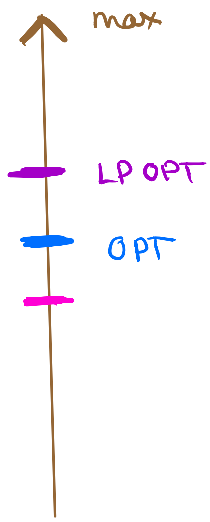
Today

Randomized rounding of SDPs

- MAX CUT
- 3-coloring



worst case ratio

$\frac{\text{LP OPT}}{\text{OPT}}$ called integrality gap.

MAXCUT

Input: $G=(V,E)$ $\quad w_{ij} \quad \forall (i,j)\in E$

Goal: partition vertex set so as to max weight of endpts crossing cut.

IP formulation of MAXCUT

$$x_i = \begin{cases} 0 & \text{on one side of partition} \\ 1 & \text{on other side} \end{cases}$$

$$z_{ij} = \begin{cases} 1 & \text{edge } (i,j) \text{ cut} \\ 0 & \text{o.w.} \end{cases}$$

$$\max \sum_{(i,j)\in E} w_{ij} z_{ij}$$

$$z_{ij} \le x_i + x_j \quad \forall (i,j)\in E$$

$$z_{ij} \le 2-(x_i+x_j) \quad \forall (i,j)\in E$$

$$x_i \in \{0,1\} \quad i\in V$$

$$z_{ij} \in \{0,1\} \quad \forall (i,j)\in E$$

* no polynomial sized LP relaxation of MAXCUT has integrality gap $> \frac{1}{2}$.

Another approach:

First, notation change

$\forall i$: $x_i \in \{-1, +1\}$

define $y_{ij} = x_i x_j \quad \forall i,j \in V$

$$\max \sum_{(i,j)\in E} w_{ij} \mathbb{1}[x_i \neq x_j]$$

Exactly captures MAX CUT!

Want $\exists x_i \ \forall i \in V$
s.t. $y_{ij} = x_i x_j \quad \forall i,j)$

$$\max \sum_{(i,j)\in E} w_{ij} \tfrac{1}{2}(1 - y_{ij})$$
$$y_{ij} = y_{ji} \qquad \forall i,j \in V$$
$$y_{ii} = 1 \qquad \forall i \in V$$

Idea: enforce brown by adding linear inequalities to purple.

SDP rounding

Intro to semi-definite programming

linear programming where vars are entries in psd matrix

Defn If A is a symmetric n by n matrix
then A is a positive semidefinite (psd) matrix $\equiv A \succeq 0$
if any of the following equivalent conditions hold

1. $\forall \vec{c} \in \mathbb{R}^n$, $c^T A c \geq 0$
2. A has nonnegative eigenvalues
3. $A = V^T V$ for some $m \times n$ matrix V, $m \leq n$
4. $A = \sum_{i=1}^{n} \lambda_i x_i x_i^T$ for some $\lambda_i \geq 0$ and orthonormal vectors $x_i \in \mathbb{R}^n$

SDP rounding

Intro to semi-definite programming

linear programming where vars are entries in psd matrix

Defn If A is a symmetric n by n matrix then A is a positive semidefinite (psd) matrix $\equiv A \succeq 0$ if any of the following equivalent conditions hold

1. $\forall \vec{c} \in \mathbb{R}^n$, $c^T A c \geq 0$
2. A has nonnegative eigenvalues
3. $A = V^T V$ for some $m \times n$ matrix V, $m \leq n$
4. $A = \sum_{i=1}^{n} \lambda_i x_i x_i^T$ for some $\lambda_i \geq 0$ and orthonormal vectors $x_i \in \mathbb{R}^n$

Semidefinite program (SDP)

max or min $\sum_{i,j} c_{ij} x_{ij}$

subject to $\sum_{i,j} a_{ijk} x_{ij} = b_k$

$x_{ij} = x_{ji} \quad \forall i,j$

$X = (x_{ij}) \succeq 0$

$\equiv$ Vector program

max or min $\sum_{i,j} c_{ij} (v_i \cdot v_j)$

subject to $\sum_{i,j} a_{ijk} (v_i \cdot v_j) = b_k$

$v_i \in \mathbb{R}^n \quad i = 1, \ldots, n$

given $X \Rightarrow X = V^T V$, set v_i to be i^{th} col of V

Key fact:

SDPs can be solved to within additive error ε in time poly(size of input, $\log(\frac{1}{\varepsilon})$)

in our discussions, we ignore additive error ε

Recap:

①

$$\boxed{\begin{array}{ll}\text{Want } \exists\, x_i & \forall i \in V \\ \text{s.t. } y_{ij} = x_i x_j & \forall i,j\end{array}}$$

$$\boxed{\begin{array}{lll}\max & \sum_{(i,j)\in E} w_{ij} \tfrac{1}{2}(1-y_{ij}) & \\ & y_{ij} = y_{ji} & \forall i,j \in V \\ & y_{ii} = 1 & \forall i \in V\end{array}}$$

Opt solution to brown + purple = Opt of MAXCUT

② Brown $\Longrightarrow$

$$(c_1\ c_2, \dots, c_n) \underbrace{\begin{pmatrix} & & \\ & y_{ij} & \\ & & \end{pmatrix}}_{Y} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} \geq 0 \quad \forall \vec{c} \in \mathbb{R}^n$$

These constraints $c^T Y c \geq 0 \quad \forall c \in \mathbb{R}^n$

$\equiv$ Y is psd matrix!

③ Yields a semidefinite programming relaxation of MAXCUT

$$\boxed{\begin{array}{lll}\max & \sum_{(i,j)\in E} w_{ij} \tfrac{1}{2}(1-y_{ij}) & \\ & y_{ij} = y_{ji} & \forall i,j \in V \\ & y_{ii} = 1 & \forall i \in V \\ \text{plus} & \boxed{Y = \begin{pmatrix} y_{11} & \cdots & y_{1n} \\ y_{n1} & \cdots & y_{nn}\end{pmatrix} \text{ psd}} & \end{array}}$$

can be solved eficiently using the ellipsoid alg. up to error ε.

We can solve this, "round" results $\Longrightarrow$ int soln

$\Longrightarrow$ prove that it gives pretty good approx.

Can equivalently write SDP relaxation as a vector program

$$\max \sum_{(i,j)\in E} w_{ij} \tfrac{1}{2}(1-y_{ij})$$

$$y_{ij} = y_{ji} \qquad \forall i,j \in V$$

$$y_{ii} = 1 \qquad \forall i \in V$$

$$Y = \begin{pmatrix} y_{11} & \cdots & y_{1n} \\ & & \\ y_{n1} & \cdots & y_{nn} \end{pmatrix} \text{ psd}$$

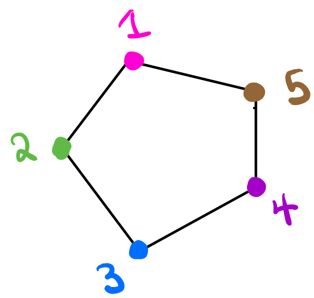
Relaxation check: $\quad \text{OPT} \leq \text{OPT}_{SDP}(G)$

$$\max \sum_{(i,j)\in E} w_{ij} \tfrac{1}{2}(1-\vec{v}_i \cdot \vec{v}_j)$$

$$v_i \cdot v_i = 1$$

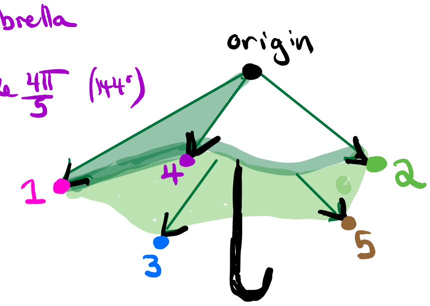
i.e. v_i's are unit vectors $\in \mathbb{R}^n$

Example



Lovász Umbrella

all edges at angle $\frac{4\pi}{5}$ (144°)



Lovász umbrella

all edges at angle $\frac{4\pi}{5}$

144°

$$\cos\left(\frac{4\pi}{5}\right) = -\frac{\phi}{2} \approx -.8$$

golden ratio $\frac{1+\sqrt{5}}{2}$

$$\max \sum_{(i,j)\in E} w_{ij} \underbrace{\tfrac{1}{2}\left(1-\cos(\text{angle}(v_i, v_j))\right)}_{\approx\ 0.9}$$

all weights equal

OPT = 4

SDP OPT ≈ 4.5

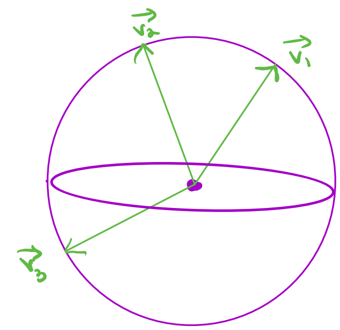
$$\text{ratio} \approx \frac{4}{4.5} \approx 0.89$$

MAXCUT

Input: $G=(V,E)$ $\quad w_{ij} \quad \forall (i,j)\in E$

Goal: partition vertex set so as to max weight of endpts crossing cut.

