

Semidefinite Programming

Grothendieck Inequalities

Fernando Mário de Oliveira Filho



Campos do Jordão, 21 November 2013

Available at: www.ime.usp.br/~fmario under talks

A class of optimization problems

$x, y \in \mathbb{R}^n$, then $\textcolor{red}{x} \cdot y = x_1y_1 + \cdots + x_ny_n$

$$S^{n-1} = \{ x \in \mathbb{R}^n : x \cdot x = 1 \}$$

$r \geq 1$ integer, $A \in R^{m \times n}$

$$\text{SDP}_r(A) = \max_{\textcolor{red}{x_i}, \textcolor{red}{y_j} \in S^{r-1}} \sum_{i=1}^m \sum_{j=1}^n A_{ij} \textcolor{red}{x_i} \cdot \textcolor{red}{y_j}$$

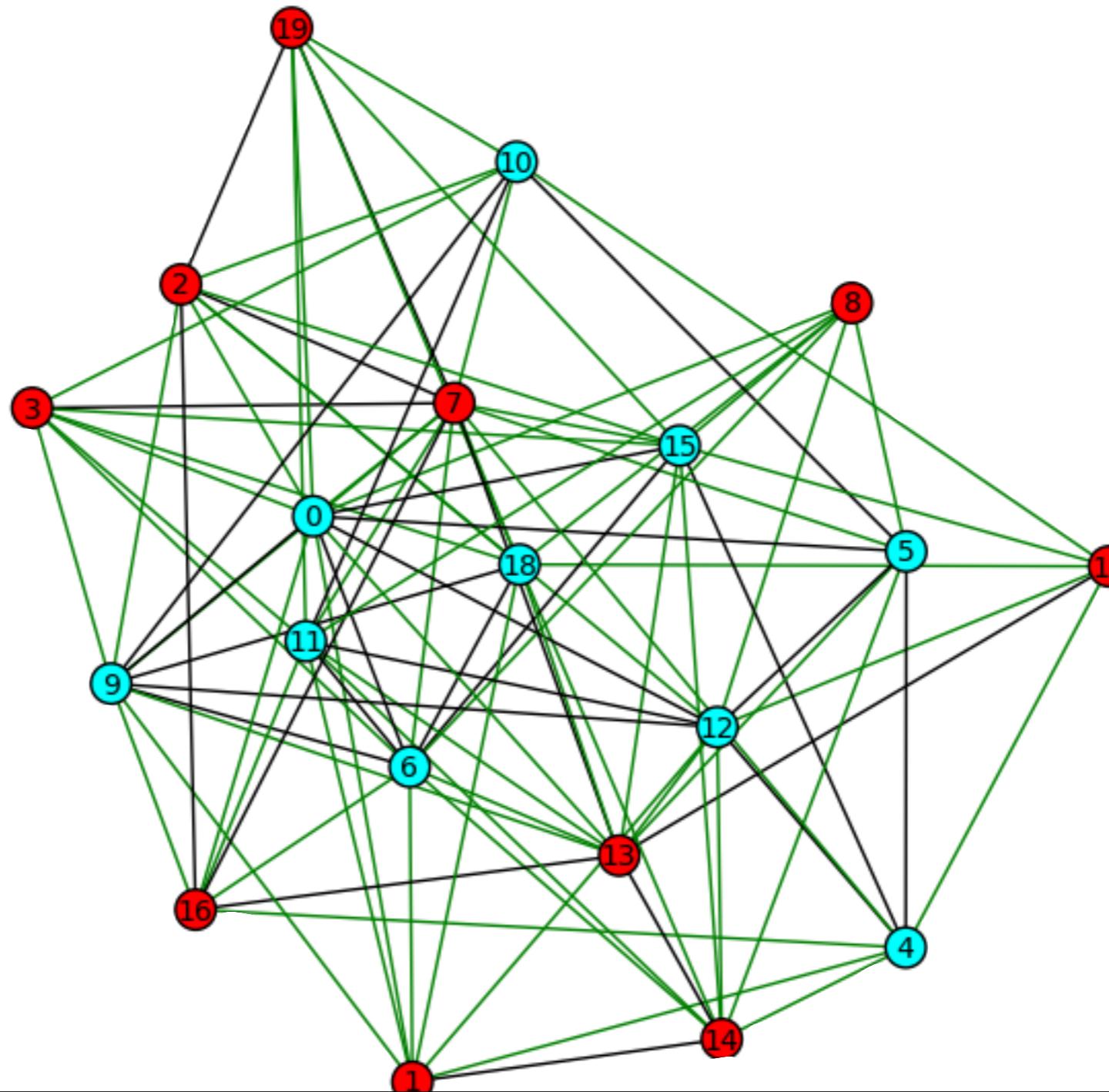
$$\text{SDP}_\infty(A) = \text{SDP}_{m+n}(A)$$

