I,J) \ge \hat{\theta}_2(I,J)$. We have the following proposition:

Proposition 7 *If there exists a feasible solution* (x^2,y^2) *to (33) with* $q(x^2,y^2) < \hat{\theta}_1(I,J)$ *and* $g^T\begin{pmatrix} x^2 \\ y^2 \end{pmatrix} > g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$ *then* $\hat{\theta}(I,J) = \hat{\theta}_2(I,J)$. *Otherwise,* $\hat{\theta}(I,J) = \hat{\theta}_1(I,J)$.

Proof First note that if every feasible solution (x^2,y^2) to (33) satisfies $q(x^2,y^2) \geq \hat{\theta}_1(I,J)$ then $\hat{\theta}_1(I,J) = \hat{\theta}_2(I,J)$, so the result holds. Thus, assume $\hat{\theta}_1(I,J) > \hat{\theta}_2(I,J)$. We break the proof into two cases.
(i) There exists a feasible solution (x^2,y^2) to (33) with $q(x^2,y^2) < \hat{\theta}_1(I,J)$ and $g^T\begin{pmatrix} x^2 \\ y^2 \end{pmatrix} > g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$:
Assume there exists another feasible solution $(\bar{x}^2,\bar{y}^2)$ to (33) with $q(\bar{x}^2,\bar{y}^2) < \hat{\theta}_1(I,J)$ and $g^T\begin{pmatrix} \bar{x}^2 \\ \bar{y}^2 \end{pmatrix} < g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$. Then a convex combination $(\hat{x}^2,\hat{y}^2)$ of (x^2,y^2) and $(\bar{x}^2,\bar{y}^2)$ is feasible in (32). By the convexity of $q(x,y)$, this point satisfies

$$q(\hat{x}^2,\hat{y}^2) \leq \max\{q(x^2,y^2), q(\bar{x}^2,\bar{y}^2)\} < \hat{\theta}_1(I,J),$$

a contradiction. Thus, every feasible solution $(\bar{x}^2,\bar{y}^2)$ to (33) with $q(\bar{x}^2,\bar{y}^2) < \hat{\theta}_1(I,J)$ is feasible in (30) so $\hat{\theta}(I,J) = \hat{\theta}_2(I,J)$.
(ii) Every feasible solution (x^2,y^2) to (33) with $q(x^2,y^2) < \hat{\theta}_1(I,J)$ satisfies $g^T\begin{pmatrix} x^2 \\ y^2 \end{pmatrix} < g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$:
Let (x^2,y^2) be a particular feasible solution to (33) satisfying $q(x^2,y^2) < \hat{\theta}_1(I,J)$ and $g^T\begin{pmatrix} x^2 \\ y^2 \end{pmatrix} < g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$. Let $(\hat{x},\hat{y})$ be feasible in (30) and infeasible in (32). Then there exists a feasible solution (x^1,y^1) to (32) that is a convex combination of $(\hat{x},\hat{y})$ and (x^2,y^2). By the convexity of $q(x,y)$, we have $q(\hat{x},\hat{y}) > q(x^1,y^1)$. It follows that the optimal solution to (30) must be feasible in (32), so $\hat{\theta}(I,J) = \hat{\theta}_1(I,J)$.

In certain situations, the value of $\hat{\theta}(I,J)$ can be determined by solving two systems of linear equations, as we illustrate in the next proposition.

Proposition 8 *If the projection of the objective quadratic* $\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}$ *onto the the constraints of problem* $\hat{\theta}_1(I,J)$ *is full rank and the constraints are linearly independent, then (30) is equivalent to a single variable optimization problem of the form*

$$\begin{aligned} &\textit{minimize}_\gamma\ \frac{1}{2}b^T\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}b\gamma^2 + \left(a^T\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}b + \begin{pmatrix} c \\ d \end{pmatrix}^T b\right)\gamma + \begin{pmatrix} c \\ d \end{pmatrix}^T a + \frac{1}{2}a^T\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}a \\ &\textit{subject to}\ \ \gamma \geq g^T\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}. \end{aligned} \tag{34}$$

