ORFE Operations Research & Financial Engineering

PRINCETON UNIVERSITY

Iterative LP and SOCP-based approximations to semidefinite and sum of squares programs

Georgina Hall
Princeton University

Joint work with:

Amir Ali Ahmadi (Princeton University)

Sanjeeb Dash (IBM)

PRINCETON UNIVERSITY

Semidefinite programming: definition

- A semidefinite program is an optimization problem of the form:

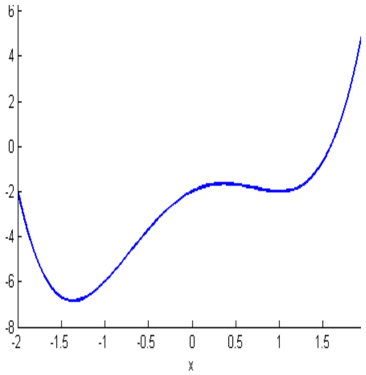
$$\min_{X \in S^{n\times n}} Tr(CX)$$
$$\text{s.t. } Tr(A_i X) = b_i, i = 1, \dots, m$$
$$X \succcurlyeq 0$$

Problem data: $C, A_i \in S^{n\times n}, b_i \in \mathbb{R}$.

PRINCETON UNIVERSITY

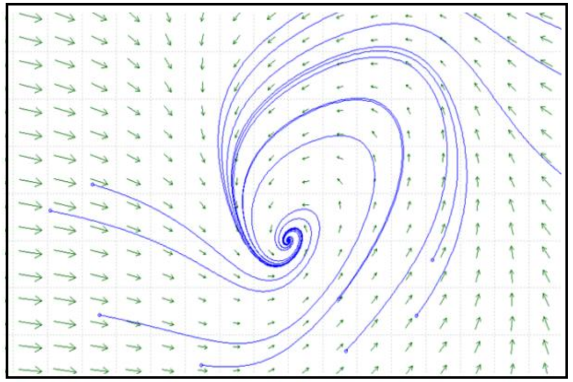
Semidefinite programming: application 1

Nonnegativity of a polynomial p of degree $2d$
$p(x)$ nonnegative
$p(x_1, \ldots, x_n) = z(x)^T Q z(x), Q \succcurlyeq 0$ ($z(x)$: monomial vector of degree d)
Polynomial optimization



$$\min_{x \in \mathbb{R}^n} p(x)$$
$$\Updownarrow$$
$$\max_{\gamma} \gamma$$
$$\text{s.t. } p(x) - \gamma \geq 0, \forall x$$
$$\Uparrow$$
$$\max_{\gamma} \gamma$$
$$\text{s.t. } p(x) - \gamma = z(x)^T Q z(x), Q \succcurlyeq 0$$

Example: Automated search for Lyapunov functions for dynamical systems



E.g., $V(x)$ nonnegative and $-\dot{V}(x)$ nonnegative
$\Rightarrow$ Global stabillity

