Quadratic factorization heuristics for copositive programming

Immanuel M. Bomze*

Florian Jarre$^\triangle$

Franz Rendl †

* University of Vienna, Austria

$^\triangle$ University of Duesseldorf, Germany

† University of Klagenfurt, Austria

Abstract

Copositive optimization problems are particular conic programs: extremize linear forms over the copositive cone subject to linear constraints. Every quadratic program with linear constraints can be formulated as a copositive program, even if some of the variables are binary. So this is an NP-hard problem class. While most methods try to approximate the copositive cone from within, we propose a method which approximates this cone from outside. This is achieved by passing to the dual problem, where the feasible set is an affine subspace intersected with the cone of completely positive matrices, and this cone is approximated from within. We consider feasible descent directions in the completely positive cone, and regularized strictly convex subproblems. In essence, we replace the intractable completely positive cone with a nonnegative cone, at the cost of a series of nonconvex quadratic subproblems. Proper adjustment of the regularization parameter results in short steps for the nonconvex quadratic programs. This suggests to approximate their solution by standard linearization techniques. Preliminary numerical results on three different classes of test problems are quite promising.

Key words: Combinatorial optimization, copositive programs, clique number

October 14, 2009

1 Copositive Programs

Copositive optimization problems are particular conic programs: extremize linear forms over the copositive cone subject to linear (equality) constraints. To be specific, let $(\mathcal{A}, b, C)$ denote the data, defining an instance of the problem. Here $b \in \mathbb{R}^m$, the m-dimensional Euclidean space; $C \in \mathcal{S}^n := \left\{Y \in \mathbb{R}^{n\times n} : Y^\top = Y\right\}$, the space of all symmetric $n \times n$ matrices; and the linear operator $\mathcal{A}$ maps $\mathcal{S}^n$ to $\mathbb{R}^m$. We refer to the end of this section for a detailed introduction of the notation.

The operator $\mathcal{A}$ is given by m matrices $A^{(i)} \in \mathcal{S}^n$ through $[\mathcal{A}(X)]_i = \langle A^{(i)}, X\rangle, \forall i$. The following primal-dual pair of conic problems are called **copositive programs**:

$$z_C = \sup\left\{b^\top y : S = C - \mathcal{A}^*(y) \in \mathcal{C}\right\}, \tag{1}$$

where $\mathcal{A}^*(y) = \sum_i y_i A^{(i)}$ maps $\mathbb{R}^m$ to $\mathcal{S}^n$, $\mathcal{C}$ is the cone of copositive matrices, and

$$z_{C^*} = \inf\left\{\langle C, X\rangle : \mathcal{A}(X) = b, X \in \mathcal{C}^*\right\} \tag{2}$$

is defined over the cone $\mathcal{C}^*$ of completely positive matrices.

We assume that the matrices $A^{(i)}$ are linearly independent and that both problems satisfy Slater's constraint qualification, which ensures that both optima are attained and

$$z_C = z_{C^*}\,.$$

In this case it is well known that $(X, y, S) \in \mathcal{C}^* \times \mathbb{R}^m \times \mathcal{C}$ is optimal for the pair of problems if, and only if

$$\mathcal{A}(X) = b\,, \quad C - \mathcal{A}^*(y) = S\,, \quad \text{and} \quad \langle S, X\rangle = 0\,.$$

Formally, these problems are very similar to **semidefinite programs**. Just replace the constraint $S \in \mathcal{C}$ by $S \in \mathcal{P}$ and $X \in \mathcal{C}^*$ by $X \in \mathcal{P}$, where $\mathcal{P}$ is the positive semidefinite cone. While semidefinite programs can be solved in polynomial time (to some fixed given precision), it is well known, and will become clear shortly, that copositive optimization is at least as hard as integer programming.

As an illustration of the modeling power of copositive optimization, we recall the following relaxations of the stability number $\alpha(G)$ of a graph G, given through its adjacency matrix A_G.

$$\alpha(G) \le \vartheta'(G) := \max\{\langle J, X\rangle : \langle A_G + I, X\rangle = 1, X \in \mathcal{P} \cap \mathcal{N}\}, \tag{3}$$

$\mathcal{N}$ being the set of elementwise nonnegative matrices, and J the all-ones-matrix. The graph parameter $\vartheta'(G)$ was investigated in [15, 19], inspired by the graph parameter $\vartheta(G)$ introduced by Lovász, see [14]. To see the validity of the inequality, it suffices to observe that a characteristic vector χ of a stable set of size $\alpha(G)$ yields the feasible solution $\frac{1}{\alpha(G)}\chi\chi^\top$ for the problem on the right hand side. It also shows that the cone $\mathcal{P} \cap \mathcal{N}$ could be replaced by $\mathcal{C}^*$, and the inequality would still be valid. In fact, if follows from [2] as well as [8] that there is equality:

$$\alpha(G) = \max\{\langle J, X\rangle : \langle A_G + I, X\rangle = 1, X \in \mathcal{C}^*\}. \tag{4}$$

This gives a first indication that copositive relaxations may have stronger modeling properties than semidefinite relaxations. The result also shows that finding an optimal solution of copositive programs is at least as difficult as integer programming.

There are several further instances of hard combinatorial optimization problems where copositive relaxations give the exact integer optimum. We refer to [11, 18] for further results along these lines. Recently, Burer [7] has shown that in fact every quadratic program with linear constraints can be formulated as a copositive program, even if some of the variables are binary; see also [4].

All this is a strong motivation to investigate copositive optimization problems. This can be done in several ways. Getting inner approximations of the copositive cone $\mathcal{C}$ (i.e., outer approximations of $\mathcal{C}^*$) leads to relaxations of the underlying problem. Using $\mathcal{P}$ instead of $\mathcal{C}^*$ amounts to using semidefinite relaxations, which have a tractable strengthening by the elementwise nonnegative semidefinite matrices:

$$\mathcal{C}^* \subseteq \mathcal{P} \cap \mathcal{N} \subseteq \mathcal{P}.$$

Further refinements of this relaxation scheme are given by Parrilo [17], who introduces a hierarchy of relaxations of increasing computational complexity which approximates $\mathcal{C}^*$ with increasing accuracy. Alternatively, one could try to solve copositive programs directly, accepting exponential running time. Recent work by Bundfuss and Dür [5, 6] provides some first insight for this approach. They provide inner and outer approximations of the copositive cone.

Finally, and now we come to the main purpose of this paper, it would also be interesting to approximate $\mathcal{C}^*$ from the inside. This approach can be viewed as a heuristic procedure for a very general class of NP-hard problems.

1.1 Outline and notation

We close this section with an overview of the paper and some notation used throughout.

First, we consider the relation of feasible descent directions in the completely positive cone with regularized strictly convex subproblems (Section 2). To solve the regularized subproblems we go back to the definition of completely positive matrices: these can be factorized into $VV^\top$ with a rectangular matrix V without negative entries. Now, if we regard V as variable, the resulting regularized subproblem becomes quadratic. By a careful selection of the regularization parameter the solution of the sbuproblem has a small norm. This suggests to approximate the solution by standard linearization techniques. The linearization can be complemented by a fixed point iteration for finding a local solution of the nonconvex subproblems. Putting these steps together results in a quadratic factorization heuristics for the completely positive program, where the most expensive computation at each iteration is the solution of a structured convex quadratic program. Section 3 proposes a method of generating hard instances of copositive programs with known solutions. Preliminary numerical results on problems of various type are reported in Section 4.