Here a, b *are two deterministic vectors.*

Proof This can be proved by examining the KKT conditions of the following program

$$\begin{aligned} \text{minimize}_{(x,y,\gamma)}\ & c^Tx + d^Ty + \frac{1}{2}\begin{pmatrix} x \\ y \end{pmatrix}^T\begin{bmatrix} Q_1 & R \\ R^T & Q_2 \end{bmatrix}\begin{pmatrix} x \\ y \end{pmatrix} \\ \text{subject to}\ \ & w_I = (q + Nx + My)_I = 0 \\ & y_J = 0 \\ & g^T\begin{pmatrix} x \\ y \end{pmatrix} = \gamma. \end{aligned} \tag{35}$$

Define $H = \begin{bmatrix} N(I,:) & M(I,:) \\ 0 & \hat{I}(J,:) \end{bmatrix} \in \Re^{(|I|+|J|)\times(n+m)}$ and $h = \begin{pmatrix} -q(I) \\ 0 \end{pmatrix}$, where $\hat{I}$ is the $m \times m$ identity matrix. The optimal solution and the KKT multipliers of program (35) must satisfy the following system of linear equations,

$$P\begin{pmatrix} x \\ y \\ u \\ \alpha \end{pmatrix} := \begin{bmatrix} Q & H^T & g \\ H & 0 & 0 \\ g^T & 0 & 0 \end{bmatrix}\begin{pmatrix} x \\ y \\ u \\ \alpha \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ h \\ \gamma \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ h \\ 0 \end{pmatrix} + \gamma\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{36}$$

Table 18 gap closed ($m = 50, k = 5, n = 10$ and condition number of $Q = 20$)

	ub	QP_{rlx}	lb (penalty)	gap closed(%)	lb from (31)	gap closed(%)	SDPT3 time	(31) time
1	-309.8868	-310.2951	-309.9126	93.68	-309.9074	94.95	1.19	1.22
2	-67.4246	-67.4938	-67.431	90.75	-67.4305	91.47	1.13	0.38
3	-138.8104	-139.6244	-138.8375	96.67	-138.8304	97.54	1.11	0.92
4	-143.4102	-143.4689	-143.4115	97.79	-143.4112	98.30	1.24	0.36
5	-264.4806	-264.5879	-264.4823	98.42	-264.4818	98.88	1.16	0.94
6	-145.4621	-145.5447	-145.4703	90.07	-145.4668	94.31	1.12	0.66
7	-20.8979	-21.7369	-21.1698	67.59	-21.1275	72.63	1.15	0.94
8	-197.8096	-197.8545	-197.8098	99.55	-197.8098	99.55	1.21	0.38
9	-217.5868	-217.6531	-217.5894	96.08	-217.5891	96.53	1.20	0.76
10	-251.3961	-251.4479	-251.4017	89.19	-251.4012	90.15	1.21	0.45
				91.98		93.43	1.17	0.70

Table 19 gap closed ($m = 100, k = 2, n = 10$ and condition number of $Q = 2$)

	ub	QP_{rlx}	lb (penalty)	gap closed(%)	lb from (31)	gap closed(%)	SDPT3 time	(31) time
1	-3.1726	-9.236	-7.4163	30.01	-7.4019	30.25	2.83	1.72
2	-30.2126	-30.5237	-30.2514	87.53	-30.2459	89.30	3.28	1.66
3	-46.1831	-46.6227	-46.2822	77.46	-46.2532	84.05	2.87	1.87
4	-6.7123	-8.7065	-7.4089	65.07	-7.3865	66.19	2.70	1.82
5	-22.5016	-23.41	-22.7475	72.93	-22.7397	73.79	2.60	1.68
6	-41.5942	-41.7507	-41.6029	94.44	-41.6007	95.85	2.92	1.78
7	-106.4055	-106.5383	-106.4105	96.23	-106.4103	96.39	2.61	1.71
8	-25.0137	-25.4995	-25.1382	74.37	-25.1231	77.48	2.90	1.67
9	-46.1831	-46.6227	-46.2822	77.46	-46.2532	84.05	2.54	1.61
10	-2.2237	-4.6037	-3.1505	61.06	-3.1095	62.78	2.47	1.66
				73.66		76.01	2.77	1.72