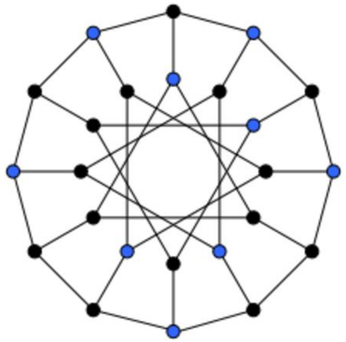
Semidefinite programming: application 2

Copositivity of a square matrix M

M copositive $\Leftrightarrow x^T M x \geq 0 \ \forall x \geq 0$

$$M = P + N$$
with $P \succcurlyeq 0, N \geq 0$

Stability number of a graph



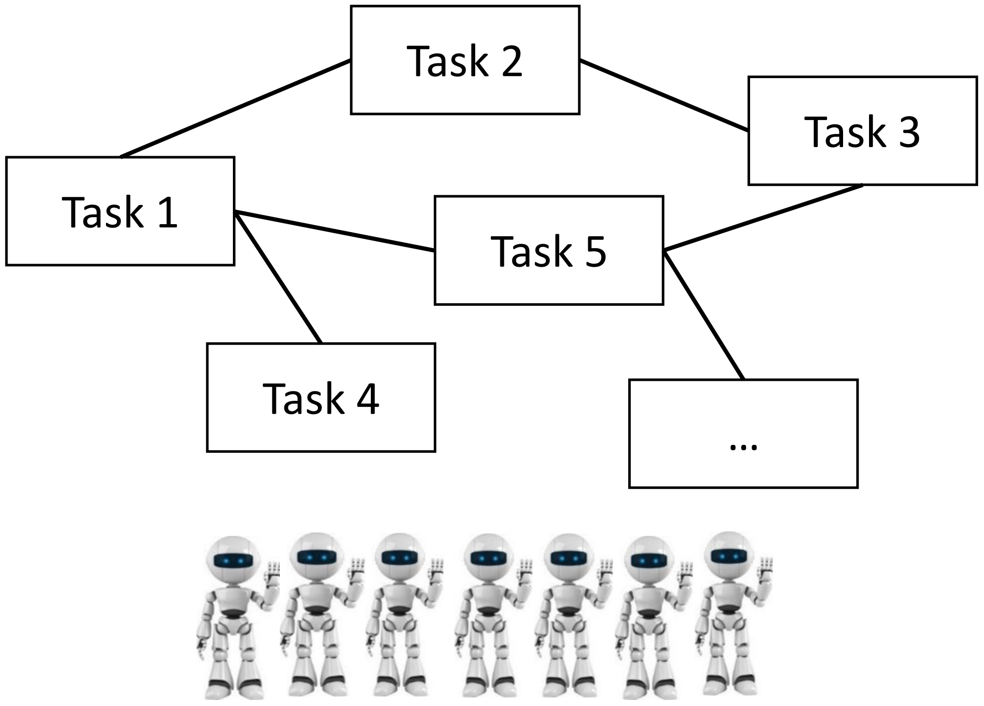
$$\alpha_G = \min \ \lambda$$
s.t. $\lambda(I + A) - J$ copositive

$$\Uparrow$$

$$\min \ \lambda$$
s.t. $\lambda(I + A) - J - \mathrm{N} \succcurlyeq 0, N \geq 0$

(De Klerk, Pasechnik)

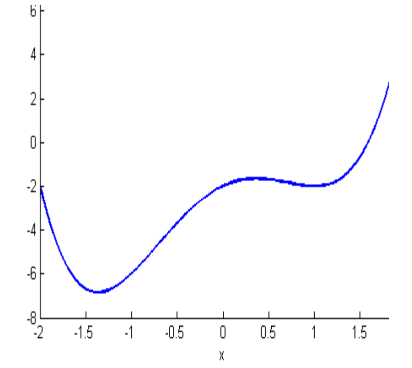
Example: task allocation for multi-agent systems

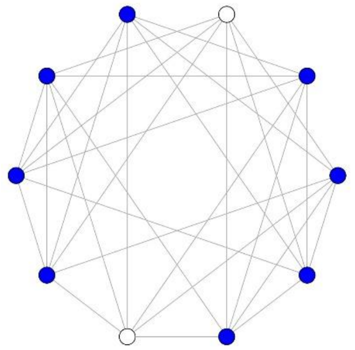


How many tasks can we accomplish simultaneously?

Semidefinite programming: pros and cons

- +: high-quality bounds for non convex problems.
- - : can take too long to solve if psd constraint is too big.