Notation: The vector of all ones (of appropriate dimension) will be denoted by e. We also use $J := ee^T$ to represent the all-ones matrix.

$$\mathcal{C}^* = \left\{ VV^\top : V \in \mathbb{R}_+^{n\times k} \text{ for some } k \leq \binom{n+1}{2} \right\}$$

is the cone of completely positive matrices. As is well known, $\mathcal{C}^* \subset \mathcal{S}^n$ is the dual cone of the cone

$$\mathcal{C} = \left\{ Y \in \mathcal{S}^n : \; x^\top Y x \geq 0 \text{ for all } x \in \mathbb{R}^n_+ \right\}$$

of all copositive matrices. This justifies the notation $\mathcal{C}^*$ (and the title of the paper). The comprehensive monograph [1] introduces basic concepts such as the cp-rank of a completely positive matrix. A recent characterization of the interior of the completely positive cone is given in [10].

The scalar product of two matrices C, X is defined as $\langle C, X \rangle = \sum_{i,j} C_{i,j} X_{i,j} =$ trace $(C^\top X)$ and defines the Frobenius norm $\|X\|^2 = \langle X, X \rangle$. We always refer to the Frobenius norm unless explicitly specified otherwise. $\mathcal{P}$ denotes the cone of positive semidefinite matrices, and $\mathcal{N}$ is the cone of elementwise nonnegative matrices. The tensor product of two matrices A and B is given by $A \otimes B = [(a_{ij} B)_{ij}]$.

1.2 Hardness certificates

We recall the following tractable relaxations of our problems.

$$z_{PN} = \sup \left\{ b^\top y : S = C - \mathcal{A}^*(y) \in \mathcal{P} + \mathcal{N} \right\} \quad \text{and} \tag{5}$$

$$z_{PN^*} = \inf \left\{ \langle C, X \rangle : \mathcal{A}(X) = b, X \in \mathcal{P} \cap \mathcal{N} \right\} . \tag{6}$$

Again, we assume strong duality to hold for these problems, so that $z_{PN} = z_{PN^*}$. Note that strict feasibility of (2) implies strict feasibility of (6) since the interior of $\mathcal{C}^*$ is contained in the interiors of $\mathcal{P}$ and $\mathcal{N}$. However, strict feasibility of (5) is not implied by strict feasibility of (1), e.g., for $\mathcal{A} = 0$ and $C \in \mathcal{C} \setminus (\mathcal{P} + \mathcal{N})$.

Since $\mathcal{P} + \mathcal{N} \subseteq \mathcal{C}$, it is clear that under our assumptions we have $z_C \geq z_{PN}$. We are particularly interested in problem instances $(\mathcal{A}, b, C)$ with

$$z_C > z_{PN} \, . \tag{7}$$

This occurs exactly if no optimal solution S of (1) is of the form $S = P + N$ with $P \in \mathcal{P}, N \in \mathcal{N}$. In [12] these copositive matrices are called 'exceptional'. Suppose that (X, y, S) is optimal for the instance given by $(\mathcal{A}, b, C)$. We call any feasible solution X of (6) a **hardness certificate** for the instance $(\mathcal{A}, b, C)$ with optimal solution (X, y, S) if

$$b^\top y > \langle C, X \rangle .$$

In this case we have

$$z_C = b^\top y > \langle C, X \rangle \geq z_{PN},$$

and therefore (7) holds.

2 A feasible descent method

The key idea exploited in this section is to define a sequence of simpler regularized subproblems that define a sequence of iterates converging to a (global) optimal solution of (2).

While the regularized subproblems are simpler in the sense that they define "short" feasible descent steps, they are still not tractable by current solvers. We therefore derive a simple iteration scheme such that the optimal solution of the regularized subproblem is a fixed point.

The regularized subproblems define an outer iteration; for each outer iteration, the fixed point problem defines the inner iterations.

2.1 Outer iteration – working in X space

Let us consider some iteration j and an iterate $X = X^{(j)}$ that satisfies the linear equations and of which we know a factorization $X^{(j)} = VV^\top$ with $V \in \mathcal{N}$. To define a feasible descent step at $X^{(j)}$ we regularize (2) based on some strictly convex squared norm $\|\Delta X\|_j^2$, where the choice of the norm $\| \,.\, \|_j$ depends on the factorization $X^{(j)} = VV^\top$. For $\varepsilon \in (0, 1)$ we obtain the *regularized problem*

$$\begin{array}{ll} \min & \varepsilon\langle C, \Delta X\rangle + (1-\varepsilon)\|\Delta X\|_j^2 \\ \text{s.t.} & \langle A^{(i)}, \Delta X\rangle = 0\,,\; i = 1 : m \\ & X^{(j)} + \Delta X \in \mathcal{C}^* \end{array} \tag{8}$$

which has a strictly convex objective function. The (unique) optimal solution of the convex problem (8) is denoted by ΔX^j. For large $\varepsilon < 1$ the point $X^{(j)}+\Delta X^j$ approaches a solution of the original problem (2).

For a fixed value of $\varepsilon \in (0, 1)$ and a given initial matrix $X^{(0)} \in \mathcal{C}^*$ that satisfies the linear equations we consider the (outer) iteration

$$X^{(j+1)} := X^{(j)} + \Delta X^{(j)}$$

given by (8) where the norm $\| \,.\, \|_j$ changes at each iteration. Computation of $\Delta X^{(j)}$ is the result of some inner iteration to be discussed below. By construction, $X^{(j)}$ is feasible for (2) for all $j \geq 0$, and the objective value $\langle C, X^{(j)}\rangle$ of $X^{(j)}$ is monotonically decreasing as $j \to \infty$. Indeed, as $\Delta X = 0$ is (8)-feasible, the optimal value is not positive, so

$$\varepsilon\langle C, X_{j+1} - X_j\rangle = \varepsilon\langle C, \Delta X_j\rangle \leq \varepsilon\langle C, \Delta X_j\rangle + (1-\varepsilon)\|\Delta X_j\|_j^2 \leq 0\,.$$

Theorem 2.1 *If the norms $\| \,.\, \|_j$ satisfy a global bound,*

$$\exists M < \infty : \qquad \|H\|_j^2 \leq M\|H\|^2 \quad \forall\, H \in \mathcal{S}^n \quad \forall\, j \tag{9}$$

then the following result holds true: Let $\bar X$ be any limit point of the sequence $X^{(j)}$. Then $\bar X$ solves problem (2).

Proof. As $X^{(j)}$ is feasible for all j and the feasible set of (2) is closed, $\bar X$ is feasible for (2) as well. Assume that X^{opt} is an optimal solution of (2) and, by contradiction, that

$$\delta := \langle C, \bar X - X^{opt}\rangle > 0\,.$$

Without loss of generality choose $M \geq \varepsilon\delta$ in (9). Put

$$\rho := \max\{1, \|\bar X - X^{opt}\|\} \quad \text{and} \quad \delta' := \frac{\varepsilon(1+\varepsilon)\delta^2}{16M\rho^2}$$

and choose j so large that $\|X^{(j)} - X^{opt}\| \leq 2\rho$ and $\langle C, X^{(j)} - \bar{X}\rangle < \delta'$. The direction

$$\Delta X := \frac{\varepsilon\delta}{8M\rho^2}(X^{opt} - X^{(j)})$$

is feasible for (8), as $\alpha = \frac{\varepsilon\delta}{8M\rho^2} \leq 1$ by construction. As $\langle C, X^{(j)}\rangle$ decreases monotonically in j, we have $\langle C, X^{(j)} - X^{opt}\rangle \geq \delta$. Further,

$$\begin{array}{rcl}
\varepsilon\langle C, X^{(j+1)} - X^{(j)}\rangle & = & \varepsilon\langle C, \Delta X^{(j)}\rangle \\
& \leq & \varepsilon\langle C, \Delta X^{(j)}\rangle + (1-\varepsilon)\|\Delta X^{(j)}\|_j^2 \\
& \leq & \varepsilon\langle C, \Delta X\rangle + (1-\varepsilon)\|\Delta X\|_j^2 \\
& \leq & -\varepsilon\frac{\varepsilon\delta}{8M\rho^2}\delta + (1-\varepsilon)M(\frac{\varepsilon\delta}{8M\rho^2}2\rho)^2 \\
& = & -\delta'\varepsilon \\
& < & \varepsilon\langle C, \bar{X} - X^{(j)}\rangle
\end{array}$$

so that $\langle C, X^{(j+1)}\rangle < \langle C, \bar{X}\rangle$, contradicting monotonicity of $\langle C, X^{(j)}\rangle$. □

