Lecture: Approximation Algorithms

Jannik Matuschke

TUM

December 12, 2016

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j \,:\, C_j \text{ is satisfied}} w_j$

$$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1} \checkmark \qquad \underbrace{\neg x_1 \vee x_3}_{C_2} \text{X} \qquad \underbrace{\neg x_3}_{C_3} \checkmark \qquad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \checkmark \qquad \underbrace{x_2 \vee \neg x_4}_{C_5} \checkmark$$

$$w_1 = 2 \qquad w_2 = 3 \qquad w_3 = 1 \qquad w_4 = 2 \qquad w_5 = 1$$

assignment: weight: 6

$x_1 = \texttt{true}$ $\quad x_2 = \texttt{true}$ $\quad x_3 = \texttt{false}$ $\quad x_4 = \texttt{true}$

Previously ...

Algorithm (Random sampling):
For each i, set $x_i = \text{TRUE}$ with proability $1/2$ (independently).

Analysis: $\Pr[C_j \text{ satisfied}] = 1 - (1/2)^{|C_j|} \geq 1/2$

- Random sampling is a randomized $\frac{1}{2}$-approximation.
- Algorithm can be derandomized (Method of Conditional Expectations).

LP rounding

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad & \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \ \geq \ z_j && \forall\, j \in [m] \\
& y_i \ \in \{0, 1\} && \forall\, i \in [n] \\
& z_j \ \in \{0, 1\} && \forall\, j \in [m]
\end{aligned}
$$

LP rounding

$$\begin{aligned}
\max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad & \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \geq z_j & \forall\, j \in [m] \\
& y_i \in \{0, 1\} & \forall\, i \in [n] \\
& z_j \in \{0, 1\} & \forall\, j \in [m]
\end{aligned}$$

LP rounding

$$\begin{aligned}
\max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \;&\geq\; z_j && \forall\, j \in [m] \\
0 \;\leq\; y_i \;&\leq\; 1 && \forall\, i \in [n] \\
0 \;\leq\; z_j \;&\leq\; 1 && \forall\, j \in [m]
\end{aligned}$$

LP rounding

$$\begin{array}{lrcll} \max & & \displaystyle\sum_{j=1}^{m} w_j z_j & & \\ \text{s.t.} & \displaystyle\sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) & \geq & z_j & \forall\, j \in [m] \\ & 0 \leq & y_i & \leq 1 & \forall\, i \in [n] \\ & 0 \leq & z_j & \leq 1 & \forall\, j \in [m] \end{array}$$

Algorithm 4:

1. Compute optimal LP solution (y^*, z^*).
2. For each $i \in [n]$, set x_i to `true` with probability y_i^*.

LP rounding

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad & \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \geq z_j & \forall\, j \in [m] \\
& 0 \leq y_i \leq 1 & \forall\, i \in [n] \\
& 0 \leq z_j \leq 1 & \forall\, j \in [m]
\end{aligned}
$$

Algorithm 4:

1. Compute optimal LP solution (y^*, z^*).
2. For each $i \in [n]$, set x_i to `true` with probability y_i^*.

Theorem

Algorithm 4 is a $(1 - 1/e)$-approximation algorithm for MAX SAT.

Choosing the better of two solutions

Let C_j be a clause of length k. From previous analysis:

- $\Pr[C_j \text{ sat. in random sampling}] \;\geq\; 1 - (1/2)^k$
- $\Pr[C_j \text{ sat. in randomized rounding}] \;\geq\; \left(1 - \left(1 - \frac{1}{k}\right)^k\right) z_j^*$

Choosing the better of two solutions

Let C_j be a clause of length k. From previous analysis:

- $\Pr[C_j \text{ sat. in random sampling}] \;\geq\; 1-(1/2)^k$
- $\Pr[C_j \text{ sat. in randomized rounding}] \;\geq\; \left(1-\left(1-\frac{1}{k}\right)^k\right) z_j^*$

Idea: Run both algorithms and take the better solution.

Choosing the better of two solutions

Let C_j be a clause of length k. From previous analysis:

- $\Pr[C_j \text{ sat. in random sampling}] \;\geq\; 1 - (1/2)^k$
- $\Pr[C_j \text{ sat. in randomized rounding}] \;\geq\; \left(1 - \left(1 - \frac{1}{k}\right)^k\right) z_j^*$

Idea: Run both algorithms and take the better solution.