Polynomial optimization	
	$p(x) = z(x)^T Q z(x), Q \succcurlyeq 0,$ (p degree $2d$ and in n vars) Q : size $\binom{n+d}{d} \times \binom{n+d}{d}$

Stability number of a graph	
	$P := \lambda(I + A) - J - N \succcurlyeq 0$ P : size $n \times n$ (n: # of nodes in the graph)

- Typical laptop cannot minimize a degree-4 polynomial in more than a couple dozen variables. (PC: 3.4 GHz, 16 Gb RAM)

LP and SOCP-based inner approximations (1/2)

Problem	Idea	Candidate (Ahmadi, Majumdar)
SDPs do not scale well	Replace psd cone by LP or SOCP representable cones	LP: A dd SOCP: A sdd

- A is diagonally dominant (dd) if

$$a_{ii} \geq \sum_{j \neq i} |a_{ij}|, \forall i.$$

- A is scaled diagonally dominant (sdd) if ∃ a diagonal matrix $D > 0$ s.t. DAD is dd.

$$DD_n \subseteq SDD_n \subseteq PSD_n$$

LP and SOCP-based inner approximations (2/2)

Replacing psd cone by dd/sdd cones:

- +: fast bounds
- - : not always as good quality (compared to SDP)

Iteratively construct a sequence of improving LP/SOCP-based cones

Initialization: Start with the DD/SDD cone

Method 1: Cholesky change of basis

Method 2: Column Generation

Method 1:Cholesky change of basis (1/4)

$$p(x) = x_1^4 - 6x_1^3x_2 + 2x_1^3x_3 + 6x_1^2x_3^2 + 9x_1^2x_2^2 - 6x_1^2x_2x_3 - 14x_1x_2x_3^2 + 4x_1x_3^3 + 5x_3^4 - 7x_2^2x_3^2 + 16x_2^4$$

$$p(x) = z^T(x)Qz(x)$$

$$Q = \begin{pmatrix} 1 & -3 & 0 & 1 & 0 & 2 \\ -3 & 9 & 0 & -3 & 0 & -6 \\ 0 & 0 & 16 & 0 & 0 & -4 \\ 1 & -3 & 0 & 2 & -1 & 2 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 2 & -6 & 4 & 2 & 0 & 5 \end{pmatrix}$$

psd but not dd

$$z(x) = (x_1^2, x_1x_2, x_2^2, x_1x_3, x_2x_3, x_3^2)^T$$

$$p(x) = \tilde{z}^T(x) \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4. \end{pmatrix} \tilde{z}(x)$$

dd in the “right basis”

$$\tilde{z}(x) = (2x_1^2 - 6x_1x_2 + 2x_1x_3 + 2x_3^2, x_1x_3 - x_2x_3, x_2^2 - \frac{1}{4}x_3^2)^T$$

Goal: iteratively improve on basis $z(x)$.

Method 1: Cholesky change of basis (2/4)

