Convex Quadratic Relaxations of Nonconvex Quadratically Constrained Quadratic Progams

John E. Mitchell[*], Jong-Shi Pang[†], and Bin Yu[‡]

Original: June 10, 2011

Abstract

Nonconvex quadratic constraints can be linearized to obtain relaxations in a well-understood manner. We propose to tighten the relaxation by using second order cone constraints, resulting in a convex quadratic relaxation. Our quadratic approximation to the bilinear term is compared to the linear McCormick bounds. The second order cone constraints are based on linear combinations of pairs of variables. With good bounds on these linear combinations, the resulting constraints strengthen the McCormick bounds. Computational results are given, which indicate that the convex quadratic relaxation can dramatically improve the solution times for some problems.
Keywords: quadratically constrained quadratic programs; second order cones; convex outer approximations.

1 Introduction

We are interested in quadratically constrained quadratic programs of the form

$$\begin{array}{llllll} \min_x & c_0^T x & + & \frac{1}{2}x^T Q^0 x & & \\ \text{subject to} & c_i^T x & + & \frac{1}{2}x^T Q^i x & \leq & g_i \qquad i = 1, \ldots, p \\ & & & Ax & \geq & b \end{array} \tag{1}$$

where x and each c_i are n-vectors, g is a p-vector, b is an m-vector, the matrices A and Q^i are dimensioned appropriately, and each Q^i is symmetric. If each Q^i is positive semidefinite

[*]Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, New York 12180-1590, U.S.A. Email:mitchj@rpi.edu. The work of this author 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.

[†]Department of Industrial and Enterprise Systems Engineering, University of Illinois, Urbana, Illinois 61801, U.S.A. Email: jspang@illinois.edu. The work of this author was supported by the National Science Foundation grant CMMI-0969600 and by the Air Force Office of Sponsored Research under grant FA9550-08-1-0061.

[‡]Department of Industrial and Systems Engineering, Rensselaer Polytechnic Institute, Troy, New York 12180-1590, U.S.A. Email:yub@rpi.edu.

then (1) is convex; otherwise in general it is nonconvex. This is perhaps the simplest type of nonconvex optimization problem and arises in many contexts. For example, Adjiman et al. [1] describe applications in chemical engineering, Al-Khayyal et al. [3] list a number of applications, Bao et al. [5] give references to several applications, and many other formulations are discussed by Floudas et al. [13]. When the constraints are all linear, the best stationary point can be found by solving a linear program with complementarity constraints; see [10, 11, 14, 27, 28], for example. The paper [14] also contains a method for handling the case when the feasible region is unbounded.

To simplify the presentation, we work with the following equivalent form which has a linear objective:

$$\begin{array}{llrcll} \min_x & c_0^T x & & & & \\ \text{subject to} & c_i^T x + \frac{1}{2}x^T Q^i x & \leq & g_i & & i = 1, \ldots, p \\ & & Ax & \geq & b & \end{array} \tag{2}$$

Quadratically constrained quadratic programs with a quadratic term $\frac{1}{2}x^T Q^0 x$ in the objective can be expressed as an equivalent problem of the form (2) using the standard technique of introducing a variable x_0 and a constraint $-x_0 + \frac{1}{2}x^T Q^0 x \leq 0$ and then modifying the objective.

Kojima and Tunçel [15] describe several successive convex relaxation procedures for determining the convex hull of the feasible region of (2). However, these routines are computationally expensive, and so there is interest in finding good valid convex relaxations using simpler methods. McCormick [20] gave some linear inequalities based only on the bounds on the variables x and Al-Khayyal et al. [3] showed that this gives the convex hull of a bilinear inequality $x_j x_k \leq \sigma_{jk}$. Belotti et al. [8] looked at linear inequalities for bilinear and higher degree products of variables. Bao et al. [5] derived linear constraints based on combining together several of the original inequalities, solving linear programs to derive the inequalities. The same authors [6] also examined various semidefinite relaxations of (2). Linderoth [18] extended results in Sherali and Alameddine [24] to give second order cone constraints that are valid on certain types of triangular regions, and showed that these inequalities give the best convex underestimators or concave overestimators of $x_j x_k$ in certain cases; he then used these inequalities in a branch-and-bound approach. Richard and Tawarmalani [21] discussed methods for extending inequalities valid for subsets of the variables to inequalities valid for all the variables. Burer [9] showed that nonconvex quadratic programs are equivalent to convex programs over the cone of completely positive matrices, and extended these results to certain types of quadratically constrained convex programs.

The general quadratic constraint in (2) can be written

$$\sum_{j=1}^{n} c_{ij} x_j + \frac{1}{2} \sum_{j=1}^{n} Q^i_{jj} x_j^2 + \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} Q^i_{jk} x_j x_k \leq g_i. \tag{3}$$

Our focus in this paper is the derivation of a method for relaxing the bilinear terms $x_j x_k$ using convex quadratic and second order cone constraints. The construction involves writing $x_j x_k$ as a difference of two squares and then using a linear overestimator for one of the squares.

We derive a family of second order cone constraints in §2 and compare it to a standard set of linear inequalities. We show how to choose good members of the family of constraints in §3. We describe an alternative convex underestimator in §4. We develop an alternative motivation of the family of constraints in §5, and present computational results in §6. The computational results demonstrate the benefit of the alternative convex underestimator, in terms of its effect on the quality of the relaxation and on the ability to determine a globally optimal solution.

2 Convex quadratic bounds on a product of variables

It is standard to linearize bilinear terms by replacing them by a new variable and then developing linear constraints relating this new variable to the original variables (see for example Balas [4], Sherali and Adams [23], Lovász and Schrijver [19], and Laurent [17]). We introduce variables y_j for $j = 1, \ldots, n$ and σ_{jk} for $1 \le j < k \le n$. Using the σ and y variables, constraint (3) is equivalent to the convex constraint

$$\sum_{j=1}^{n} c_{ij} x_j + \tfrac{1}{2} \sum_{Q^i_{jj}>0, j=1}^{n} Q^i_{jj} x_j^2 + \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} Q^i_{jk} \sigma_{jk} \;\le\; g_i + \tfrac{1}{2} \sum_{Q^i_{jj}<0, j=1}^{n} |Q^i_{jj}| y_j \tag{4}$$

together with the nonconvex constraints

$$\begin{aligned} x_j x_k &= \sigma_{jk} \qquad \text{for } 1 \le j < k \le n & (5)\\ x_j^2 &= y_j \qquad \text{for } j = 1, \ldots, n. & (6)\end{aligned}$$

Constraints (5) and (6) can then be approximated using convex constraints, to give a convex relaxation of (2). We assume the linear constraints $Ax \ge b$ imply bounds $x^L \le x \le x^U$. (These bounds are either given to us, or we can calculate them by minimizing and maximizing the individual components of x subject to the linear constraints.) The concave upper envelope of (6) is

$$(x_j^L + x_j^U) x_j - x_j^L x_j^U \ge y_j. \tag{7}$$

McCormick[20] introduced the following valid linear constraints to relax (5):

$$\left.\begin{array}{rcl} x_k^L x_j + x_j^L x_k & \le & \sigma_{jk} + x_j^L x_k^L \\ x_k^U x_j + x_j^U x_k & \le & \sigma_{jk} + x_j^U x_k^U \\ x_k^L x_j + x_j^U x_k & \ge & \sigma_{jk} + x_j^U x_k^L \\ x_k^U x_j + x_j^L x_k & \ge & \sigma_{jk} + x_j^L x_k^U \end{array}\right\} \quad 1 \le j < k \le n. \tag{8}$$

It was proved in [3] that these inequalities give the convex lower envelope and concave upper envelope of (5) when only simple bound constraints are available. These constraints are exploited in packages for nonconvex optimization, including BARON [22, 25], αBB [2], and COUENNE [7]. Tightening the bounds x_j^L and x_j^U can be very useful [7]. Inequalities combining together terms for several indices j have been investigated by Bao et al. [5]. In this paper, we show that inequalities that strengthen (8) can be derived when the variables x are restricted to lie in a more complicated convex set than the simple box.