Vector programming relaxation



$$\max \quad \frac{1}{2}\sum_{(i,j)\in E} w_{ij}\,(1-\vec{v}_i\cdot\vec{v}_j)$$

$$\vec{v}_i\cdot\vec{v}_i = 1 \qquad \forall i\in V$$

$$\vec{v}_i \in \mathbb{R}^n$$

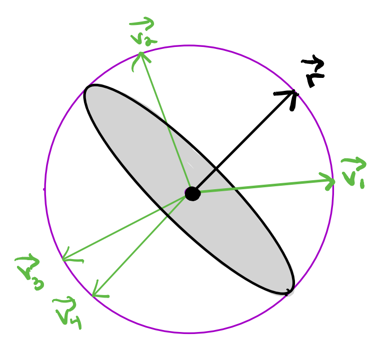
Can solve SDP in poly time.

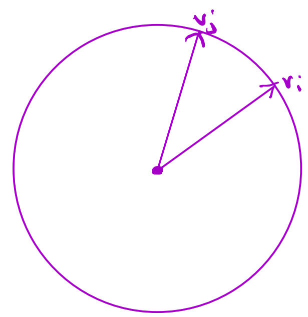
$\boxed{\text{Claim:}}$ MAXCUT OPT $\leq$ SDP OPT

But how to round? get large contribution to OPT when $v_i\cdot v_j$ very -ve

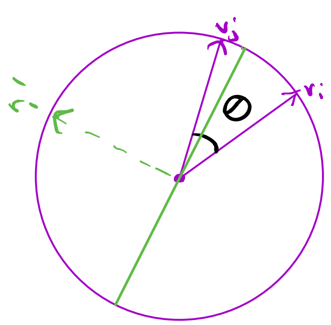
Random hyperplane rounding

Solve SDP $\longrightarrow v_1^*, v_2^*, \ldots, v_n^*$
pick random hyperplane thru origin
partition vertices based on which side of hyperplane





v_j
r_i



v_j
r_i
θ
r_i'

Hardness

① If $\exists$ approx alg for MAX CUT with approx ratio ≥ 0.941, then $P = NP$.

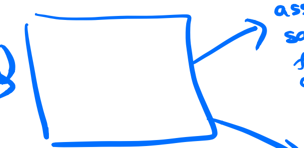
② If the "unique games conjecture" is true, there is no approx alg for MAXCUT with approx ratio better than 0.878

Unique Games Conjecture

$\forall \varepsilon > 0$, $\exists$ prime q s.t.

$(1-\varepsilon, \varepsilon)$ Gap version

MAX2LIN(q) instance



if $\exists$ assignment satisfying $\geq 1-\varepsilon$ fraction of eqns output Yes

if no assignment satisfies $> \varepsilon$ fraction output NO

admits no poly time soln unless $P = NP$

MAX2LIN(q)

q prime

input: linear equations mod q w/ unknowns $x_1, \ldots, x_n \in \{0, 1, \ldots, q-1\}$

(form $x_i - x_j = c$)

$x_3 - x_{11} \equiv 87 \pmod{97}$

$x_7 - x_{22} \equiv 3 \pmod{97}$

$\vdots$

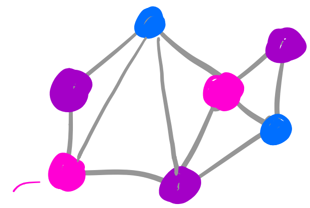
$x_7 - x_{19} \equiv 56 \pmod{97}$

Problem: Find assignment of x_i's that satisfies max possible # of eqns

③ Int gap of the [GW] SDP $= 0.878\ldots$

③ Every poly sized LP relaxation of MAXCUT has integrality gap of $\frac{1}{2}$. Janes

3-Coloring a 3-colorable graph



Given graph $G=(V,E)$
& promise that it is 3-colorable

What is min k s.t. we can find a k-coloring of G in poly time?

Simple results:

① A graph with max degree Δ can be colored with $\leq \Delta+1$ colors

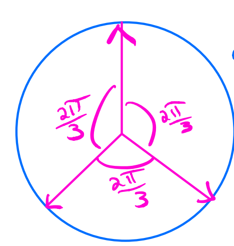
② A 3-colorable graph can be colored with $O(\sqrt{n})$ colors.