SDP to solve

$\min Tr(CX)$
s.t. $Tr(A_i X) = b_i,$
$X \succcurlyeq 0$

Initialize

Solve: $\min Tr(CX)$
s.t. $Tr(A_i X) = b_i,$
X dd/sdd

Step 1

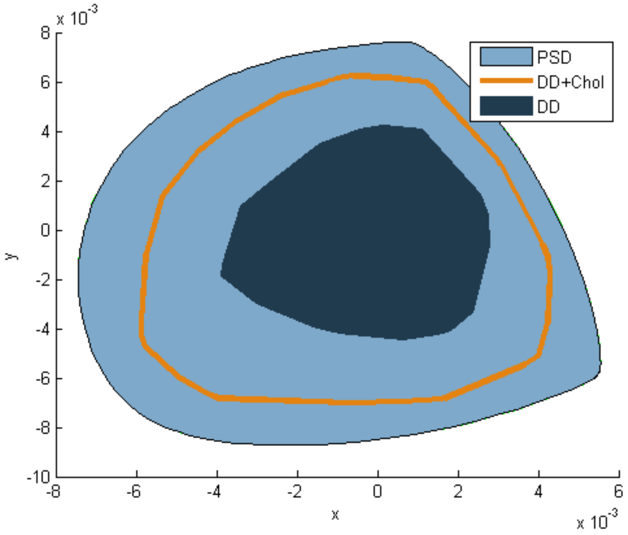
Compute:
$U_k = chol(X^*)$

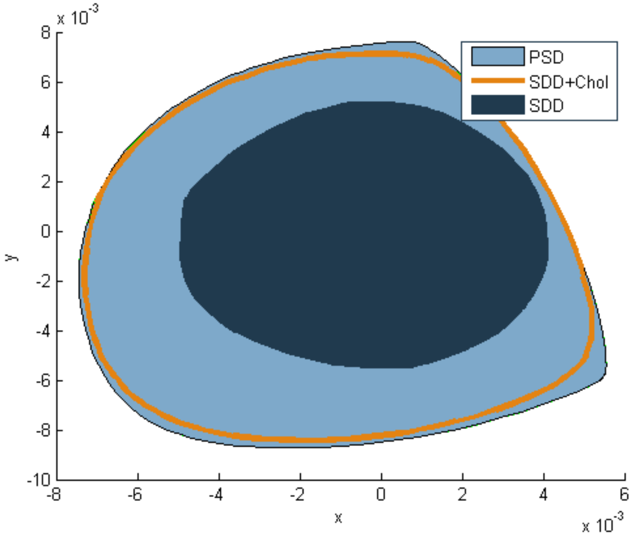
Step 2

Solve: $\min Tr(CX)$
s.t. $Tr(A_i X) = b_i,$
$X = U_k^T Q U_k,$ Q dd/sdd

$\boldsymbol{k := k+1}$

One iteration of this method
on $I + xA + yB$,
A, B fixed 10×10,
generated randomly