The derivation will exploit the convex set

$$\mathcal{P} := \{x : x^L \le x \le x^U,\ Ax \ge b,\ c_0^T x \le \bar{g}_0,\ \exists\, y, \sigma \text{ satisfying } (4), (7), \text{ and } (8)\}$$

and its projection $\mathcal{P}^{jk}$ onto (x_j, x_k) space, where $\bar{g}_0$ is a valid upper bound on the optimal value of (2). We would like to use the structure of $\mathcal{P}^{jk}$ to obtain a convex outer approximation of the set

$$\Xi^{jk} := \{(x_j, x_k, \sigma_{jk}) : (x_j, x_k) \in \mathcal{P}^{jk},\ \sigma_{jk} = x_j x_k\}$$

which is tighter than that provided by (8). An important observation is that constraint (5) is equivalent to the following constraint, for any $\alpha \neq 0$,

$$(x_j + \alpha x_k)^2 = 4\alpha\sigma_{jk} + (x_j - \alpha x_k)^2 \tag{9}$$

as can be seen by expanding the squares. This constraint can be relaxed to a convex quadratic constraint if upper and lower bounds α^U and α^L on $x_j - \alpha x_k$ are known. In particular, we obtain the valid convex constraint

$$(x_j + \alpha x_k)^2 \le 4\alpha\sigma_{jk} + (\alpha^L + \alpha^U)(x_j - \alpha x_k) - \alpha^L\alpha^U \tag{10}$$

from the concave upper envelope (7) of $(x_j - \alpha x_k)^2$. Positive choices for α give lower bounds on σ_{jk}, and negative choices give upper bounds. We define

$$f_\alpha(x_j, x_k) := (x_j + \alpha x_k)^2 - (\alpha^L + \alpha^U)(x_j - \alpha x_k) + \alpha^L\alpha^U \tag{11}$$

so (10) can be written

$$f_\alpha(x_j, x_k) \le 4\alpha\sigma_{jk}. \tag{12}$$

Valid upper and lower bounds on $x_j - \alpha x_k$ can be constructed trivially as

$$\alpha^L = x_j^L - \alpha x_j^U \quad \text{and} \quad \alpha^U = x_j^U - \alpha x_k^L. \tag{13}$$

However, using such weak bounds in (10) will not improve on (8), since the latter constraints define the convex envelope. We summarize this in the following lemma.

Lemma 1. *If the bounds (13) are used for $x_j - \alpha x_k$ then (10) is implied by (8).*

It follows that it is essential to improve on the bounds (13), perhaps by solving the second order cone programs

$$\min_{x,y,\sigma}\{\pm(x_j - \alpha x_k) : Ax \ge b,\ c_0^T x \le \bar{g}_0,\ (4), (7), \text{ and } (8)\}.$$

For example, assume the projection $\mathcal{P}^{jk}$ is given by

$$x_j^L \le x_j \le x_j^U, \quad x_k^L \le x_k \le x_k^U, \quad L^{jk} \le x^j - x^k \le U^{jk}$$

and the bounds are such that the projection can be represented as illustrated in Figure 1. The McCormick lower bounds (8) agree with the bilinear term $x_j x_k$ on the horizontal and vertical boundaries of the region in the figure. On the diagonal boundaries, the lower bound given by (10) is equal to the bilinear term, as we show in the following lemma.

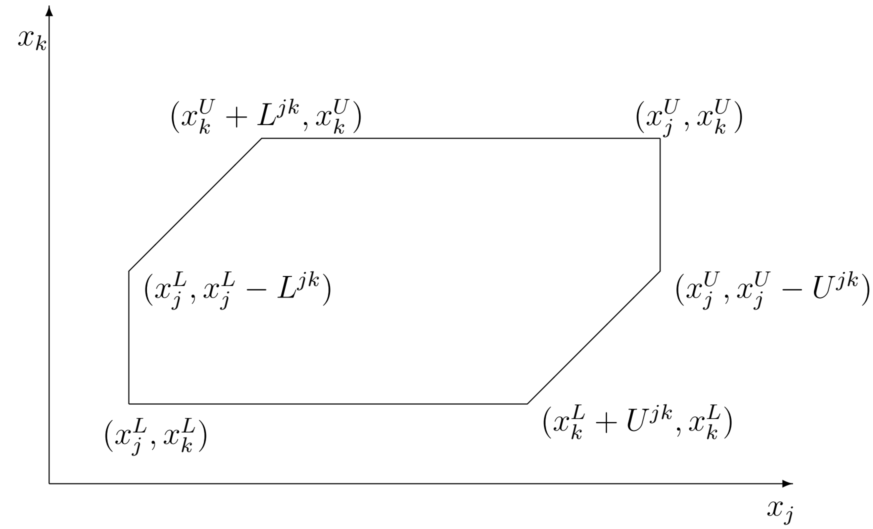

Figure 1: An outer approximation of the feasible region in (x_j, x_k)-space.

Lemma 2. *Let $f_\alpha(x_j, x_k)$ be given by (11). If $x_j - \alpha x_k = U^{jk}$ or $x_j - \alpha x_k = L^{jk}$ then $f_\alpha(x_j, x_k) = 4\alpha x_j x_k$.*

Proof. If $x_j - \alpha x_k = U^{jk}$ then

$$\begin{aligned} 4\alpha x_j x_k &= (x_j + \alpha x_k)^2 - (x_j - \alpha x_k)^2 \\ &= (x_j + \alpha x_k)^2 - (U^{jk})^2 \\ &= (x_j + \alpha x_k)^2 - ((U^{jk})^2 + L^{jk}U^{jk} - U^{jk}L^{jk}) \\ &= f_\alpha(x_j, x_k). \end{aligned}$$

The case of $x_j - \alpha x_k = L^{jk}$ follows similarly. □

We have the following corollary for facet-defining inequalities of $\mathcal{P}^{jk}$ that follows directly from the lemma.

Corollary 1. *Constraint (10) is a convex underestimator of the inequality $x_j x_k \leq \sigma_{jk}$ if $\alpha > 0$, and it is a concave overestimator of $x_j x_k \geq \sigma_{jk}$ if $\alpha < 0$. Further, if $x_j - \alpha x_k = U^{jk}$ or $x_j - \alpha x_k = L^{jk}$ is a facet-defining inequality of $\mathcal{P}^{jk}$ then constraint (10) is tight on the facet.*

The quadratic function strengthens the McCormick bounds (8). Assume for example the feasible region $\mathcal{P}^{jk}$ is of the form

$$0 \leq x_j \leq 3, \quad 0 \leq x_k \leq 3, \qquad -1 \leq x_k - x_j \leq 1$$

At the point $(x_j, x_k) = (1, 2)$, the McCormick bounds only constrain $\sigma_{jk} \geq 0$. Conversely, the quadratic function bound with $\alpha = 1$ restricts $\sigma_{jk} \geq 2$, exactly agreeing with the function

