Introduction to Semidefinite Programming I: Basic properties and variations on the Goemans-Williamson approximation algorithm for max-cut

MFO seminar on Semidefinite Programming

Monique Laurent - CWI, Amsterdam, and Tilburg University

May 30, 2010
Positive semidefinite matrices

**Definition:** For a symmetric $n \times n$ matrix $X$, the following conditions are equivalent:

1. $X$ is *positive semidefinite* (written $X \succeq 0$) if all eigenvalues of $X$ are nonnegative.
2. $u^T X u \geq 0$ for all $u \in \mathbb{R}^n$.
3. $X = UU^T$ for some matrix $U \in \mathbb{R}^{n \times p}$.
4. For some vectors $v_1, \ldots, v_n \in \mathbb{R}^p$, $X_{ij} = v_i^T v_j$ ($i, j \in [n]$). Say that $X$ is the *Gram matrix* of the $v_i$’s.
5. All principal minors of $X$ are nonnegative.

**Definition:** $X$ is *positive definite* (written $X \succ 0$) if all eigenvalues of $X$ are positive.
Notation

- $S_n$: the space of $n \times n$ symmetric matrices.
- $S_n^+$: the cone of positive semidefinite matrices.
- $S_n^{++}$: the cone of positive definite matrices.
- $S_n^{++}$ is the interior of the cone $S_n^+$.

Trace inner product on $S_n$:

$$A \cdot B = \text{Tr}(A^T B) = \sum_{i,j=1}^{n} A_{ij} B_{ij}$$

The PSD cone is self-dual: For $X \in S_n$,

$$A \in S_n^+ \iff A \cdot B \geq 0 \quad \forall B \in S_n^+$$
Primal/dual semidefinite programs

Given matrices $C, A_1, \ldots, A_m \in S_n$ and a vector $b \in \mathbb{R}^m$

**Primal SDP:**

\[
p^* := \max_X C \cdot X \text{ such that } A_j \cdot X = b_j \ (j = 1, \ldots, m), \ X \succeq 0
\]

**Dual SDP:**

\[
d^* := \min_y b \cdot y \text{ such that } \sum_{j=1}^m y_j A_j - C \succeq 0
\]

**Weak duality:** $p^* \leq d^*$

**Pf:** If $X$ is *primal feasible* and $y$ is *dual feasible*, then

\[
0 \leq \left( \sum_{j=1}^m y_j A_j - C \right) \cdot X = \sum_{j} y_j (A_j \cdot X) - C \cdot X = b \cdot y - C \cdot X
\]
Analogy between LP and SDP

Given vectors $c, a_1, \ldots, a_m \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$

**Primal/dual LP:**

$$\max_x c \cdot x \text{ such that } a_j \cdot x = b_j \ (\forall j \leq m), \ x \in \mathbb{R}_+^n$$

$$\min_y b \cdot y \text{ such that } \sum_{j=1}^m y_j a_j - c \geq 0$$

**Primal SDP:**

$$\max_X C \cdot X \text{ such that } A_j \cdot X = b_j \ (j = 1, \ldots, m), \ X \in S_n^+$$

- SDP is the analogue of LP, replacing $\mathbb{R}_+^n$ by $S_n^+$.
- Get LP when $C, A_j$ are diagonal matrices.
Strong duality: $p^* = d^*$?

Strong duality holds for LP, but we need some regularity condition (e.g., Slater condition) to have strong duality for SDP!

**Primal (P) / Dual (D) SDP’s:** $p^* \leq d^*$

(P) $p^* = \sup C \cdot X$ s.t. $A_j \cdot X = b_j$ ($j = 1, \ldots, m$), $X \succeq 0$

(D) $d^* = \inf b \cdot y$ s.t. $\sum_{j=1}^m y_j A_j - C \succeq 0$

**Strong duality Theorem:**

1. If (P) is strictly feasible ($\exists X \succ 0$ feasible for (P)) and bounded ($p^* < \infty$), then $p^* = d^*$ and (D) attains its infimum.