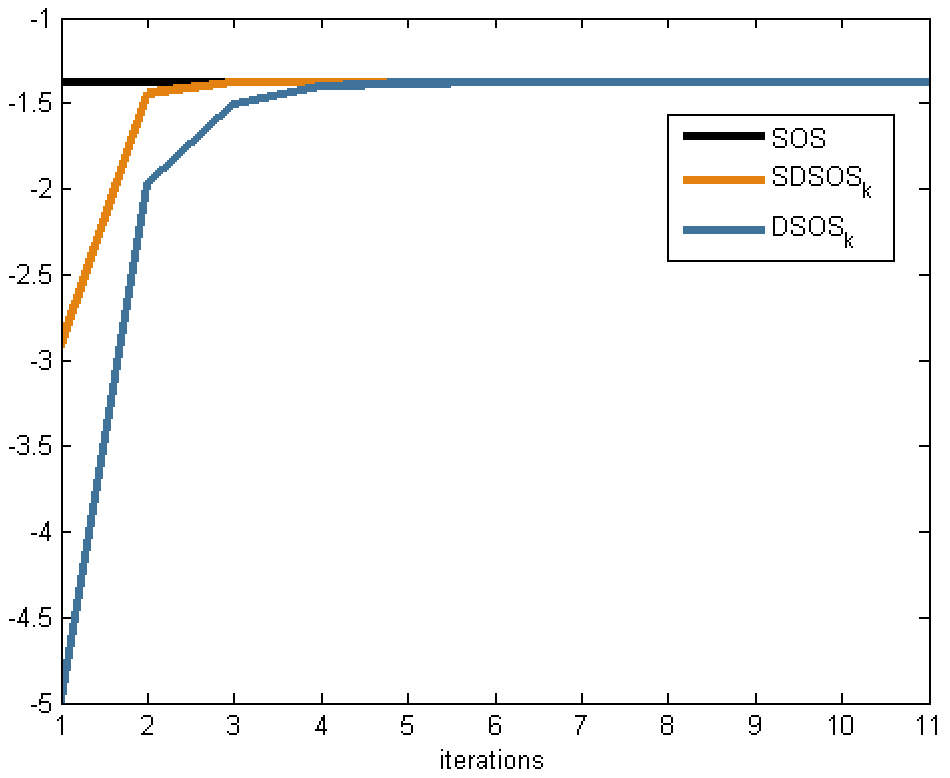




PRINCETON UNIVERSITY

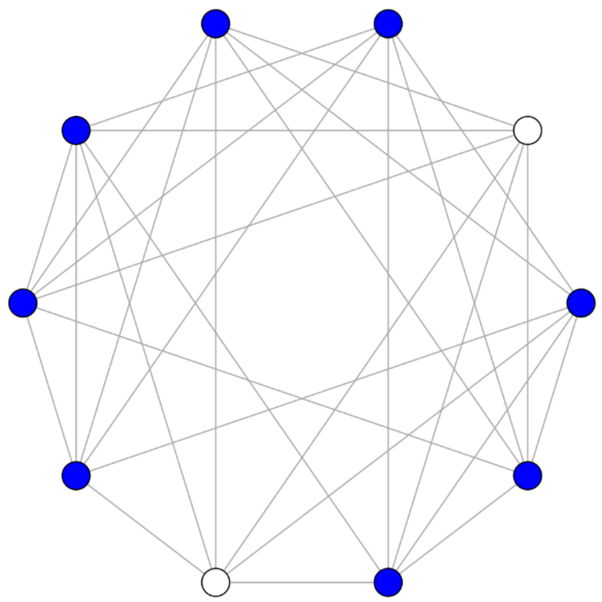
Method 1: Cholesky change of basis (3/4)

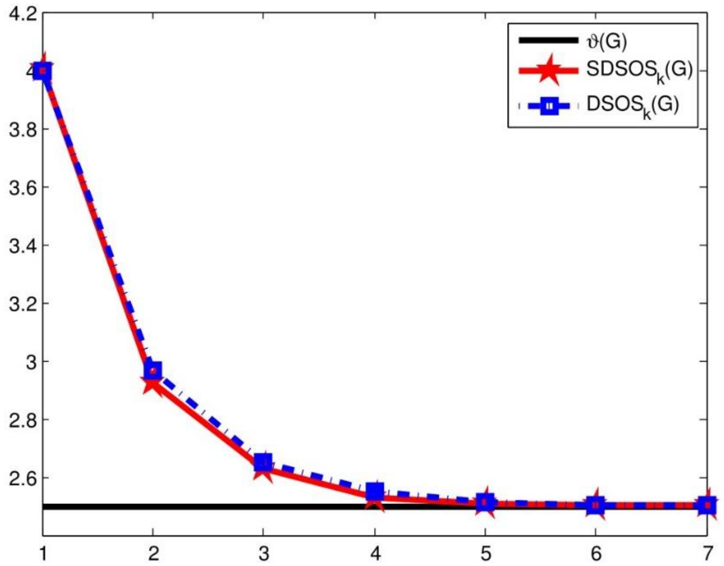
- Example 1: minimizing a degree-4 polynomial in 4 variables



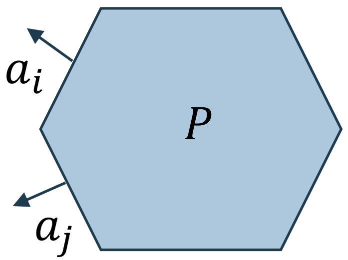
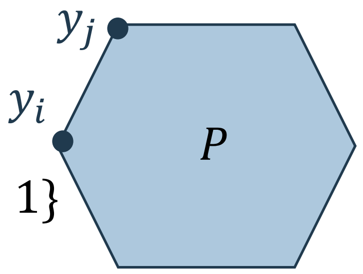
Method 1: Cholesky change of basis (4/4)

- Example 2: stability number of the Petersen graph





Method 2: Column generation

Polytope P	
Facet description $P = \{x \mid Ax \le b\}$, where $A = \begin{bmatrix} a_1^T \\ \vdots \\ a_m^T \end{bmatrix}$, $b \in \mathbb{R}^m$ 	**Corner description** $P = \{\sum_i \alpha_i y_i \mid \alpha_i \ge 0, \sum_i \alpha_i = 1\}$ 

