Approximation Algorithms

* Intro.
* GW-Max Cut
* Hardness of approximation, PCP theorem

* Intro: Max Cut
  Input: Graph $G=(V,E)$ undirected
  Output: Partition of G: $U, V\setminus U$ s.t. max edges are cut.

- NP-hard : Solving optimally $\longrightarrow$ Solving SAT
- We will consider solving approximately
- $Opt(G)$ = value of optimal solution
  We will find solution U of value $C(U) \geq \alpha\, Opt(G)$
  α = approximation ratio $\in [0,1]$ (larger the better)

- Alg 1: Choose U randomly : $\forall i \; Pr[i \in U] = 1/2$ iid
  Claim: will give expected ratio $1/2$
  Proof: $Opt(G) \leq |E|$
  $$E[c(U)] = \sum_{e \in E} e \text{ is Cut} = \frac{1}{2}|E| \geq \frac{1}{2} Opt(G)$$

- Alg 2: derandomize alg 1:
  Will find distribution over cuts U with poly. support
  s.t. $E_U[c(U)] \geq \frac{1}{2} Opt(G)$

  Pairwise independent hashing:
  Let $|U| = 2^k$ consider $GF[2^k]$ let $F = \{ f_{a,b} \mid a,b \in GF[2^k] \}$
  $$f_{a,b}(x) = (ax+b) \bmod 2$$
  For each a,b f_{ab} defines a Cut U: $U_{a,b} = \{ x \mid f_{a,b}(x) = 0 \}$
  We have: $\forall_x \forall_y \, Pr_{a,b}[f_{ab}(x) \neq f_{a,b}(y)] = 1/2$ thus:

- Pick random Cut in F : $E_{ab}[c(U_{a,b})] = \sum_{e \in E} e \text{ is cut} = \frac{1}{2}|E|$

* Alg 3: SDP [Goemans Williamson]

- will give ratio $\alpha \sim 0.87$
- Can one do better? No, under "unique label cover".

- SDP: $A_{n \times n}$ symmetric matrix

  A is PSD: all eigen values ≥ 0

- Fact 1: A PSD iff: $\exists n$ vectors in $\mathbb{R}^n$ $v_1 \dots v_n$ st $A = [a_{ij}]$ & $a_{ij} = \langle v_i, v_j \rangle$

Proof: Let $v_1 \dots v_n$ be eigen vectors with values $\lambda_1 \dots \lambda_n$

$$A \begin{bmatrix} v_1 \cdots v_n \end{bmatrix} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \begin{bmatrix} v_1 \cdots v_n \end{bmatrix} \qquad \text{take } v_1 \cdots v_n \text{ orthonormal}$$

$$A = \left(\begin{bmatrix} v_1 \cdots v_n \end{bmatrix} \begin{bmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{bmatrix} \right) \left(\begin{bmatrix} v_1 \cdots v_n \end{bmatrix} \begin{bmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{bmatrix} \right)^T \Rightarrow A = B^T B$$

$$A = B^T B \Rightarrow x^T A x = x^T B^T B x = \langle Bx, Bx \rangle \geq 0$$

$$v \text{ eigen vector} \quad v^T A v = v^T \lambda v = \lambda \|v\| \geq 0 \Rightarrow \lambda \geq 0$$

This gives a geometric interp. of SD matrices.

- Fact 2: LP:

$$\left.\begin{aligned} &\text{Max} \ \textstyle\sum c_i x_i \\ &\text{st} \quad Ax \leq b \\ &\qquad\ x \geq 0 \end{aligned}\right\} \text{ solvable in poly time (\# variables, size of coefficients)}$$

SDP:

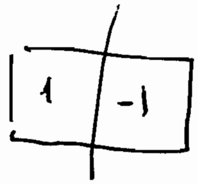
$$\left.\begin{aligned} &\text{Max} \ \textstyle\sum c_i x_i \\ &\text{s.t.} \quad Ax \leq b \\ &\qquad\ \underset{\text{PSD}}{[x]} \succeq 0 \end{aligned}\right\} \text{ "solvable" in poly time by "Elipsoid method"}$$