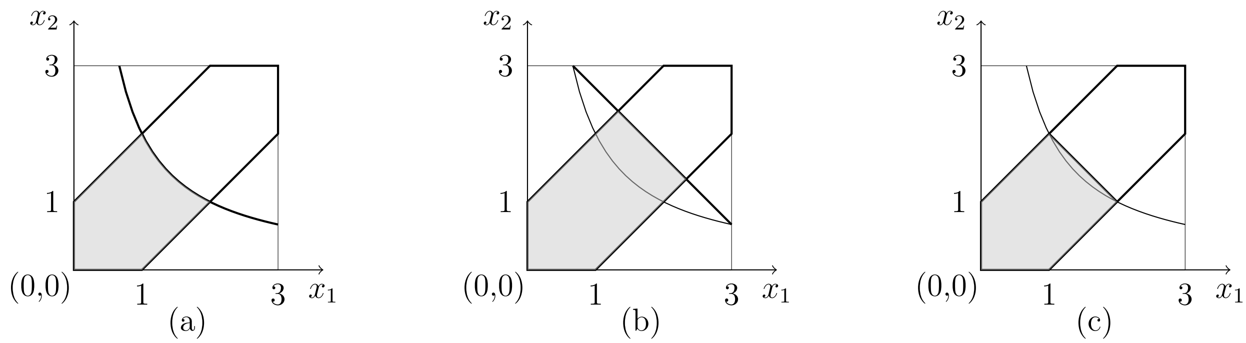

Figure 2: (a) Feasible region. (b) Relaxation with the McCormick inequalities. (c) Relaxation with constraint (10).

value. In this example, the constraint (11) characterizes the lower envelope of Ξ^{jk} (see Proposition 1 below). The lower bounding constraints on σ_{jk} are

$$\begin{aligned} \sigma_{jk} &\geq 0 && (14)\\ \sigma_{jk} &\geq 3x_k + 3x_j - 9 && (15)\\ 4\sigma_{jk} &\geq (x_j + x_k)^2 - 1 && (16) \end{aligned}$$

with (14) and (15) specializations of (8), and (16) coming from (10) with $\alpha = 1$. Knowing the lower envelope of Ξ^{jk} enables the solution of the quadratically constrained problem

$$\begin{array}{lrcrcll} \min_{x_1,x_2} & -x_1 & - & x_2 & & & \\ \text{subject to} & & & x_1x_2 & \leq & 2 & \\ & x_1 & - & x_2 & \leq & 1 & \\ & -x_1 & + & x_2 & \leq & 1 & \\ & 0 & \leq & x_i & \leq & 3 & i = 1,2 \end{array}$$

with feasible region shaded in Figure 2(a). Omitting the quadratic constraint gives a relaxation with optimal value -6, achieved at $(3, 3)$. Adding the constraints (14) and (15) gives the linear program

$$\begin{array}{lrcrcll} \min_{x_1,x_2} & -x_1 & - & x_2 & & & \\ \text{subject to} & & & \sigma_{12} & \leq & 2 & \\ & & & \sigma_{12} & \geq & 0 & \\ & 3x_1 & + & 3x_2 & \leq & \sigma_{12} + 9 & \\ & 3x_1 & - & \sigma_{12} & \geq & 0 & \\ & -\sigma_{12} & + & 3x_2 & \geq & 0 & \\ & x_1 & - & x_2 & \leq & 1 & \\ & -x_1 & + & x_2 & \leq & 1 & \\ & 0 & \leq & x_i & \leq & 3 & i = 1,2 \end{array}$$

whose feasible region is shaded in Figure 2(b). This LP has optimal value $-3\frac{2}{3}$, achieved at $x = (\frac{11}{6}, \frac{11}{6})$, $\sigma_{12} = 2$. Adding the further constraint (16) then gives the feasible region shaded in Figure 2(c) with optimal value -3, achieved at the extreme points $x = (1, 2)$ and $x = (2, 1)$ of the relaxation, with $\sigma_{12} = 2$, which are optimal in the original problem.

Proposition 1. *Assume $x_j^U - x_j^L = x_k^U - x_k^L$ and $\mathcal{P}^{jk}$ has six extreme points, at (x_j^L, x_k^L), $(x_j^L + \gamma, x_k^L)$, $(x_j^U, x_k^U - \gamma)$, (x_j^U, x_k^U), $(x_j^U - \gamma, x_k^U)$, and $(x_j^L, x_k^L + \gamma)$. Then the lower convex underestimator of σ_{jk} over $\mathcal{P}^{jk}$ is given by (8) together with (10) with $\alpha = 1$.*

Proof. Note that by the definition of the bounds x_j^L, x_j^U, x_k^L, and x_k^U, we must have that γ is a positive constant smaller than $x_j^U - x_j^L$. Any point $\bar{x}$ in $\mathcal{P}^{jk}$ is a convex combination of two points on the boundary of $\mathcal{P}^{jk}$ of the form $\bar{x}^1 := \bar{x} + \theta(1, -1)$ and $\bar{x}^2 := \bar{x} - \nu(1, -1)$ for some $\theta, \nu \geq 0$. At each of these boundary points $\bar{x}^1$ and $\bar{x}^2$, the lower bound on σ_{jk} given by the constraints (8) and (10) is equal to $x_j x_k$, so the convex underestimator is tight. Further, the same constraint is active at both $\bar{x}^1$ and $\bar{x}^2$ and it is linear on the line segment between the two points, so the lower convex underestimator of σ_{jk} at $\bar{x}$ cannot be larger than that given by the constraint that is active at $\bar{x}^1$ and $\bar{x}^2$. □

The following proposition gives a concave overestimator of σ_{jk} for a particular $\mathcal{P}^{jk}$. The proof is similar to that for Proposition 1 and is therefore omitted.

Proposition 2. *Assume $\mathcal{P}^{jk} = \{(x_j, x_k) : x_j^L \leq x_j \leq x_j^U, x_k^L \leq x_k \leq x_k^U, x_j^L + x_k^L + \gamma \leq x_j + x_k \leq x_j^U + x_k^U - \gamma\}$ for some positive constant $\gamma < 0.5(x_j^U + x_k^U - x_j^L - x_k^L)$ Then the upper concave overestimator of σ_{jk} over $\mathcal{P}^{jk}$ is given by (8) together with (10) with $\alpha = -1$.*

Propositions 1 and 2 exploit the linearity of (10) along line segments orthogonal to the line $x_j - \alpha x_k = \beta$ for some constant β, when $\alpha = \pm 1$. If $\alpha \neq \pm 1$ then this property no longer holds, so constraint (10) will not in general be a tight underestimator or overestimator of the convex hull of Ξ^{jk} throughout the line segment.

Nonetheless, Proposition 1 can be generalized to regions $\mathcal{P}^{jk}$ of the form

$$x_j^L \leq x_j \leq x_j^U, \; x_k^L \leq x_k \leq x_k^U, \; L^{jk} \leq x_j - \alpha x_k \leq U^{jk}$$

when $\alpha > 0$, $x_j^U - x_j^L = \alpha(x_k^U - x_k^L)$, and $L^{jk} + U^{jk} = x_j^L + x_j^U - \alpha(x_k^L + x_k^U)$. In this case, each point in $\mathcal{P}^{jk}$ with $x_j - \alpha x_k = L^{jk}$ can be joined to a point in $\mathcal{P}^{jk}$ with $x_j - \alpha x_k = U^{jk}$ along a direction where $x_j + \alpha x_k$ is constant, and conversely. Now, (10) is linear along such line segments, so it provides the tightest possible convex underestimator of the the bilinear form. Hence we obtain the tightest convex underestimator over the whole of $\mathcal{P}^{jk}$, using an argument similar to the earlier proposition. This is illustrated in Figure 3. Similarly, Proposition 2 can be generalized to regions $\mathcal{P}^{jk}$ of the form

$$x_j^L \leq x_j \leq x_j^U, \; x_k^L \leq x_k \leq x_k^U, \; L^{jk} \leq x_j - \alpha x_k \leq U^{jk}$$

when $\alpha < 0$, $x_j^U - x_j^L = -\alpha(x_k^U - x_k^L)$, and $L^{jk} + U^{jk} = x_j^L + x_j^U - \alpha(x_k^L + x_k^U)$. We summarize this in the following proposition.

