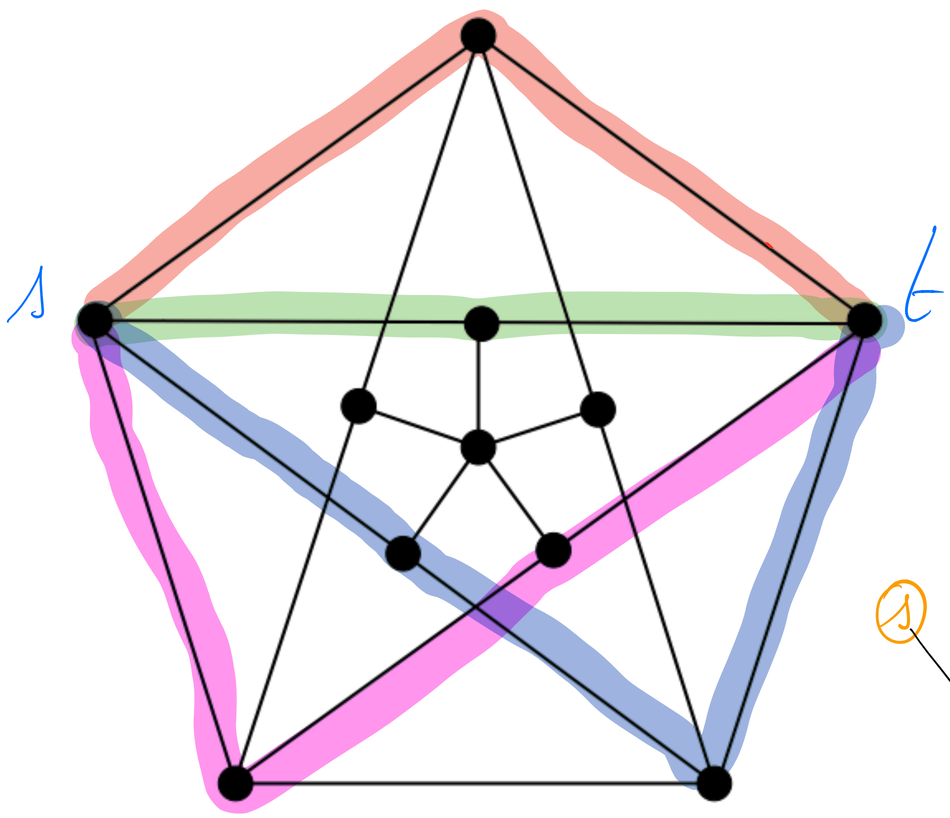
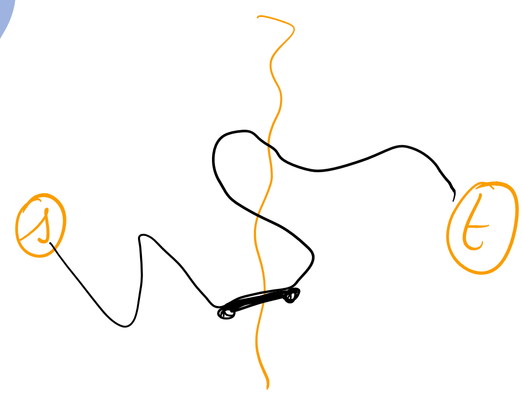
How to cut a graph



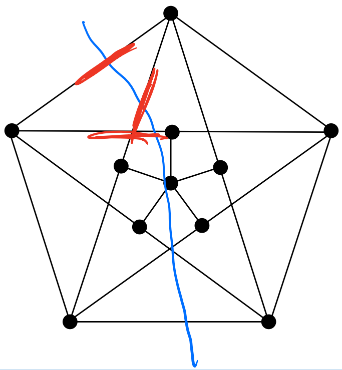


min s-t cut = 4

4 edge-disjoint s-t paths

Two Similar (?) Problems

The Minimum and Maximum Cut Problems



Cut in a graph.

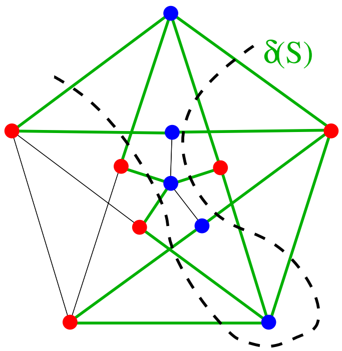
Given graph $G = (V, E)$, for any $S \subset V$, cut

$$\delta(S) := \{(i, j) \in E | i \in S \Leftrightarrow j \notin S\}.$$

For given weight $w \in \mathbb{R}^E$, weight of cut: $w(\delta(S)) := \sum_{e \in \delta(S)} w_e$

Two Similar (?) Problems

The Minimum and Maximum Cut Problems



$|E| = 20$, cut weight=15

Cut in a graph.

