Copositive Programming and Combinatorial Optimization

Franz Rendl

http://www.math.uni-klu.ac.at

Alpen-Adria-Universität Klagenfurt

Austria

joint work with I.M. Bomze (Wien) and F. Jarre (Düsseldorf)

Overview

- What is Copositive Programming ?
- Copositivity and combinatorial optimization ?
- Relaxations based on CP
- Heuristics based on CP

Completely Positive Matrices

Let $A = (a_1, \ldots, a_k)$ be a nonnegative $n \times k$ matrix, then

$$X = a_1 a_1^T + \ldots + a_k a_k^T = AA^T$$

is called completely positive.

$$COP = \{X : X \text{ completely positive}\}$$

COP is closed, convex cone. From the definition we get

$$COP = \text{conv}\{aa^T : a \geq 0\}.$$

For basics, see the book: A. Berman, N. Shaked-Monderer: Completely Positive Matrices, World Scientific 2003

Copositive Matrices

Dual cone COP^* of COP in S_n (sym. matrices):

$$Y \in COP^* \iff \text{tr} XY \geq 0 \;\; \forall X \in COP$$

$$\iff a^T Y a \geq 0 \;\; \forall \text{ vectors } a \geq 0.$$

By definition, this means Y is copositive.

$$CP = \{Y : a^T Y a \geq 0 \; \forall a \geq 0\}$$

CP is dual cone to COP!

Copositive Matrices

Dual cone COP^* of COP in S_n (sym. matrices):

$$Y \in COP^* \iff \text{tr}XY \geq 0 \;\; \forall X \in COP$$

$$\iff a^T Y a \geq 0 \;\; \forall \text{ vectors } a \geq 0.$$

By definition, this means Y is copositive.

$$CP = \{Y : a^T Y a \geq 0 \, \forall a \geq 0\}$$

CP is dual cone to COP!
Bad News: $X \notin CP$ is NP-complete decision problem.

Semidefinite matrices PSD: $Y \in PSD \iff a^T Y a \geq 0 \;\; \forall a$.
Well known facts: • $PSD^* = PSD$ (PSD cone is selfdual.)
• $COP \subset PSD \subset CP$

Semidefinite and Copositive Programs

Problems of the form

$$\max \langle C, X \rangle \text{ s.t. } A(X) = b,\ X \in PSD$$

are called Semidefinite Programs.

Semidefinite and Copositive Programs

Problems of the form

$$\max\langle C, X\rangle \text{ s.t. } A(X) = b,\ X \in PSD$$

are called Semidefinite Programs.

Problems of the form

$$\max\langle C, X\rangle \text{ s.t. } A(X) = b,\ X \in CP$$

or

$$\max\langle C, X\rangle \text{ s.t. } A(X) = b,\ X \in COP$$

are called Copositive Programs, because the primal or the dual involves copositive matrices.

Overview

- What is Copositive Programming ?
- Copositivity and combinatorial optimization ?
- Relaxations based on CP
- Heuristics based on CP

Why Copositive Programs ?

Copositive Programs can be used to solve combinatorial optimization problems.

Why Copositive Programs ?

Copositive Programs can be used to solve combinatorial optimization problems.

- Stable Set Problem:

Let A be adjacency matrix of graph, J be all ones matrix.
Theorem (DeKlerk and Pasechnik (SIOPT 2002))

$$\alpha(G) = \max\{\langle J, X\rangle : \langle A + I, X\rangle = 1, \quad X \in COP\}$$

$$= \min\{y : y(A + I) - J \in CP\}.$$

This is a copositive program with only one equation (in the primal problem).
This is a simple consequence of the Motzkin-Straus Theorem.

Proof (1)

$$\frac{1}{\alpha(G)} = \min\{x^T(A+I)x : x \in \Delta\}$$ (Motzkin-Straus Theorem)

$\Delta = \{x : \ \sum_i x_i = 1, \ x \geq 0\}$ is standard simplex.

Proof (1)

$$\frac{1}{\alpha(G)} = \min\{x^T(A+I)x : x \in \Delta\}$$ (Motzkin-Straus Theorem)

$\Delta = \{x : \ \sum_i x_i = 1, \ x \geq 0\}$ is standard simplex. We get

$$0 = \min\{x^T(A + I - \frac{ee^T}{\alpha})x : \ x \in \Delta\}$$

$$= \min\{x^T(\alpha(A+I) - J)x : \ x \geq 0\}.$$

This shows that $\alpha(A+I) - J$ is copositive.