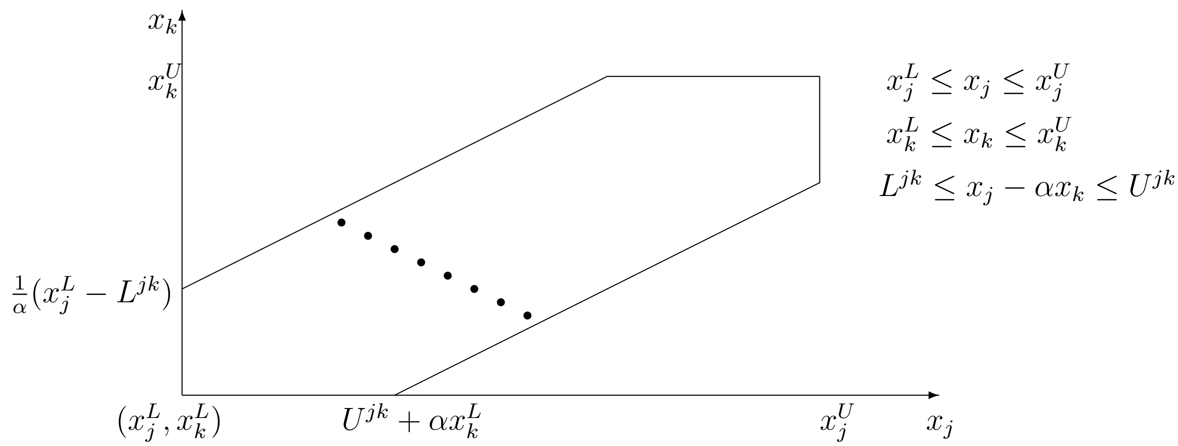

Figure 3: The feasible region $\mathcal{P}^{jk}$. Constraint (10) gives a tight linear underestimator of the bilinear form along the dotted line $x_j + \alpha x_k = \beta$ for some β.

Proposition 3. *Let $\mathcal{P}^{jk}$ have the form*

$$x_j^L \le x_j \le x_j^U, \; x_k^L \le x_k \le x_k^U, \; L^{jk} \le x_j - \bar{\alpha} x_k \le U^{jk}$$

1. *If $\bar{\alpha} > 0$, $x_j^U - x_j^L = \bar{\alpha}(x_k^U - x_k^L)$, and $L^{jk} + U^{jk} = x_j^L + x_j^U - \bar{\alpha}(x_k^L + x_k^U)$ then the lower convex underestimator of σ_{jk} over $\mathcal{P}^{jk}$ is given by (8) together with (10) with $\alpha = \bar{\alpha}$.*
2. *If $\bar{\alpha} < 0$, $x_j^U - x_j^L = -\bar{\alpha}(x_k^U - x_k^L)$, and $L^{jk} + U^{jk} = x_j^L + x_j^U - \bar{\alpha}(x_k^L + x_k^U)$ then the upper concave overestimator of σ_{jk} over $\mathcal{P}^{jk}$ is given by (8) together with (10) with $\alpha = \bar{\alpha}$.*

If $\mathcal{P}^{jk}$ is diamond-shaped then combining the two parts of this proposition shows that inequality (10) gives the convex envelope of the function $x_j x_k$. In particular, we have the following proposition.

Proposition 4. *Let $\mathcal{P}^{jk}$ have the form*

$$L_1^{jk} \le x_j - \bar{\alpha} x_k \le U_1^{jk}, \; L_2^{jk} \le x_j + \bar{\alpha} x_k \le U_2^{jk}$$

for some $\bar{\alpha} > 0$. If $\mathcal{P}^{jk}$ is nonempty then the convex envelope of Ξ^{jk} is given by (10) with $\alpha = \pm\bar{\alpha}$.

3 Choosing constraints

Every positive value of α leads to a convex lower bound (10) on σ_{jk} for $x \in \mathcal{P}^{jk}$, and each negative value leads to a concave upper bound. In this section, we discuss methods for determining which values of α to use. First note that the facets of $\mathcal{P}^{jk}$ lead to nondominated bounds, from Corollary 1. If $x_j - \bar{\alpha} x_k = \beta$ is a facet for some choice of β then the corresponding inequality (10) provides a tight bound throughout the facet, and no other value of α gives a constraint that is tight at any interior point of the facet.

Values of α that do not correspond to facets of $\mathcal{P}^{jk}$ may also lead to nondominated inequalities. In what follows, we assume $\mathcal{P}^{jk}$ is a polyhedron. Each extreme point of $\mathcal{P}^{jk}$ is either $\arg\min\{x_j - \alpha x_k : (x_j, x_k) \in \mathcal{P}^{jk}\}$ or $\arg\max\{x_j - \alpha x_k : (x_j, x_k) \in \mathcal{P}^{jk}\}$ for an interval of values of α of positive length. Let (α_1, α_2) be an open interval not containing the origin such that there are two extreme points x^1 and x^2 of $\mathcal{P}^{jk}$ with $x^1 = \arg\min\{x_j - \alpha x_k : (x_j, x_k) \in \mathcal{P}^{jk}\}$ and $x^2 = \arg\max\{x_j - \alpha x_k : (x_j, x_k) \in \mathcal{P}^{jk}\}$ for each α in the interval. It follows that for each such α, we have $\alpha^L = x_j^1 - \alpha x_k^1$ and $\alpha^U = x_j^2 - \alpha x_k^2$, so from (11) and (12) we obtain the following bound on σ_{jk}:

$$
\begin{aligned}
\frac{1}{4\alpha} f_\alpha(x_j, x_k) &= \frac{1}{4\alpha}((x_j + \alpha x_k)^2 - (x_j^1 - \alpha x_k^1 + x_j^2 - \alpha x_k^2)(x_j - \alpha x_k) \\
&\qquad + (x_j^1 - \alpha x_k^1)(x_j^2 - \alpha x_k^2)) \\
&= \frac{1}{4}\alpha(x_k - x_k^1)(x_k - x_k^2) + \frac{1}{4\alpha}(x_j - x_j^1)(x_j - x_j^2) \\
&\qquad + \frac{1}{4}(x_j(x_k + x_k^1 + x_k^2) + x_k(x_j + x_j^1 + x_j^2) - x_j^1 x_k^1 - x_j^2 x_k^2).
\end{aligned}
$$

This is a lower bound on σ_{jk} if $\alpha > 0$ and an upper bound if $\alpha < 0$. We want to determine the value of α in the interval that gives the best bound at a given point (x_j, x_k). The only possible stationary point in the interval is

$$
\bar{\alpha} = \sqrt{\frac{(x_j - x_j^1)(x_j - x_j^2)}{(x_k - x_k^1)(x_k - x_k^2)}} \tag{17}
$$

where the sign of the square root coincides with the sign of the interval. For the square root to exist and be in the interval, the numerator and denominator of the fraction must be of the same sign and nonzero. We argue using second order conditions that they must both be negative if $\bar{\alpha}$ is to give the best bound in the interval (α_1, α_2). Consider first the case that the interval is contained in the positive half-line, so we are looking for the greatest lower bound. In this case we want the second derivative to be negative, that is

$$
\frac{(x_j - x_j^1)(x_j - x_j^2)}{\bar{\alpha}^3} < 0.
$$

Since $\bar{\alpha} > 0$, we obtain the conditions that

$$
(x_j - x_j^1)(x_j - x_j^2) < 0 \text{ and } (x_k - x_k^1)(x_k - x_k^2) < 0 \tag{18}
$$

for $\bar{\alpha}$ to exist and be a minimizer. If $\alpha_1 < \bar{\alpha} < \alpha_2$ then $\bar{\alpha}$ gives the best lower bound at the point (x_j, x_k) over all constraints of the form (10).