Given graph $G = (V, E)$, for any $S \subset V$, cut

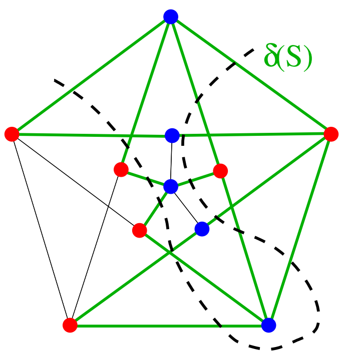
$$\delta(S) := \{(i,j) \in E | i \in S \Leftrightarrow j \notin S\}.$$

For given weight $w \in \mathbb{R}^E$, weight of cut: $w(\delta(S)) := \sum_{e \in \delta(S)} w_e$

unit weight $w_e = 1 \quad \forall e \in E$

Two Similar (?) Problems

The Minimum and Maximum Cut Problems



$|E| = 20$, cut weight=15

Cut in a graph.

Given graph $G = (V, E)$, for any $S \subset V$, cut

$$\delta(S) := \{(i,j) \in E | i \in S \Leftrightarrow j \notin S\}.$$

For given weight $w \in \mathbb{R}^E$, weight of cut: $w(\delta(S)) := \sum_{e \in \delta(S)} w_e$

For nonnegative weights $w \in \mathbb{R}^E_+$:

$MaxCut: \max_{S \subset V} w(\delta(S))$

$MinCut: \min_{\emptyset \subsetneq S \subsetneq V} w(\delta(S))$

Computational Complexity (18.404)

Decision problem: Yes/No answer

MINCUT: Given a graph $G=(V,E)$, integer k

Q: Is there a cut of value $\leq k$?

MAXCUT: Given a graph $G=(V,E)$, integer k

Q: Is there a cut of value $\geq k$?

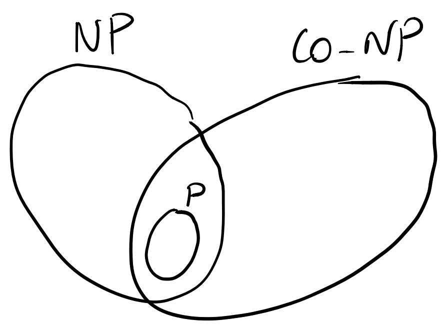
NP: Yes instances have a "short" certificate

MINCUT, MAXCUT in NP

CO-NP: No instances have a "short" certificate

?

$NP \cap$ co-NP: ("well characterized")



P : ("easy")

NP-complete ("hard")

Clay millenium prize for $P \overset{?}{\neq} NP$

Which is easy ?

MINCUT ? Easy — yes ✓

MAXCUT ? Hard — No ✗

extreme values:

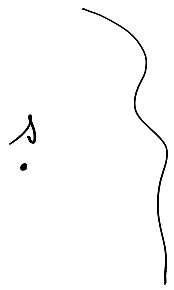
mincut = 0 for disconnected graphs

all edges in maxcut for bipartite (bicolorable) graphs

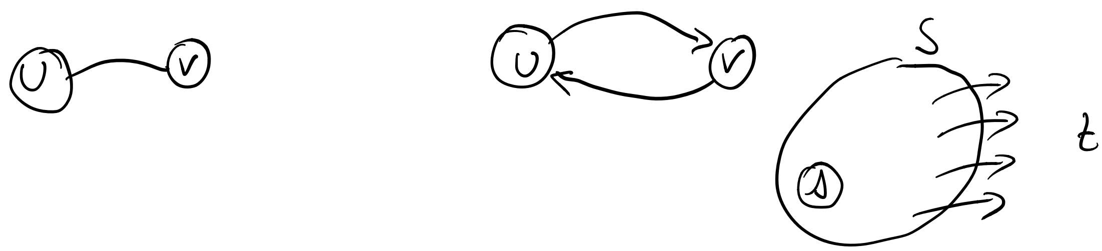
MINCUT : well-characterized

$$\min_{\emptyset \neq S \neq V} w(\delta(S)) = \min_{t : s \neq t} \underbrace{\min_{s \in S \subseteq V \setminus \{t\}} w(\delta(S))}_{\text{min } s\text{-}t \text{ cut problem}}$$