	DD cone	SDD cone
Facet description	$DD_n = \{M \mid M_{ii} \ge \sum_{j \neq i} \lvert M_{ij} \rvert, \forall i\}$	$SDD_n = \{M \mid \exists$ diagonal $D > 0$ s.t. DMD dd$\}$
Extreme ray description	$DD_n = \left\{ \sum_i v_i \, \alpha_i \, v_i^T \mid \alpha_i \ge 0 \right\}$ v_i has at most two nonzero components $= \pm 1$ $V_{DD} := \{v_i\}$	$SDD_n = \left\{ \sum_i V_i \, \Lambda_i \, V_i^T \mid \Lambda_i \succcurlyeq 0 \right\}$ V_i is all zeros except for one component in each row=1 $V_{SDD} := \{V_i\}$

Method 2: Column Generation

- Main idea: **add new atoms** to the DD_n/SDD_n cones.

Initial SDP

$$\max_{y \in \mathbb{R}^m} b^T y$$
$$s.t.\ \ C - \sum_{i=1}^{m} y_i A_i \succcurlyeq 0$$

Inner approximate with DD_n

$$\max_{y \in \mathbb{R}^m} b^T y$$
$$s.t.\ \ C - \sum_{i=1}^{m} y_i A_i = \sum \alpha_i v_i v_i^T$$
$$\alpha_i \geq 0, v_i \in V_{DD}$$

Take the dual

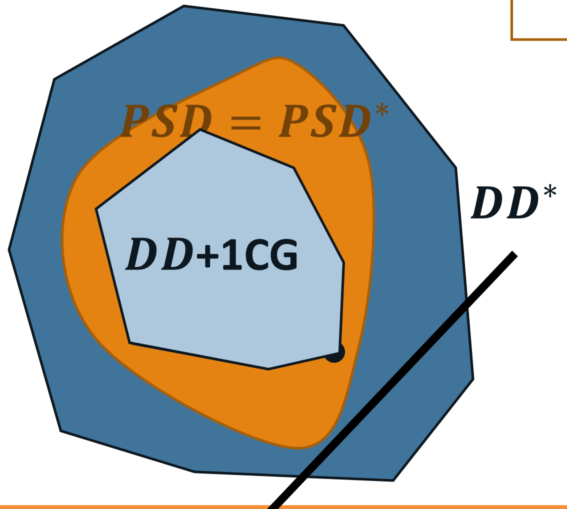
$$\min_{X \in S^n} tr(CX)$$
$$s.t.\ \ tr(A_i X) = b_i$$
$$v_i^T X v_i \geq 0$$

Find a new "atom"

Pick v s.t. $v^T X v < 0$.

Add v to primal DD inner approx.

$$\max_{y \in \mathbb{R}^m} b^T y$$
$$s.t.\ \ C - \sum_{i=1}^{m} y_i A_i = \sum \alpha_i v_i v_i^T + \alpha v v^T$$
$$\alpha_i \geq 0, v_i \in V_{DD}, \alpha \geq 0$$



Method 2: Column Generation

- <u>Main idea</u>: **add new atoms** to the DD_n/SDD_n cones.

Initial SDP

$$\max_{y\in\mathbb{R}^m} b^T y \quad s.t.\ \ C - \sum_{i=1}^m y_i A_i \succcurlyeq 0$$

Inner approximate with SDD_n

$$\max_{y\in\mathbb{R}^m} b^T y \quad s.t.\ \ C - \sum_{i=1}^m y_i A_i = \sum \boldsymbol{V_i \Lambda_i V_i^T}$$

$$\boldsymbol{\Lambda_i \succcurlyeq 0, V_i \in V_{SDD}}$$

Take the dual

$$\min_{X\in S^n} tr(CX) \quad s.t.\ \ tr(A_i X) = b_i$$

$$\boldsymbol{V_i^T X V_i \succcurlyeq 0}$$

Find a new "atom"