2. If (D) is strictly feasible ($\exists y$ with $\sum_j y_j A_j - C \succ 0$) and bounded ($d^* > -\infty$), then $p^* = d^*$ and (P) attains its supremum.
Proof of 2. Assume \( d^* \in \mathbb{R} \) and \( \sum_j \tilde{y}_j A_j - C \succ 0 \) \( \exists \tilde{y} \)

\[
p^* = \max_{X \succeq 0} \quad C \cdot X \\
\geq d^* = \inf_y \quad b \cdot y \\
A_j \cdot X = b_j \\
\sum_j y_j A_j - C \succeq 0
\]

**Goal:** There exists \( X \) feasible for (P) with \( C \cdot X \geq d^* \).

**WMA** \( b \neq 0 \) (else, \( b = 0 \) implies \( d^* = 0 \) and choose \( X = 0 \)). Set

\[
\mathcal{M} := \left\{ \sum_j y_j A_j - C \mid y \in \mathbb{R}^m, \ b \cdot y \leq d^* \right\}.
\]

**Fact:** \( \mathcal{M} \cap S_n^{++} = \emptyset \).

**Pf:** Otherwise, let \( y \) for which \( b \cdot y \leq d^* \) and \( \sum_j y_j A_j - C \succ 0 \). Then one can find \( y' \) with \( b \cdot y' < b \cdot y \leq d^* \) and \( \sum_j y'_j A_j - C \succ 0 \). \( \rightsquigarrow \) This contradicts the minimality of \( d^* \).
Sketch of proof for 2. (continued)

As $\mathcal{M} \cap S_{n}^{++} = \emptyset$, there is a hyperplane separating $\mathcal{M}$ and $S_{n}^{++}$. That is, there exists $Z \succeq 0$ non-zero with $Z \cdot Y \leq 0 \ \forall Y \in \mathcal{M}$, i.e.,

$$b \cdot y \leq d^* \implies Z \cdot \left( \sum_{j} y_j A_j - C \right) \leq 0$$

By Farkas’ lemma, there exists $\mu \in \mathbb{R}_+$ for which

$$(Z \cdot A_j)_j = \mu b \ \text{and} \ \mu d^* \leq Z \cdot C$$

If $\mu = 0$, then $0 \geq Z \cdot \left( \sum_{j} \tilde{y}_j A_j - C \right) > 0$, a contradiction.

Hence $\mu > 0$ and $Z/\mu$ is feasible for (P) with $C \cdot (Z/\mu) \geq d^*$. QED.
An example with duality gap

\[ p^* = \min x_{12} \text{ s.t. } \begin{pmatrix} 0 & x_{12} & 0 \\ x_{12} & x_{22} & 0 \\ 0 & 0 & 1 + x_{12} \end{pmatrix} \succeq 0 \]

\[ = \min \frac{1}{2} E_{12} \cdot X \text{ s.t. } \begin{align*}
E_{11} \cdot X &= 0 \quad \Rightarrow a \\
E_{13} \cdot X &= 0 \quad \Rightarrow b \\
E_{23} \cdot X &= 0 \quad \Rightarrow c \\
(E_{33} - \frac{1}{2} E_{12}) \cdot X &= 1 \quad \Rightarrow y \\
X &\succeq 0
\end{align*} \]

\[ d^* = \max y \text{ s.t. } \frac{1}{2} E_{12} - aE_{11} - bE_{13} - cE_{23} - y(E_{33} - \frac{1}{2} E_{12}) \succeq 0 \]

\[ = \max y \text{ s.t. } \begin{pmatrix} -a & \frac{y+1}{2} & -b \\ \frac{y+1}{2} & 0 & -c \\ -b & -c & -y \end{pmatrix} \succeq 0 \]

Thus, \( p^* = 0, \ d^* = -1 \Rightarrow \text{non-zero duality gap} \)
Recall: An LP with rational data has a *rational* optimum solution whose *bit size is polynomially bounded* in terms of the bit length of the input data.

Not true for SDP:

- $\sqrt{2} = \max x \text{ s.t. } \begin{pmatrix} 1 & x \\ x & 2 \end{pmatrix} \succeq 0$
- Any solution to
  
  $\begin{pmatrix} x_1 - 2 & 0 \\ 0 & 1 \end{pmatrix} \succeq 0, \quad \begin{pmatrix} x_2 & x_1 \\ x_1 & 1 \end{pmatrix} \succeq 0, \ldots, \begin{pmatrix} x_n & x_{n-1} \\ x_{n-1} & 1 \end{pmatrix} \succeq 0$

  satisfies $x_1 \geq 2^{2^{n-1}}$.  

**Theorem:** SDP can be solved in polynomial time to an arbitrary prescribed precision. [Assuming certain technical conditions hold.]