Proof (1)

$$\frac{1}{\alpha(G)} = \min\{x^T(A+I)x : x \in \Delta\}$$ (Motzkin-Straus Theorem)

$\Delta = \{x : \ \sum_i x_i = 1, \ x \geq 0\}$ is standard simplex. We get

$$0 = \min\{x^T(A + I - \frac{ee^T}{\alpha})x : \ x \in \Delta\}$$

$$= \min\{x^T(\alpha(A+I) - J)x : \ x \geq 0\}.$$

This shows that $\alpha(A+I) - J$ is copositive. Therefore

$$\inf\{y : \ y(A+I) - J \in CP\} \leq \alpha.$$

Proof (2)

Weak duality of copositive program gives:

$$\sup\{\langle J, X\rangle : \ \langle A+I, X\rangle = 1, \ X \in COP\} \leq$$

$$\leq \inf\{y : \ y(A+I) - J \in CP\} \leq \alpha.$$

Proof (2)

Weak duality of copositive program gives:

$$\sup\{\langle J, X\rangle : \ \langle A+I, X\rangle = 1, \ X \in COP\} \leq$$

$$\leq \inf\{y : \ y(A+I) - J \in CP\} \leq \alpha.$$

Now let ξ be incidence vector of a stable set of size α. The matrix $\frac{1}{\alpha}\xi\xi^T$ is feasible for the first problem. Therefore

$$\alpha \leq \sup\{\ldots\} \leq \inf\{\ldots\} \leq \alpha.$$

This shows that equality holds throughout and sup and inf are attained.
The recent proof of this result by DeKlerk and Pasechnik does not make explicit use of the Motzkin Straus Theorem.

Connections to theta function

Theta function (Lovasz (1979)):

$$\vartheta(G) = \max\{\langle J, X\rangle : x_{ij} = 0 \ ij \in E, \mathrm{tr}(X) = 1, X \succeq 0\} \geq \alpha(G).$$

Motivation: If ξ characteristic vector of stable set, then $\frac{1}{\xi^T\xi}\xi\xi^T$ is feasible for above SDP.

Connections to theta function

Theta function (Lovasz (1979)):

$$\vartheta(G) = \max\{\langle J, X\rangle : x_{ij} = 0 \; ij \in E, \text{tr}(X) = 1, X \succeq 0\} \geq \alpha(G).$$

Motivation: If ξ characteristic vector of stable set, then $\frac{1}{\xi^T\xi}\xi\xi^T$ is feasible for above SDP.

Schrijver (1979) improvement: include $X \geq 0$
In this case we can add up the constraints $x_{ij} = 0$ and get

$$\vartheta'(G) = \max\{\langle J, X\rangle : \langle A, X\rangle = 0, \text{tr}(X) = 1, X \geq 0, X \succeq 0\}.$$

($A \ldots$ adjacency matrix). We have $\vartheta(G) \geq \vartheta'(G) \geq \alpha(G)$.
Replacing the cone $X \geq 0, X \succeq 0$ by $X \in COP$ gives $\alpha(G)$, see before.

A general copositive modeling theorem

Burer (2007) shows the following general result for the power of copositive programming:
The optimal values of P and C are equal: opt(P) = opt(C)

$$(P) \qquad \min x^T Q x + c^T x$$

$$a_i^T x = b_i,\ x \geq 0,\ x_i \in \{0,1\}\ \forall i \leq m.$$

Here $x \in \mathbb{R}^n$ and $m \leq n$.

$$(C) \qquad \min \mathrm{tr}(QX) + c^T x,\ \text{s.t. } a_i^T x = b_i,$$

$$a_i^T X a_i = b_i^2,\ X_{ii} = x_i\ \forall i \leq m,\ \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \in COP$$

Overview

- What is Copositive Programming ?
- Copositivity and combinatorial optimization ?
- Relaxations based on CP
- Heuristics based on CP

Approximating COP

We have now seen the power of copositive programming.

Since optimizing over CP is NP-Hard, it makes sense to get approximations of CP or COP.

- To get relaxations, we need supersets of COP, or inner approximations of CP (and work on the dual cone). The Parrilo hierarchy uses Sum of Squares and provides such an outer approximation of COP (dual viewpiont!).

- We can also consider inner approximations of COP. This can be viewed as a method to generate feasible solutions of combinatorial optimization problems (primal heuristic!).