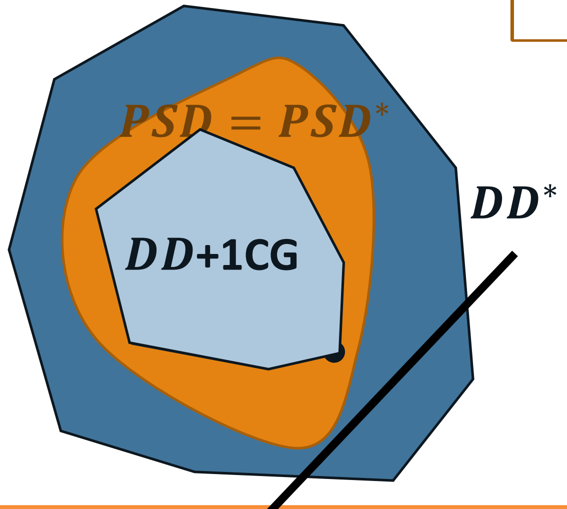
Pick $V \in \mathbb{R}^{n\times 2}$

s.t. $\boldsymbol{V^T X V \not\succcurlyeq 0}$.

Add V to primal SDD inner approx.

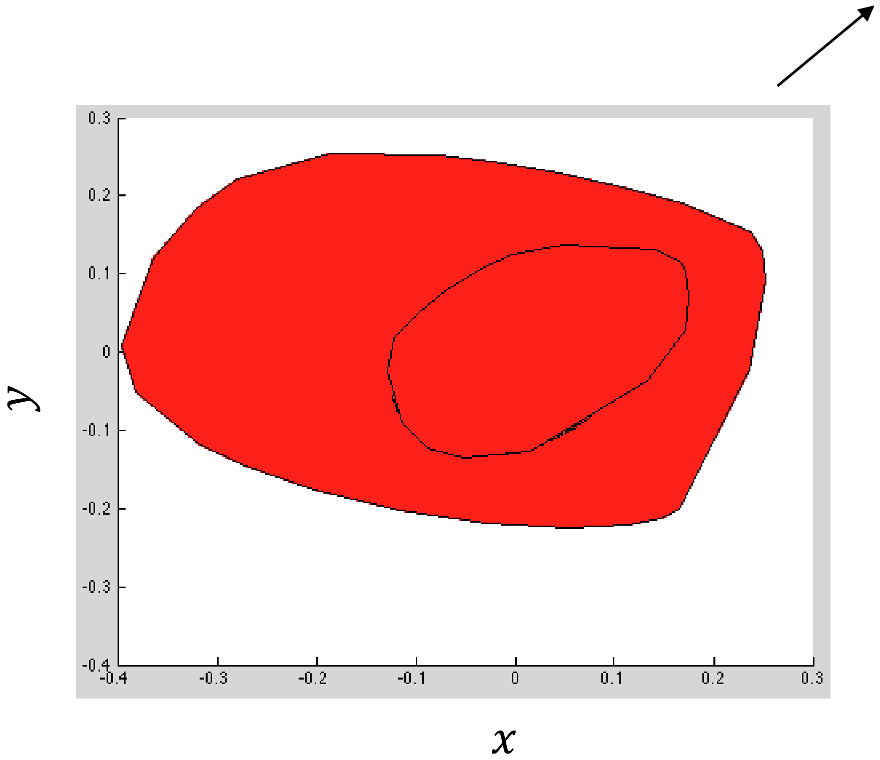
$$\max_{y\in\mathbb{R}^m} b^T y \quad s.t.\ \ C - \sum_{i=1}^m y_i A_i = \sum \boldsymbol{V_i \Lambda_i V_i^T} + \boldsymbol{V \Lambda V^T}$$

$$\boldsymbol{\Lambda_i \succcurlyeq 0, V_i \in V_{SDD}, \Lambda \succcurlyeq 0}$$



Method 2: Column Generation

Five iterations of column generation for $I + xA + yB$, A, B fixed 10×10, generated randomly