Remark 2.1 *The proof of Theorem 2.1 is also true, when $\varepsilon = \varepsilon_j \in (0,1)$ is changed at each iteration as long as there exists a uniform lower bound $\bar{\varepsilon} > 0$ such that $\varepsilon_j \geq \bar{\varepsilon}$ for all j.*

Remark 2.2 *If $\varepsilon > 0$ is sufficiently small, and if the norm $\| \,.\, \|_j$ is the Frobenius norm, then by convexity of $\mathcal{C}^*$ one may interpret the iterates $X^{(j)}$ as approximations to a continuous steepest descent path projected onto $\mathcal{C}^*$.*

2.2 Inner iteration – working in V space

It is the aim of this subsection to compute solutions of the regularized subproblems (8). The key idea is to work in the space of V variables instead of X variables, arriving at a quadratic reformulation of (8) that does not use the cone $\mathcal{C}^*$.

Let an iterate $X^{(j)} = VV^\top$ be given where $V \in \mathbb{R}^{n\times k}$ with $k \geq n(n+1)/2$. Then, the next iterate $X^{(j+1)} = X^{(j)} + \Delta X \in \mathcal{C}^*$ can be written as $X^{(j+1)} = (V + \Delta V)(V + \Delta V)^\top$ with $V + \Delta V \geq O$. Defining

$$\Delta X = \Delta X(\Delta V) := V(\Delta V)^\top + (\Delta V)V^\top + (\Delta V)(\Delta V)^\top, \qquad (10)$$

problem (8) is equivalent to

$$\begin{array}{rl}
\min & \varepsilon\langle C, \Delta X(\Delta V)\rangle + (1-\varepsilon)\|\Delta X(\Delta V)\|_j^2 \\
\text{s.t.} & \langle A^{(i)}, \Delta X(\Delta V)\rangle = 0\,,\ i = 1:m \\
& V + \Delta V \geq O\,.
\end{array} \qquad (11)$$

Here, the intractable cone $\mathcal{C}^*$ of (8) has been replaced with the nonnegative orthant. We stress a certain paradoxon: by construction, (8) and (11) are equivalent. (8) has a unique local – and global – optimal solution. However, (11) may have multiple local (nonglobal) solutions; the equivalence only refers to the global solution of (11).

We now discuss the numerical solution of (11). An update $X^{(j+1)} = (V + \Delta V)(V + \Delta V)^\top$ of $X^{(j)}$ is (2)-feasible if $V + \Delta V$ has no negative entries and if the quadratic constraints

$$\langle A^{(i)}, VV^\top\rangle + 2\langle A^{(i)}V, \Delta V\rangle + \langle A^{(i)}, \Delta V(\Delta V)^\top\rangle = b_i \tag{12}$$

are satisfied. Given V, the slacks $s_i := b_i - \langle A^{(i)}, VV^\top\rangle$ are zero when $X^{(j)}$ satisfies the linear equations, else some slacks are nonzero.

For small $\varepsilon > 0$, the norm of the solution ΔX of (8) is small, and as shown in Corollary 2.1 below, when $V > 0$ has full rank there exists a ΔV such that $\Delta X = \Delta X(\Delta V)$ and such that the norm of ΔV is small as well. (ΔV is not unique in general.)

Theorem 2.1 suggests to replace the norm $\|\Delta X(\Delta V)\|_j$ with the norm (for singular V this is only a semi-norm)

$$\|\Delta V\|_V := \|V\Delta V^\top + \Delta V V^\top\| \approx \|V\Delta V^\top + \Delta V V^\top + \Delta V \Delta V^\top\| = \|\Delta X(\Delta V)\|.$$

$\|\Delta V\|_V$ is not a norm of the perturbation $\Delta X(\Delta V)$ but some other form of regularization. Indeed it is possible to maintain the statement of Theorem 2.1 for certain nonquadratic regularizations as well. In our numerical implementation we replace $\|\Delta V\|_V$ with $\|\Delta V\|$ since this leads to subproblems that can be solved extremely efficiently.

Based on (12) we arrive at the quadratic subproblem

$$\begin{array}{rl} \min & \varepsilon\,[2\langle CV, \Delta V\rangle + \langle C\Delta V, \Delta V\rangle] + (1-\varepsilon)\|\Delta V\|^2 \\ \text{s.t.} & \langle A^{(i)}\Delta V, \Delta V\rangle + 2\langle A^{(i)}V, \Delta V\rangle = s_i\,,\ i = 1:m \\ & V + \Delta V \in \mathbb{R}^{n\times k}_+ \end{array} \tag{13}$$

which is equivalent to (8) and (11) when $\|\Delta X\|_j$ is replaced with the regularization term $\|\Delta V\|$. Since $\|\Delta V\|$ is small when $\varepsilon > 0$ is small, we linearize the quadratic constraints in (13) in order to obtain a good approximation of the solution of (13).

2.3 An implementable method for (13)

When computing a local solution of (13) one needs to observe that a slight infeasibility ($\tilde{s} \neq 0$) could allow a large reduction of the objective function; this condition number is not known. In Algorithm 2.1 we therefore outline an approach that generates a *feasible* local solution of (13) in the limit, based on linearization and Tikhonov regularization.

Before outlining Algorithm 2.1 next, we note that the approach chosen here does not guarantee to find the global optimal solution of (13), see Section 2.4 below.

Algorithm 2.1 (Iterative solution of (13)) .
Input: $\varepsilon \in (0,1)$ *and* $V \geq O$ *with* $\langle A^{(i)}, VV^\top\rangle = b_i$ *for all* i.

1. *Set* $\Delta V^{\text{old}} := O$ *and* $\tilde{s}_i := 0$ *for all* i. *Set* $l = 1$ *and* $\tau_1 := 1 - \varepsilon$.

2. *Solve*

$$\begin{array}{rl} \min & \varepsilon\langle 2C(V + \Delta V^{\text{old}}), \Delta V\rangle + (1-\varepsilon)\|\Delta V^{\text{old}} + \Delta V\|^2 + \tau_l\|\Delta V\|^2 \\ \text{s.t.} & \langle 2A^{(i)}(V + \Delta V^{\text{old}}), \Delta V\rangle = \tilde{s}_i\,,\ i = 1:m \\ & V + \Delta V^{\text{old}} + \Delta V \in \mathbb{R}^{n\times k}_+\,, \end{array} \tag{14}$$

and denote the optimal solution by ΔV^l*.*

3. *Update* $\Delta V^{\mathrm{old}} := \Delta V^{\mathrm{old}} + \Delta V^l$ *and* $\tilde{s}_i := b_i - \langle A^{(i)}(V + \Delta V^{\mathrm{old}}), V + \Delta V^{\mathrm{old}} \rangle$*.*
4. *If* $\|\Delta V^{\mathrm{old}}\| > 1$ *set* $\varepsilon = \varepsilon/2$ *and* $\Delta V^{\mathrm{old}} = \Delta V^{\mathrm{old}}/2$*.*
5. *If* $\|\Delta V^l\| \approx 0$*: Stop,* ΔV^{old} *approximately solves (13) locally.*
6. *Else update* $l := l + 1$*,* $\tau_l := 1.5 * \tau_{l-1}$*, and go to Step 2.*

The parameter $\tau_l \to \infty$ controls the step size $\|\Delta V^l\|$ at each iteration l; for small $\|\Delta V^l\|$ the linearization error $\|\tilde{s}\|$ is small. If $\|\Delta V^l\| \to 0$ for $l \to \infty$, then, evidently, ΔV^{old} is feasible for (13) in the limit, and in fact it is a local solution of (13) in the limit. We do not elaborate on the stopping criterion in Step *5.* but assume that the equations of (13) are satisfied to machine precision.