Here matrix Q is positive semidefinite, u and α are the KKT multipliers corresponding to the constraints in program (35). As the projection of Q onto the constraints is of full rank and the constraints are linearly independent, program (35) has a unique optimal solution for any γ. Thus the unique optimal solution of program (35) can be written as $a + b\gamma$, with $a = (a_x, a_y), b = (b_x, b_y)$ resulting respectively from

$$P \begin{pmatrix} a_x \\ a_y \\ u \\ \alpha \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ h \\ 0 \end{pmatrix}, \tag{37}$$

and

$$P \begin{pmatrix} b_x \\ b_y \\ \bar{u} \\ \bar{\alpha} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{38}$$

Substituting for (x, y) in (30) gives (35).

We report some computational results in Tables 18, 19 and 20. Our method is tested on QPCCs with full rank quadratic objectives and with equality side constraints only; however, it also works for QPCCs with inequality constraints. In the test, we first add the penalty term $y^T Dw$ to the objective functions of the QPCCs, then we use (31) to obtain the lower bounds. We report the lower bounds and the percentage of gap closed to the upper bounds of the QPCCs. All reported running times are in seconds. We considered QPCCs with 50, 100 and 300 pairs of complementary variables respectively. In the testing, we use $|I_0| = 10$ indices of the entries where the initial infeasible solution $(\bar{y}, \bar{w})$ violates the complementarity constraints.

By adding the penalty term $y^T Dw$ to the objective function of the QPCC only, we can close as much as 90% of the gap on average when the size of the QPCCs is not large. The QP lower bounding technique helps to close 18.08%, 8.92% and 3.55% of the remaining gaps respectively if we combine the two methods. By using just the QP lower bounding technique, we could close slightly less than 50% of the gap in the best of the above 3 cases.

Table 20 gap closed ($m = 300, k = 5, n = 10$ and condition number of $Q = 1$)

	ub	QP_{rlx}	lb (penalty)	gap closed(%)	lb from (31)	gap closed(%)	SDPT3 time	(31) time
1	-101.6011	-102.1054	-101.7406	72.34	-101.7294	74.56	93.99	10.96
2	-62.5099	-68.4999	-64.2666	70.67	-64.1999	71.79	69.27	10.91
3	-5.2152	-9.7033	-7.0351	59.45	-6.9769	60.75	77.90	11.27
4	-2.5884	-29.8765	-25.8613	14.71	-25.6694	15.42	77.11	11.57
5	-256.6023	-258.1193	-256.7252	91.90	-256.721	92.18	74.18	11.27
6	-6.0225	-9.7167	-7.3488	64.10	-7.3013	65.38	79.37	10.58
7	-13.2958	-20.2235	-17.0647	45.60	-16.9537	47.20	75.10	11.08
8	-21.8755	-26.3378	-23.3456	67.06	-23.2272	69.71	72.86	11.08
9	-58.8571	-60.7365	-59.2903	76.95	-59.244	79.41	83.19	11.55
10	-2.9081	-8.4514	-5.7658	48.45	-5.7569	48.61	69.80	11.05
				61.12		62.50	69.20	11.13

7 Concluding Remarks and Future Work

Our computational results demonstrate that we can successfully find globally optimal solutions to convex quadratic programs with at least 100 complementarity constraints, or provide certificates that no such solutions exist. Our method effectively combines two algorithms for disjunctive programs (namely, branch-and-cut and logical Benders decomposition) and involves two refinements: preconditioning by adding a quadratic penalty term to the objective, and improved lower bound generation. Our results can be contrasted with those obtained by a local optimization nonlinear programming package, which is incapable of providing any information on the quality of the solution obtained when the method terminates successfully, or offering certificates for infeasible or unbounded problems. Such existing software is particularly unreliable for solving QPCCs with unbounded QP relaxations.

From computational experience, the current MIQP solvers have difficulties solving large-sized QPCCs where the variables have large finite bounds. The main reason for computational difficulty is running out of memory. If the box constrained QPCCs could be solved more efficiently, then the global resolution of large-scale QPCCs could be enhanced.