Analysis: Run either algorithm with probability $1/2$.
Then clause C_j is satisfied with probability at least

$$\frac{1}{2}\left(1 - 2^{-k}\right) z_j^* + \frac{1}{2}\left(1 - \left(1 - \frac{1}{k}\right)^k\right) z_j^* \;\geq\; \frac{3}{4} z_j^*.$$

Choosing the better of two solutions

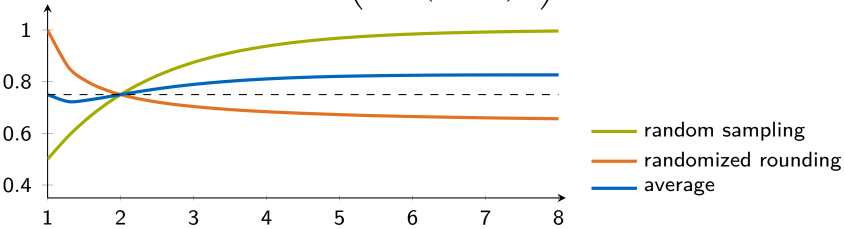
Let C_j be a clause of length k. From previous analysis:

- $\Pr[C_j \text{ sat. in random sampling}] \geq 1 - (1/2)^k$
- $\Pr[C_j \text{ sat. in randomized rounding}] \geq \left(1 - \left(1 - \frac{1}{k}\right)^k\right) z_j^*$

Idea: Run both algorithms and take the better solution.

Analysis: Run either algorithm with probability $1/2$.
Then clause C_j is satisfied with probability at least

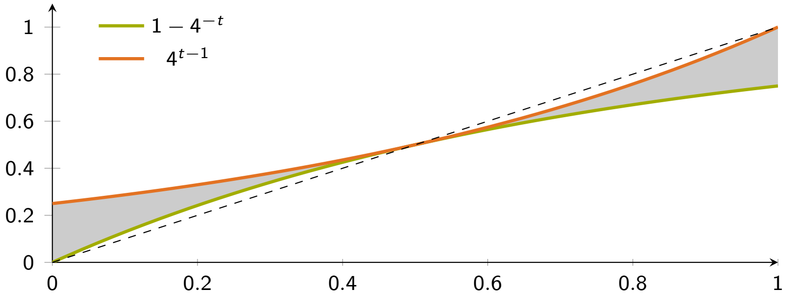
$$\frac{1}{2}\left(1 - 2^{-k}\right) z_j^* + \frac{1}{2}\left(1 - \left(1 - \frac{1}{k}\right)^k\right) z_j^* \geq \frac{3}{4} z_j^*.$$



Non-linear randomized rounding

Let $f : [0,1] \to [0,1]$ be a function with

$$1 - 4^{-t} \le f(t) \le 4^{t-1} \qquad \text{for all } t \in [0,1].$$



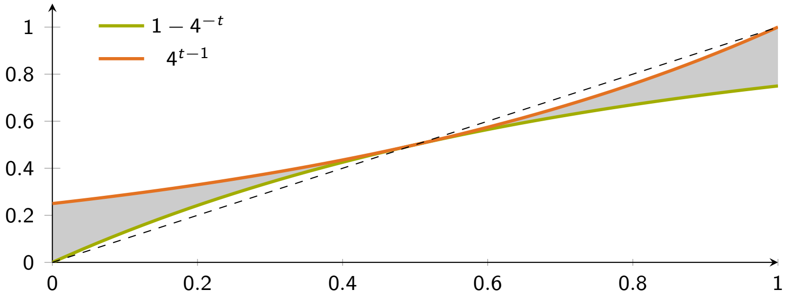
Algorithm 5

1. Compute optimal LP solution (y^*, z^*).
2. For each $i \in [n]$, set x_i to `true` with probability $f(y_i^*)$.

Non-linear randomized rounding

Let $f : [0, 1] \to [0, 1]$ be a function with

$$1 - 4^{-t} \leq f(t) \leq 4^{t-1} \qquad \text{for all } t \in [0, 1].$$



Algorithm 5

1. Compute optimal LP solution (y^*, z^*).
2. For each $i \in [n]$, set x_i to `true` with probability $f(y_i^*)$.

Theorem

Algorithm 5 is a randomized 3/4-approximation for MAX SAT.

Integrality gap

$$
\begin{aligned}
Z^* := \max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad & \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \geq z_j && \forall\, j \in [m] \\
& 0 \leq y_i \leq 1 && \forall\, i \in [n] \\
& 0 \leq z_j \leq 1 && \forall\, j \in [m]
\end{aligned}
$$

We have analyzed two algorithms with ALG $\geq \frac{3}{4} Z^*$.

Can we do better using this LP?

Integrality gap

$$
\begin{aligned}
Z^* := \max \quad & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} \quad & \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \ \geq \ z_j && \forall\, j \in [m] \\
& 0 \ \leq \ y_i \ \leq \ 1 && \forall\, i \in [n] \\
& 0 \ \leq \ z_j \ \leq \ 1 && \forall\, j \in [m]
\end{aligned}
$$

We have analyzed two algorithms with ALG $\geq \frac{3}{4} Z^*$.
Can we do better using this LP? No.