Fix s



min s-t cut problem in undirected graph
special case of min s-t cut problem in directed graph



min s-t cut problem in directed graphs
= max s-t flow value in graph

For (undirected) graph with unit weights

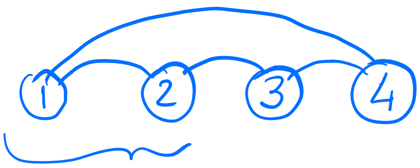
Menger's theorem:

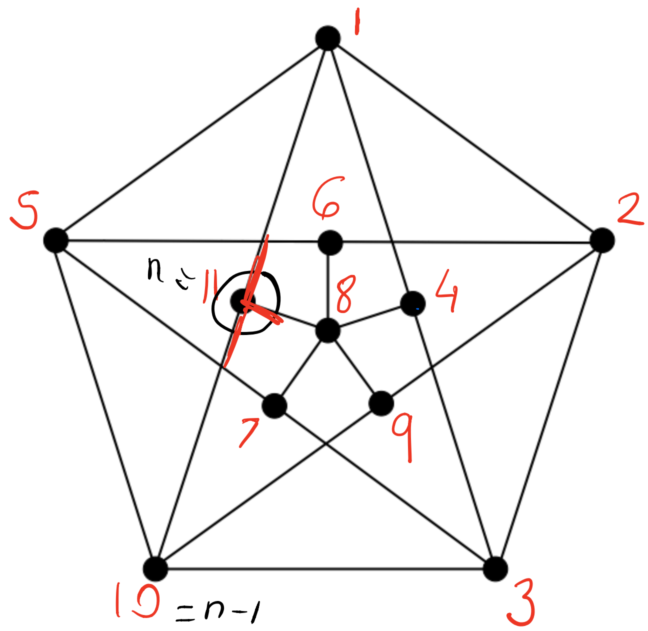
In any graph $G=(V,E)$, for any $s,t \in V$

max # of edge-disjoint paths between s and t = min cut value separating s and t

MINCUT "easy"

Maximum adjacency ordering of Nagamochi - Ibaraki
(for mincut)
(not for s-t mincut)





Rule: start from any vertex (call it 1)
at every step, choose next vertex which has the most edges to the previously labelled vertices

Theorem: Suppose $|V| = n$, and max adj ordering is $1, 2, 3, \ldots, n-1, n$. Then min cut separating $n-1$ from n is given by cut induced by $\{n\}$.

global min cut
- either separates $n-1$ and n ✓
- or does not
  contract $n-1$ and n. Start again

Coping with Intractibility

One approach: Approximation Algorithms

Design polynomial-time algorithms that deliver solution within a provable guarantee

Approximation algorithms often rely on first solving a **relaxation** of the problem: optimizing over larger space

Approximating MaxCut

How well can MaxCut be approximated in polynomial-time? For any weighted graph, find z such that

$$\alpha z \leq MaxCut \leq z$$

- $\alpha \leq 1$: approximation ratio.
- α-approximation algorithm if also produces cut $\delta(S)$ with $w(\delta(S)) \geq \alpha MaxCut$.

Approximating MaxCut

How well can MaxCut be approximated in polynomial-time? For any weighted graph, find z such that

$$\alpha z \leq MaxCut \leq z$$

- $\alpha \leq 1$: approximation ratio.
- α-approximation algorithm if also produces cut $\delta(S)$ with $w(\delta(S)) \geq \alpha MaxCut$.

Easy to get $\alpha = \frac{1}{2}$

- Choose S uniformly at random

$$\Rightarrow \mathbb{E}[w(\delta(S))] = \sum_{e \in E} \Pr[e \in \delta(S)] w_e = \frac{1}{2} \sum_{e \in E} w_e \geq \frac{1}{2} MaxCut$$

- For deterministic algorithm, repeatedly condition on whether $v \in S$ $\rightarrow$ greedy $\frac{1}{2}$-approximation algorithm.