SDP:

$$\left.\begin{aligned} &\text{Max} \ \textstyle\sum c_{ij} \langle v_i, v_j \rangle \\ &\quad A \big[\langle v_i, v_j \rangle\big] \leq b \\ &\quad v_1 \cdots v_n \in \mathbb{R}^n \end{aligned}\right\} \text{ also "solvable" in poly}$$

- Max - Cut :

$$\left.\begin{array}{l} \text{Max} \sum_{ij \in E} \frac{1 - x_i x_j}{2} \\ x_i \in \{-1, 1\} \end{array}\right\}$$



NP - hard to solve

$$\left.\begin{array}{l} \text{Max} \sum_{ij \in E} \frac{1 - \langle v_i, v_j \rangle}{2} \\ v_i \in R^1 \\ \|v\| = 1 \end{array}\right\}$$

as above

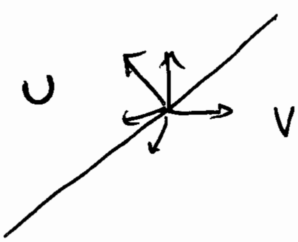
$$\left.\begin{array}{l} \text{Max} \sum_{ij \in E} \frac{1 - \langle v_i, v_j \rangle}{2} \\ \|v_i\| = 1 \\ v_i \in R^n \end{array}\right\}$$

can solve! $\Rightarrow$ Solution = vectors in R^n

How do we get a cut from above solution:

Take random hyperplane through origin (Normal is a random vector in unit sphere)

U



$V \setminus U \equiv$ Cut

Analysis: $E[\text{edges cut}] = \sum_{ij \in E} E[ij \text{ is cut}] = \sum_{ij \in E} \Pr[ij \text{ is cut}] =$

$$= \sum_{ij \in E} \Pr[\,v_i, v_j \text{ cut by random hyperplane}\,] \overset{*}{\geq} \sum_{ij \in E} \alpha \left(\frac{1 - \langle v_i, v_j \rangle}{2} \right) =$$

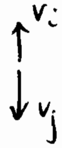
$$= \alpha \, SDP \geq \alpha \, Opt(G)$$

$$\alpha \sim 0.878$$

- Let v_i, v_j be vectors

- Need to prove: $\Pr[v_i\, v_j \text{ cut by random hyperplane}] \geq \alpha \left(\frac{1 - \langle v_i v_j \rangle}{2} \right)$

- Intuition:

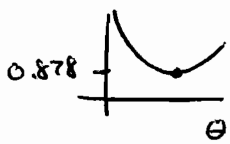


$\Pr = 1$ $\qquad \frac{1 - \langle v_i v_j \rangle}{2} = 1$

$\Pr = 0$ $\qquad \frac{1 - \langle v_i v_j \rangle}{2} = 0$

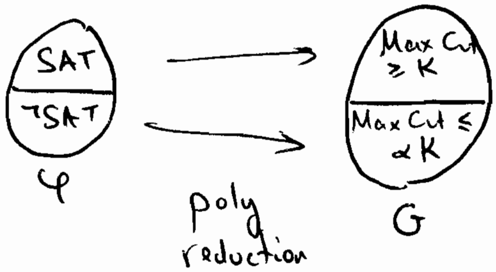
$\Pr[v_i\, v_j \text{ cut}] = \frac{\theta_{ij}}{\pi}$ (symmetry)

Find $\min_{\theta} \frac{\theta}{\pi} \frac{2}{(1 - \cos\theta)}$



- Alg 3: Solve SDP + round using random hyperplane gives expected app. ratio $\alpha \sim 0.878$, can derandomize.