Relaxations

Inner approximation of CP.

$$CP := \{M : x^T M x \geq 0 \ \forall x \geq 0\}$$

Parrilo (2000) and DeKlerk, Pasechnik (2002) use the following idea to approximate CP from inside:

$$M \in CP \text{ iff } P(x) := \sum_{ij} x_i^2 x_j^2 m_{ij} \geq 0 \ \ \forall x.$$

A sufficient condition for this to hold is that
$P(x)$ has a sum of squares (SOS) representation.

Theorem Parrilo (2000) : $P(x)$ has SOS iff $M = P + N$, where $P \succeq 0$ and $N \geq 0$.

Parrilo hierarchy

To get tighter approximations, Parrilo proposes to consider SOS representations of

$$P_r(x) := (\sum_i x_i^2)^r P(x)$$

for $r = 0, 1, \ldots$. (For $r = 0$ we get the previous case.) Mathematical motivation by an old result of Polya.

Theorem Polya (1928):
If M strictly copositive then $P_r(x)$ is SOS for some sufficiently large r.

Parrilo hierarchy (2)

Parrilo characterizes SOS for $r = 0, 1$:

$P_0(x)$ is SOS iff $M = P + N$, where $P \succeq 0$ and $N \geq 0$.

$P_1(x)$ is SOS iff $\exists M_1, \ldots, M_n$ such that

$$M - M_i \succeq 0$$

$$(M_i)_{ii} = 0 \; \forall i \quad (M_i)_{jj} + 2(M_j)_{ij} = 0 \; \forall i \neq j$$

$$(M_i)_{jk} + (M_j)_{ik} + (M_k)_{ij} \geq 0 \; \forall i < j < k$$

The resulting relaxations are SDP. But the $r = 1$ relaxation involves n matrices and n SDP constraints to certify SOS. This is computationally challenging.

Overview

- What is Copositive Programming ?
- Copositivity and combinatorial optimization ?
- Relaxations based on CP
- Heuristics based on CP

Inner approximations of COP

We consider

$$\min\langle C, X\rangle \text{ s.t. } A(X) = b,\ X \in COP$$

Remember: COP= $\{X : X = VV^T,\ V \geq 0\}$.

Some previous work by:

• Bomze, DeKlerk, Nesterov, Pasechnik, others:
Get stable sets by approximating COP formulation of the stable set problem using optimization of quadratic over standard simplex, or other local methods.

• Bundschuh, Dür (2008): linear inner and outer approximations of CP

Incremental version

A general feasible descent approach:
Let $X = VV^T$ with $V \geq 0$ be feasible. Consider the regularized, and convex descent step problem:

$$\min \epsilon\langle C, \Delta X\rangle + (1-\epsilon)\|\Delta X\|^2,$$

such that $A(\Delta X) = 0, \quad X + \Delta X \in COP.$

Incremental version

A general feasible descent approach:
Let $X = VV^T$ with $V \geq 0$ be feasible. Consider the regularized, and convex descent step problem:

$$\min \epsilon\langle C, \Delta X\rangle + (1-\epsilon)\|\Delta X\|^2,$$

$$\text{such that } A(\Delta X) = 0, \quad X + \Delta X \in COP.$$

For small $\epsilon > 0$ we approach the true optimal solution, because we follow the continuous steepest descent path, projected onto COP.

Unfortunately, this problem is still not tractable. We approximate it by working in the V-space instead of the X-space.

Incremental version: V-space

$$X^+ = (V + \Delta V)(V + \Delta V)^T \text{ hence },$$

$$X = VV^T$$

$$\Delta X = \Delta X(\Delta V) = V\Delta V^T + \Delta V V^T + (\Delta V)(\Delta V^T).$$

Now linearize and make sure ΔV is small.

Incremental version: V-space

$$X^+ = (V+\Delta V)(V+\Delta V)^T \text{ hence },$$

$$X = VV^T$$

$$\Delta X = \Delta X(\Delta V) = V\Delta V^T + \Delta V V^T + (\Delta V)(\Delta V^T).$$

Now linearize and make sure ΔV is small. We get the following subproblem:

$$\min \epsilon\langle 2CV, \Delta V\rangle + (1-\epsilon)\|\Delta V\|^2 \text{ such that}$$

$$\langle 2A_i, \Delta V\rangle = b_i - \langle A_i V, V\rangle \ \ \forall i,$$

$$V + \Delta V \geq 0$$

This is convex approximation of nonconvex version in $\Delta X(\Delta V)$.