When the interval is contained in the negative half-line, the sufficient condition for $\bar{\alpha}$ to give the least upper bound is that the second derivative be positive. This gives again the requirement (18).

There may be a continuum of values of α that lead to nondominated inequalities. In the construction above, the value of $\bar{\alpha}$ is determined by the point (x_j, x_k). If the point is changed slightly then $\bar{\alpha}$ would be changed slightly, and still be in the interval. This observation implies that such $\bar{\alpha}$ must occur in intervals of positive length, as we summarize in the following proposition.

Proposition 5. *Let $\bar{\alpha} \neq 0$. If $x_j - \bar{\alpha} x_k = \beta$ is not a facet of $\mathcal{P}^{jk}$ for any β and if there is a point $(\bar{x}_j, \bar{x}_k) \in \mathcal{P}^{jk}$ where (10) using $\bar{\alpha}$ uniquely provides the best bound on σ_{jk} over all inequalities of the form (8) or (10) then there is an interval $(\alpha^1, \alpha^2) \ni \bar{\alpha}$ such that each α in the interval gives an inequality (10) that uniquely realizes the best bound on σ_{jk} for some point $(x_j, x_k) \in \mathcal{P}^{jk}$ over all inequalities of the form (8) or (10).*

We now give an example where a non-facet defining α gives the best bound.

Example 1. *Take $\mathcal{P}^{jk}$ to be given by $0 \leq x_j \leq 6$, $0 \leq x_k \leq 3$, $x_j - 6x_k \geq -12$. Taking $\bar{\alpha} = 2\sqrt{2}$ gives the greatest lower bound of $2.5 - \sqrt{2}$ on σ_{jk} at $\bar{x} = (4, 1)$. The best lower bound from (8) at this point is 0. The lower bound from (10) with $\alpha = 6$ is $2/3$, corresponding to the facet defining inequality.*

A parametric pivoting approach for finding the facets and extreme points can be used when $\mathcal{P}^{jk}$ is a polyhedron. The finite set of facets can be used to define an initial set of constraints (10). Additional constraints of this form can be added in a cutting plane or cutting surface framework. In particular, if the solution to the current relaxation of (2) violates (5) for some triple (x_j, x_k, σ_{jk}) then equation (17) can be used to obtain a better bound on σ_{jk}. The use of the formula requires determination of the points x^1 and x^2, which can be obtained by examining pairs of the extreme points of $\mathcal{P}^{jk}$ found during the parametric pivoting approach. Heuristic methods could be used to speed up the search for an appropriate pair of points.

4 Second order cone constraints and triangles

In the following example, the constraints (8) and (10) do not suffice to give the lower convex underestimator of σ_{jk} on $\mathcal{P}^{jk}$. Assume the feasible region $\mathcal{P}^{jk}$ is

$$0 \leq x_j \leq 5, \quad 0 \leq x_k \leq 6, \qquad x_k - x_j \leq 1$$

as illustrated in Figure 4. The maximum value of $x_j - x_k$ is achieved at $(5, 0)$. Equation (10) with $\alpha = 1$ holds at equality with $\sigma_{jk} = x_j x_k$ on the line $x_j - x_k = 5$ through this point. The constraint underestimates the convex hull of σ_{jk} at points in the interior of $\mathcal{P}^{jk}$ where $x_j + x_k \neq 5$ and $x_j - x_k < 5$. In order to rectify this, a constraint can be constructed that is linear between any feasible point on the line $x_j - x_k = -1$ and the point $(5, 0)$. Linderoth [18] showed that this constraint can be represented as a rotated second order cone constraint, which gives the best possible convex underestimator on the triangle with vertices at $(-1, 0)$, $(5, 0)$, and $(5, 6)$.

Given a facet of $\mathcal{P}^{jk}$, the polyhedron $\mathcal{P}^{jk}$ can be embedded in a right-angled triangle, with the facet lying on the hypotenuse of the triangle. A valid second order cone constraint

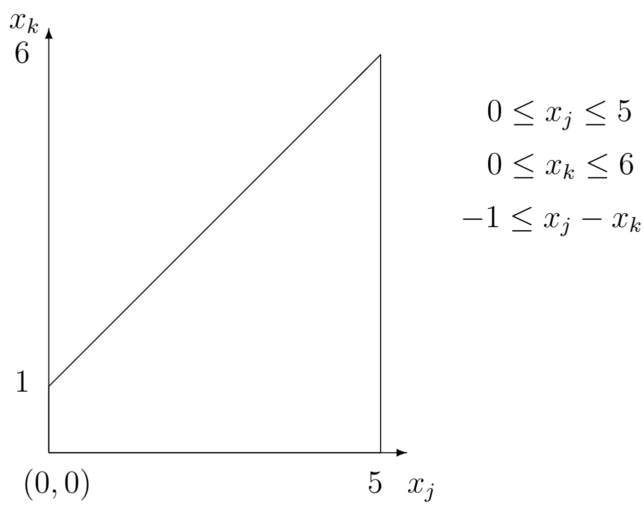

Figure 4: The feasible region of a nonsymmetric example in (x_j, x_k)-space.

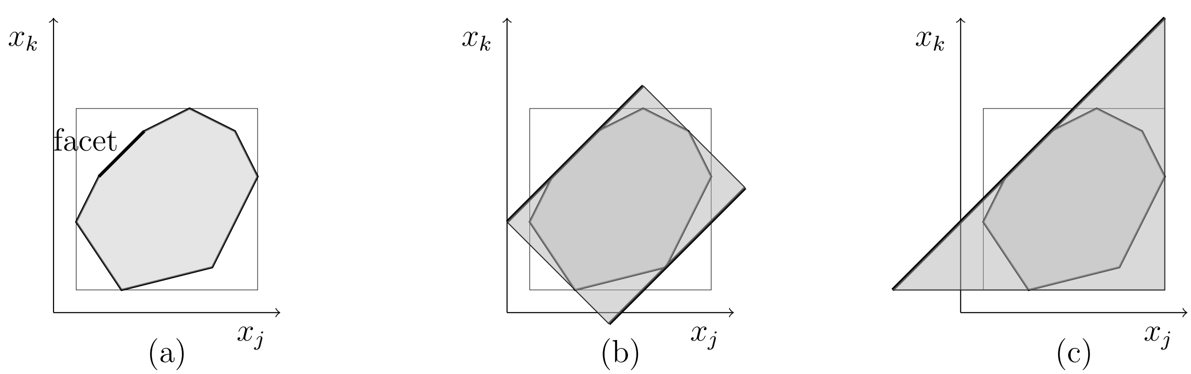

Figure 5: Illustration of (10) and the constraint from [18] derived from one facet of $\mathcal{P}^{jk}$. (a) Feasible region $\mathcal{P}^{jk}$. (b) Constraint (10) gives the best convex underestimator that is valid throughout the shaded rectangle. (c) The second order cone constraint from [18] gives the best convex underestimator that is valid throughout the shaded triangle.

can then be constructed from the results in [18]. This construction is illustrated in Figure 5. Both the constraint on the triangle and the constraint (10) are constructed so that they are valid throughout a particular polyhedron that typically strictly contains $\mathcal{P}^{jk}$, with the result that in general they do not give the best convex underestimator of the function $x_j x_k$ on $\mathcal{P}^{jk}$. In the general case, neither inequality dominates the other. In the case of Example 1, the lower bound from the rotated second order cone constraint at the point $(4, 1)$ is 1.5.

