Lecture 25 - The PCP Theorem

[A proof sketch]

A gem in TCS. Circa 1990s
FGLSS/AS/ALMSS early '90s

Dinur '03: A new proof which is very short if you already have a lot of TCS tools in your back pocket. And we do! Didn't plan this for the course, but it recently occurred to me.

PCP Theorem: \( \exists \varepsilon_0 > 0 \) s.t. \((1 - \varepsilon_0)\)-approximating Max-3Sat is NP-hard.

[Handwritten note: \( \varepsilon_0 \) can be any \( \varepsilon \approx 10^{-20} \). Needs many more tools or something]

More precisely, finitely many Cook-Levin-type reductions from Circuit-SAT to Max-3Sat.

\[
\begin{align*}
\text{size } n & \Rightarrow \text{poly}(n) \text{-time alg } r \\
\text{poly}(n) \text{-time alg } r & \Rightarrow y_1, y_2, \ldots, y_{\text{poly}(n)} \text{ s.t. } \sum y_i = \text{poly}(n) \\
\text{poly}(n) \text{-time alg } r & \Rightarrow \text{every asgn to } y_i \text{ falsifies } \geq \varepsilon_0 \text{ frac. of clauses}
\end{align*}
\]

Why cool? "Peys viewpoin": Can verify NP statements w.h.p. while only looking at \( O(1) \) bits of a proof.

Merlin: "Hey Arthur, check out this circuit C. It's satisfiable!"
Arthur: "Oh yeah? Prove it." Merlin: "Sure, here's a sat. asgn for x's"
Arthur: "I'm feeling lazy today. Give me a sat. asgn for y's instead."
Merlin: "Okay, here. I wrote it on this piece of paper."
Arthur: *Picks one of the m cons of C at random* \( y \), say \( y_1 \)
Arthur: "Hmm, I'll read your asgn just to y_2, y_3, y_4" Merlin: "Checks out. I guess C was satisfiable."

"Wait, maybe I was cheated. Well I'll just do \( O(\varepsilon_0) \) more spot checks..."
The $\varepsilon_0 = 0$ ($\varepsilon_0 = \frac{1}{4m} = \frac{1}{4}$) case is just NP-hardness of 3-Sat.

Let its proof: Some of the $y$'s can be $x$'s. \( \Rightarrow \) 3-CNF contains the usual "proof" 1 gate values

= Encode gate consist, "output = 1" as 3Sat consists.

\[ x \text{ and } y \text{ and } z \rightarrow (x \vee y \vee z) \]

Nothing special act. 3Sat in PCP Thm. Any NP-hard CSP \( \text{const. arity (domain)} \) is ok. \( \Rightarrow \) changes.

Say \( |x|, |y|, |z| \) PCP Thm. proved with 3-Coloring (3-CNF).

```
   \[ C \rightarrow \text{color}(v_1) + \text{color}(v_2) \]
   \[ \text{color}(v_3) + \text{color}(v_4) \]
```

"gadget" \( e \) code \( v_4 \) with 2 $y$'s

\[ e \rightarrow 00 \]
\[ e \rightarrow 01 \]
\[ e \rightarrow 10 \]

Every coloring blows 4 3CNFs.

\[ \text{encode const. w/ at least 4CNFs} \]
\[ 3 \text{-var. CNFs, } \geq 8 \text{ 3CNFs} \]

\[ \Rightarrow \text{in each, at least } \frac{1}{8} \text{ 3CNFs violated} \]

\[ \Delta \rightarrow \frac{1}{8} \text{ 3CNFs violated} \]

\[ \text{from: And w.w.:} \text{ 3Sat} \rightarrow 3\text{col} \text{!} \]

\[ \text{On to proof!} \]

Define: Let $x$ be a CSP. \( \text{unsat}(x) \) = free of costs.

\[ \text{on arg.} \rightarrow \text{violated by best argn. to $x$} \]

"gadget" we'll consider, hence "$x$" be the graph.
$I \in \mathbb{R}$, $K < \infty$

Direct from $0 → 0$ is poly-time alg mapping $\mathcal{G} \rightarrow \mathcal{G}'$

s.t.: 1. $\text{Size}(\mathcal{G}') \leq k \cdot \text{Size}(\mathcal{G})$ 2. $\text{size} = \# \text{bits}$ 3. if $\text{unsat}(\mathcal{G}) < \varepsilon_0$ then $\text{unsat}(\mathcal{G}') > 2 \cdot \text{unsat}(\mathcal{G})$

PCP Theorem:

- Ck-set
- 3-Coloring
- $\log N$ times
- gadget to $\text{Sat}$ if you wish.

Remark: $K$ does not "know" $\text{unsat}$ vals. But "spreads out the difficulty."

Proof:


- $\mathcal{G} \rightarrow \mathcal{G}'$

- Side effects: $\text{Size}' \leq K \cdot \text{Size}$

- $|\Omega'| = 3$

- $\text{unsat}(\mathcal{G}') = 0 \Rightarrow \text{unsat}(\mathcal{G}) = 0$

- $\text{unsat}(\mathcal{G}') > \text{unsat}(\mathcal{G})$ 4. [all steps have this poly]

- $C_1 = \text{const.}$

- $8$-regular

- Side effects: same
3. Increase \( \Omega \) : \text{unsat}.