Conisder this instance:

$x_1 \vee x_2 \qquad x_1 \vee \neg x_2 \qquad \neg x_1 \vee x_2 \qquad \neg x_1 \vee \neg x_2 \qquad w \equiv 1$

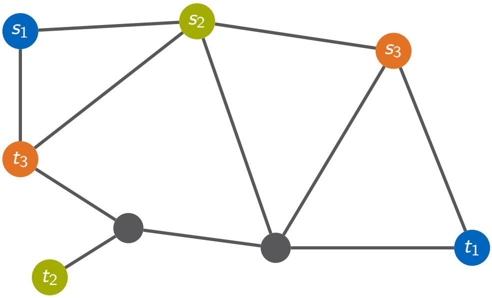
$$\text{OPT} = 3 \qquad Z^* = 4 \ (y_i = 1/2 \text{ for all } i)$$

Chernoff Bounds: Integer Multicommodity Flows

Integer Multicommodity Flow

Input: graph $G = (V, E)$, k terminal pairs $s_i, t_i \in V$

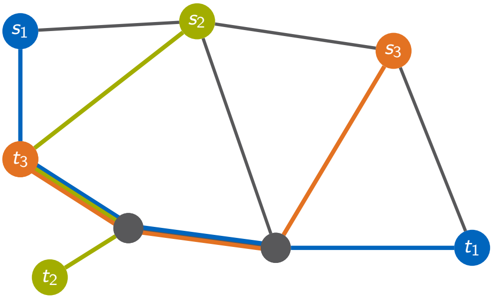
Task: find set an s_i-t_i-path P_i for each $i \in [k]$, minimizing $\max_{e \in E} |\{i \in [k] : e \in P_i\}|$



Integer Multicommodity Flow

Input: graph $G = (V, E)$, k terminal pairs $s_i, t_i \in V$

Task: find set an s_i-t_i-path P_i for each $i \in [k]$, minimizing $\max_{e \in E} |\{i \in [k] : e \in P_i\}|$



LP relaxation

$\mathcal{P}_i := \{P \subseteq E \ : \ P \text{ is } s_i\text{-}t_i\text{-path}\}$ $\qquad$ $\mathcal{P} := \bigcup_{i\in[k]} \mathcal{P}_i$

$$\begin{aligned}
\min \quad & W & & \\
\text{s.t.} \quad & \sum_{P\in\mathcal{P}_i} x_P = 1 & & \forall\ i \in [k] \\
& \sum_{P\in\mathcal{P}: e\in P} x_P \le W & & \forall\ e \in E \\
& x_P \in \{0,1\} & & \forall\ P \in \mathcal{P}
\end{aligned}$$

LP relaxation

$\mathcal{P}_i := \{P \subseteq E \ : \ P \text{ is } s_i\text{-}t_i\text{-path}\}$ $\qquad \mathcal{P} := \bigcup_{i \in [k]} \mathcal{P}_i$

$$
\begin{aligned}
\min \quad & W & & \\
\text{s.t.} \quad \sum_{P \in \mathcal{P}_i} x_P &= 1 & & \forall\, i \in [k] \\
\sum_{P \in \mathcal{P} : e \in P} x_P &\leq W & & \forall\, e \in E \\
x_P &\in \{0, 1\} & & \forall\, P \in \mathcal{P}
\end{aligned}
$$

LP relaxation

$\mathcal{P}_i := \{P \subseteq E \;:\; P \text{ is } s_i\text{-}t_i\text{-path}\}$ $\qquad$ $\mathcal{P} := \bigcup_{i\in[k]} \mathcal{P}_i$

$$\begin{aligned}
\min \quad & W & & \\
\text{s.t.} \quad & \sum_{P\in\mathcal{P}_i} x_P = 1 & & \forall\, i \in [k] \\
& \sum_{P\in\mathcal{P}:e\in P} x_P \le W & & \forall\, e \in E \\
& x_P \ge 0 & & \forall\, P \in \mathcal{P}
\end{aligned}$$

LP relaxation

$\mathcal{P}_i := \{P \subseteq E \ : \ P \text{ is } s_i\text{-}t_i\text{-path}\}$ $\qquad \mathcal{P} := \bigcup_{i \in [k]} \mathcal{P}_i$

$$
\begin{aligned}
\min \quad & W & & \\
\text{s.t.} \quad \sum_{P \in \mathcal{P}_i} x_P &= 1 & & \forall\, i \in [k] \\
\sum_{P \in \mathcal{P} : e \in P} x_P &\leq W & & \forall\, e \in E \\
x_P &\geq 0 & & \forall\, P \in \mathcal{P}
\end{aligned}
$$

Algorithm:

1. Compute optimal LP solution (x^*, W^*).
2. For each i, let $P_i = P \in \mathcal{P}_i$ with probability x_P^*.

LP relaxation

$\mathcal{P}_i := \{P \subseteq E \; : \; P \text{ is } s_i\text{-}t_i\text{-path}\} \qquad \mathcal{P} := \bigcup_{i \in [k]} \mathcal{P}_i$

$$
\begin{array}{llcll}
\min & W & & & \\
\text{s.t.} & \sum_{P \in \mathcal{P}_i} x_P & = & 1 & \forall\, i \in [k] \\
 & \sum_{P \in \mathcal{P} : e \in P} x_P & \leq & W & \forall\, e \in E \\
 & x_P & \geq & 0 & \forall\, P \in \mathcal{P}
\end{array}
$$