$$\mathbb{E}[w(\delta(S))] = \mathbb{E}[w(\delta(S)) \mid 1 \text{ is blue}] \cdot \underbrace{\mathbb{P}[1 \text{ blue}]}_{=\frac{1}{2}} + \mathbb{E}[w(\delta(S)) \mid 1 \text{ is red}] \cdot \underbrace{\mathbb{P}[1 \text{ red}]}_{=\frac{1}{2}}$$

$$\leq \max\left(\mathbb{E}[w(\delta(S)) \mid 1 \text{ is blue}], \mathbb{E}[w(\delta(S)) \mid 1 \text{ is red}]\right)$$

Successively fix colouring of each vertex without decreasing the expectation

Formulating MaxCut

Convex

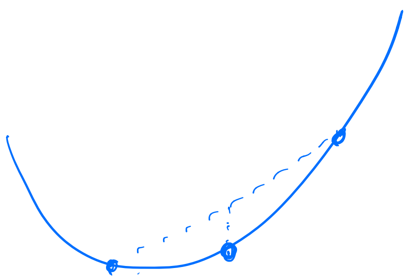
$$= \max_{x \in [-1,1]^V} \frac{1}{4} x^T L x$$

$$MaxCut = \max_{x \in \{-1,1\}^V} \frac{1}{4} \sum_{(u,v) \in E} w_{uv}(x_u - x_v)^2 = \max_{x \in \{-1,1\}^V} \frac{1}{4} x^T L x$$

where the Laplacian $L = L(w) = \sum_{e \in E} w_e L_e$ with

$L_{\{u,v\}} = (1_u - 1_v)(1_u - 1_v)^T$.

$$\begin{array}{c|cc} & u & v \\ \hline u & 1 & -1 \\ v & -1 & 1 \end{array}$$



$$x^T L_e x = x_u^2 + x_v^2 - 2 x_u x_v$$

Eigenvalue Bound for MaxCut

$$MaxCut = \max_{x \in \{-1,1\}^V} \frac{1}{4} x^T L x$$

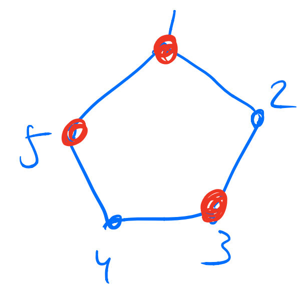
Eigenvalue Bound:

$$MaxCut \leq \frac{1}{4} \max_{x \in \mathbb{R}^V : x^T x = n} x^T L x = \frac{n}{4} \lambda_{max}(L)$$

(λ_{max}: largest eigenvalue)

Example: cycle on 5 vertices C_5

$$L = \begin{pmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{pmatrix}$$



with eigenvalues $2(1 - \cos(2\pi k/5))$ for $k = 0, 1, 2, 3, 4$

MaxCut ≤ 4.52254. Here, $\frac{MaxCut}{\lambda_{max}(L)} = \frac{32}{25+5\sqrt{5}} = 0.88445\cdots$

Refined Eigenvalue Bound

$$MaxCut = \max_{x \in \{-1,1\}^V} \frac{1}{4} x^T L x = \max_{x \in \{-1,1\}^V} \frac{1}{4} \left(x^T (L + Diag(u)) x - \sum_i u_i \right)$$

[Delorme-Poljak 1993]

$$MaxCut \leq \frac{n}{4} \left[\min_{u \in \mathbb{R}^V : \sum u_i = 0} \lambda_{max}(L + Diag(u)) \right]$$

- How does one compute efficiently best u??
- Worst-case gap conjectured by Delorme and Poljak to be $\frac{32}{25+5\sqrt{5}} = 0.88445 \cdots$ for the 5-cycle

Semidefinite Programming Relaxation for MaxCut

[G. and Williamson '94]

$$MaxCut = \max_{x \in \{-1,1\}^V} \frac{1}{4} x^T L x = \max_{x \in \{-1,1\}^V} \frac{1}{4} \langle L, xx^T \rangle$$

where the Laplacian $L = L(w) = \sum_{e \in E} w_e L_e$ with

$L_{\{u,v\}} = (1_u - 1_v)(1_u - 1_v)^T$.

$$(x^T)(L)(x)$$

$$= \mathrm{Tr}(L \cdot xx^T)$$

rank-1 matrix

Semidefinite Programming Relaxation for MaxCut

[G. and Williamson '94]

$$MaxCut = \max_{x \in \{-1,1\}^V} \frac{1}{4} x^T L x = \max_{x \in \{-1,1\}^V} \frac{1}{4} \langle L, xx^T \rangle \leq \max_{Y} \frac{1}{4} \langle L, Y \rangle$$

1., 3., 4.