5 Using diagonalization

In this section, we present an alternative approach to obtain convex quadratic relaxations of nonconvex quadratic constraints and relate this approach to the constraints (10). The approach is to look at a change of variables, under which the quadratic constraints in (2) can be replaced by quadratic constraints having no bilinear terms.

Each of the symmetric matrices Q^i could be diagonalized, so $Q^i = V^i D^i {V^i}^T$ where D^i is a diagonal matrix and V^i is an orthogonal matrix. Introducing variables $x^i = {V^i}^T x$, the general quadratic constraint in (2) is equivalent to the constraint

$$c_i^T x + \frac{1}{2} \sum_{D^i_{jj} > 0, j=1}^{n} D^i_{jj} (x^i_j)^2 \leq g_i + \frac{1}{2} \sum_{D^i_{jj} < 0, j=1}^{n} |D^i_{jj}| (x^i_j)^2$$

where the right hand side can be upper-bounded using a linear function of the form (7), resulting in a convex quadratic constraint.

It is interesting to compare this approach to the approach based on (10). The diagonalization approach introduces an extra pn variables in the form of the x^i terms. The bilinear term approach introduces an extra $n(n-1)/2$ variables σ_{jk}. The bilinear term approach can exploit sparsity: if $Q^i_{jk} = 0$ for all i then there is no need to introduce σ_{jk}. The extra linear constraints $x^i = {V^i}^T x$ in the diagonalization approach will typically be dense. The strength of the diagonalization approach depends on the strength of the bounds on the x^i_j terms where $D^i_{jj} < 0$; such bounds can be calculated using $\mathcal{P}$.

For example, for a single quadratic constraint of the form $x_j x_k \leq b$, the diagonalization approach would result in the following:

$$\begin{aligned}
b &\geq x_j x_k \\
&= \frac{1}{2} \begin{bmatrix} x_j & x_k \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_j \\ x_k \end{bmatrix} \\
&= \frac{1}{4} \begin{bmatrix} x_j & x_k \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_j \\ x_k \end{bmatrix} \\
&= \frac{1}{4} \begin{bmatrix} x_j & x_k \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} x_j \\ x_k \end{bmatrix} - \frac{1}{4} \begin{bmatrix} x_j & x_k \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} x_j \\ x_k \end{bmatrix} \\
&= \frac{1}{4}(x_j + x_k)^2 - \frac{1}{4}(x_j - x_k)^2
\end{aligned}$$

which is equivalent to (10) with $\alpha = 1$. Thus, in this case the diagonalization approach will exploit only upper and lower bounds on $x_j - x_k$, not bounds using more general α. More generally, the diagonalization approach exploits upper and lower bounds on the specific linear combinations of the original variables that correspond to eigenvectors of Q^i with negative eigenvalues, and is unable to exploit more general linear combinations of the variables.

6 Computational results

Computational tests were conducted on three classes of problems. For these tests, a feasible solution was found which provides an upper bound $\bar{g}_0$ on the optimal value, and then a relaxation consisting of the constraints $Ax \geq b$ together with (4), (7), and (8) was solved to obtain a lower bound. The lower bound relaxation was then tightened in two ways: by tightening the bounds x^L and x^U used in (8) and by adding constraints of the form (10). The bounds used in these tightenings were obtained by solving linear programs or second order cone programs which include the constraint $c_0^T x \leq \bar{g}_0$. We report the proportion of the gap between the upper and lower bounds that was closed using these two tightenings.

CPLEX 11.0 was used to solve the second order cone programs. When possible, the student version of KNITRO was used to find a feasible solution to (2). Default settings were used for the solvers. The tests in §6.1 and §6.2 on the quality of the relaxations were run on one core of an Apple Mac Pro with a 2x2.8 GHz Quad-Core Intel Xeon processor, using AMPL to call the solvers. The global optimization tests in §6.1 were performed using the NEOS server [12]. The tests in §6.3 were run on a single core of an AMD Phenom II X4 955@3.2GHZ with 4GB memory, using C++ with callable CPLEX. All times are reported in seconds.

6.1 Diamond problems

The first class of problems are quadratically constrained quadratic programs, with $0 \leq x_i \leq 1$, and with linear constraints $-0.5 \leq x_i - x_j \leq 0.5$ and $0.5 \leq x_i + x_j \leq 1.5$ for each pair of variables $1 \leq i < j \leq n$. All entries in $c_i, i = 0, \ldots, p$ are uniformly generated between -1 and 1. All entries above the leading diagonal in $Q^i, i = 1, \ldots, p$ are uniformly generated between -0.5 and 0.5, the leading diagonals are zero, and the entries below the leading diagonal are chosen to make the matrices symmetric. The right hand side parameters g_i are chosen so that $x = 0.5e$ is feasible, with a slack of 0.1 in each quadratic constraint (here e denotes the vector of ones).

6.1.1 Global optimality

Smaller instances were solved to global optimality using the solver BARON [26], using the default parameters, including a tolerance of 10% on the optimality gap. A time limit of 1000 seconds was placed on each run. In these tests, the problems were solved in two ways:

1. The original formulation (2) was submitted.

		Original formulation			Modified formulation				
n	p	# solved	time	BaR	# solved	time	BaR	gapH	gapO
15	5	5	47.9	2503	5	76.8	363	48%	74%
20	1	3	291.4	8708	4	51.1	52	55%	82%
20	5	1	378.9	8227	4	251.0	358	57%	81%

Table 1: Solving diamond problems to optimality. Times are measured in seconds. Columns headed "BaR" give the number of BaR iterations reported by BARON. The column headed "gapH" gives the percentage of the gap between the heuristic solution and the relaxation with McCormick constraints (8) that was closed using the constraints (10). Similarly, the column headed "gapO" gives the percentage of the gap between the upper bound on the optimal value reported by BARON and the relaxation with McCormick constraints (8) that was closed using the constraints (10).

2. Variables σ_{jk} are added for $1 \leq j < k \leq n$, the quadratic constraints are replaced by versions of (4), the linear constraints (8) are used for $1 \leq j < k \leq n$, the convex quadratic constraints (10) are included for $\alpha = \pm 1$ and $1 \leq j < k \leq n$, and the nonlinear equality constraints (5) are explicitly included.

The two formulations gave identical upper and lower bounds on the optimal value, when they were both able to solve the problem. The results can be found in Table 1. Each row represents an average of five instances. The averages are taken over the instances actually solved by the code. BARON solved the modified formulation for 13 of the 15 problems in the 1000 second time limit, whereas BARON was only able to solve 9 of the original formulations. The modified formulation led to greatly reduced runtimes for the larger instances. The number of iterations required by BARON was dramatically reduced through the use of the convex quadratic constraints (10). It should also be noted from the table that the modified formulation is able to close about 80% of the optimality gap. Thus, these constraints are very effective for these problems.