$$\text{SDP}_1(A) \leq \text{SDP}_\infty(A)$$

Maximum cuts in graphs...

Given: Graph $G = (V, E)$, weights $w: E \rightarrow \mathbb{R}_+$

Find: Set $U \subseteq V$ maximizing $w(\delta(U))$



And how to find one

$$V = \{1, \dots, n\}, A \in \mathbb{R}^{n \times n}$$

$$A_{ij} = \begin{cases} -w_{ij} & \text{if } i \neq j \\ w(\delta(i)) & \text{otherwise} \end{cases}$$

$$\text{maxcut} = \max_{x_i \in \{-1, 1\}} \frac{1}{4} \sum_{i,j=1}^n A_{ij} x_i x_j$$

If A is PSD, then $x_i = y_i$ in $\text{SDP}_r(A)$

A is Laplacian \implies A is PSD \implies $\text{maxcut} = \frac{1}{4} \text{SDP}_1(A)$

The r -vector model

Two sets $U = \{1, \dots, m\}$ and $V = \{1, \dots, n\}$ of particles

$$A_{ij} = \begin{cases} 0 & \text{if no interaction} \\ > 0 & \text{if ferromagnetic interaction} \\ < 0 & \text{if antiferromagnetic interaction} \end{cases}$$

Particles have vector-valued spin $\color{red}{x_i, y_j \in S^{r-1}}$

Energy: Hamiltonian $H(f) = - \sum_{i=1}^m \sum_{j=1}^n A_{ij} x_i \cdot y_j$

Ground state: minimizes total energy = $\color{red}{\text{SDP}_r(A)}$

Semidefinite programming

$$\text{SDP}_r(A) = \max_{\substack{x_i, y_j \in S^{r-1}}} \sum_{i=1}^m \sum_{j=1}^n A_{ij} x_i \cdot y_j$$

$A \in \mathbb{R}^{n \times n}$ is PSD
of rank k $\iff \exists x_1, \dots, x_n \in \mathbb{R}^k$ s.t. $A_{ij} = x_i \cdot x_j$

$$\text{SDP}_r(A) = \max \quad \sum_{i=1}^m \sum_{j=1}^n A_{ij} S_{ij}$$
$$Y = \begin{pmatrix} R & S \\ S^T & T \end{pmatrix} \succeq 0$$

hard constraint

$$R_{ii} = 1, T_{ii} = 1$$

$$R \in \mathbb{R}^{m \times m}, S \in \mathbb{R}^{m \times n}, T \in \mathbb{R}^{n \times n}$$

Y has rank r

Gram matrix

Semidefinite programming

Easy consequence:

If $A \in \mathbb{R}^{m \times n}$, then $\text{SDP}_\infty(A) = \text{SDP}_{m+n}(A)$

Complexity:

- $r = 1$: NP-hard (from MAXCUT)
- $r < m + n$: good question...
- $r = m + n$: polytime (ellipsoid method, with caveat)

$$\text{SDP}_1(A) \leq \text{SDP}_\infty(A) \leq K \cdot \text{SDP}_1(A)$$

Hard

Easy

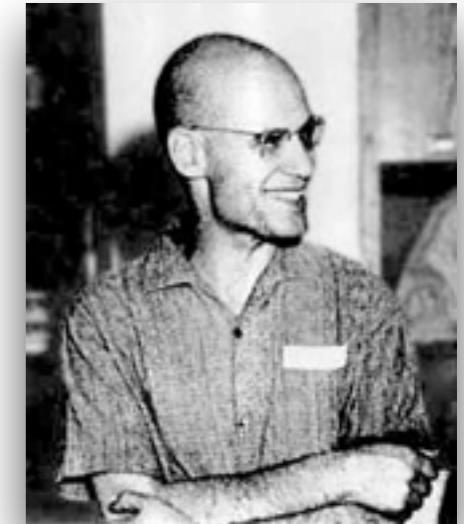


integrality gap

Grothendieck inequality

Theorem. Exists a constant K such that

$$\text{SDP}_1(A) \leq \text{SDP}_\infty(A) \leq K \cdot \text{SDP}_1(A)$$



Grothendieck
1953 (in SP)

Matrix is...	Bound	Inverse bound	Author
Any	< 1.782...	> 0.561...	Krivine (1977), Naor, Makarychev x2
Positive semidefinite	1.570...	0.639...	Nesterov (1998)
Laplacian	1.138...	0.878...	Goemans & Williamson (1995)

Hyperplane rounding

Solution $x_i, y_j \in S^{m+n-1}$
of $\text{SDP}_\infty(A)$



Solution $\alpha_i, \beta_j \in \{-1, +1\}$
of $\text{SDP}_1(A)$

Rounding procedure:

1. Pick uniformly at random: $z \in S^{m+n-1}$
2. Set $\alpha_i = \text{sgn}(x_i \cdot z), \beta_j = \text{sgn}(y_j \cdot z)$

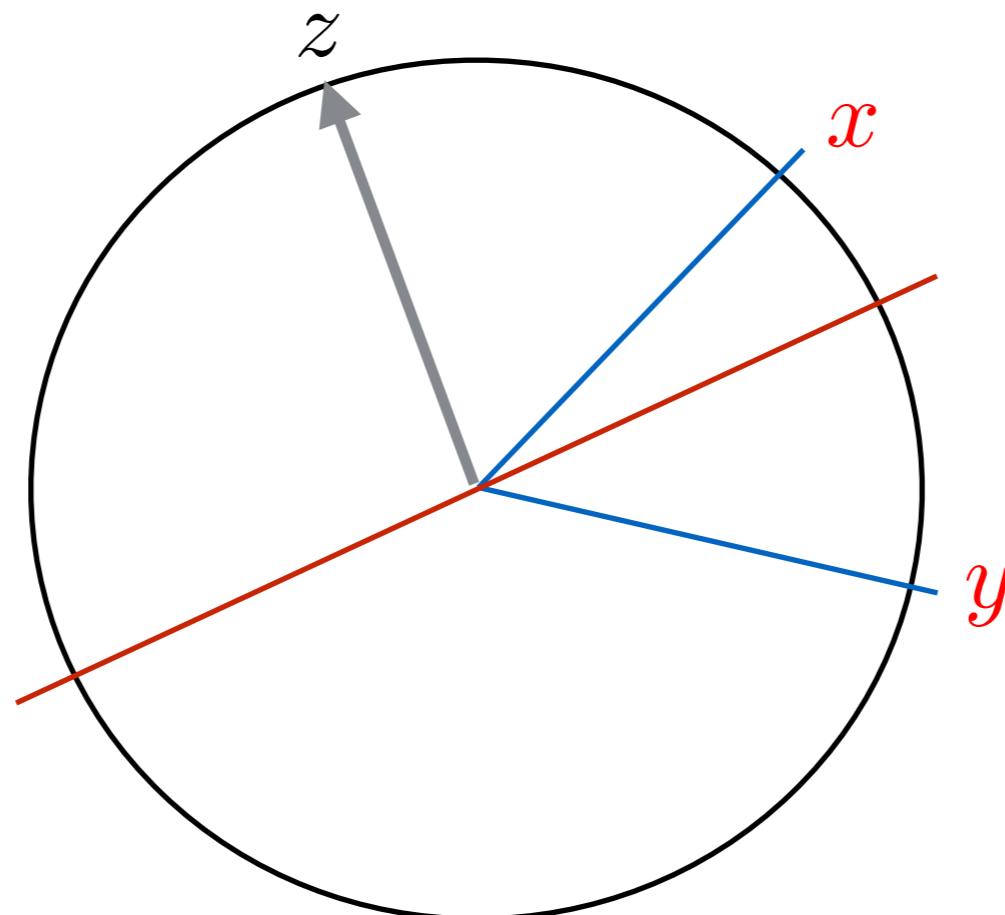
$$\mathbb{E} \left[\sum_{i=1}^m \sum_{j=1}^n A_{ij} \alpha_i \beta_j \right] = \sum_{i=1}^m \sum_{j=1}^n A_{ij} \mathbb{E}[\alpha_i \beta_j]$$