where the Laplacian $L = L(w) = \sum_{e \in E} w_e L_e$ with

$L_{\{u,v\}} = (1_u - 1_v)(1_u - 1_v)^T.$

- Let $Y = xx^T$ (where $x \in \{-1,1\}^V$). Thus

  1. $Y \succeq 0$ (positive semidefinite)
  2. $\text{rank}(Y) = 1$ $\leftarrow$ relax!
  3. $Y_{ii} = 1$ for all $i \in V$
  4. weight of cut $= \frac{1}{4} \langle L, Y \rangle$

Semidefinite Programming (SDP) relaxation

$$MaxCut \quad \leq \quad SDP = \text{Max} \quad \frac{1}{4}\langle L, Y \rangle \qquad \left(= \sum_{(i,j)\in E} w_{ij} \frac{1 - Y_{ij}}{2} \right)$$

$$(SDP) \qquad \text{s.t.} \qquad Y_{ii} = 1 \qquad \forall i$$

$$Y \succeq 0$$

can be computed in time polynomial in size of graph

Semidefinite Programming (SDP)

$$\begin{aligned} \text{Sup} \quad & \langle C, X \rangle & \\ \text{s.t.} \quad & \langle A_i, X \rangle = b_i & i = 1, \cdots, m \\ & X \succeq 0 & \end{aligned}$$

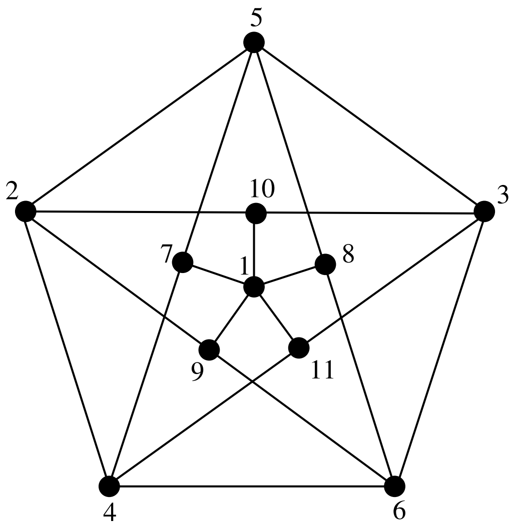
- Generalizes linear programming
- **Convex** optimization. Special case of conic programming (over cone K of positive semidefinite matrices): real SDP, complex SDP, hyperbolic optimization, ...
- Strong duality (under regularity condition) $\quad \sup \cdots = \inf \cdots$
- Can be solved up to ϵ in polynomial time (in input size and $\log \frac{1}{\epsilon}$) by (off-the-shelf or specialized) interior-point algorithms (as $-\log\det(Y)$ is convex over the convex SDP cone)

GW Randomized Approximation Algorithm

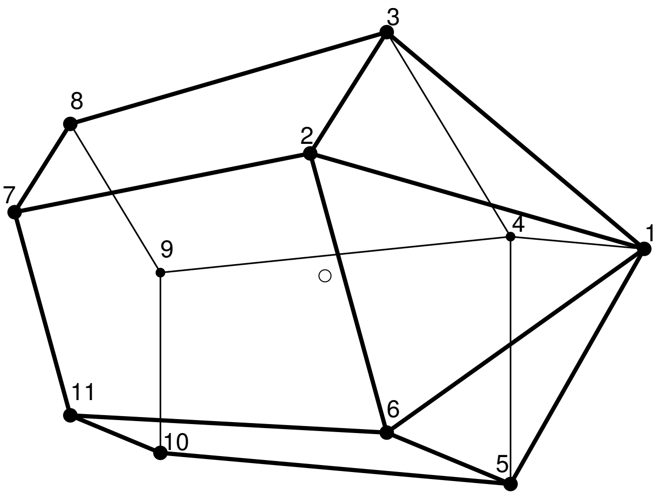
How do you recover a good cut from this convexification?