Algorithm: basic version

Input: $(A, b, C),\ 0 < \epsilon < 1,\ V \geq 0$

- repeat 'until done'
  - Solve subproblem

$$\min \epsilon \langle 2CV, \Delta V \rangle + (1-\epsilon) \|\Delta V\|^2$$

s. t. $\langle 2A_iV, \Delta V \rangle = b_i - \langle A_iV, V \rangle\ \forall i, \quad V + \Delta V \geq 0.$

  - Update: $V \leftarrow V + \Delta V$
- end repeat

Possible stopping condition:
$\|\Delta V\|$ and $\|b - A(X)\|$ should be small.

Correcting the linearization

The linearization $V\Delta V^T + \Delta V V^T$ introduces an error both in the cost function and in the constraints $A(X) = b$. We therefore include corrector iterations of the form

$$\Delta V = \Delta V_{old} + \Delta V_{corr}$$

before the actual update $V \leftarrow V + \Delta V$.

Correcting the linearization

The linearization $V\Delta V^T + \Delta V V^T$ introduces an error both in the cost function and in the constraints $A(X) = b$. We therefore include corrector iterations of the form

$$\Delta V = \Delta V_{old} + \Delta V_{corr}$$

before the actual update $V \leftarrow V + \Delta V$.

Final Subproblem: Additional input: ΔV_{old} (initially set to 0)

$$\min \epsilon \langle 2C(V + \Delta V_{old}), \Delta V \rangle + (1 - \epsilon)\|\Delta V\|^2$$

s.t. $\langle 2A_i(V + \Delta V_{old}), \Delta V \rangle = b_i - \langle A(V + \Delta V_{old}), V + \Delta V_{old} \rangle,$

$$V + \Delta V_{old} + \Delta V \geq 0.$$

Convex quadratic subproblem

The convex subproblem is of the following form, after appropriate redefinition of data and variables $x = vec(V + \Delta V), \ldots$.

$$\min c^T x + \rho x^T x \text{ such that } Rx = r, \; x \geq 0.$$

Convex quadratic subproblem

The convex subproblem is of the following form, after appropriate redefinition of data and variables $x = vec(V + \Delta V), \ldots$.

$$\min c^T x + \rho x^T x \text{ such that } Rx = r, \; x \geq 0.$$

If V is $n \times k$ (k columns generating V), x if of dimension nk and there are m equations.

Convex quadratic subproblem

The convex subproblem is of the following form, after appropriate redefinition of data and variables $x = vec(V + \Delta V), \ldots$.

$$\min c^T x + \rho x^T x \text{ such that } Rx = r,\ x \geq 0.$$

If V is $n \times k$ (k columns generating V), x if of dimension nk and there are m equations.
Since Hessian of cost function is identity matrix, this problem can be solved efficiently using interior-point methods. (convex quadratic with sign constraints and linear equations)

Main effort is solving a linear system of order m, essentially independent of n and k.

Test data sets

COP problems coming from formulations of NP-hard problems are too difficult (Stable Set, Coloring, Quadratic Assignment) to test new algorithms.

Would like to have:

Data (A, b, C) and (X, y, Z) such that

- (X, y, Z) is optimal for primal and dual (no duality gap).
- COP is nontrivial (optimum not given by optimizing over semidefiniteness plus nonnegativity)
- generate instances of varying size both in n and m.

Hard part: Z provably copositive !!

A COP generator

Basic idea: Let G be a graph, and (X_G, y_G, Z_G) be optimal solution of

$$\omega(G) = \max\{\langle J, X\rangle : \langle A_G + I, X\rangle = 1,\ X \in COP\}$$

$$= \min\{y : y(A_G + I) - J \in CP\}.$$

A COP generator

Basic idea: Let G be a graph, and (X_G, y_G, Z_G) be optimal solution of

$$\omega(G) = \max\{\langle J, X\rangle : \langle A_G + I, X\rangle = 1,\ X \in COP\}$$

$$= \min\{y : y(A_G + I) - J \in CP\}.$$

Would like to have

$$\omega(G) < \max\{\langle J, X\rangle : \langle A_G + I, X\rangle = 1, X \succeq 0, X \geq 0\} = \vartheta'(G).$$

This insures that SDP is not enough to solve the problem.

Optimal complementary pair: $Z_G = y_G(A_G + I) - J$ and $X_G = \sum_i \lambda_i \xi_i \xi_i^T$ with ξ_i characteristic vectors of max clique.