Grothendieck's identity

$x, y \in S^{N-1}$

z picked uniformly at random in S^{N-1}

$$\mathbb{E}[\operatorname{sgn}(x \cdot z) \operatorname{sgn}(y \cdot z)] = \frac{2}{\pi} \arcsin x \cdot y$$



Hyperplane rounding

Solution $x_i, y_j \in S^{m+n-1}$
of $\text{SDP}_\infty(A)$


rounding

Solution $\alpha_i, \beta_j \in \{-1, +1\}$
of $\text{SDP}_1(A)$

Rounding procedure:

1. Pick uniformly at random: $z \in S^{m+n-1}$
2. Set $\alpha_i = \text{sgn}(x_i \cdot z), \beta_j = \text{sgn}(y_j \cdot z)$

$$\mathbb{E} \left[\sum_{i=1}^m \sum_{j=1}^n A_{ij} \alpha_i \beta_j \right] = \sum_{i=1}^m \sum_{j=1}^n A_{ij} \mathbb{E}[\alpha_i \beta_j]$$

$$= \frac{2}{\pi} \sum_{i=1}^m \sum_{j=1}^n A_{ij} \arcsin(x_i \cdot y_j) \quad ? \quad \frac{2}{\pi} \sum_{i=1}^m \sum_{j=1}^n A_{ij} x_i \cdot y_j$$

Krivine's lemma

Given:

Unit vectors $x_1, \dots, x_m, y_1, \dots, y_m$

There are:

Unit vectors $x'_1, \dots, x'_m, y'_1, \dots, y'_m \in S^{m+n-1}$ with

$$\arcsin(x'_i \cdot y'_j) = \ln(1 + \sqrt{2})x_i \cdot y_j$$

New rounding procedure:

1. Pick uniformly at random: $z \in S^{m+n-1}$
2. Compute x'_i and y'_j
3. Set $\alpha_i = \operatorname{sgn}(x'_i \cdot z)$ and $\beta_j = \operatorname{sgn}(y'_j \cdot z)$

Krivine's bound

$$\arcsin(x'_i \cdot y'_j) = \ln(1 + \sqrt{2})x_i \cdot y_j$$

$$\mathbb{E} \left[\sum_{i=1}^m \sum_{j=1}^n A_{ij} \alpha_i \beta_j \right] = \sum_{i=1}^m \sum_{j=1}^n A_{ij} \mathbb{E}[\alpha_i \beta_j]$$

$$\stackrel{\text{AI}}{=} \frac{2}{\pi} \sum_{i=1}^m \sum_{j=1}^n A_{ij} \arcsin(x'_i \cdot y'_j)$$

$$= \frac{2}{\pi} \ln(1 + \sqrt{2}) \sum_{i=1}^m \sum_{j=1}^n A_{ij} x_i \cdot y_j$$