G.-Williamson '94:

1. Solve SDP relaxation $\Rightarrow Y$
2. Cholesky decomposition $\Rightarrow v_i \in \mathbb{R}^n$ for $i \in V$ with $Y_{ij} = \langle v_i, v_j \rangle$
3. Let r be sampled uniformly from unit sphere S^{n-1} (or r Gaussian)
4. Cut $\delta(S)$ defined by $S = \{i : \langle r, v_i \rangle \geq 0\}$



$|E| = 20$



$SDP = 17.18, \mathbb{E}[|\delta(S)|] = 15.11$

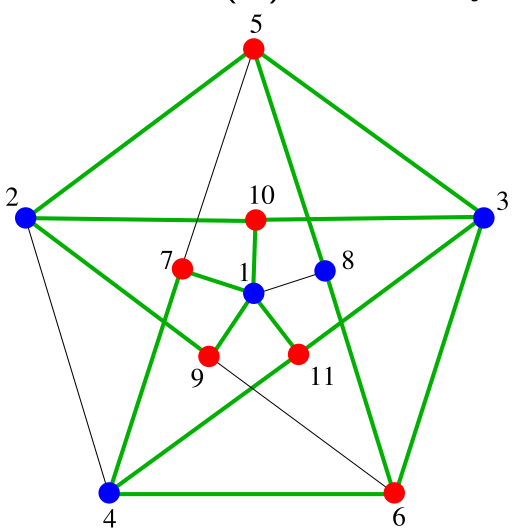
GW Randomized Approximation Algorithm

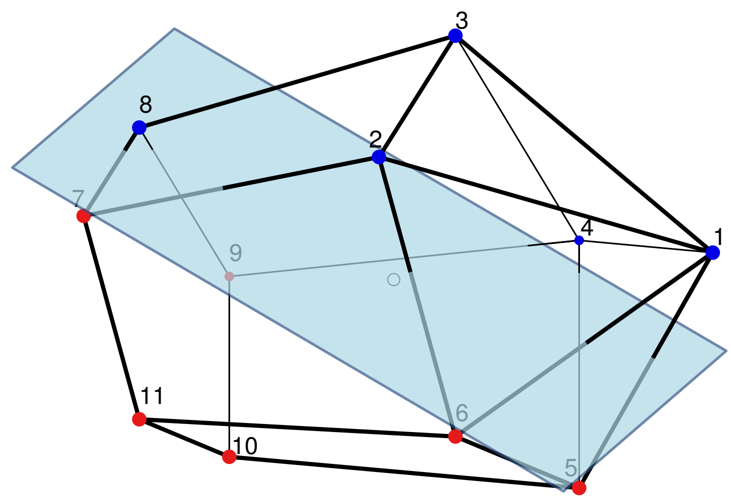
How do you recover a good cut from this convexification?

$d : \frac{d(d+1)}{2} \leq n = |V|$

G.-Williamson '94:

1. Solve SDP relaxation $\Rightarrow Y$
2. Cholesky decomposition $\Rightarrow v_i \in \mathbb{R}^n$ for $i \in V$ with $Y_{ij} = \langle v_i, v_j \rangle$
3. Let r be sampled uniformly from unit sphere S^{n-1} (or r Gaussian)
4. Cut $\delta(S)$ defined by $S = \{i : \langle r, v_i \rangle \geq 0\}$





$|E| = 20$, $\textsc{Cut}=16$, $SDP = 17.18, \mathbb{E}[|\delta(S)|] = 15.11$

GW Randomized Approximation Algorithm

How do you recover a good cut from this convexification?

G.-Williamson '94:

1. Solve SDP relaxation $\Rightarrow Y$
2. Cholesky decomposition $\Rightarrow v_i \in \mathbb{R}^n$ for $i \in V$ with $Y_{ij} = \langle v_i, v_j \rangle$
3. Let r be sampled uniformly from unit sphere S^{n-1} (or r Gaussian)
4. Cut $\delta(S)$ defined by $S = \{i : \langle r, v_i \rangle \geq 0\}$

Theorem [GW '94] For $w \geq 0$

$$\mathbb{E}[w(\delta(S)] \geq \alpha SDP \geq \alpha MaxCut$$

where $\alpha = \min_{-1<\rho<1} \frac{\arccos(\rho)/\pi}{(1-\rho)/2} = \min_{0<\theta<\pi} \frac{\theta/\pi}{(1-\cos\theta)/2} = 0.87856\cdots$

My new license plate 3 weeks *before* this result: 