6.1.2 Improvement in the quality of the relaxations

We also examined the quality of the relaxations for larger instances, which were too large to solve to optimality. The upper bound for these problems is given by the point $x = 0.5e$. (The student version of KNITRO was unable to solve larger versions of these problems because they have many linear constraints.) Refining the bounds used in the constraints (8) was not helpful for these problems. The constraints (10) were used with $\alpha = \pm 1$ for each pair of variables. Five problems were run for each of five sizes, with $n = p = 25, 50, 75, 100, 125$, and each entry in Table 2 is an average over the five problems of that size. It is notable that the error in the objective function value is reduced by approximately 70%, with the results improving as the problem instances get larger. This is the gap compared to the initial solution $0.5e$ (labelled "gapH" in Table 1), so the gap with respect to the optimal solution ("gapO" in Table 1) would be reduced by a greater amount. Further, the error in (5) is

size	%gap closed	relative error	LB time	SOCP time	%gap closed tri
25	61.9%	0.195	0.09	0.47	45.3%
50	69.6%	0.210	1.60	5.01	49.5%
75	70.7%	0.213	12.12	23.30	49.9%
100	72.3%	0.226	97.40	72.14	50.0%
125	72.8%	0.229	768.17	199.08	50.0%

Table 2: Computational results for QCQPs with diamond constraints for larger instances. The columns give the proportion of the duality gap closed, the value of $\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}|\sigma_{jk} - x_j x_k|$ after adding (10) relative to this quantity before adding these constraints, the time required to solve the initial relaxation, the time required to solve the final relaxation, and the percentage of the initial gap closed using the inequalities in [18].

reduced by at least 77%, as can be seen from the third column. Even the relative runtime for the SOCP relaxation with (10) improves as the size of the problems grows (although this may be due to the difference in tolerances in CPLEX for linear programs as opposed to second order cone programs, and that CPLEX uses simplex for the LPs and an interior point method for the quadratically constrained problems). The constraints (10) give the convex envelope of $x_j x_k$ for these problems (see Proposition 4), and it can be seen that these constraints dramatically outperform the triangle-based constraints from [18] for these problems.

6.2 Random QCQP's

The second class of problems constrain $0 \leq x_i \leq 1$ for $i = 1, \ldots, n$. The entries in c_i and Q^i were generated as in §6.1. The right hand side coefficients g_i were uniformly distributed between 1 and 100, as in [5]. The entries in the $m \times n$ matrix A were uniformly generated between 0 and 1, and the entries in b uniformly between 1 and 10. The origin is feasible in these problems, and we used KNITRO to find a better feasible solution and an upper bound. The initial lower bound is the optimal value of the convex QCQP using the McCormick linear constraints without refining the bounds.

Refining the bounds used in (8) was somewhat helpful for these problems. We only refined the bounds for variables that were at least 0.02 from their boundaries; this weakened the lower bound on the optimal value slightly, but greatly reduced the computational time. We generated second order cone constraints (10) for five choices of α, namely, $\pm 1, \pm 3$ and $\frac{1}{3}$. It is expensive to find the upper and lower bounds α^U and α^L for all of the pairs of variables, so we only generated the constraints for the pairs where (5) was most violated in the solution to the relaxation with the refined bounds on the variables. This limited constraint generation appeared to be about as effective as generating the constraints for all pairs. For these problems, the constraints from [18] were about as effective as (10), and combining the two sets of constraints is slightly better than either set alone; because of the small difference, these results are omitted. The computational results are contained in

n	p	m	%gap closed		relative error		(8) time	(10) time
			(8)	(8)+(10)	(8)	(8)+(10)		
20	20	80	63	71	0.44	0.16	0.13	0.96
20	50	160	68	77	0.55	0.26	0.13	0.95
40	20	80	53	73	0.49	0.23	0.48	7.72
40	50	160	61	78	0.66	0.30	0.43	10.4
60	20	80	36	55	0.64	0.42	2.48	77.8
60	50	160	26	41	1.55	1.15	1.27	57.5

Table 3: Computational results for QCQPs with approximately 30% of entries in each Q^i being nonzero. The columns give the proportion of the duality gap closed and the value of $\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}|\sigma_{jk} - x_j x_k|$ after refining the bounds used in the McCormick constraints and after adding the convex quadratic constraints (10), relative to this quantity before refining the bounds used in the McCormick constraints, the time required to find the refined bound from (8), and the time required to find the bounds for (10) and solve the resulting second order cone program.

Table 3.

It is clear that refining the bounds used for the McCormick constraints is helpful, and that the constraints (10) can help considerably with closing the remaining gap. Constraint (10) also succeeds in decreasing the relative error in $\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}|\sigma_{jk} - x_j x_k|$ when compared to the value after using the refined McCormick constraints. Refining the McCormick constraints does not always help this measure, possibly because of some outlier coordinates.

6.3 Linear programs with complementarity constraints

The third test set of problems arise from linear programs with linear complementarity constraints. A standard formulation for such a problem has variables $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, and is the following:

$$\begin{array}{lll} \underset{(x,y)}{\text{minimize}} & c^T x + d^T y & \\ \text{subject to} & Ax + By \geq b & \\ \text{and} & 0 \leq y \perp q + Nx + My \geq 0, & \end{array} \tag{19}$$

in which $N \in \mathbb{R}^{m\times n}$, $M \in \mathbb{R}^{m\times m}$, $q \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, $d \in \mathbb{R}^m$, $b \in \mathbb{R}^k$, $A \in \mathbb{R}^{k\times n}$ and $B \in \mathbb{R}^{k\times m}$. This is equivalent to the quadratically constrained quadratic program

$$\begin{array}{ll} \underset{(x,y)}{\text{minimize}} & c^T x + d^T y \\ \text{subject to} & Ax + By \geq b \\ & 0 \leq y,\ q + Nx + My \geq 0, \\ & y^T(q + Nx + My) \leq 0. \end{array} \tag{20}$$

Note that this QCQP has just a single quadratic constraint, which can be expressed equivalently as

$$q^T y + \tilde{y}^T x + \frac{1}{2} y^T (M + M^T) y \leq 0, \quad \tilde{y} = N^T y.$$

In the problems we generated, $M + M^T$ is positive semidefinite and $n = 2$, so there are only two bilinear terms. Problems of this type arise, for example, in parameter selection with support vector regression [16]. Burer [9] has shown that problems of this type are equivalent to convex optimization problems over the cone of completely positive matrices, provided y and $q + Nx + My$ are bounded.

The parameters in (19) were generated as follows. The variables x and y are constrained to be nonnegative. The entries in c and d are uniformly distributed integers between 0 and 9, which ensures the problem is not unbounded. The entries in A, B, and N are uniformly generated integers between -5 and 5, with a proportion of the entries zeroed out. The matrix $\frac{1}{2}(M + M^T)$ is set equal to LL^T where L is an $m \times r$ matrix whose entries are uniformly generated integers between -5 and 5, with a proportion of the entries zeroed out. This construction ensures that $\frac{1}{2}(M + M^T)$ is positive semidefinite. The matrix M is then obtained from $\frac{1}{2}(M + M^T)$ by adjusting the nonzero off-diagonal entries by a uniformly distributed random integer between -2 and 2. To ensure feasibility of (19), a solution $\bar{x}$, $\bar{y}$ is generated. The entries in $\bar{x}$ are integers uniformly distributed between 0 and 9. Two thirds of the entries in $\bar{y}$ are set equal to zero, and the remainder are integers uniformly distributed between 0 and 9. The entries in the right hand side b are chosen so that each slack with the generated solution is an integer uniformly distributed between 1 and 10. The third of the entries in q corresponding to the positive components of $\bar{y}$ are chosen so that complementarity is satisfied. Another third of the entries in q are chosen so that the corresponding components of $q + N\bar{x} + M\bar{y}$ are zero. The final third of the entries of q are chosen so that the corresponding slack in $q + N\bar{x} + M\bar{y} \geq 0$ is an integer uniformly distributed between 1 and 10. The construction is designed so that it is unlikely that the generated solution $\bar{x}$, $\bar{y}$ is optimal.