Algorithm:

1. Compute optimal LP solution (x^*, W^*).
2. For each i, let $P_i = P \in \mathcal{P}_i$ with probability x^*_P.

Define random variable $Y_e := |\{i \; : \; e \in P_i\}|$. Then

$$\mathbb{E}[Y_e] \;=\; \sum_{i \in [k]} \sum_{P \in \mathcal{P}_i : e \in P} \Pr[P_i = P] \;=\; \sum_{i \in [k]} \sum_{P \in \mathcal{P}_i : e \in P} x^*_P \;\leq\; W^*.$$

$$\mathbb{E}[\mathsf{ALG}] = \mathbb{E}[\max_{e \in E} Y_e]$$

Know: $\mathbb{E}[Y_e] \leq W^* \leq \mathsf{OPT}$

$$\mathbb{E}[\mathsf{ALG}] = \mathbb{E}[\max_{e \in E} Y_e]$$

Know: $\mathbb{E}[Y_e] \leq W^* \leq \mathsf{OPT}$

Caution: $\mathbb{E}[\max_{e \in E} Y_e] \neq \max_{e \in E} \mathbb{E}[Y_e]$

Chernoff bound

Theorem

Let $X_1, \ldots, X_k$ be independent random variables in $\{0, 1\}$ and $U \geq \mathbb{E}[\sum_{i=1}^{k} X_i]$. Then for $0 \leq \delta \leq 1$:

$$\Pr\left[\sum_{i=1}^{k} X_e^i \geq (1+\delta)U\right] \leq \exp\left(-\frac{U\delta^2}{3}\right).$$

Chernoff bound

Theorem

Let $X_1, \ldots, X_k$ be independent random variables in $\{0, 1\}$ and $U \geq \mathbb{E}[\sum_{i=1}^{k} X_i]$. Then for $0 \leq \delta \leq 1$:

$$\Pr\left[\sum_{i=1}^{k} X_e^i \geq (1+\delta)U\right] \leq \exp\left(-\frac{U\delta^2}{3}\right).$$

Proof. Use Markov's inequality; see Williamson & Shmoys Section 5.10.

Chernoff bound

Theorem

Let $X_1, \ldots, X_k$ be independent random variables in $\{0, 1\}$ and $U \geq \mathbb{E}[\sum_{i=1}^{k} X_i]$. Then for $0 \leq \delta \leq 1$:

$$\Pr\left[\sum_{i=1}^{k} X_e^i \geq (1+\delta)U\right] \leq \exp\left(-\frac{U\delta^2}{3}\right).$$

Apply to Randomized Rounding for IMF:

Random variable $X_e^i = \begin{cases} 1 & \text{if path } P_i \text{ contains } e \\ 0 & \text{otherwise} \end{cases}$

Then $Y_e = \sum_{i=1}^{k} X_e^i$ with $\mathbb{E}[Y_e] \leq W^*$.

Chernoff bound

Theorem

Let $X_1, \ldots, X_k$ be independent random variables in $\{0, 1\}$ and $U \geq \mathbb{E}[\sum_{i=1}^k X_i]$. Then for $0 \leq \delta \leq 1$:

$$\Pr\left[\sum_{i=1}^{k} X_e^i \geq (1+\delta)U\right] \leq \exp\left(-\frac{U\delta^2}{3}\right).$$

Apply to Randomized Rounding for IMF:

Random variable $X_e^i = \begin{cases} 1 & \text{if path } P_i \text{ contains } e \\ 0 & \text{otherwise} \end{cases}$

Then $Y_e = \sum_{i=1}^k X_e^i$ with $\mathbb{E}[Y_e] \leq W^*$.

With $\delta = 1$ and $U = c\ln(m)W^*$:

$$\Pr[Y_e \geq 2c\ln(m)W^*] \;\leq\; \exp\left(-\frac{c}{3}\ln(m)W^*\right) \;\leq\; m^{-c/3}$$

Analysis

Theorem

ALG $< c \ln(m) W^*$ with probability at least $1 - m^{1-c/3}$.

Analysis

Theorem

ALG $< c \ln(m) W^*$ with probability at least $1 - m^{1-c/3}$.

Proof. Know: $\Pr[Y_e \geq 2c \ln(m) W^*] \ \leq \ m^{-c/3}$

Theorem

$\text{ALG} < c\ln(m)W^*$ with probability at least $1 - m^{1-c/3}$.

Proof. Know: $\Pr[Y_e \geq 2c\ln(m)W^*] \;\leq\; m^{-c/3}$

$$\Pr[\text{ALG} \geq c\ln(m)W^*] \;=\; \Pr[\exists\, e \in E:\; Y_e \geq c\ln(m)W^*]$$

Analysis

Theorem

ALG $< c\ln(m)W^*$ with probability at least $1 - m^{1-c/3}$.