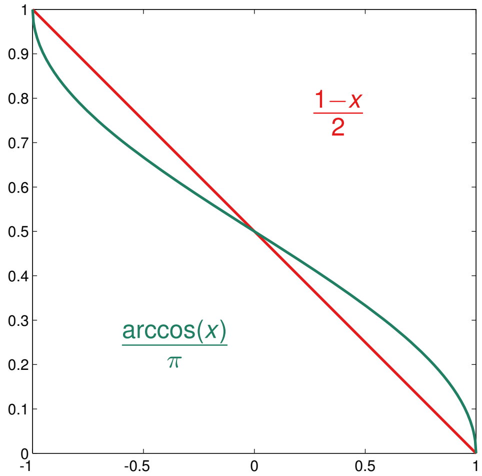
(Shockingly) Elementary Analysis

Lemma

For $v_i, v_j \in S^{d-1}$ and r Gaussian,

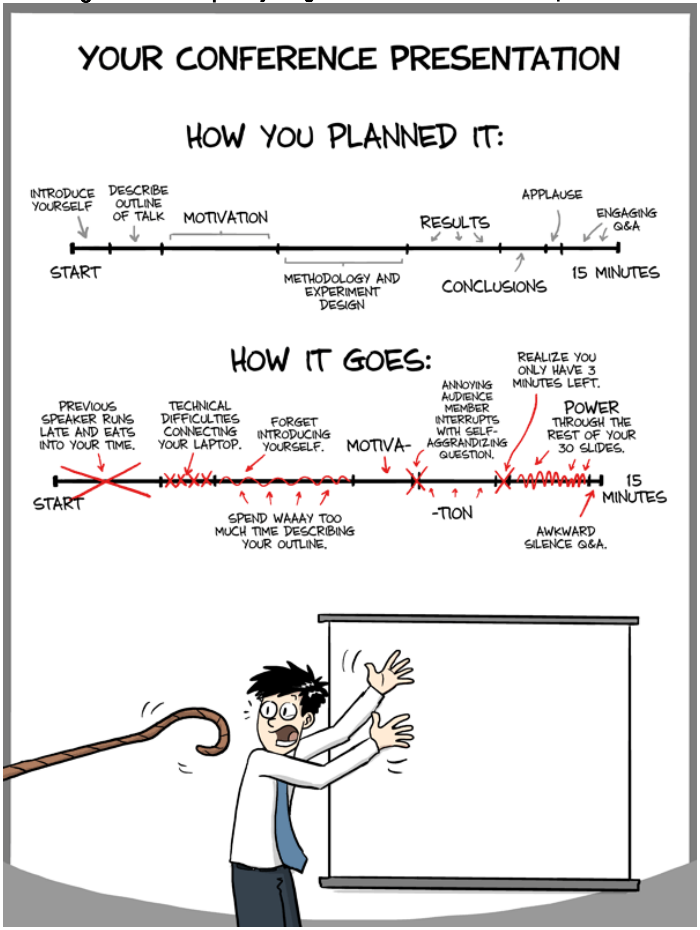
$$\Pr[\operatorname{sgn}\langle r, v_i\rangle \neq \operatorname{sgn}\langle r, v_j\rangle] = \frac{1}{\pi}\arccos\langle v_i, v_j\rangle$$

$$\rightarrow \mathbb{E}[w(\delta(S))] = \sum_{(i,j)\in E} w_{ij}\frac{\arccos Y_{ij}}{\pi} \text{ while } SDP = \sum_{(i,j)\in E} w_{ij}\frac{1-Y_{ij}}{2}$$



$$\frac{\arccos(x)}{\pi} \geq 0.87856\ldots\frac{1-x}{2}$$

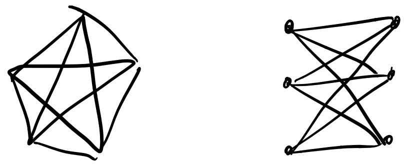
Piled Higher and Deeper ***by Jorge Cham*** **www.phdcomics.com**



title: "How your Conference Presentation Goes" - originally published 1/21/2013

Max cut in planar graph is polynomially solvable

↳ no K_5 and $K_{3,3}$ minors (Kuratowski)



If take planar dual (vertices ⟷ faces)
problem reduces to matching type
problem

- Can even solve maxcut in graphs with
no K_5 minor
(e.g. by Linear Programming)

Dual of SDP Relaxation: Refined Eigenvalue Bound!

$$SDP = \frac{n}{4}\left[\min_{u \in \mathbb{R}^V : \sum u_i = 0} \lambda_{max}(L + Diag(u))\right]$$

Rounding scheme based on this eigenvalue formulation?