The key step at each iteration l of Algorithm 2.1 is the solution of problem (14) which is discussed next. Setting $\tilde{C} = 2\varepsilon C(V + \Delta V^{\mathrm{old}}) + 2(1-\varepsilon)\Delta V^{\mathrm{old}}$, $\tilde{A^{(i)}} = 2A^{(i)}(V + \Delta V^{\mathrm{old}})$, and $\rho = \tau + 1 - \varepsilon > 0$, we arrive at the condensed form of (14):

$$\begin{array}{rl} \min & \langle \tilde{C}, \Delta V \rangle + \rho \|\Delta V\|^2 \\ \text{s.t.} & \langle \tilde{A^{(i)}}, \Delta V \rangle = \tilde{s}_i \,,\ i = 1:m \,, \\ & V + \Delta V \in \mathbb{R}_+^{n \times k} \,. \end{array} \tag{15}$$

This is a strictly convex quadratic problem over a polyhedron and thus tractable also for large n and k, as long as m is moderately small. In vector form, $x = \mathrm{vec}(\Delta V)$, $v = \mathrm{vec}(V)$, $\tilde{c} = \mathrm{vec}(\tilde{C})$, $\tilde{a}_i = \mathrm{vec}(\tilde{A^{(i)}})$, problem (15) can be written as a problem in $\mathbb{R}^{nk}$,

$$\min \left\{ \tilde{c}^\top x + \rho x^\top x : \tilde{a}_i^\top x = \tilde{s}_i \,,\ i = 1:m \,,\ x + v \geq o \right\} . \tag{16}$$

Note that the Hessian of the objective function is the identity. This greatly simplifies the use of interior-point methods for this problem.

2.4 Local, nonglobal solutions

Here, we discuss the "disturbing fact" that (13) may have local solutions that do not correspond to minimizers of the convex problem (8), even when ε – and hence $\|\Delta V\|$ and $\|\Delta X\|$ – are very small. In the extreme case, where V does not have full rank there may be a lack of Lipschitz continuity of ΔV as a function of ΔX. This can be illustrated with a simple example:

Let

$$\min \{ \langle C, X \rangle : \langle J, X \rangle = 1 \,,\ X \in \mathcal{C}^* \} \tag{17}$$

be the Motzkin-Straus reformulation of the max-clique problem, i.e. $C = -A_G$ where A_G is the adjacency matrix of a graph G. Let χ, $\bar{\chi}$ be the characteristic vectors of two cliques in G with $\chi^\top e < \bar{\chi}^\top e$ and set $v = \chi / e^\top \chi$, $\bar{v} = \bar{\chi} / e^\top \bar{\chi}$. Let $V = [v, o]$ and $\Delta V = [-v, \bar{v}]$, then $X = VV^\top$ is feasible for (17) and

$$\Delta X := V \Delta V^\top + \Delta V V^\top + \Delta V \Delta V^\top = -vv^\top - vv^\top + vv^\top + \bar{v}\bar{v}^\top = \bar{v}\bar{v}^\top - vv^\top$$

is a feasible descent direction for (17) and also for the regularized problem of the form (8). A line search along $\lambda \Delta X$ will prove nonoptimality of X for any $\lambda \in (0, 1]$. However, a line search along $\lambda \Delta V$ shows that

$$\langle C, (V + \lambda \Delta V)(V + \lambda \Delta V)^\top \rangle = (1 - \lambda)^2 \langle C, vv^\top \rangle + \lambda^2 \langle C, \bar{v}\bar{v}^\top \rangle$$

is increasing for small $\lambda > 0$ (because $\langle C, vv^\top \rangle < 0$) and thus, ΔV is not a descent direction for (13). ($\lambda \Delta V$ is not a feasible step for $\lambda < 1$ either.)

Observe that the descent step $\Delta X_\delta := \delta(\bar{v}\bar{v}^\top - vv^\top)$ for $\delta \in (0,1)$ corresponds to ΔV_δ with

$$\Delta X_\delta = V \Delta V_\delta^\top + \Delta V_\delta V^\top + \Delta V_\delta \Delta V_\delta^\top \tag{18}$$

and $\Delta V_\delta = [-\rho v, \sqrt{\delta}\bar{v}]$ with $\rho := 1 - \sqrt{1-\delta} \approx \delta/2$. Here, the second column of ΔV_δ is not a Lipschitz continuous function of δ at $\delta = 0$, and, as implied by Proposition 2.1 below, there does not exist any other Lipschitz continuous function ΔV_δ satisfying (18) either, if χ is a maximal clique, (i.e., χ is a clique that is not contained in a larger clique – but there may be other cliques of larger cardinality).

Proposition 2.1 *We consider the simple version (17) of (2). When χ is a maximal clique, $v = \chi / e^\top \chi$, $V = [v, o]$, and $\varepsilon > 0$ is sufficiently small, then $\Delta V = O$ is a local minimizer of (13).*

Proof. Without loss of generality, let $\chi = (1, \ldots, 1, 0 \ldots, 0)^\top$ and $p := n - \chi^\top e$ be the number of zeros in χ. The KKT conditions of (13) (case $m = 1$) at $\Delta V = O$ are

$$\exists y \in \mathbb{R},\ S \in \mathbb{R}^{n \times 2}: \quad 2\varepsilon CV + 2yAV - S = O, \quad S \geq O, \quad \langle S, V \rangle = 0.$$

Here,

$$CV = \frac{-1}{\chi^\top e - 1} \begin{pmatrix} 1, \ldots, 1, \xi_1 \ldots, \xi_p \\ 0, \ldots, 0 \end{pmatrix}^\top$$

where $\xi_l \leq 1$ (since, by assumption, the clique χ is maximal), and

$$AV = \begin{pmatrix} 1, \ldots, 1 \\ 0, \ldots, 0 \end{pmatrix}^\top.$$

Choosing

$$y = \frac{\varepsilon}{\chi^\top e - 1} \quad \text{and} \quad S = \frac{2\varepsilon}{\chi^\top e - 1} \begin{pmatrix} 0, \ldots, 0, \zeta_1 \ldots, \zeta_p \\ 0, \ldots, 0 \end{pmatrix}^\top,$$

where $\zeta_l = 1 - \xi_l \geq 0$ completes the KKT conditions.

Note that y is proportional to ε so that the Lagrangian is positive definite for sufficiently small $\varepsilon > 0$, and thus, $\Delta V = O$ also satisfies the second order sufficient conditions for a local minimizer.