A COP generator (2)

Reverse Engineering: Select $A_i (i = 1, \ldots, m)$ and set $b = A(X_G)$, i.e. $b_i = \langle A_i, X_G \rangle$. Select $y \in I\!R^m$ and set $C = Z_G - A^T(y)$

Now (X_G, y, Z_G) is optimal for data (A, b, C).

A COP generator (2)

Reverse Engineering: Select $A_i (i = 1, \ldots, m)$ and set $b = A(X_G)$, i.e. $b_i = \langle A_i, X_G \rangle$. Select $y \in I\!R^m$ and set $C = Z_G - A^T(y)$

Now (X_G, y, Z_G) is optimal for data (A, b, C).

To generate bigger instances, we select $G = H * K$ (strong graph product) where K is a perfect graph and H is such that $\omega(H) < \vartheta'(H)$.
We get $A_G = A_H \otimes I + I \otimes A_K + A_H \otimes A_K$.
Facts: $\omega(G) = \omega(H)\omega(K)$ and $\vartheta'(G) = \vartheta'(H)\vartheta'(K)$.
Hence, we have optimal solution

$$y_G = \omega_H \omega_K, Z_G = y_G(A_G + I) - J, X_G = X_H \otimes X_K.$$

In our generator we set $H = C_5$ (5-cycle)

Computational results

A sample instance with $n = 60,\ m = 100$.

$z_{sdp} = -9600, 82,\quad z_{sdp+nonneg} = -172.19, z_{cop} = -69.75$

Computational results

A sample instance with $n = 60,\ m = 100$.

$z_{sdp} = -9600, 82,\ \ z_{sdp+nonneg} = -172.19, z_{cop} = -69.75$

it	|b-A(X)|	f(x)
1	0.002251	-68.7274
5	0.000014	-69.5523
10	0.000001	-69.6444
15	0.000001	-69.6887
20	0.000000	-69.6963

The number of inner iterations was set to 5, column 1 shows the outer iteration count.

Computational results

A sample instance with $n = 60,\ m = 100$.

$z_{sdp} = -9600,82, \quad z_{sdp+nonneg} = -172.19, z_{cop} = -69.75$

it	|b-A(X)|	f(x)
1	0.002251	-68.7274
5	0.000014	-69.5523
10	0.000001	-69.6444
15	0.000001	-69.6887
20	0.000000	-69.6963

The number of inner iterations was set to 5, column 1 shows the outer iteration count.

But starting point: $V0 = .95$ Vopt $+ .05$ rand

Computational results (2)

Example (continued). recall $n = 60,\ m = 100$.

$z_{sdp} = -9600, 82,\quad z_{sdp+nonneg} = -172.19, z_{cop} = -69.75$

start	iter	|b-A(X)|	f(x)
(a)	20	0.000000	-69.696
(b)	20	0.000002	-69.631
(c)	50	0.000008	-69.402

Different starting points:

(a) V = .95 * V_{opt} + .05 * rand

(b) V = .90 * V_{opt} + .10 * rand

(c) V = rand(n, 2n)

Random Starting Point

Example (continued), $n = 60, \; m = 100$.

$z_{sdp} = -9600, 82, \quad z_{sdp+nonneg} = -172.19, z_{cop} = -69.75$

it	|b-A(X)|	f(x)
1	6.121227	1831.5750
5	0.021658	101.1745
10	0.002940	-43.4477
20	0.000147	-67.0989
30	0.000041	-68.7546
40	0.000015	-69.2360
50	0.000008	-69.4025

Starting point: $V0 = rand(n, 2n)$

More results

n	m	opt	found	$\|b - A(X)\|$
50	100	314.48	314.90	$4 \; 10^{-5}$
60	120	-266.99	-266.48	$4 \; 10^{-5}$
70	140	-158.74	-157.55	$3 \; 10^{-5}$
80	160	-703.75	-701.68	$5 \; 10^{-5}$
100	100	-659.65	-655.20	$8 \; 10^{-5}$

Starting point in all cases: rand(n,2n)
Inner iterations: 5
Outer iterations: 30

The code works on random instances. Now some more serious experiments.

Some experiments with Stable Set

$$\max\langle J, X\rangle \text{ such that } \mathrm{tr}(X) = 1,\ \mathrm{tr}(A_G X) = 0,\ X \in COP$$

Only two equations but many local optima.