- Theoretically: Use the ellipsoid method [since checking whether $X \succeq 0$ is in P, e.g. with Gaussian elimination]
- Practically: Use e.g. interior-point algorithms.

**More precisely:** Let $K$ denote the feasible region of the SDP. Assume we know $R \in \mathbb{N}$ s.t. $\exists X \in K$ with $\|X\| \leq R$ if $K \neq \emptyset$.

Given $\varepsilon > 0$, the ellipsoid based algorithm, either finds $X^*$ at distance at most $\varepsilon$ from $K$ such that $C \cdot X^* \succeq C \cdot X - \varepsilon \ \forall X \in K$ at distance at least $\varepsilon$ from the border, or claims: there is no such $X$.

The running time is polynomial in $n$, $m$, the bit size of $A_j$, $C$, $b$, $\log R$, and $\log(1/\varepsilon)$.
Feasibility of SDP

**Feasibility SDP problem (F):** Given integer $A_0, A_j \in S_n$, decide whether there exists $x \in \mathbb{R}^m$ s.t. $A_0 + \sum_{j=1}^m x_j A_j \succeq 0$?

- $(F) \in \text{NP} \iff (F) \in \text{co-NP}$.  
  \[\text{[Ramana 97]}\]

- $(F) \in \text{P}$ for fixed $n$ or $m$.  
  \[\text{[Porkolab-Khachiyan 97]}\]