□

Note that in this proof we considered the case $k = 2$, but the argument can directly be applied to any $k \geq 1$. In a weaker form it also extends to the case when V is very close to a rank-1-matrix. As a consequence of Proposition 2.1, problem (13) may have local solutions in spite of the regularization term $(1-\varepsilon)\|\Delta V\|^2$. In the next section we identify a situation where the Lipschitz continuity of ΔV as a function of ΔX is maintained, and thus the negative example of this section is no longer applicable.

2.5 Local Lipschitz continuity of the corrections in V-space

Example (18) considers a matrix V of two columns and a perturbation ΔX_δ of rank two. We ignore the linear constraints for the moment and consider more general perturbations ΔX_δ of rank n. Clearly, if rank $(V) < \frac{n}{2}$ then the mapping $\Delta V \mapsto V(\Delta V)^\top + (\Delta V)V^\top$ is not surjective, and hence, there does not exist a Lipschitz continuous function $\Delta V_\delta \mapsto \Delta X_\delta$ as searched for in equation (18). A negative example such as (18), however, can be excluded when $V > 0$ has full rank:

Lemma 2.1 *If V is a rectangular matrix such that $VV^\top \succ 0$, then the map $\Delta V \mapsto V(\Delta V)^\top + (\Delta V)V^\top$ is surjective.*

Proof. To keep this paper self-contained we give a short proof of this well-known result. Assume that $VV^\top \succ 0$. Then there exist n columns of V that are linearly independent. By setting $\Delta V = 0$ for the remaining columns, we may assume without loss of generality that V is a full rank square matrix. Let $V = U\Sigma\tilde{U}$ be the singular value decomposition of V with a positive definite diagonal matrix Σ. Then, the map

$$\Delta V \mapsto U\Sigma\tilde{U}(\Delta V)^\top + (\Delta V)(U\Sigma\tilde{U})^\top$$

is surjective on the space $\mathcal{S}^n$, if and only if the map

$$\Delta V \mapsto U^\top \left(U\Sigma\tilde{U}(\Delta V\tilde{U})^\top + (\Delta V\tilde{U})(U\Sigma\tilde{U})^\top \right) U$$

is surjective. The right hand side of the latter is $\Sigma(\Delta V)^\top + (\Delta V)\Sigma$. Since Σ is a positive diagonal matrix, this map is trivially surjective. □

Corollary 2.1 *If $V > 0$ is a rectangular matrix such that $VV^\top \succ 0$ and if $\|\Delta X\|$ is sufficiently small, then there exists a function $\Delta V = \Delta V(\Delta X)$ such that $V(\Delta V)^\top + (\Delta V)V^\top + (\Delta V)(\Delta V)^\top = \Delta X$. Moreover, ΔV is locally Lipschitz continuous at $\Delta X = O$, and thus, $V + \Delta V \geq 0$.*

Proof. After identifying a linear subspace $\mathcal{H}$ such that the map $\mathcal{H} \to \mathcal{S}^n : \Delta V \mapsto V(\Delta V)^\top + (\Delta V)V^\top$ of Lemma 2.1 is bijective, the proof is an immediate consequence of the implicit function theorem. □

Thus, whenever $V > O$ has rank n the matrix V allows the computation of Lipschitz continuous descent steps ΔV_δ. The full rank condition for V is certainly satisfied for any $X = VV^\top$ in the interior $(\mathcal{C}^*)^\circ$ of $\mathcal{C}^*$.

2.6 Limits of the factorization approach

We reformulate the statement of Corollary 2.1: Given a point $X^{(j)} = V^{(j)}(V^{(j)})^\top \succ O$, $V^{(j)} > O$ there exist $\varepsilon_j > 0$ and $M_j < \infty$ such that for any $\Delta X^{(j)} = (\Delta X^{(j)})^\top$ with $\|\Delta X^{(j)}\| \leq \varepsilon_j$ a matrix $\Delta V^{(j)}$ can be found such that $V^{(j+1)} := V^{(j)} + \Delta V^{(j)} \geq O$, $\|\Delta V^{(j)}\| \leq M_j \|\Delta X^{(j)}\|$, and

$$X^{(j+1)} := X^{(j)} + \Delta X^{(j)} = (V^{(j)} + \Delta V^{(j)})(V^{(j)} + \Delta V^{(j)})^\top. \qquad (19)$$

Unfortunately, this result does not imply that any point in $\mathcal{C}^*$ can be "reached" (in the sense of convergence to) along a sequence of points $X^{(j)}$ in $\mathcal{C}^*$ based on the iteration (19). In fact, the assumptions in Corollary 2.1 can be satisfied for any $k \geq n$. However, there exist matrices $\hat{X}$ in $\mathcal{C}^*$ that have cp-rank $\geq \frac{n^2}{4}$, i.e. the dimension k of $V \in \mathbb{R}^{k \times n}$ must satisfy $k \geq \frac{n^2}{4}$ in order to allow for a representation $X = VV^\top$ with $V \geq O$. In fact, $\hat{X}$ can be chosen as the center of a full-dimensional ball of matrices all having cp-rank $\geq \frac{n^2}{4}$. This observation makes clear: Based on the result of Corollary 2.1 one cannot guarantee to "reach" such a matrix $\hat{X}$ using updates as in (19).

If rows k and l of the initial matrix V^0 coincide, $V^0_{k,*} = V^0_{l,*}$, then rows k and l will remain identical throughout the algorithm implemented in Section 4. Hence, adding a large number of identical rows in order to satisfy $k \geq \frac{n^2}{4}$ does not help. Moreover, it is unclear whether there exists a choice of a finite (polynomial) number of rows that enables convergence to any given $\hat{X} \in \mathcal{C}^*$.

These two observations greatly limit the potential of the factorization heuristics. Nevertheless, for the Motzkin-Straus reformulation of the max-clique problem the cp-rank of at least one optimal solution is one, and hence the algorithm can be implemented with $k \ll \frac{n^2}{4}$ without necessarily preventing convergence to an optimal solution; this approach is taken in Section 4.

2.7 Starting point

Above we have assumed that an initial matrix X and a factorization $X = VV^\top$ are given such that $V \geq O$ and X satisfies all linear constraints. For the case $m = 1$ and $A = ee^\top$ this assumption is easily satisfied by rescaling the matrix $I + ee^\top$, but in general, satisfying this assumption may be as hard as solving problems of the form (2).

If Slater's condition is satisfied for (2), the following heuristics will eventually (i.e. for large p) provide a feasible solution: Choose random vectors $v^j \in [0,1]^n$ for $1 \leq j \leq p$, set $X_j = v^j (v^j)^\top$, and define $\beta_{i,j} = \langle A^{(i)}, X_j \rangle$. Then find $x_j \geq 0$ minimizing

$$\sum_{i=1}^{m} \left(b_i - \sum_j \beta_{i,j} \, x_j \right)^2 .$$

When the optimal value is zero, $V = [\sqrt{x_1} v^1, \ldots, \sqrt{x_p} v^p]$ provides a feasible starting point.

For the numerical examples in Section 4, Algorithm 2.1 was modified as to allow for $\tilde{s}_i \neq 0$ in *Step 1.*; the algorithm could thus be started with some random starting point. It turned out that for the problems tested here, feasibility was not an issue.

3 Generating random instances

In order to test our approaches, we have to find instances of somehow intermediate complexity: on one hand, we have to ensure that the solution cannot be reduced to the tractable approximation $\mathcal{P} \cap \mathcal{N}$ of $\mathcal{C}^*$, or $\mathcal{P} + \mathcal{N}$ of $\mathcal{C}$, so (7) should hold. On the other hand, the harmlessly looking problem

$$\begin{aligned} \omega(G) &= \min\{t : S = t(I + A_{\bar{G}}) - J \in \mathcal{C}\} \\ &= \max\{\langle J, X\rangle : \langle I + A_{\bar{G}}, X\rangle = 1, X \in \mathcal{C}^*\} \end{aligned} \tag{20}$$

with only one equation will be extremely difficult to solve, unless P=NP, due to the inapproximability results for the clique number $\omega(G)$.