For any fixed \( t \):
\[
\text{size}' \leq 2^{2^{2t+1}} \cdot \text{size}
\]
\[
\left| \Omega' \right| \approx 3^{8^t}
\]

\[\text{[enormous domain, weird const.]}\]

(weird \( \Omega \) and costs on huge domain)

\[\text{unsat} \geq \frac{C_t}{\text{unsat}} \text{ if unsat} \leq \varepsilon_0.\]

4. 

\[\Box\]

Some reduces domain size.

\[\left| \Omega' \right| = 3, \quad \text{size}' \leq 2^{2^{2t+1}} \cdot \text{size}.
\]

\[\text{unsat}' \geq \frac{\text{unsat}}{c_3} \]

Put together:
\[\text{size}'''' \leq 2^{2^{2t+1}} \cdot \text{size}''''
\]
\[\left| \Omega'''' \right| = 3
\]

\[\text{unsat}'''' \geq 0 \text{ if unsat} \leq \varepsilon_0.
\]

Take \( t = \frac{c_1 c_2 c_3}{c_3} = O(1). \)
(Sketch) Replacement of each vertex by expander $I$

Let $A'$ be best assign for $G'$; $x'$ gets $\varepsilon = \text{unsat'} < \text{frac of consns wrong}. \text{ Define } A \text{ for } G \text{ by letting } A(v) = \text{label}(A'(v))$

A must violate $\Omega(\text{unsat}) < \text{frac of consns in } G$

For each constraint $c_i$ in $G$, either it's violated by $A'$ or $v_i$ or $v'_i$ gets a minority label for its cloud.

$\Rightarrow \Omega(\text{unsat}) < \text{frac of inter-cloud edges in } G'$

$\Rightarrow \Omega(\text{unsat}) \geq \text{frac of intra-cloud equal consns violated by } A'$

Remark: Only reason to use an expander here and not a random graph is so that the overall NP-hardness reduction is dense.

But not a huge deal to me to use a randomized reduction. Implies $(1-\varepsilon, 1)$-approx Max-3 sat $\in P$ $\Rightarrow NP \subseteq \mathbb{R}^P$. Good enough for me, Class I
Expanderize: Slap a $d$-regular expander on $G$, put equality constraint on edges.

Becomes an $8$-regular, half-as-good expander.
Input:

\[ G = \text{parity-2, domain } \{0, 1\} = 3, \text{ } \frac{1}{d} \text{-regular expander.} \]

\[ G' : \text{variable/true set: same as } G, \] \[ \text{edges/constit in } G' \equiv \text{paths of length } \ell \text{ in } G \]

\[ \text{new degree } \geq 8^\ell \]

\[ \text{new domain: } \Omega' = \Omega + 8^2 + \ldots + 8^\ell \quad (\text{size } 3 \times 3^\ell) \]

We think of a "label" for \( u \in G' \) as giving an opinion label (red, green, color) to its old distinct nbs

\( \ell = 3 \)

Constraints: between path endpoints:

"test everything"

In particular, for each edge \( (v, u) \) on a path \( (v, g) \), test that \( a \)'s color opinion for \( u \), together with \( g \)'s color opinion for \( u \) satisfies the old \((u, v)\) constraint in \( G \)
size' = \binom{n}{1} \cdot \text{size } V

\text{unsat} ?? Sketch...

Let \( A' \) be best assignment for \( G' \). Write \( A'(w)_v \) for \( w \) and \( A' \) of \( v \) in color.

Define assign \( A \) for \( G \) as follows:

\[ A(v) = \text{plurality of } \{ A'(w)_v \}. \]

A violates \( e : = \min(\text{unsat, } w : \text{dist}(w) = 1) \) frac of \( G \) constr.

Let \( F \) be the edges of \( G \) it violates

Goal: \( A' \) viol about \( \Omega(1) \) frac of all path constrs in \( G' \).

Let \( u \rightarrow v \rightarrow b \) be a path in \( G' \).

If it passes through an \( F \) edge \((uv)\), there is "const. chance that"

\[ A'(u)_v = A(a), \quad A'(v)_b = A(b) \]

because \( u \approx_t (1/2) \approx_t \approx_t (1/2) \approx_t \) and \( A \) does plurality vote.

This \( \Rightarrow A' \) violates the path constr. on \( u \rightarrow v \rightarrow b \).

\( \Delta \) unsat \( \approx 1/\alpha \) (frac of paths passing thru \( F \))

\( F \) is an unsat frac of \( G \) edges

path of len \( \alpha t \)

\( G \) is an expander

so edges are \( \approx \) "randomly" distr.

\[ \text{Prob path misses } F \approx_{\alpha t \cdot \text{unsat}} \]

\( \Rightarrow \) frac of paths hitting \( F \) \( \approx \text{unsat} \)