Proof. Know: $\Pr[Y_e \geq 2c\ln(m)W^*] \leq m^{-c/3}$

$$\begin{aligned}\Pr[\text{ALG} \geq c\ln(m)W^*] &= \Pr[\exists\, e \in E:\ Y_e \geq c\ln(m)W^*] \\ &\underset{\bigcup\text{-bound}}{\leq} \sum_{e\in E} \Pr[Y_e \geq c\ln(m)W^*]\end{aligned}$$

Theorem

ALG $< c\ln(m)W^*$ with probability at least $1 - m^{1-c/3}$.

Proof. Know: $\Pr[Y_e \geq 2c\ln(m)W^*] \;\leq\; m^{-c/3}$

$$\begin{aligned}\Pr[\mathrm{ALG} \geq c\ln(m)W^*] \;&=\; \Pr[\exists\, e \in E:\; Y_e \geq c\ln(m)W^*] \\ &\underset{\bigcup\text{-bound}}{\leq} \sum_{e\in E} \Pr[Y_e \geq c\ln(m)W^*] \\ &\leq\; m \cdot m^{-c/3} \qquad \square\end{aligned}$$

Analysis

Theorem

ALG $< c\ln(m)W^*$ with probability at least $1 - m^{1-c/3}$.

Proof. Know: $\Pr[Y_e \geq 2c\ln(m)W^*] \leq m^{-c/3}$

$$\begin{aligned}\Pr[\mathrm{ALG} \geq c\ln(m)W^*] &= \Pr[\exists\, e \in E:\ Y_e \geq c\ln(m)W^*] \\ &\underset{\bigcup\text{-bound}}{\leq} \sum_{e\in E} \Pr[Y_e \geq c\ln(m)W^*] \\ &\leq m \cdot m^{-c/3} \quad \square\end{aligned}$$

Remark: If $W^* \geq c\ln(m)$, we can get a better approximation guarantee; see Theorem 5.29 in Williamson & Shmoys.

Semidefinite Programming

Semidefinite programming

$X \in \mathbb{R}^{n \times n}$ positive semidifinte (psd): $\quad y^T X y \geq 0 \quad \forall\, y \in \mathbb{R}^n$

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} X_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ijk} X_{ij} = b_k \qquad \forall\, k \in [\ell] \\
& X_{ij} = X_{ji} \quad \forall\, i, j \in [n] \\
& X \text{ psd}
\end{aligned}
$$

Solving SDPs

- convex program
- separation oracle: compute smallest eigenvalue of X
- can be solved in polynomial time with additive error (under some technical conditions)

SDP $\rightarrow$ Vector program

Lemma

$$X \in \mathbb{R}^{n\times n} \text{ is symmetric and psd}$$
$$\Leftrightarrow$$
$$X = V^T V \text{ for some } V \in \mathbb{R}^{m\times n} \text{ with } m \leq n$$

$$\begin{aligned}
\max \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}X_{ij} & & \\
\text{s.t.} \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ijk}X_{ij} = b_k & & \forall\, k \in [\ell] \\
& X_{ij} = X_{ji} & & \forall\, i,j \in [n] \\
& X \text{ psd} & &
\end{aligned}$$

SDP $\to$ Vector program

Lemma

$$X \in \mathbb{R}^{n\times n} \text{ is symmetric and psd}$$
$$\Leftrightarrow$$
$$X = V^T V \text{ for some } V \in \mathbb{R}^{m\times n} \text{ with } m \leq n$$

$\rightsquigarrow$ replace X_{ij} by $v_i^T v_j$ for $v_i, v_j \in \mathbb{R}^n$

$$\begin{aligned}
\max \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} X_{ij} & & \\
\text{s.t.} \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ijk} X_{ij} = b_k & & \forall\, k \in [\ell] \\
& X_{ij} = X_{ji} & & \forall\, i,j \in [n] \\
& X \text{ psd} & &
\end{aligned}$$

SDP $\to$ Vector program

Lemma

$$X \in \mathbb{R}^{n \times n} \text{ is symmetric and psd}$$
$$\Leftrightarrow$$
$$X = V^T V \text{ for some } V \in \mathbb{R}^{m \times n} \text{ with } m \leq n$$

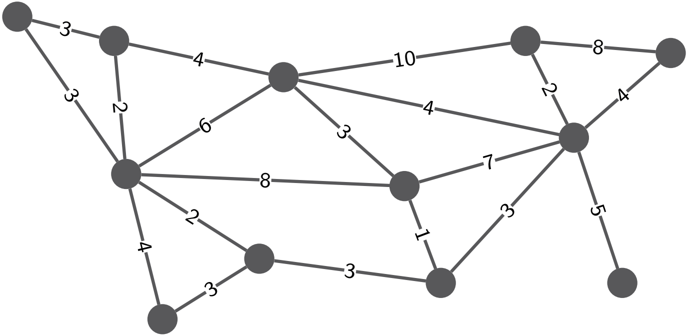
$\leadsto$ replace X_{ij} by $v_i^T v_j$ for $v_i, v_j \in \mathbb{R}^n$