- Testing existence of a *rational* solution is in P for fixed dimension $m$.  
  \[\text{[Porkolab-Khachiyan 97]}\]

~~~  More on complexity and algorithms for SDP in other lectures.
Use SDP to express convex quadratic constraints

Consider the convex quadratic constraint:

\[ x^T A x \leq b^T x + c \]

where \( A \succeq 0 \).

- Write \( A = B^T B \) for some \( B \in \mathbb{R}^{p \times n} \).

- Then:

\[ x^T A x \leq b^T x + c \iff \begin{pmatrix} I_p & Bx \\ x^T B^T & b^T x + c \end{pmatrix} \succeq 0 \]

\[ \iff \text{Use Schur complement: Given } C \succ 0, \]

\[ \begin{pmatrix} C & B \\ B^T & A \end{pmatrix} \succeq 0 \iff A - B^T C^{-1} B \succeq 0 \]
Introduction to Semidefinite Programming I: Basic properties and variations on the Goemans-Williamson approximation algorithm for max-cut

The S-lemma [Yakubovich 1971]

Consider the quadratic polynomials:

\[ f(x) = x^T Ax + 2a^T x + \alpha = (1 \ x^T) \begin{pmatrix} \alpha & a^T \\ a & A \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} \]

\[ g(x) = x^T Bx + 2b^T x + \beta = (1 \ x^T) \begin{pmatrix} \beta & b^T \\ b & B \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} \]

**Question:** Characterize when

\[ (*) \ f(x) \geq 0 \implies g(x) \geq 0 \]

**Answer:** Assume \( f(x) > 0 \) for some \( x \). Then, \( (*) \) holds IFF

\[ \begin{pmatrix} \beta & b^T \\ b & B \end{pmatrix} - \lambda \begin{pmatrix} \alpha & a^T \\ a & A \end{pmatrix} \succeq 0 \quad \text{for some } \lambda \geq 0 \]
Testing sums of squares of polynomials

**Question:** How to check whether a polynomial
\[ p(x) = \sum_{\alpha \in \mathbb{N}^n, |\alpha| \leq 2d} p_\alpha x^\alpha \] can be written as a sum of squares:
\[ p(x) = \sum_{j=1}^{m} (u_j(x))^2 \]
for some polynomials \( u_j \)?

**Answer:** Use semidefinite programming:

- Write \( u_j(x) = (a_j)^T [x]_d \) \[ [x]_d = (x^\alpha)_{|\alpha| \leq d} \]
- \[ \sum_j (u_j(x))^2 = ([x]_d)^T \left( \sum_j a_j a_j^T \right) [x]_d \]

\( \Rightarrow \) Test feasibility of SDP:
\[ \sum_{\beta, \gamma : |\beta|, |\gamma| \leq d, \beta + \gamma = \alpha} A_{\beta, \gamma} = p_\alpha \ (|\alpha| \leq 2d), \ A \succeq 0 \]
Two milestone applications of SDP to combinatorial optimization

- Approximate maximum stable sets and minimum vertex coloring with the theta number.
  
  Work of Lovász [1979], Grötschel-Lovász-Schrijver [1981]

- (First non-trivial) 0.878-approximation algorithm for max-cut of Goemans-Williamson [1995]
The theta number

Let $G = (V, E)$ be a graph.

- $S \subseteq V$ is a stable set if $S$ contains no edge.
- $\alpha(G) := \text{maximum size of a stable set}$, $\rightarrow \text{stability number}$
- $\chi(G) := \text{minimum number of colors needed to color the vertices so that adjacent vertices receive distinct colors.}$

$\rightarrow$ Computing $\alpha(G), \chi(G)$ is an NP-hard problem.

- **The theta number of Lovász [1979]:**

  $\vartheta(G) := \max J \cdot X \text{ s.t. } Tr(X) = 1, \ X_{ij} = 0 \ (ij \in E), \ X \succeq 0$

- **Lovász 'sandwich' theorem:** $\alpha(G) \leq \vartheta(G) \leq \chi(\bar{G})$.

$\rightarrow$ Can compute $\alpha(G), \chi(\bar{G})$ via SDP for graphs with $\alpha(G) = \chi(\bar{G})$. 

---

Monique Laurent - CWI, Amsterdam, and Tilburg University

Introduction to Semidefinite Programming I: Basic properties
Maximum cuts in graphs

\[ G = (V, E), \ n = |V|, \ w = (w_e)_{e \in E} \text{ edge weights.} \]