$$= \frac{2}{\pi} \ln(1 + \sqrt{2}) \mathbf{SDP}_\infty(A)$$

Proof of Krivine's lemma

$$A, B \in \mathbb{R}^{m \times n}$$

Hadamard product: $(A \odot B)_{ij} = A_{ij}B_{ij}$

If $A, B \in \mathcal{S}_n$ are PSD, then $A \odot B$ is PSD

Unit vectors $x_1, \dots, x_m, y_1, \dots, y_m$

Unit vectors $x'_1, \dots, x'_m, y'_1, \dots, y'_m \in S^{m+n-1}$ with

$$x'_i \cdot y'_j = \sin(\textcolor{red}{c} x_i \cdot y_j)$$

$$A = \begin{pmatrix} R & S \\ S^T & T \end{pmatrix} \succeq 0 \quad \begin{aligned} R_{ij} &= x_i \cdot x_j \\ S_{ij} &= x_i \cdot y_j \\ T_{ij} &= y_i \cdot y_j \end{aligned}$$

Proof (cont'd)

$$\sin(\textcolor{red}{c}t) = \sum_{k=0}^{\infty} (-1)^k \frac{\textcolor{red}{c}^{2k+1}}{(2k+1)!} t^{2k+1}$$

$$\sinh(\textcolor{red}{c}t) = \sum_{k=0}^{\infty} \frac{\textcolor{red}{c}^{2k+1}}{(2k+1)!} t^{2k+1}$$

$$\xi_k = (-1)^k \sqrt{\frac{\textcolor{red}{c}^{2k+1}}{(2k+1)!}}, \quad \eta_k = \sqrt{\frac{\textcolor{red}{c}^{2k+1}}{(2k+1)!}}$$

$$\textcolor{red}{C}_k = \begin{pmatrix} X_k & Y_k \\ Y_k^T & Z_k \end{pmatrix} \succeq 0$$
$$(X_k)_{ij} = \xi_k \xi_k$$
$$(Y_k)_{ij} = \xi_k \eta_k$$
$$(Z_k)_{ij} = \eta_k \eta_k$$

Finishing...

$$\textcolor{red}{A} = \begin{pmatrix} R & S \\ S^T & T \end{pmatrix} \succeq 0 \quad \begin{array}{ll} R_{ij} = x_i \cdot x_j & T_{ij} = y_i \cdot y_j \\ S_{ij} = x_i \cdot y_j & \end{array}$$

$$\textcolor{red}{C}_k = \begin{pmatrix} X_k & Y_k \\ Y_k^T & Z_k \end{pmatrix} \succeq 0 \quad \begin{array}{ll} (X_k)_{ij} = \xi_k \xi_k & (Z_k)_{ij} = \eta_k \eta_k \\ (Y_k)_{ij} = \xi_k \eta_k & \end{array}$$

$$\sum_{k=0}^{\infty} C_k \odot A^{\odot(2k+1)} \succeq 0$$

Decomposition $x'_1, \dots, x'_m, y'_1, \dots, y'_m$ with

$$x'_i \cdot y'_j = \sin(\textcolor{red}{c} x_i \cdot y_j)$$

$$x'_i \cdot x'_i = y'_j \cdot y'_j = \sinh(\textcolor{red}{c})$$

Unit vectors $\implies \textcolor{red}{c} = \sinh^{-1}(1) = \ln(1 + \sqrt{2})$

Krivine vs. Goemans-Williamson

Simple approximation algorithm:

Put $x \in V$ in \mathcal{U} ind. with probability $1/2$

Estimates of integrality gap of SDP:

A Laplacian matrix of weighted graph

Goemans-Williamson (1995):

$$\text{SDP}_1(A) \geq 0.878 \dots \text{SDP}_\infty(A)$$

Krivine (1977):

$$\text{SDP}_1(A) \geq 0.56 \dots \text{SDP}_\infty(A)$$

Summary and coming soon...

Grothendieck inequalities:

- Many applications: physics, quantum information theory, combinatorial optimization, Banach spaces, tensor products...
- Algorithmic interpretation: due to Alon and Naor.
- Generalization to higher ranks (recent development)