We are now going to show how to generate an infinite family of instances satisfying (7), provided we have available one instance with hardness certificate.

Lemma 3.1 *Let $\tilde{Z}$ be a hardness certificate for the instance $(\tilde{\mathcal{A}}, \tilde{b}, \tilde{C})$ with optimal solution $(\tilde{X}, \tilde{y}, \tilde{S})$. Set $S := \tilde{S}, X := \tilde{X}, Z := \tilde{Z}$ and select $y \in \mathbb{R}^m$ arbitrarily. Select matrices $A^{(i)} \perp X - Z$. Finally set $b := \mathcal{A}(X)$ and $C := S + \mathcal{A}^*(y)$. Then (X, y, S) is optimal for the instance $(\mathcal{A}, b, C)$ and (7) holds.*

Proof. Let $(\tilde{X}, \tilde{y}, \tilde{S})$ be optimal for (1), (2) with data $(\tilde{\mathcal{A}}, \tilde{b}, \tilde{C})$ and hardness certificate $\tilde{Z}$. This means we have

$$\tilde{\mathcal{A}}(\tilde{X}) = \tilde{b}, \tilde{X} \in \mathcal{C}, \tilde{S} = \tilde{C} - \tilde{\mathcal{A}}^*(\tilde{y}) \in \mathcal{C}^*, \langle \tilde{X}, \tilde{S}\rangle = 0$$

and $\tilde{Z} \in \mathcal{P} \cap \mathcal{N}$ satisfies $\tilde{\mathcal{A}}(\tilde{Z}) = \tilde{b}, \langle \tilde{C}, \tilde{Z}\rangle < \tilde{b}^\top \tilde{y}$.

By construction, (X, y, S) solves the copositive program with data $(\mathcal{A}, b, C)$. We show that (7) holds for this problem. To see this, let (u, T) be any feasible solution of (5), so $T = C - \mathcal{A}^*(u) \in \mathcal{P} + \mathcal{N}$. We have, using $b = \mathcal{A}(X) = \mathcal{A}(Z)$,

$$b^\top(u - y) = \langle Z, \mathcal{A}^*(u) - \mathcal{A}^*(y)\rangle = \langle Z, S - T\rangle \le \langle Z, S\rangle,$$

because $\langle Z, T\rangle \ge 0$. But

$$\langle Z, S\rangle = \langle Z, \tilde{C} - \tilde{\mathcal{A}}^*(\tilde{y})\rangle = \langle \tilde{Z}, \tilde{C}\rangle - \tilde{b}^\top \tilde{y} < 0,$$

yielding $b^\top u < b^\top y$ and so (7) holds for the data $(\mathcal{A}, b, C)$. □

We use the problem (20) to generate the desired triple (X, y, S). Thus we need some graph G where we know $\omega(G)$ together with a characteristic vector x for a maximum clique and a hardness certificate Z such that $\omega(G) < \langle J, Z\rangle$ and $\langle I + A_{\bar{G}}, Z\rangle = 1, Z \in \mathcal{P} \cap \mathcal{N}$. In this case $(\frac{1}{\omega(G)} xx^t, \omega(G), \omega(G)(I + A_{\bar{G}}) - J)$ solves (20) and (7) holds for (20).

To simplify the construction of a graph G with the desired properties, we use the strong graph product. This approach is inspired by the study [3] on multiplicativity of the $\theta'(G)$ bound [14, 15, 19] for $\omega(G)$; see also [13]. We recall that

$$\theta'(G) = \max\{\langle J, X\rangle : \langle I, X\rangle = 1, \langle A_{\bar{G}}, X\rangle = 0, X \in \mathcal{P} \cap \mathcal{N}\}.$$

We note that G has a hardness certificate if and only if $\theta'(G) > \omega(G)$.

If H and K are two graphs with adjacency matrices A_H and A_K, then the graph G with adjacency matrix

$$A_G = A_H \otimes A_K + I \otimes A_K + A_H \otimes I$$

is called the strong graph product of H and K, $G = H * K$ for short. It is well known that $\omega(G) = \omega(H)\omega(K)$. Note also that

$$I + A_G = (I + A_H) \otimes (I + A_K).$$

We are going to show now that if one of the factors, say H, has a hardness certificate, then so does the strong graph product.

Lemma 3.2 *Let K and H be graphs and suppose that H has a hardness certificate. Then $G = H * K$ also has a hardness certificate.*

Proof. Since H has a hardness certificate, there exists $Z_H \in \mathcal{P} \cap \mathcal{N}$ such that $\langle I, Z_H\rangle = 1, \langle A_{\bar{H}}, Z_H\rangle = 0$ and $\langle J, Z_H\rangle > \omega(H)$. Let x_K be the incidence vector of a clique of size $\omega(K)$ in K and set $Z_K = \frac{1}{\omega(K)} x_K x_K^\top$, so $\langle I, Z_K\rangle = 1, \langle A_{\bar{K}}, Z_K\rangle = 0, \langle J, Z_K\rangle = \omega(K)$.

We now show that $Z_G = Z_H \otimes X_K$ is a hardness certificate for G. Clearly, $Z_G \in \mathcal{P} \cap \mathcal{N}$. Note that, using $I + A_H + A_{\bar{H}} = J$, we have $\langle I + A_H, Z_H\rangle = \langle J, Z_H\rangle$, and similarly $\langle I + A_K, Z_K\rangle = \langle J, Z_K\rangle$. Therefore

$$\langle I + A_G, Z_G\rangle = \langle I + A_H, Z_H\rangle\langle I + A_K, Z_K\rangle = \langle J, Z_H\rangle\langle J, Z_K\rangle = \langle J, Z_G\rangle,$$

and so $\langle A_{\bar{G}}, Z_G\rangle = 0$. We therefore have $\langle I + A_{\bar{G}}, Z_G\rangle = \langle I, Z_G\rangle = 1$, and

$$\langle J, Z_G\rangle = \langle J, Z_H\rangle\langle J, Z_K\rangle > \omega(H)\omega(K) = \omega(G),$$

as desired. □

We now summarize the generation of random instances as follows.

1. Select a graph H with incidence vectors $x_1, \ldots, x_h$ of cliques of size $\omega(H) < \vartheta'(H)$. In our generator, we take the 5-cycle, $H = C_5$, which satisfies $\omega(C_5) = 2 < \vartheta'(C_5) = \sqrt{5}$. Select Z_H with $\langle J, Z_H\rangle > \omega(H)$ and $\langle I + A_{\bar{H}}, Z_H\rangle = 1, Z_H \in \mathcal{P} \cap \mathcal{N}$. (We use $Z_H = \frac{1}{5}I + \frac{1}{9}A_H$.)

2. Select a perfect graph K with incidence vectors $\xi_1, \ldots, \xi_k$ of cliques of size $\omega(K)$. In our case K is selected to be an interval graph. Set $Z_K = \frac{1}{\omega(K)}\xi_1\xi_1^\top$.

3. Let $z_1, \ldots, z_{hk}$ be a set of maximum cliques in $G = H * K$, given by $x_i \otimes \xi_j, \forall i, j$. Set $\omega(G) = \omega(H)\omega(K)$ and let $X_G = \frac{1}{\omega(G)} \sum_i \lambda_i z_i z_i^\top$ where $\lambda_i \geq 0, \sum_i \lambda_i = 1$ can be chosen arbitrarily. Finally, $S_G = \omega(G)(I + A_{\bar{G}}) - J$ and $Z_G = Z_H \otimes Z_K$.