- Trevisan '08: Spectral partitioning gives 0.531 guarantee.
- Soto '09: Guarantee improved to 0.614

Alternative Rounding Scheme II: Sticky Brownian Rounding

Sticky Brownian Rounding à la Bansal:

[Abbasi-Zadeh, Bansal, Guruganesh, Nikolov, Schwartz & Singh '18]:

- Maintain $X_t \in [-1, 1]^n$. Initially $X_0 = 0$, eventually $X_T \in \{-1, 1\}^n$.
- Sticky: As soon as $(X_t)_i$ is close to ± 1, stays there
- Step from X_t to X_{t+1} is scaled Gaussian step with covariance matrix Y, restricted to non-fixed indices/vertices

Using elliptic PDEs and complex analysis, show guarantee of 0.861

Variant with slowdown (modifying also covariance matrix):

- Abbasi-Zadeh et al.: 0.878 (but < 0.8785)
- Eldan and Naor '19: can match GW bound, $0.87856\cdots$, with *Krivine diffusions*

Analysis of GW Hyperplane Rounding is Tight

Theorem [Karloff '99], [Alon & Sudakov '00] :
For $-1 \leq \rho \leq 0$, construction of MaxCut instances with $SDP = MaxCut = (1-\rho)/2$ but GW rounding provides hyperplane cuts having value 'only' $\arccos(\rho)/\pi$

- Karloff's construction (works for some range of ρ): Kneser-type graphs
- Alon and Sudakov's construction:
  - $V = \{-1,1\}^m$ and $E = \{(x,y) : H(x,y) = (1-\rho)m/2\}$
  - Bose-Mesner algebra of the Hamming association scheme: Eigenvectors/functions are
    $$\chi_I(x) = \prod_{i \in I} x_i \qquad \text{for all } I \subseteq [m]$$
    and eigenvalues given by Krawtchouk polynomials
  - Natural embedding of $\{-1,1\}^m$ in S^{m-1} has $\langle v_i, v_j \rangle = \rho$.
    $\Rightarrow SDP \geq \frac{1}{2}(1-\rho)|E|$
    $+$ spectral analysis $\Rightarrow SDP = \frac{1}{2}(1-\rho)|E|$
  - Expectation of random hyperplane cut value: $(\arccos(\rho)/\pi)|E|$.

Tight Gap between SDP and MaxCut

Feige and Schechtman '02

Theorem [Feige and Schechtman '02]

$$\inf_{G,w} \frac{MaxCut}{SDP} = \alpha = \min_{-1<\rho<0} \frac{\arccos(\rho)/\pi}{(1-\rho)/2} (= 0.87856\cdots)$$

Construction: Fix any $-1 < \rho < 0$

1. V dense set of points on S^d
2. $E = \{(x, y) | \rho - \epsilon \leq \langle x, y \rangle \leq \rho\}$

Tight Gap between SDP and MaxCut

Feige and Schechtman '02

Theorem [Feige and Schechtman '02]

$$\inf_{G,w} \frac{MaxCut}{SDP} = \alpha = \min_{-1<\rho<0} \frac{\arccos(\rho)/\pi}{(1-\rho)/2} (= 0.87856\cdots)$$

Construction: Fix any $-1 < \rho < 0$

1. V dense set of points on S^d
2. $E = \{(x,y) | \rho - \epsilon \leq \langle x, y \rangle \leq \rho\}$

- Given embedding $\Rightarrow SDP \geq \frac{1}{2}(1-\rho)|E|$
- Need to show maximum cut given by hyperplane cuts

Proof from isoperimetric inequality on continuous analogue, and concentration.

Tight Gap between SDP and MaxCut

Feige and Schechtman '02

Theorem [Feige and Schechtman '02]

$$\inf_{G,w} \frac{MaxCut}{SDP} = \alpha = \min_{-1<\rho<0} \frac{\arccos(\rho)/\pi}{(1-\rho)/2} (= 0.87856\cdots)$$

Construction: Fix any $-1 < \rho < 0$

1. V dense set of points on S^d
2. $E = \{(x,y) | \rho - \epsilon \leq \langle x, y \rangle \leq \rho\}$

- Given embedding $\Rightarrow SDP \geq \frac{1}{2}(1-\rho)|E|$
- Need to show maximum cut given by hyperplane cuts

Proof from isoperimetric inequality on continuous analogue, and concentration.

Theorem

Among all sets $A \subseteq S^d$ with $\mu(A) = a$, a cap has the largest $\mu_\rho(A) := \mu^2(\{(x,y) \in S^d \times S^d | x \in A, y \notin A \text{ and } \langle x, y \rangle \geq \rho\})$ for any $\rho \in [-1, 1]$.