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cdot v_i^T v_j \\ \text{s.t.} \quad & \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ijk} \cdot v_i^T v_j = b_k \qquad \forall\, k \in [\ell] \\ & v_i \in \mathbb{R}^n \qquad \forall\, i \in [n] \end{aligned}$$

Rounding an SDP by choosing a random hyperplane: The Maximum Cut Problem

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$

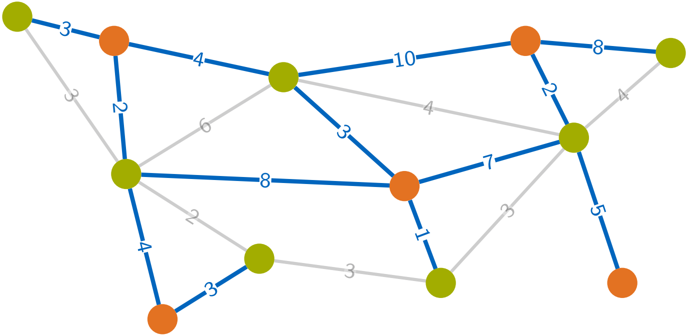
Task: find $S \subseteq V$ maximizing $\sum_{e \in \delta(S)} w(e)$



Max Cut

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$

Task: find $S \subseteq V$ maximizing $\sum_{e \in \delta(S)} w(e)$



SDP for MAX CUT

Quadratic program:

$$\begin{aligned} \max \quad & \frac{1}{2} \cdot \sum_{\{i,j\} \in E} w_{ij}(1 - x_i x_j) & & \text{w.l.o.g.: } V = [n] \\ \text{s.t.} \quad & x_i \in \{-1, +1\} & \forall\, i \in [n] & \end{aligned}$$

SDP for MAX CUT

Quadratic program:

$$\max \quad \frac{1}{2} \cdot \sum_{\{i,j\} \in E} w_{ij}(1 - x_i x_j) \qquad \text{w.l.o.g.: } V = [n]$$

$$\text{s.t.} \quad x_i \in \{-1, +1\} \qquad \forall\, i \in [n]$$

Relaxation:

$$Z^* := \max \quad \frac{1}{2} \cdot \sum_{\{i,j\} \in E} w_{ij}(1 - v_i^T v_j)$$

$$\text{s.t.} \quad v_i^T v_i = 1 \qquad \forall\, i \in [n]$$

$$v_i \in \mathbb{R}^n \qquad \forall\, i \in [n]$$

Selecting a random hyperplane

$$
\begin{aligned}
Z^* \;:=\; \max \;\; & \frac{1}{2} \cdot \sum_{\{i,j\} \in E} w_{ij}(1 - v_i^T v_j) & \\
\text{s.t.} \;\; & v_i^T v_i \;=\; 1 & \forall\, i \in [n] \\
& v_i \;\in \mathbb{R}^n & \forall\, i \in [n]
\end{aligned}
$$

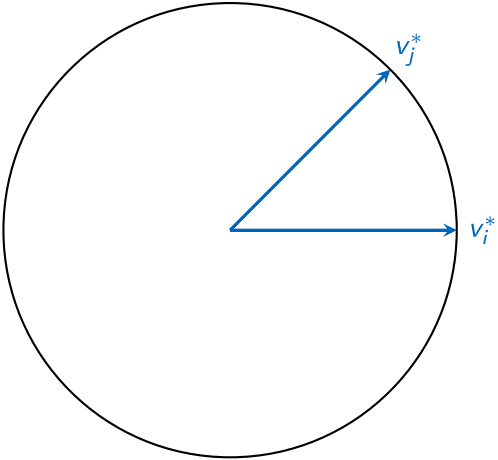
Algorithm

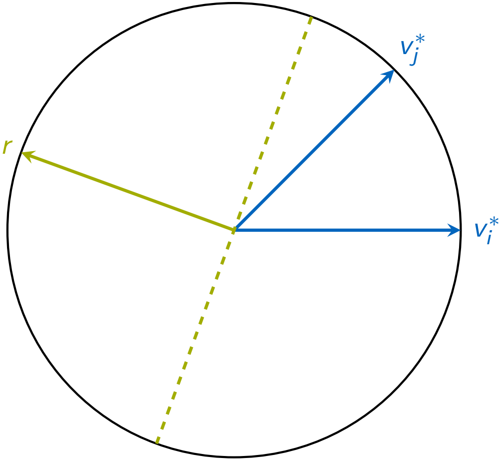
1. Compute optimal solution v^* to SDP.
2. Choose $r \in \mathbb{R}^n$ with $r^T r = 1$ uniformly at random.
3. Return $S := \{i \in [n] : r^T v_i^* \geq 0\}$.