Some experiments with Stable Set

$$\max\langle J, X\rangle \text{ such that } \mathrm{tr}(X) = 1,\ \mathrm{tr}(A_G X) = 0,\ X \in COP$$

Only two equations but many local optima. We consider a selection of graphs from the DIMACS collection. Computation times in the order of a few minutes.

name	n	ω	clique found
keller4	171	11	9
brock200-4	200	17	14
c-fat200-1	200	12	12
c-fat200-5	200	58	58
brock400-1	400	27	24
p-hat500-1	500	9	8

Sufficient condition for Non-copositivity

To show that $M \notin CP$, consider

$$\min\{x^T M x : e^T x = 1,\ x \geq 0\}$$

and try to solve this through

$$\min\{\langle M, X\rangle : e^T x = 1, \langle J, X\rangle = 1, \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \in COP\}.$$

Our method is local, but once we have feasible solution with negative value, we have a certificate for $M \notin CP$.

We apply this to get another heuristic for stable sets.

Stable sets - second approach

If we can show that for some integer t

$$Q = t(A + I) - J$$

is not copositive, then $\alpha(G) \geq t + 1.$

Stable sets - second approach

If we can show that for some integer t

$$Q = t(A + I) - J$$

is not copositive, then $\alpha(G) \geq t + 1$. Hence we consider $\min\{x^T Qx : e^T x = 1, x \geq 0\}$ and translate this into

$$\min\{\langle Q, X\rangle : e^T x = 1, \langle J, X\rangle = 1, \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \in COP\}.$$

If we find solution with value < 0, then we have certificate that $\alpha(G) \geq t + 1$.

Stable sets - second approach

If we can show that for some integer t

$$Q = t(A + I) - J$$

is not copositive, then $\alpha(G) \geq t + 1$. Hence we consider $\min\{x^T Q x : e^T x = 1, x \geq 0\}$ and translate this into

$$\min\{\langle Q, X\rangle : e^T x = 1, \langle J, X\rangle = 1, \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \in COP\}.$$

If we find solution with value < 0, then we have certificate that $\alpha(G) \geq t + 1$.

• Note that we prove existence of stable set of size $t + 1$, without actually providing such a set.

Some preliminary experiments

We look at easy instances, given by interval graphs, where stability number is known.

n	ω	# max cliques	new method
100	20	181	20
500	50	451	50
1000	100	901	100

Some preliminary experiments

We look at easy instances, given by interval graphs, where stability number is known.

n	ω	# max cliques	new method
100	20	181	20
500	50	451	50
1000	100	901	100

Some DIMACS graphs

name	n	ω	clique found
keller4	171	11	11
brock200-4	200	17	15
brock400-1	400	27	23
p-hat500-1	500	9	9

Generating Copositive Cuts

We can also use the heuristic test for non-copositivity to generate CP-cuts. Under strong duality assumptions we have:

$$\min\{\langle C, X\rangle : A(X) = b, X \in COP\} =$$

$$\max\{b^T y : C - A^T(y) = Z \in CP\}$$

From our convex QP-subproblem, we get multipliers y for the equations $A(X) = b$.

If our heuristic finds that $C - A^T(y) \notin CP$, it provides $v \geq 0$ such that $v^T(C - A^T(y))v < 0$.

Generating Copositive Cuts (2)

We add

$$\langle C - A^T(y), vv^T \rangle \geq 0$$

to the relaxed dual

$$\min\{b^T y : C - A^T(y) = Z \in PSD, \langle C - A^T(y), vv^T \rangle \geq 0\}$$

to get better relaxation.
This allows to estimate the quality of our heuristics.

Work in progress, currently no practical experiments.

Research Topics

- Explore the potential of copositive cuts

- Explore this approach in the context of (nonconvex) quadratic optimization with binary variables (heuristics and bounds)

- The complexity status of checking whether $M \in COP$ is open. An obvious observation:

$M \in COP \Leftrightarrow \exists X = (x_{ij})$ such that, with $Y = (x_{ij}^2), M = YY^T$.

This is a system of order 4 polynomial equations. Can you do better?

Last Slide

We have seen the power of copositivity.

Relaxations: The Parrilo hierarchy is computationally too expensive. Other way to approximate CP?

Heuristics: Unfortunately, the subproblem may have local solutions, which are not local minima for the original descent step problem.

The number of columns of V does not need to be larger than $\binom{n+1}{2}$, but for practical purposes, this is too large.

Further technical details in a forthcoming paper by I. Bomze, F. Jarre and F. R.: Quadratic factorization heuristics for copositive programming, technical report, (2008).