References

1. Audet, C., Savard, G., Zghal, W.: New branch-and-cut algorithm for bilevel linear programming. Manage. Sci. **134**, 353–370 (2007).
2. Audet, C., Hansen, P., Jaumard, B., Savard, G.: Links between linear bilevel and mixed 0-1 programming problems. J. Optim. Theory Appl. **93**, 273–300 (1997).
3. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming: Theory and Algorithms. John Wiley& Sons, Inc., New Jersey (2006).
4. Bienstock, D.: Eigenvalue techniques for proving bounds for convex objective, nonconvex programs. IPCO 29-42 (2010).
5. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. Found. Comput. Math. **9**, 717–772 (2009).
6. Chen, X., Ye, J.: A class of quadratic programs with linear complementarity constraints. Set-Valued and Variational Ana. **17**, 113–133 (2010).
7. Codato, G., Fischetti, M.: Combinatorial Benders' cuts for mixed-integer linear programming. Oper. Res. **54**, 738–766 (2006).
8. Eaves, B.C.: On quadratic programing. Manage. Sci. **17**, 698–711 (1971).
9. Fazel, M.. Matrix rank minimization with applications. Ph.D. thesis, Electrical Engineering Department, Stanford University (2002).
10. Fazel, M., Hindi, H., Boyd, S.: Rank minimization and applications in system theory. Proceedings of the 2004 American Control Conference, Boston, Massachusetts June 30–July 2, 2004. 3273–3278 (2004).
11. Han, Z.: A SAT solver implemented in MATLAB. http://www.mathworks.com/matlabcentral/fileexchange/22284-satisfiability-solver. Accessed September 2011.
12. Hooker, J.N.: Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction. Wiley, (2000).
13. Hooker, J.N.: Integrated Methods for Optimization. Springer, New York (2006).
14. Hooker, J.N., Ottosson, G.: Logic-based Benders decomposition. Math. Program. **96**, 33–60 (2003).
15. Hu, J. On linear programs with linear complementarity constraints. Ph.D. thesis, Mathematical Sciences, Rensselaer Polytechnic Institute (2009).
16. Hu, J., Mitchell, J.E., Pang, J.S., Bennett, K.P., Kunapuli, G.: On the global reolution of linear programs with linear complementarity constraints. SIAM J. on Optimization. **19**, 445–471 (2008).
17. Hu, J., Mitchell, J.E., Pang, J.S., Yu, B.: On linear programs with linear complementarity constraints. J. Global Optim. doi: 10.1007/s10898-010-9644-3.
18. Hu, J., Mitchell, J.E., Pang, J.S.: An LPCC approach to nonconvex quadratic programs. Math. Program. doi: 10.1007/s10107-010-0426-y.
19. Jiang, H., Ralph, D.: QPECgen, a MATLAB generator for mathematical programs with quadratic objectives and affine variational inequality constraints. Comput. Optim. Appl. **13**, 25–59 (1999).
20. Luo, Z.Q., Pang, J.S., Ralph, D.: Mathematical Programs With Equilibrium Constraints. Cambridge University Press, New York (1996).
21. Mitchell, J.E., Pang, J.S., Yu,B.: Obtaining tighter relaxation of linear programs with complementarity constraints. *Proceedings of the Conference on Modeling and Optimization, Theory and Applications* (2010).

22. Pang, J.S.: Three modeling paradigms in Mathematical Programming. Math. Program. **125**, 297–323 (2010).
23. Toh, K.C., Todd, M.J., Tutuncu, R.H.: SDPT3—a Matlab software package for semidefinite programming. Optim. Methods and Soft. **11**, 545–581 (1999).
24. Yu, B.. A branch and cut approach to linear programs with linear complementarity constraints. Ph.D. thesis, Mathematical Sciences, Rensselaer Polytechnic Institute (2011).
25. "MacMPEC Test Problems." http://wiki.mcs.anl.gov/leyffer/index.php/MacMPEC. Accessed September 2011.