KNITRO was used to find a feasible solution to (20). The LP relaxation of (19) was solved to give an initial lower bound. This bound was improved in three stages:

1. Use the constraints (8).

2. Refine the bounds on x and $\tilde{y}$ used in (8).

3. Add constraints of the form (10) for the same five choices of α as in §6.2, namely ± 1, ± 3, and $\frac{1}{3}$.

Refining the bounds on x and $\tilde{y}$ was very effective for this class of problems, so we repeatedly refined, for a total of 8 refinements.

Computational results are contained in Table 4. Problems with $m = 100$, 150, and 200 complementarities were solved. The matrices A, B, N, and L were either 20% or 70% dense. The rank r of L was either 30 or 60 for $m = 100$, either 30 or 100 for $m = 150$, and either 30 or 120 for $m = 200$. Five problems were solved for each choice of m, sparsity, and rank, leading

	% gap closed	% time (secs)			sufficient
		100	150	200	
McCormick	47.0	0.2	0.5	1.2	0
+ 8 refines	88.2	9.7	29.4	67.6	6
+ quadratic	88.4	10.0	30.3	69.3	8

Table 4: Computational results for linear programs with complementarity constraints

to a total of 60 problems. For one problem with $m = 150$ and four problems with $m = 200$, CPLEX reported that the matrix $M + M^T$ was not positive semidefinite, due to numerical errors, and so these problems were not solved. The results did not appear to be greatly affected by the choice of sparsity or rank, so the results are aggregated in the table, and each entry in the "time" columns represents a mean of 20, 19, or 16 instances. The effectiveness of the approach also did not appear to be greatly affected by the problem size m, so the "% gap closed" column reports the average value over all 55 solved instances. We were able to determine the optimal solution for these problems using a branch-and-cut approach, so the "gap" is the gap between the optimal value and the value of the LP relaxation.

For these problems, refining the McCormick bounds is very powerful, with constraint (10) providing some additional help. As shown in the last column of the table, this approach is able to determine the optimal solution for 8 of the 60 test problems. The nonconvex quadratic constraint $y^T w \leq 0$ is a powerful modeling tool for these problems. Even with just the McCormick bounds, we are able to close 47% of the gap; refining the bounds and using constraint (10) closes most of the remaining gap.

7 Conclusions

Convex quadratic relaxations of nonconvex quadratic constraints can be powerful computationally, often closing much of the gap between a lower and upper bound on the optimal value of a nonconvex quadratically constrained quadratic program. They allow the solution to global optimality of some instances that cannot be solved directly within a time limit. For certain classes of problems, the constraints introduced in this paper give the convex envelope of a bilinear function. Care is still needed with the use of these constraints, because general codes for convex quadratic programming problems are not quite as reliable and highly developed as codes for linear programming and these problems may not satisfy constraint qualifications, so it would perhaps be necessary to build in a tolerance before employing these constraints in a branch-and-bound code to find an optimal solution.

Acknowledgements

The work of the first author 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. The work of the second author was supported by the Air Force Office of Sponsored Research under grant FA9550-08-1-0061 and by the National Science Foundation under grant CMMI-0969600.

References

[1] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas. Global optimization of MINLP problems in process synthesis and design. *Computers and Chemical Engineering*, 21:S445–S450, 1997.

[2] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas. A global optimization method, αBB, for general twice-differentiable constrained NLPs — I. Theoretical advances. *Computers and Chemical Engineering*, 22(9):1137–1158, 1998.

[3] F. A. Al-Khayyal and J. E. Falk. Jointly constrained biconvex programming. *Mathematics of Operations Research*, 8(2):273–286, 1983.

[4] E. Balas. Intersection cuts — a new type of cutting planes for integer programming. *Operations Research*, 19:19–39, 1971.

[5] X. Bao, N. V. Sahinidis, and M. Tawarmalani. Multiterm polyhedral relaxations for nonconvex, quadratically-constrained quadratic programs. *Optimization Methods and Software*, 24(4–5):485–504, 2009.

[6] X. Bao, N. V. Sahinidis, and M. Tawarmalani. Semidefinite relaxations for quadratically constrained quadratic programs: A review and comparisons. *Mathematical Programming*, online first, 2011.

[7] P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wachter. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software*, 24(4–5):597–634, 2009.

[8] P. Belotti, A. J. Miller, and M. Namazifar. Valid inequalities and convex hulls for multilinear functions. *Electronic Notes in Discrete Mathematics*, 36:805–812, 2010.

[9] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120(2):479–495, 2009.

[10] S. Burer and D. Vandenbussche. A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations. *Mathematical Programming*, 113(2):259–282, 2008.

[11] S. Burer and D. Vandenbussche. Globally solving box-constrained nonconvex quadratic programs with semidefinite-based finite branch-and-bound. *Computational Optimization and Applications*, 43(2):181–195, 2009.

[12] E. D. Dolan, R. Fourer, J. J. Moré, and T. S. Munson. Optimization on the NEOS server. *SIAM News*, 35(6):4, 8–9, July/August 2002.

[13] C. A. Floudas, P. M. Pardalos, C. Adjiman, W. R. Esposito, Z. H. Gümüs, S. T. Harding, J. L. Klepeis, C. A. Meyer, and C. A. Schweiger. *Handbook of test problems in local and global optimization*. Number 33 in Nonconvex Optimization and its Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.

[14] J. Hu, J. E. Mitchell, and J.S. Pang. An LPCC approach to nonconvex quadratic programs. *Mathematical Programming*, online first, 2011.

[15] M. Kojima and L. Tuncel. Cones of matrices and successive convex relaxations of nonconvex sets. *SIAM Journal on Optimization*, 10(3):750–778, 2000.

[16] G. Kunapuli, J. Hu, K. P. Bennett, and J.S. Pang. Classification model selection via bilevel programming. *Optimization Methods and Software*, 23(4):475–489, 2008.

[17] M. Laurent. A comparison of the Sherali-Adams, Lovász-Schrijver and Lasserre relaxations for $0-1$ programming. *Mathematics of Operations Research*, 28(3):470–496, 2003.

[18] J. T. Linderoth. A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Mathematical Programming*, 103(2):251–282, 2005.

[19] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, 1(2):166–190, 1991.

[20] G. P. McCormick. Computability of global solutions to factorable nonconvex programs: part I — convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

[21] J.-P. P. Richard and M. Tawarmalani. Lifted inequalities: a framework for generating strong cuts for nonlinear progams. *Mathematical Programming*, 121(1):61–104, 2010.

[22] N. Sahinidis. BARON: A general purpose global optimization software package. *Journal of Global Optimization*, 8:201–205, 1996.

[23] H. D. Sherali and W. D. Adams. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Number 31 in Nonconvex Optimization and its Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[24] H. D. Sherali and A. R. Alameddine. A new reformulation linearization technique for bilinear programming problems. *Journal of Global Optimization*, 2:379–410, 1992.

[25] M. Tawarmalani and N. Sahinidis. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*. Kluwer, Dordrecht, The Netherlands, 2002.

[26] M. Tawarmalani and N. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103(2):225–249, 2005.

[27] D. Vandenbussche and G. L. Nemhauser. A branch-and-cut algorithm for nonconvex quadratic programs with box constraints. *Mathematical Programming*, 102(3):559–575, 2005.

[28] D. Vandenbussche and G. L. Nemhauser. A polyhedral study of nonconvex quadratic programs with box constraints. *Mathematical Programming*, 102(3):531–557, 2005.