Theorem

The algorithm is a randomized 0.878-approximation for Max Cut.

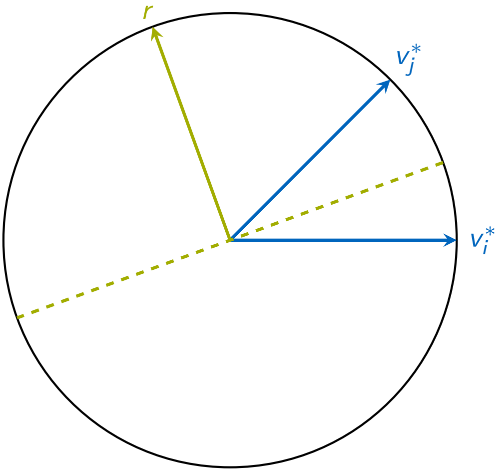
Analysis





$i \notin S, j \notin S$

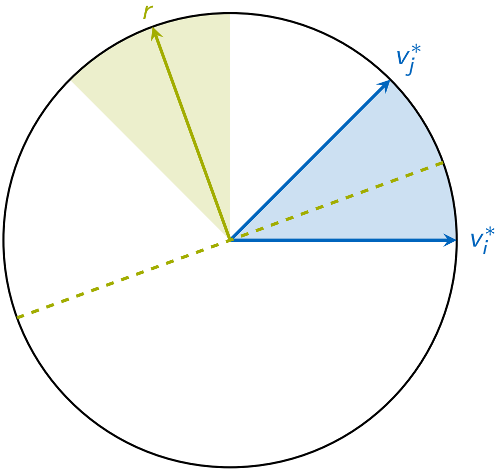
$\{i, j\} \notin \delta(S)$



$i \notin S, j \notin S$

$\{i, j\} \in \delta(S)$

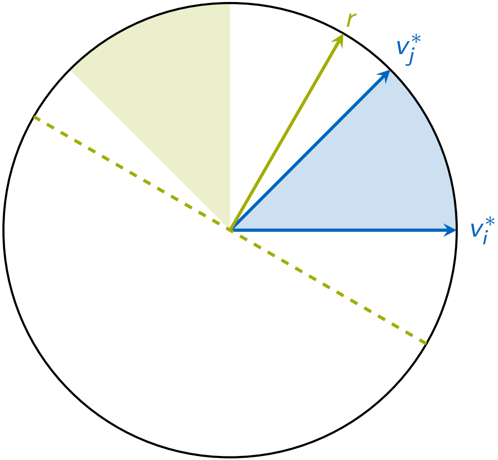
Analysis



$i \notin S, j \notin S$

$\{i, j\} \in \delta(S)$

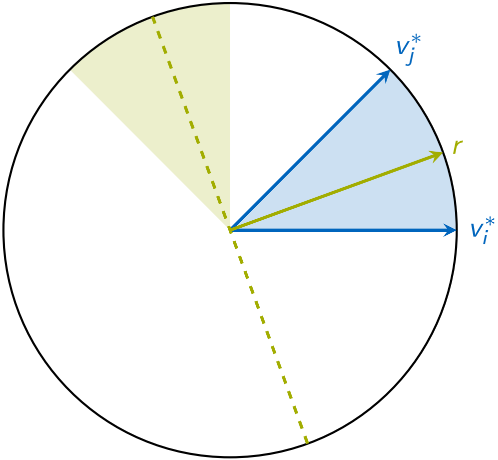
Analysis



$i \in S, j \in S$

$\{i, j\} \notin \delta(S)$

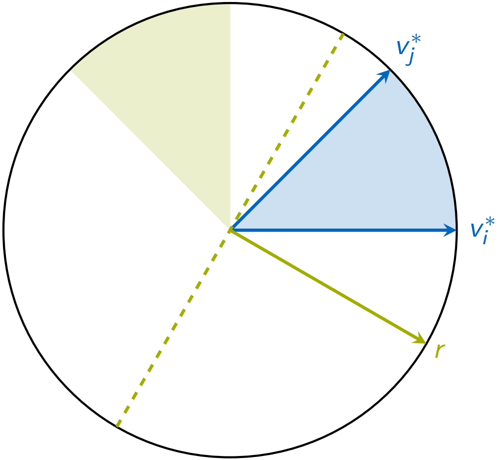
Analysis



$i \in S, j \in S$

$\{i, j\} \notin \delta(S)$

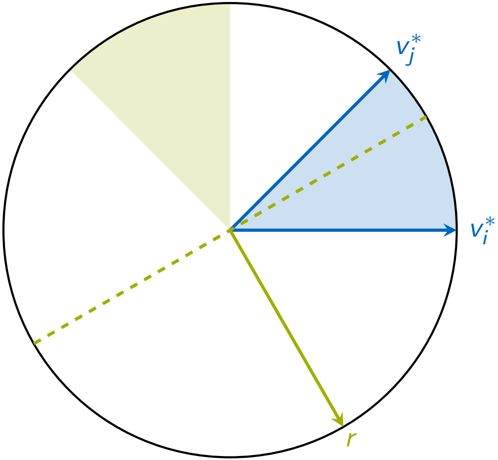
Analysis



$i \in S, j \in S$

$\{i, j\} \notin \delta(S)$

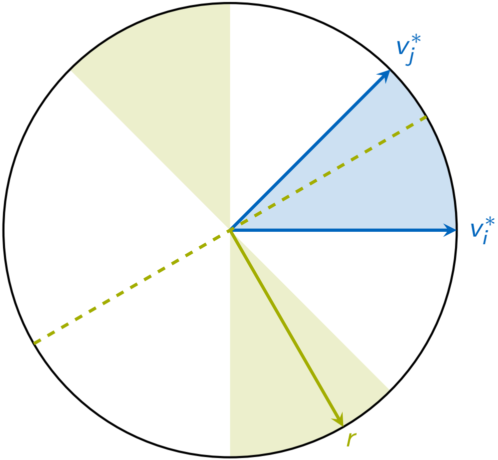
Analysis



$i \notin S, j \in S$

$\{i, j\} \in \delta(S)$

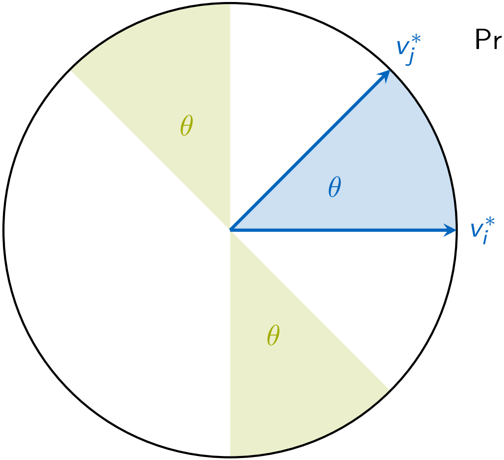
Analysis



$i \notin S, j \in S$

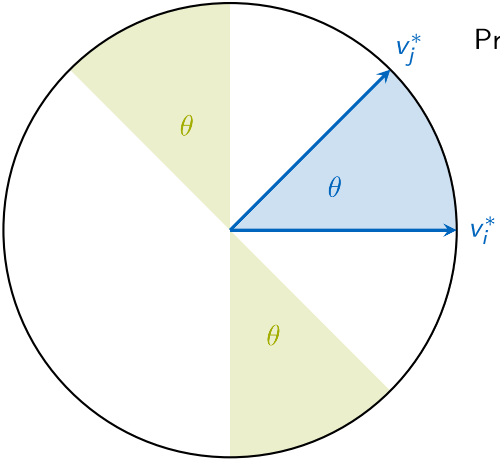
$\{i, j\} \in \delta(S)$

Analysis