4. Select $y \in \mathbb{R}^m$ and also $A^{(i)} \perp X_G - Z_G$ and set $b = \mathcal{A}(X), C = S_G + \mathcal{A}^*(y)$.

Then (X_G, y, S_G) is an optimal solution of (1), (2) with data (A, b, C) and (7) holds.

4 Computational results

In this section we present some computational results of our algorithm on a variety of copositive programs. The only nontrivial part of our algorithm is the solution of the convex quadratic problem (15) which has nk sign constrained variables ΔV and m equality constraints. The Hessian of the objective function is a multiple of the identity; therefore interior point methods are well suited to solve the subproblem. We use a standard primal-dual Matlab implementation.

α	initial value	final value
0.00	27330	425.9687
0.70	9131	371.8865
0.80	6105	371.8532
0.90	3007	371.8363
0.95	1611	371.8284

Table 1: Starting points $V(\alpha) = \alpha V_{opt} + (1 - \alpha)R$; optimal value is 371.8261.

k	worst	average	best
n	387.2253	377.9922	372.5890
$2n$	376.1712	373.1243	372.0279
$3n$	375.0452	372.4333	371.9574

Table 2: Starting points V_0 are of size $n \times k$, the final values are from 10 runs, giving the best worst and average values, the optimal value is 371.8261

4.1 Randomly generated instances

As a first experiment we consider random copositive programs that cannot be reduced to programs over the intersection of the positive semidefinite cone and the nonnegative cone. The generation was described in the previous section. A matlab implementation makes these instances reproduceable.

We first experiment with a single instance. It was generated to have matrix order $n = 50$ and $m = 80$ constraints. The optimal objective function value is 371.8261. As a first experiment we verify that starting close to the optimal solution, our algorithm actually recovers the optimal solution. In Table 1 we show results of our algorithm using starting points $V(\alpha) = \alpha V_{opt} + (1 - \alpha)R$. Here, V_{opt} is the optimal solution with $k = 30$ columns, and R is a matrix with entries uniformly drawn from the unit interval and then scaled to the same norm as V_{opt}. To allow convergence also for the case $\alpha = 0$ we chose a large value $\epsilon = 0.9$ and 100 outer iterations, with 10 inner iterations each. We clearly see that starting close to V_{opt} allows us to recover the optimal solution with only small error. Also the solution times are shorter in this case.

As a second experiment, we consider random starting points on the same instance, again with $\epsilon = 0.9$ and 100 outer iterations, with 10 inner iterations each. We now investigate the effect of the random starting point, depending on the number k of columns. In Table 2, we provide for $k \in \{n, 2n, 3n\}$ the final objective function values. We generated 10 starting points with the Matlab command `V = rand(n,k)`, and provide the best, the worst and the average final objective function value. It comes as no surprise that the quality of the solutions improves as we increase k. We also note that using larger values of k takes us very close to the true (but in general unknown) optimum. We also note that the fluctuation of the final result decreases with increasing k.

Up to now we have experimented with just a single instance. In Table 3 we show computational results for random instances of various sizes. We include n, m and the optimal value, and provide computational results of our algorithm with $k = 3n$ columns in V. The values listed in column f_{alg} refer to feasible

n	m	f_{opt}	f_{alg}	sdp & nonneg.	sdp
50	25	14.30	14.49	-156.25	-15573.51
50	50	-83.99	-83.87	-242.13	-13362.76
50	100	-181.73	-181.73	-300.44	-11083.89
75	40	316.14	317.72	8.01	-32952.28
75	75	14.45	16.22	-270.97	-30693.11
75	150	-906.21	-906.02	-1109.05	-27108.43
100	50	168.07	172.09	-298.98	-61129.04
100	100	575.95	578.85	141.12	-52848.75
100	200	565.29	565.98	231.43	-47266.46

Table 3: Starting points V_0 are of size $n \times 3n$, the final values are from a single run

solutions of the completely positive program. For comparison we also list the results obtained from the semidefinite relaxation of the dual (copositive) problem in column "sdp", and from the semidefinite relaxation plus the nonnegative orthant in column "sdp & nonneg". For these examples, the computation times of "sdp & nonneg" were slightly higher than for f_{alg} while the accuracy of the approximation was considerably lower. The plain semidefinite relaxation in column "sdp" gives a very poor approximation of f_{opt}. The rate of convergence of Algorithm 2.1 for these problems is rather low requiring a large number of outer iterations. To reduce this large number we selected a large value $\epsilon = 0.97$. We then restricted the algorithm to 100 outer iterations, with 10 inner iterations each.

4.2 Box constrained nonconvex quadratic problems

Burer [7] shows that quadratic problems with linear constraints in nonnegative, and possibly binary variables can be reformulated as copositive programs. The perhaps simplest case is box-constrained nonconvex quadratic optimization, given as follows:

$$(BQP) \qquad \max \frac{1}{2}x^T Q x + c^T x \text{ such that } 0 \le x \le e.$$

We assume that Q is symmetric, but not negative semidefinite, so the resulting problem is NP-hard. The reformulation of [7] asks to introduce slack variables for the upper bounds, resulting in

$$x + s = e, \; x \ge 0, s \ge 0.$$

The reformulation is based on

$$\begin{pmatrix} 1 \\ x \\ s \end{pmatrix} \begin{pmatrix} 1 \\ x \\ s \end{pmatrix}^T .$$

Therefore, let $Z \in \mathcal{S}^{2n+1}$ be partitioned as follows

$$Z = \begin{pmatrix} 1 & x^T & s^T \\ x & X & R^T \\ s & R & S \end{pmatrix},$$

instance	n	f_{opt}	f_{alg}
spar020-100-1	20	706.50	706.41
spar020-100-2	20	856.50	855.49
spar020-100-3	20	772.00	772.00
spar030-060-1	30	706.00	705.76
spar030-060-2	30	1377.17	1376.59
spar030-060-3	30	1293.50	1288.41
spar030-080-1	30	952.73	952.70
spar030-080-2	30	1597.00	1597.00
spar030-080-3	30	1809.78	1808.34
spar040-030-1	40	839.50	824.58
spar040-030-2	40	1429.00	1427.94
spar040-030-3	40	1086.00	1084.37

Table 4: Box-constrained quadratic programming as copositive program.

where $X, S \in \mathcal{S}^n$. The resulting copositive program is now given as follows.

$$\max \frac{1}{2}\langle Q, X\rangle + c^T x \text{ such that } x_i + s_i = 1, x_{ii} + s_{ii} + 2r_{ii} = 1, \forall i \; Z \in \mathcal{C}^*$$

In Table 4 we present computational results on a series of instances, solved to optimality in [20]. For these examples we choose the number of columns in V as $k = 10$, for all n. We set $\epsilon = 0.5$ and use 30 inner iterations for each outer iteration; the number of outer iterations is set to 100. For this problem class, we know that there exists an optimal solution of rank 1. This knowledge is exploited as follows: After 100 outer iterations a subset of large columns is identified and the algorithm is restarted for another 50 iterations using the large columns of V only. Then, the best column is selected and scaled to satisfy $0 \leq x \leq 1$ providing a feasible solution for the initial problem. Even with such a small value of $k = 10$ (allowing to attack larger problems as well) we observe that in many cases we come very close to the true optimal solution. There are however also instances where we get stuck in local optima. The table includes the number n of the variables in (BQP). The dimension of the resulting CP is $2n + 1$, with $2n + 1$ equality constraints.