\[ S \subseteq V \mapsto \text{cut } \delta(S) := \text{all edges cut by the partition } (S, V \setminus S). \]

**Max-Cut problem:** Find a cut of maximum weight. \[ \mapsto \text{mc}(G) \]

- Max-Cut is NP-hard.
  No \((16/17 + \varepsilon)\)-approximation algorithm unless P=NP.

- Max-Cut is in P for graphs with no \(K_5\) minor, since it can be computed with the LP:

\[
\begin{align*}
\text{max } & w^T x \\
\text{s.t. } & x_{ij} - x_{ik} - x_{jk} \leq 0, \ x_{ij} + x_{ik} + x_{jk} \leq 2 \ \forall i, j, k \in V
\end{align*}
\]

**An easy 1/2-approximation algorithm** for \(w \geq 0\): Consider the random partition \((S, V \setminus S)\), where \(i \in S\) with prob. 1/2:

\[ E(w(S)) = w(E)/2 \geq \text{mc}(G)/2 \]
Goemans-Williamson approximation algorithm for Max-Cut

- Encode a partition \((S, V \setminus S)\) by a vector \(x \in \{\pm 1\}^n\).
- Encode the cut \(\delta(S)\) by the matrix \(X = xx^T\).
- Reformulate Max-Cut:
  \[
  \max \frac{1}{2} \sum_{ij \in E} w_{ij}(1 - x_ix_j) \quad \text{s.t.} \quad x \in \{\pm 1\}^n
  \]
- Solve the SDP relaxation:
  \[
  \max \frac{1}{2} \sum_{ij \in E} w_{ij}(1 - X_{ij}) \quad \text{s.t.} \quad X \succeq 0, \quad \text{diag}(X) = e
  \]
  \(v_1, \ldots, v_n\) unit vectors s.t. \(X = (v_i^Tv_j)\) is opt. for SDP.
- Randomized rounding: Pick a random hyperplane \(H\) with normal \(r\).
  \(\leadsto\) partition \((S, V \setminus S)\) depending on the sign of \(v_i^Tr\).
**Theorem:** For $w \geq 0$,

$$mc(G) \geq E(w(\delta(S))) \geq 0.878 \text{sdp}(G) \geq 0.878 \text{mc}(G).$$

**Basic lemma:** $\text{Prob}(ij \in \delta(S)) = \frac{\arccos(v_i^Tv_j)}{\pi}$.

$$E(w(\delta(S))) = \sum_{ij \in E} w_{ij} \text{Prob}(ij \in \delta(S)) = \sum_{ij \in E} w_{ij} \frac{\arccos(v_i^Tv_j)}{\pi} = \sum_{ij \in E} w_{ij} \frac{(1-v_i^Tv_j)}{2} \frac{2 \arccos(v_i^Tv_j)}{\pi 1 - v_i^Tv_j} \geq \alpha_{GW} \sim 0.878$$

$\geq \text{sdp}(G) \alpha_{GW}$
Extension to $\pm 1$ quadratic programming

Given $A \in S_n$

**Integer problem:** $\text{ip}(A) := \max \ x^T A x \ \text{s.t.} \ x \in \{\pm 1\}^n$

**SDP relaxation:** $\text{sdp}(A) := \max \ A \cdot X \ \text{s.t.} \ X \succeq 0, \ \text{diag}(X) = e.$

- $A = \frac{1}{4} L_w,$ $L_w$: Laplacian matrix of $(G, w)$ $\Rightarrow$ Max-Cut
  where $L_w(i, i) = w(\delta(i)), \ L_w(i, j) = -w_{ij}$.

  $\Rightarrow$ When $A \succeq 0, \ Ae = e, \ A_{ij} \leq 0 (i \neq j) \Rightarrow 0.878$-approx. alg.

  - $\Rightarrow$ When $A \succeq 0 \Rightarrow \frac{2}{\pi} (\sim 0.636)$-approx. alg. [Nesterov 97]

  $\Rightarrow$ When $\text{diag}(A) = 0 \Rightarrow$ Grothendieck constant
Nesterov $\frac{2}{\pi}$-approximation algorithm

- **Solve SDP:** Let $v_1, \ldots, v_n$ unit vectors s.t. $X = (v_i^T v_j)$ maximizes $A \cdot X$.
- **Random hyperplane rounding:** Pick a random unit vector $r$.
  
  $\Rightarrow$ random ±1 vector: $x = (\text{sgn}(r^T v_i))_{i=1}^n$

**Lemma 1 [identity of Grothendieck]** $E(xx^T) = \frac{2}{\pi} \arcsin X$.

**Proof:**

$$E(\text{sgn}(r^T v_i) \text{sgn}(r^T v_j)) = 1 - 2 \text{Prob}(\text{sgn}(r^T v_i) \neq \text{sgn}(r^T v_j))$$

$$= 1 - 2 \frac{\arccos(v_i^T v_j)}{\pi} = \frac{2}{\pi} \left( \frac{\pi}{2} - \arccos(v_i^T v_j) \right)$$

$$= \frac{2}{\pi} \arcsin(v_i^T v_j).$$
Global performance analysis

**Lemma 2:** \( \arcsin X - X \geq 0 \)

**Proof:** \( \arcsin x - x = \sum_k a_k x^{2k+1} \) where \( a_k \geq 0 \).

**Global analysis:**

\[
E(x^T A x) = A \cdot E(xx^T) = A \cdot (E(xx^T) - \frac{2}{\pi} X) + \frac{2}{\pi} A \cdot X
\]

\[