* Hardness of Approximation:
  - How can we prove ~~that~~ upper bounds on approx. ratio.
  - General paradigm: Reductions



Distinguishing whether Max-Cut $\geq K$ or $\leq \alpha K$ is NP-hard $\xRightarrow{\text{implies}}$ approximating within ratio greater than α is NP-hard.

$\varphi \to G \to U$: $\begin{array}{l} c(U) \geq \alpha K \Rightarrow \varphi \in SAT \\ c(U) < \alpha K \Rightarrow \varphi \notin SAT \end{array}$

- "All we need" is a gap reduction.
- Will show main tool in these reductions: "PCP theorem".
- Remark: new result [Dinur 05] presents new reductions without PCP theorem.

* PCP: Probabilistically Checkable Proofs

- Typical proof for SAT: assignment $a_1 \cdots a_n$

SAT = variables $x_1 \cdots x_n$
clauses $(x_1 \cup \bar{x}_2 \cup x_5)$
$(x_7 \cup x_3 \cup \bar{x}_2) \cdots$

Typical verifier of proof: ① Reads assignment
② Decides in poly time

$$\exists V \quad \forall \varphi \quad \varphi \in SAT \quad \underset{\text{proof}}{\exists \pi} \quad V(\pi)=1$$
$$\varphi \notin SAT \quad \forall \pi \quad V(\pi)=0$$

- We will relax verification process: V random.

$$\exists V \quad \forall \varphi \quad \varphi \in SAT \quad \exists \pi \quad V(\pi)=1 \quad \text{w.p. } 1 \quad (\text{with probability } 1)$$
$$\varphi \notin SAT \quad \forall \pi \quad V(\pi)=0 \quad \text{w.p} > 1/2$$

But will require: V: ① Reads only const. number of bits of π
② Decides
$\Big\}$ picking location + decision in poly time.

- We will prove a "weak" theorem in which such V exists but π is exponential in length. The "strong" version (π polynomial) will not be shown.

- Will now show connection between PCP & Approximation

Connection between PCP & Approximation

$\exists$ Strong PCP $\Rightarrow$ 3-SAT hard to approximate beyond some constant

Let V be Verifier of PCP: $V(q_1 \ldots q_K)$ (K constant)

$V \equiv \hat{\varphi}_{V,r} \in$ 3-SAT (express V using clauses of length 3)

[assume V uses randomness for choosing queries only]

Let $\hat{\varphi}$ = concatination of $\hat{\varphi}_{V,r}$ for every random string r.

Now: $\varphi \in SAT \Rightarrow \hat{\varphi} \in SAT$

$\varphi \notin SAT \Rightarrow \hat{\varphi} \notin SAT$ & const. fraction of clauses of $\hat{\varphi}$ are not Satisfied.

3-SAT hard to approximate $\Rightarrow$ Strong PCP

Define V: Given φ let $\hat{\varphi}$ be s.t. $\varphi \in SAT \rightarrow \hat{\varphi} \in SAT$

$\varphi \notin SAT \rightarrow \hat{\varphi} \notin SAT$ & const. frac. of clauses of $\hat{\varphi}$ not sat.

Now V picks clause of $\hat{\varphi}$ at random and queries proof $q_1\, q_2\, q_3$

If Clause $(q_1\, q_2\, q_3) = T$ $V = 1$

$\varphi \in SAT \Rightarrow V = 1$ always

$\varphi \notin SAT \Rightarrow$ No matter what π is there is a const. chance that V will pick unsat. clause.

Proof of weak PCP:

① Will show proof system for QP-SAT: Input: m poly's Q_i of deg. 2 in $x_1 \ldots x_n$
Output: yes if Q_i have common root.