4.3 CP heuristic for the clique number

As a final experiment, we consider the copositive formulation of the clique number (20)

$$\max\langle J, X\rangle \text{ such that } \langle I, X\rangle = 1, \langle A_{\bar{G}}, X\rangle = 0, X \in \mathcal{C}^*.$$

It is well known that computing $\omega(G)$ or even getting reasonable approximations of $\omega(G)$ is an extremely difficult task, unless P=NP. So these completely positive programs should be a challenge for our approach. In Table 5 we apply our algorithm to some of the graphs from the DIMACS collection. Again we know that there exists an optimal solution of rank one, so we experiment with the following starting points. We noticed that $V_0 = I$ was a good point to start, and we compare against random starting points having n, and those having 20, columns.

Problem	n	$\omega(G)$	$V_0 = I$	$V_0 = rand(n)$	$V_0 = rand(n, 20)$
brock200-1	200	21	20	20	19
brock200-2	200	12	11	10	10
brock200-3	200	15	13	14	12
brock200-4	200	17	16	15	14
c-fat200-1	200	12	6(12)	6(12)	12
c-fat200-2	200	24	12(24)	11(23)	12(24)
c-fat200-5	200	58	58	58	58
san200-0.7-1	200	30	30	17	30
san200-0.7-2	200	18	16	13	12
san200-0.9-1	200	70	70	70	70
san200-0.9-2	200	60	60	60	38
san200-0.9-3	200	44	35(36)	34(35)	33
phat300-1	300	8	8	8	8
phat300-2	300	25	24	25	25
phat300-3	300	36	33	34	33
brock400-1	400	27	23	24	22
brock400-2	400	29	23	23	22
brock400-3	400	31	23	23	21
brock400-4	400	33	23	23	22
san400-0.5-1	400	13	13	8	7(8)
san400-0.7-1	400	40	22	20	21
san400-0.7-2	400	30	30	16	16
san400-0.7-3	400	22	16	16	13
san400-0.9-1	400	100	52	56	55
sanr400-0.5	400	13	12	12	12
sanr400-0.7	400	21	19	20	18

Table 5: Comparison, different starting points V_0

For these problems, feasibility is not an issue (there is only one linear equation that can trivially be satisfied by rescaling a given solution). This allows to choose large values of ϵ without risking loss of feasibility, and it turned out that larger values of ϵ gave better results in the average. The numbers reported in Table 5 refer to $\epsilon = 0.985$. The number of outer iterations could be reduced to 20 (since ϵ is large), and the number of inner iterations was also set to 20.

In Table 5 we list the largest clique that could be identified from the final result of Algorithm 2.1. In some cases, the algorithm also provided a certificate that there exists a clique of larger cardinality; these certificates are added in brackets following the largest clique that could be identified.

While we do find the largest clique in some cases, there are (as expected) also quite a few instances, where our algorithm gets stuck in rather poor local optima.

5 Summary and conclusions

We have proposed a procedure which finds approximate solutions of the NP-hard problem (2). The method is based on a linearization of nonconvex subproblems and solves a sequence of convex quadratic optimization problems with diagonal Hessian.

There exists many families of optimization problems which can be cast in the form (2). In order to have instances, where we can arbitrarily select the order of the matrix space, and the number of equality constraints, we have set up a generator for instances with known optimal solution, which possess a hardness certificate in the sense of (7).

The computational results show that our method typically finds feasible solutions which are within a small fraction of the optimal value. As a second class of problems we consider box constrained quadratic optimization with nonconvex objective function, which are well known to be NP-hard. These problems have a simple equivalent formulation of the form (2). We apply our method to instances from the literature, where the optimal objective function value is known. Our method again finds feasible solutions with objective function value very close to the optimal value.

To explore the limits of our method we also apply it to the copositive formulation of the clique number ω of a graph. Determining ω is in general not only NP-hard, but even getting good approximations is difficult. Thus it should come as no surprise that our algorithm gets trapped in local solutions. Our cliques sometimes differ by a fair amount from the maximum cliques. We point out however, that we do not do any problem-specific postprocessing, such as local exchanges to find better cliques, but simply use our method in black-box style. We find it remarkable that our method serves as heuristic for such diverse problems as the ones mentioned above.

References

[1] Berman, A., and N. Shaked-Monderer, *Completely positive matrices.* World Scientific, Singapore 2003.

[2] Bomze, I.M., M. Dür, E. de Klerk, A. Quist, C. Roos, and T. Terlaky (2000), “On copositive programming and standard quadratic optimization problems,” *Journal of Global Optimization* **18**, 301–320.

[3] Bomze, I.M., F. Frommlet, and M. Locatelli (2009), “Gap, cosum, and product properties of the θ' bound on the clique number,” to appear in *Optimization.*

[4] Bomze, I.M., and F. Jarre (2009), “A note on Burer's copositive representation of mixed-binary QPs,” Technical Report TR-ISDS **2009-04**, Department of Statistics and Decision Support Systems, University of Vienna, Austria, submitted (August 2009). Available as `http://www.optimization-online.org/DB_HTML/2009/08/2368.html`, last accessed October 9, 2009.

[5] Bundfuss, S., and M. Dür (2008), “Algorithmic Copositivity Detection by Simplicial Partition,” *Linear Algebra and its Applications* **428** 1511–1523.

[6] Bundfuss, S., and M. Dür (2009), “An Adaptive Linear Approximation Algorithm for Copositive Programs,” to appear in *SIAM Journal on Optimization.*

[7] Burer, S. (2009), "On the copositive representation of binary and continuous nonconvex quadratic programs," to appear in *Mathematical Programming.*

[8] de Klerk, E., and D.V. Pasechnik (2002), "Approximation of the stability number of a graph via copositive programming," *SIAM Journal on Optimization* **12**, 875–892.

[9] DIMACS Implementation Challenges, ftp://dimacs.rutgers.edu/pub/challenge/graph

[10] Dür, M., and G. Still (2008), "Interior points of the completely positive cone," *Electronic Journal of Linear Algebra* **17**, 48–53.

[11] Dukanovic, I., and F. Rendl (2010), "Copositive programming motivated bounds on the stability and chromatic numbers," *Mathematical Programming* **121**, 249–268.

[12] Johnson, C.R., and R. Reams (2008), "Constructing copositive matrices from interior matrices," *Electronic Journal of Linear Algebra* **17**, 9–20.

[13] Knuth, D.E. (1994), "The sandwich theorem," *Electronic Joournal of Combinatorics* **22**, 1–48.

[14] Lovász, L. (1979), "On the Shannon capacity of a graph," *IEEE Transactions on Information Theory*, **IT–25**, 1–7.

[15] McEliece, R.J., E.R. Rodemich, and H.C. Rumsey (1978), "The Lovász' bound and some generalizations," *Journal of Combinatorics, Information & System Sciences* **3**, 134–152.

[16] Nesterov, Y., and A. Nemirovski, *Interior Point Polynomial Algorithms in Convex Programming.* SIAM Publications, Philadelphia, USA 1994.

[17] Parrilo, P., *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization.* PhD thesis, California Institute of Technology, USA, 2000.

[18] Povh, J., and F. Rendl (2009), "Copositive and semidefinite relaxations of the quadratic assignment problem," *Discrete Optimization*, **6**, 231–241.

[19] Schrijver, A. (1979), "A comparison of the Delsarte and Lovasz bounds," *IEEE Transactions on Information Theory*, **IT–25**, 425–429.

[20] Vandenbussche, D., and G. Nemhauser (2005), "A branch-and-cut algorithm for nonconvex quadrati programs with box constraints," *Mathematical Programming* **102**, 559-575.