= \underbrace{A \cdot \left( \frac{2}{\pi} \arcsin X - \frac{2}{\pi} X \right)}_{\geq 0} + \frac{2}{\pi} A \cdot X
\]

\[
\geq \frac{2}{\pi} A \cdot X.
\]

**Therefore:** For \( A \succeq 0 \), \( \text{ip}(A) \geq \frac{2}{\pi} \text{sdp}(A) \).
Assume $\text{diag}(A) = 0$.
The support graph $G_A$ has as edges the pairs $ij$ with $A_{ij} \neq 0$.

**Definition:** The Grothendieck constant $K(G)$ of a graph $G$ is the smallest constant $K$ for which

$$\text{sdp}(A) \leq K \cdot \text{ip}(A) \quad \text{for all} \quad A \in S_n \quad \text{with} \quad G_A \subseteq G.$$ 

**Theorem:** ([Gr. 53] [Krivine 77] [Alon-Makarychev(x2)-Naor 05])

- For $G$ complete bipartite, $\frac{\pi}{2} \leq K(G) \leq \frac{\pi}{2} \frac{1}{\ln(1+\sqrt{2})} \approx 1.782$
- $\Omega(\log(\omega(G))) \leq K(G) \leq O(\log(\vartheta(\bar{G})))$. 

Monique Laurent - CWI, Amsterdam, and Tilburg University

Introduction to Semidefinite Programming I: Basic properties and variations on the Goemans-Williamson approximation algorithm for max-cut
Sketch of proof for Krivine’s upper bound: \( \frac{\pi}{2} \frac{1}{\ln(1+\sqrt{2})} \)

Show: \( K(K_n,m) \leq \frac{\pi}{2} \frac{1}{\ln(1+\sqrt{2})} =: \frac{\pi}{2c} \), \( c := \ln(1 + \sqrt{2}) \)

1. Let \( A \in \mathbb{R}^{n \times m} \). Let \( u_i \ (i \leq n) \) and \( v_j \ (j \leq m) \) be unit vectors in \( H \) maximizing \( \text{sdp}(A) = \sum_{i \leq n, j \leq m} a_{ij} u_i \cdot v_j \).

2. **Construct** new unit vectors \( S(u_i), T(v_j) \in \hat{H} \) satisfying
   \[
   \arcsin(S(u_i) \cdot T(v_j)) = c \ u_i \cdot v_j
   \]

3. Pick a random unit vector \( r \in \hat{H} \). Define the \( \pm 1 \) vectors \( x, y \)
   \[
   x_i = \text{sgn}(r^T S(u_i)), \quad y_j = \text{sgn}(r^T T(v_j))
   \]

4. **Analysis:**
   \[
   E(x^T A y) = \sum_{i,j} a_{ij} \quad E(x_i y_j) = \sum_{i,j} a_{ij} \frac{2}{\pi} \arcsin(S(u_i) \cdot T(v_j))
   \]
   \[
   = \sum_{i,j} a_{ij} \frac{2}{\pi} \ c \ u_i \cdot v_j = \frac{2}{\pi} c \ \text{sdp}(A).
   \]
Step 2. Given unit vectors $u, v \in H$, construct $S(u), T(v) \in \hat{H}$ satisfying
\[
\arcsin(S(u) \cdot T(v)) = c \ u \cdot v.
\]

- $\sin x = \sum_{k \geq 0} (-1)^k \frac{x^{2k+1}}{(2k+1)!}, \quad \sinh x = \sum_{k \geq 0} \frac{x^{2k+1}}{(2k+1)!}$

- Set $c := \sinh^{-1}(1) = \ln(1 + \sqrt{2})$.

- Set $S(u) = \left( \sqrt{\frac{c^{2k+1}}{(2k+1)!}} u \otimes (2k+1) \right)_k \in \hat{H} := \bigoplus_{k \geq 0} H \otimes (2k+1)$,
- $T(v) = \left( (-1)^k \sqrt{\frac{c^{2k+1}}{(2k+1)!}} v \otimes (2k+1) \right)_k \in \hat{H}$.

- Then, $S(u) \cdot T(v) = \sum_k (-1)^k \frac{c^{2k+1}}{(2k+1)!} (u \cdot v)^{2k+1} = \sin(c \ u \cdot v)$.

Thus: $\arcsin(S(u) \cdot T(v)) = c \ u \cdot v$. 
A reformulation of the theta number

**Theorem** [Alon-Makarychev(x2)-Naor 05]
The smallest constant $C$ for which

$$\text{sdp}(-A) \leq C \text{sdp}(A)$$

for all $A \in S_n$ with $G_A \subseteq G$

is $C = \vartheta(\tilde{G}) - 1$.

**Geometrically:**
- $\mathcal{E}_n := \{ X \in S_n | X \succeq 0, \text{diag}(X) = e \} \leadsto \text{the elliptope}$

  ➤ $\mathcal{E}(G) \subseteq \mathbb{R}^E$: the projection of $\mathcal{E}_n$ onto the edge set of $G$.

  ➤ **Theorem [AMMN]**: $-\mathcal{E}(G) \subseteq (\vartheta(\tilde{G}) - 1) \mathcal{E}(G)$. 

---

Monique Laurent - CWI, Amsterdam, and Tilburg University
Link with matrix completion