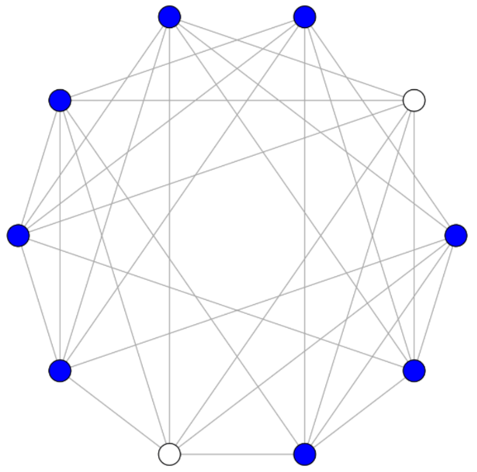
- Column generation method: **adds** atoms to the initial cones
- Cholesky change of basis method: "**rotates**" existing atoms

$$v_i \to U^T v_i, \text{ where } U = chol(X^*)$$

PRINCETON UNIVERSITY

Method 2: Column Generation (3/3)

- Example 1: stability number of the Petersen graph

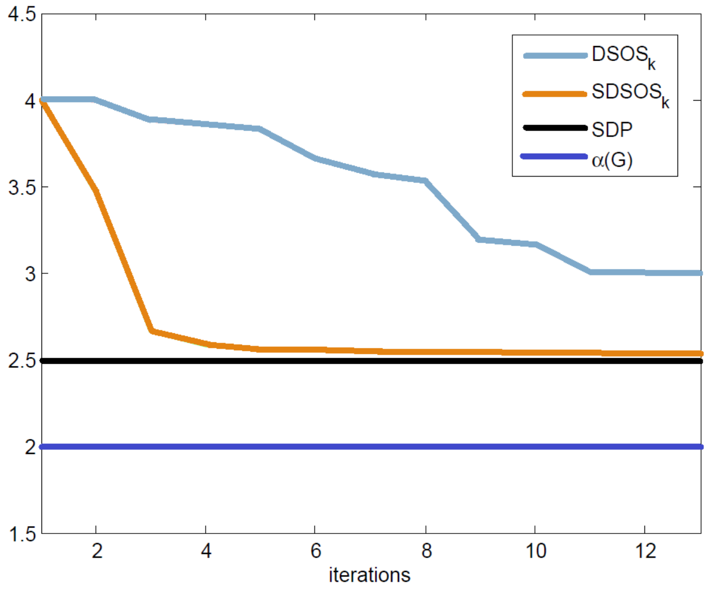


Petersen Graph

Upper bound on the stability number through **SDP**:

$$\min \lambda$$
$$\text{s.t. } \lambda(I + A) - J - N \succcurlyeq 0, N \geq 0$$

Apply Column Generation technique to this SDP



Method 2: Column Generation (3/3)

- Example 2: minimizing a degree-4 polynomial

	$n = 15$		$n = 20$		$n = 25$		$n = 30$		$n = 40$	
	bd	t(s)	bd	t(s)	bd	t(s)	bd	t(s)	bd	t(s)
DSOS	-10.96	0.38	-18.01	0.74	-26.45	15.51	-36.52	7.88	-62.30	10.68
SDSOS	-10.43	0.53	-17.33	1.06	-25.79	8.72	-36.04	5.65	-61.25	18.66
$DSOS_k$	-5.57	31.19	-9.02	471.39	-20.08	600	-32.28	600	-35.14	600
SOS	-3.26	5.60	-3.58	82.22	-3.71	1068.66	NA	NA	NA	NA

PRINCETON UNIVERSITY

Main messages

- Can **improve** on initial DD and SDD inner approximations to the PSD cone.
- This is done with **LPs and SOCPs** and **iteratively**.
- We presented two methods for this: **cholesky change of basis** and **column generation.**

Thank you for listening

Questions?

Want to learn more?
http://scholar.princeton.edu/ghall