Approximate Graph Coloring by Semidefinite Programming

David Karger, Rajeev Motwani, Madhu Sudan

Presented by:
Dan Pelleg

First: M symm. PD $\Rightarrow \exists$ square real U s.t. $U^t \cdot U = M$.

Look at cols of U as vecs v_i. Then $m_{ij} = \langle v_i, v_j \rangle$.

Also, U can be found in polytime.

So, SDP actually:

$$\text{minimize:} \quad \sum c_{ij} \cdot \langle v_i, v_j \rangle$$

s.t.:

$$\textstyle\sum_k a_{k\ell} \cdot \langle v_i, v_j \rangle \quad \leq \quad b_\ell \qquad\qquad 1 \leq \ell \leq m$$

Given the graph $G = (V, E)$, our SDP will be:

$$\text{minimize} \quad \alpha$$

s.t.:

$$\begin{array}{rcll} \langle v_i, v_j \rangle & \leq & \alpha & \forall (i,j) \in E \\ v_i & \in & S_n & \forall i \in V \end{array}$$

From now on, assume G is 3-colorable. Theorem:

$$\boxed{\alpha \leq -\frac{1}{2}}$$

Proof:

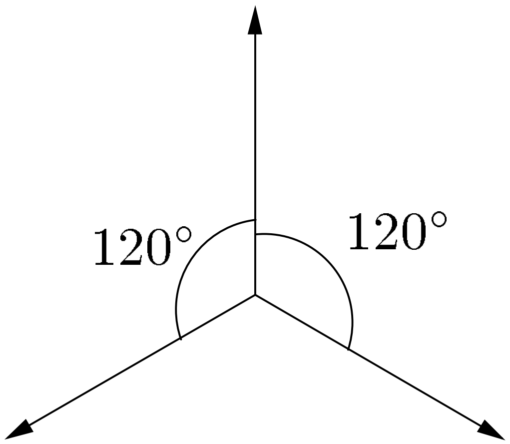
1. Show there are 3 unit vectors $\in \mathcal{R}^n$ s.t. dot product of any pair is $\leq -\frac{1}{2}$.
2. Each monochromatic set is assigned one such vector. We get a feasible solution with value $\leq -\frac{1}{2}$.

Quick reminder: dot product of unit vectors = cosine of angle between them.

$$\cos(0) = 1 \qquad \cos(\pi = 90^\circ) = 0$$

$$\cos(2\pi = 180^\circ) = -1 \quad \cos(\tfrac{2}{3}\pi = 120^\circ) = -\tfrac{1}{2}$$

So we can show 3 vectors in the plane (obviously will work for $n > 2$):



What do we have up to now?

- G is 3-colorable.
- Each $i \in V$ has a vector v_i s.t.

$$\langle v_i, v_j \rangle \leq -\frac{1}{2} \qquad \forall (i,j) \in E$$

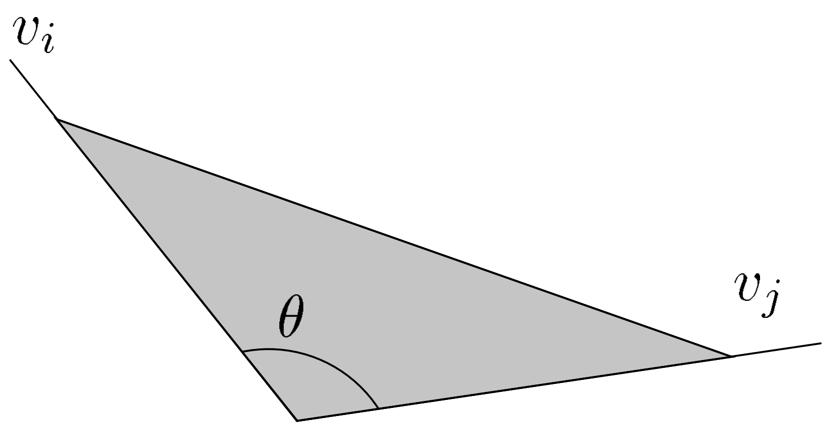
- Our SDP guarantees that adjacent vertices in G repel each other.

Now what? CUT!

Choose random hyperplane C through the origin. Partition by graph nodes to those "above" and "below" the hyperplane. More precisely, C "Separates" i, j if $\mathrm{sgn}(c \cdot v_i) \neq \mathrm{sgn}(c \cdot v_j)$.

- (Notoriously misleading 3-D example).
- This is another way to split a 3-colorable graph into two pieces so that neither has a triangle.

Why would cutting help? Look at plane defined by v_i, v_j, origin:



Prob(C seperates i, j) = Prob(C in shaded area) = $\frac{\theta}{\pi}$.

Since $\alpha = \cos\theta \le -\frac{1}{2}$, we are guaranteed $\frac{\theta}{\pi} \ge \frac{2}{3}$.

Algorithm: Choose $r = 2 + \lceil \log_3 d \rceil$ cutting hyperplanes independently. Get at most 2^r regions. Assign a distinct color to each.

Each edge "cut" with probability:

$$1 - \left(\frac{1}{3}\right)^r = 1 - \frac{1}{9d}$$

Expected $\sharp$ edges which are not cut is:

$$\frac{m}{9d} \leq \frac{n}{18} < \frac{n}{8} \qquad \text{(since } m \leq \tfrac{nd}{2}\text{)}$$

Markov: $\Pr(\sharp \text{ uncut edges} > \frac{n}{4}) \leq \frac{1}{2}$.

For each uncut edge, delete one endpoint arbitrarily (don't paint it).

Where are we?

- With pr. $> \frac{1}{2}$ have coloring for at least 3/4 of nodes.
- No. of colors used $= 2^r = O(d^{\log_3 2})$.

For remaining nodes, apply same method using fresh palette.

Let $C_n = \sharp$ colors used on n-node graph. Then have

$$C_n \leq C_{n/4} + d^{\log_3 2}$$

Solved by:

$$C_n \leq d^{\log_3 2} \log n$$

A different analysis: use Wigderson degree-reduction technique.

Our goal is n^β colors (and we can actually achieve it using just SDP, for not-so-good β). Following Wigderson, we assume that $d < n^{1-\beta}$.

$$
\begin{aligned}
n^\beta &\leq \left(\frac{n}{4}\right)^\beta + \left(n^{1-\beta}\right)^{\log_3 2} \\
\beta &< (1-\beta)\log_3 2 \\
\beta &= 0.631/1.631 \\
&= 0.387
\end{aligned}
$$

Summary

- Ask SDP to place adjacent nodes far from each other.
  - Will succeed, since graph is k-colorable.
- Cutting hyperplanes are our "randomized rounding". Each region formed has a different color.
- Good probability that adjacent vertices will have different color. Good enough for a "semicoloring" that takes care of some of the nodes.
- For rest of nodes, re-apply method using new colors.
- Analyze to get $O(n^{0.387})$ coloring.
  - Best known is $O(n^{3/14}) = O(n^{0.21})$.