$\Pr[\{i,j\} \in \delta(S)] = \frac{2\theta}{2\pi}$

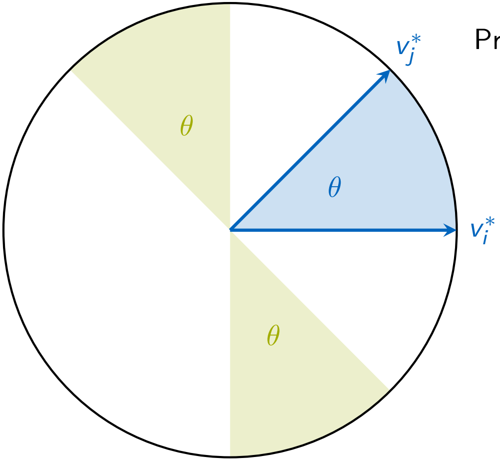
Analysis



$$\Pr[\{i,j\} \in \delta(S)] = \frac{2\theta}{2\pi}$$

$$\theta = \arccos({v_i^*}^T v_j^*)$$

Analysis



$\Pr[\{i,j\} \in \delta(S)] = \frac{2\theta}{2\pi}$

$\theta = \arccos({v_i^*}^T v_j^*)$

Lemma

$$\frac{1}{\pi}\arccos(x) \;\geq\; 0.878 \cdot \frac{1}{2}(1-x) \qquad \forall\, x \in [-1,1]$$

Hardness

There is an instance of MAX CUT with $\text{OPT} = 0.878 \cdot Z^*$.

Theorem

Unless $P = NP$ or the Unique Games Conjecture fails, there is no α-approximation algorithm for the maximum cut problem for any

$$\alpha > \min_{x \in [-1,1]} \frac{\frac{1}{\pi} \arccos(x)}{\frac{1}{2}(1 - x)} \geq 0.878.$$

Summary: Randomized rounding

Randomization can give us new perspectives!

It is also helpful to know some basic ...

- calculus (arithmetic vs. geometric mean)
- probability theory (Chernoff bound)
- geometry (picking a random hyperplane)