Find a vertex of deg $\geq \sqrt{n}$
Use 3 colors to color it & its neighbors (neighborhood 2-colorable)
Remove it & its neighbors from graph

An SDP-based alg

$$\begin{aligned} \min\ & \lambda \\ \text{s.t. } & \vec{v}_i \cdot \vec{v}_j \leq \lambda \quad \forall (i,j) \in E \\ & \vec{v}_i \cdot \vec{v}_i = 1 \quad \forall i \in V \\ & \vec{v}_i \in \mathbb{R}^n \end{aligned}$$

Claim:
if graph is 3-colorable
$\lambda \leq -\frac{1}{2}$



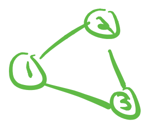
$\cos\left(\frac{2\pi}{3}\right) = -\frac{1}{2}$

$$\begin{aligned} &\min \ \lambda \\ \text{s.t. } &\vec{v}_i \cdot \vec{v}_j \le \lambda \quad \forall (i,j) \in E \\ &\vec{v}_i \cdot \vec{v}_i = 1 \quad \forall i \qquad (*) \\ &\vec{v}_i \in \mathbb{R}^n \quad \forall i \end{aligned}$$

Claim:
if graph is 3-colorable
$\lambda \le -\frac{1}{2}$

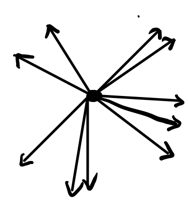
Aside: If G has a triangle, then optimal soln to SDP has $\lambda^* \ge -\frac{1}{2}$

Proof: Suppose



$$0 \le \left(\vec{v}_1+\vec{v}_2+\vec{v}_3, \vec{v}_1+\vec{v}_2+\vec{v}_3\right) = \vec{v}_1\cdot\vec{v}_1 + \vec{v}_2\cdot\vec{v}_2 + \vec{v}_3\cdot\vec{v}_3 + \vec{v}_1\cdot\vec{v}_2 + \vec{v}_1\cdot\vec{v}_3 + \vec{v}_2\cdot\vec{v}_1 + \vec{v}_2\cdot\vec{v}_3 + \vec{v}_3\cdot\vec{v}_1 + \vec{v}_3\cdot\vec{v}_2$$

Algorithm



1. Solve SDP $(*)$ $\Rightarrow$ v_i^* $i=1,\dots,n$
2. Choose t random hyperplanes thru origin
3. Color vertices in each region w/ diff color
4. remove any edges properly colored
5. Repeat steps 2-4 until have proper coloring

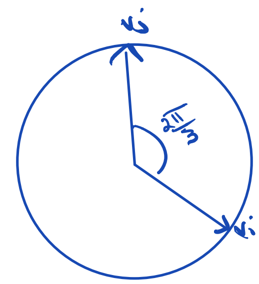
One execution of step 2 uses 2^t colors.

Goal: produce $\underbrace{\text{semi-coloring}}$ w.p. $\geq \frac{1}{2}$ $(*)$

coloring of nodes s.t.
$\leq \frac{n}{4}$ edges have same color at both endpts
$\Rightarrow$ at least $\frac{n}{2}$ vertices properly colored.

Observation: if k-colors sufficient to get semi-coloring,
$\Rightarrow$ graph can be properly colored with $O(k \log n)$ colors

What should t be to guarantee $(*)$?



Fix $(i,j) \in E$
$\Pr(i \,\&\, j$ get same color$)$

$\Rightarrow E(\#$ edges with same color$)$

Let Δ^* be a parameter

1. Pick a vertex of deg $\geq \Delta^*$ & 3-color it & neighbors
2. Repeat step 1 until all vertices have degree $\leq \Delta^*$

$\Big\}$ $\leq 3\frac{n}{\Delta^*}$ colors

3. Run SDP-based alg to color rest $\Big\}$ $\tilde{O}(\Delta^{*\,\log_3 2})$ colors

Choose Δ^* to minimize $\quad \dfrac{3n}{\Delta^*} + (\Delta^*)^{\log_3 2}$

$$\Rightarrow \quad \Delta^* = n^{\log_6 3} \quad \Rightarrow \quad \tilde{O}(n^{0.39})$$

Current best: $\quad O(n^{0.199})$

NP-hard to color with 4 colors

$\boxed{\text{Huge open problem:}}$ Is there an alg for 3-coloring a 3-colorable graph that uses polylog n colors?

Next time: we'll use linear programming duality

- lower bounds on randomized alg complex
- design randomized algs for online problems