QP-SAT $\equiv$ SAT

② Actually will show proof system for single poly. P and then show how to obtain ① above:

Find V st. $\exists a\ P(a) = 0 \rightarrow \exists \pi\ V(\pi) = 1$ always

$\forall a\ P(a) \neq 0 \rightarrow \forall \pi\ V(\pi) = 0$ w.p. $> 1/2$

(3) Proof idea:

* Let $a_1 \cdots a_n$ be root of P.
* Let $A_a(x) = \sum a_i x_i$ $\quad (x = x_1 \cdots x_n)$

  Let $B_a(y) = \sum a_i a_j y_{ij}$ $\quad (y = y_{11} \cdots y_{nn})$
* If $P(a) = 0$ & $\pi = A_a(x), B_a(x)$

  Then 2 queries suffice to verify that $P(a) = 0$

  $P(a) = S_0 + \sum s_i a_i + \sum s_{ij} a_i a_j$

  In A query position $S_p = s_1 \cdots s_n$

  " B " " $S_q : s_{11} \cdots s_{nn}$

* So we will construct V st:

  I: Given π check if $\pi = A_a(x)\ B_a(y)$ for some $a = a_1 \cdots a_n$

  II: If yes ask 2 queries and decide.

(4) Details: <u>Test 1:</u> Linearity test:

Test will pass if A, B linear fail whp if "not" ← far from linear

Test = pick random $X_1 = (x_1^1 \cdots x_n^1)$

$X_2 = (x_1^2 \cdots x_n^2)$

check if $A_a(x_1) + A_a(x_2) = A(x_1 + x_2)$

Nice proof using Discrete Fourier transform.

<u>Test 2:</u> Consistency check:

We know: A close to $\hat{A}_a$ $\quad a = a_1 \cdots a_n$

B close to $\hat{B}_b$ $\quad b = b_{11} \cdots b_{nn}$

Test if $b_{ij} = a_i a_j$

will pass if $b_{ij} = a_i a_j$

will not pass whp if $\exists ij$ st. $a_i a_j \neq b_{ij}$

Test = "access $\hat{A}_a$ at X_1 & X_2" [should be

Let $\alpha = \hat{A}_a(X_1) \cdot \hat{A}_a(X_2)$ $\quad \alpha = \sum_{ij} a_i a_j x_i^1 x_j^2$]

"access" $\hat{B}_b$ at $y = \{y_{ij} = x_i^1 x_j^2\}$

$\beta = \hat{B}_b(y)$ [should be $\beta = \sum b_{ij} x_i^1 x_j^2$]

* Will a linear code work?

All linear codes ~~are~~ have rate $R < 1-H(p)$ (Korn bound).

Now check $\alpha = \beta$.

If $M_1 = [a_i a_j]$ $\quad \alpha = \cancel{x_1^T M_1 x_2}\ x_1^T M_1 x_2$
$M_2 = [b_{ij}]$ $\quad \beta = x_1^T M_2 x_2$

If $\exists ij$ $a_i a_j \neq b_{ij}$ & $x_1 + x_2$ chosen at random $\Rightarrow$
w.p. $1/4$ $\alpha \neq \beta$

How to access $\hat{A}$, $\hat{B}$.

To access x, chose random r access $A(r)$, $A(r+x)$
and add : $A(r) + A(r+x) \stackrel{?}{=} \hat{A}(x)$

A close to $\hat{A}$ $\Rightarrow$ w.h.p. $A(r) = \hat{A}(r)$, $A(r+x) = \hat{A}(r+x)$ $\Rightarrow A(r) + A(r+x) = \hat{A}(x)$

Test 3: Evaluate:

We know A close to $\hat{A}_a$
B close to $\hat{B}_a$

Access $\hat{A}_a(S_1)$
$\hat{B}_a(S_2)$

compute $P(a)$

(5) How to obtain result for QP-SAT. Cannot query each Q_i
Chose random $r = r_1 \dots r_n$ define $P(x) = \sum r_i Q_i(x)$

$\exists a \ \forall i \ Q_i(a) = 0 \rightarrow \exists a \ P(a) = 0$

$\forall a \ \exists i \ Q_i(a) \neq 0 \rightarrow \forall a \ (P(a) \neq 0$ w.p. $1/2)$