Matrix completion: Given a partial $n \times n$ matrix, whose entries are specified on the diagonal (say equal to 1) and on a subset $E$ of the positions (given by $x \in \mathbb{R}^E$), decide whether it can be completed to a PSD matrix.

Equivalently, decide whether $x \in \mathcal{E}(G)$?

- A necessary condition: Each fully specified principal submatrix is PSD. [Clique condition]
- The clique condition is sufficient IFF $G$ is a chordal graph (i.e. no induced circuit of length $\geq 4$).

\[
\begin{pmatrix}
1 & 1 & a? & -1 \\
1 & 1 & 1 & b? \\
a? & 1 & 1 & 1 \\
-1 & b? & 1 & 1
\end{pmatrix}
\]

is not completable to PSD
Another necessary condition

Fact: \[
\begin{pmatrix}
1 & \cos a & \cos b \\
\cos a & 1 & \cos c \\
\cos b & \cos c & 1
\end{pmatrix} \preceq 0 \iff \begin{cases}
a + b + c \leq 2\pi \\
a - b - c \leq 0 \\
-a + b - c \leq 0 \\
-a - b + c \leq 0
\end{cases}
\]

- Write \( x = \cos a \) for some \( a \in [0, \pi]^E \).
  If \( x \in \mathcal{E}(G) \) then
  \[
  a(F) - a(C \setminus F) \leq \pi(|F| - 1) \quad \forall \ C \text{ circuit, } F \subseteq C \text{ odd}.
  \]

- The metric condition is sufficient IFF \( G \) has no \( K_4 \) minor.
  \[
  \begin{pmatrix}
  1 & -1/2 & -1/2 & -1/2 \\
  -1/2 & 1 & -1/2 & -1/2 \\
  -1/2 & -1/2 & 1 & -1/2 \\
  -1/2 & -1/2 & -1/2 & 1
  \end{pmatrix} \not\succeq 0
  \]
  while \( \frac{2\pi}{3} (1, 1, 1, 1) \) satisfies the triangle inequalities.
Geometrically

- \( \text{CUT}^{\pm 1}(G) \subseteq \mathcal{E}(G) \), with equality IFF \( G \) has no \( K_3 \) minor.

- \( \mathcal{E}(G) \subseteq \cos(\pi \; \text{MET}^0(G)) \), with equality IFF \( G \) has no \( K_4 \) minor.

- \( \mathcal{E}(G) \subseteq \cos(\pi \; \text{CUT}^0(G)) \), with equality IFF \( G \) has no \( K_4 \) minor.

- The Goemans-Williamson randomized rounding argument shows: If \( v_1, \ldots, v_n \) are unit vectors and \( a_{ij} := \arccos(v_i^T v_j) \) are their pairwise angles, then

\[
\sum_{1 \leq i < j \leq n} c_{ij} a_{ij} \leq \pi \; c_0
\]

if \( c \cdot z \leq c_0 \) is any inequality valid for the cuts of \( K_n \).
Extension to max $k$-cut [Frieze Jerrum 95]

**Max $k$-cut:** Given $G = (V, E)$, $w \in \mathbb{R}_+^E$, $k \geq 2$, find a partition $\mathcal{P} = (S_1, \ldots, S_k)$ maximizing $w(\mathcal{P}) = \sum_{e \in E | e \text{ is cut by } \mathcal{P}} w_e$.

- Pick unit vectors $a_1, \ldots, a_k \in \mathbb{R}^k$ with $a_i^T a_j = -\frac{1}{k-1}$ for $i \neq j$.

\[\text{Model max } k\text{-cut:}\]

\[
mc_k(G) = \max \frac{k-1}{k} \sum_{ij \in E} w_{ij} (1 - x_i^T x_j)
\]

s.t. $x_1, \ldots, x_n \in \{a_1, \ldots, a_k\}$.

- **SDP relax.**

\[
sdp_k(G) = \max \frac{k-1}{k} \sum_{ij \in E} w_{ij} (1 - v_i^T v_j)
\]

s.t. $v_i$ unit vectors, $v_i^T v_j \geq -\frac{1}{k-1}$.

- **Randomized rounding:** Pick $k$ independent random unit vectors $r_1, \ldots, r_k \rightsquigarrow$ partition $\mathcal{P} = (S_1, \ldots, S_k)$ where $S_h = \{i \mid v_i^T r_h \geq v_i^T r_{h'} \ \forall h'\}$.
The probability that edge $ij$ is not cut, i.e., $v_i, v_j$ are both closer to the same $r_h$, is equal to $k$ times a function $f(v_i^T v_j)$.

$$E(w(P)) = \sum_{ij \in E} w_{ij} (1 - kf(v_i^T v_j))$$

$$= \sum_{ij \in E} w_{ij} \left( 1 - \frac{k}{k-1} \frac{1 - v_i^T v_j}{k-1} \right) \geq \alpha_k := \min_{\frac{1}{k-1} \leq t \leq 1} \frac{1 - kf(t)}{1-t} \frac{k}{k-1}$$

$$\geq \alpha_k \ sdp_k(G).$$

- $\alpha_2 = \alpha_{GW} \sim 0.878$: GW approximation ratio for max-cut.
- $\alpha_3 = \frac{7}{12} + \frac{3}{4\pi^2} \arccos^2(-1/4) > 0.836$ [de Klerk et al.]
- $\alpha_{100} > 0.99$. 