# CS 221 LECTURE 18 — 3/29/2018

TODAY: ~~PCP~~ • NP $\subseteq$ PCP$(O(n^2), 20)$

• Overview of Dinur's Proof of NP = PCP$(\log n, 1)$

---

**Recall PCP$(r(n), q(n))$**

• Verifier $V$ tosses $r(n)$ coins, reads $x$, determines $q$ queries $i_1, \ldots i_q \in [\ell]$ & predicate $f : \{0,1\}^q \to \{0,1\}$

• Queries $y_{i_1}, \ldots y_{i_q}$ & accepts iff $f(y_{i_1} \cdots y_{i_q}) = 1$.

---

**Today: PCP for Quadratic SAT**

• Input = polynomials $P_1, \ldots P_m \in \mathbb{F}_2[x_1 \ldots x_n]$
$\deg P_i \leq 2$ (

• Goal: $\exists\, a_1 \ldots a_n$ s.t. $\forall j$ $P_j(a_1 \ldots a_n) = 0$.

---

**Claim 1**: Quadratic SAT is NP-complete.

**Proof**: Exercise (Use AND - $\oplus$ - circuits are complete).

# PCP for Quadratic SAT

- Prover "Expected" to provide:

  1. $l(\bar{a})$ for every linear function $l$ of $x_1 .. x_n$
  2. $q(\bar{a})$ for every quadratic function $q$ of $x_1 .. x_n$

- Prover gives $\left(T_1[l]\right)_{l \in \mathbb{F}[x_1 .. x_n],\ \deg(l) \leq 1}$

  $\left(T_2[q]\right)_{q \in \ ..\ \deg(q) \leq 2}$.

- Verifier: (Ideally) Needs to check $\exists\, a$ s.t.

  for all $l$ $\quad T_1[l] = l(a)$

  for all $q$ $\quad T_2[q] = q(a)$

  for all $j \in [m]$ $\quad T_2[P_j] = 0$.

  - Unfortunately $\forall$ quantifiers are problems.
  - Will settle for weaker guarantees....

Revised Goals : Verify $\exists a$ s.t.

1. $\Pr_{\ell}\left[T_1[\ell] \neq \ell(a)\right] \leq .01$

2. $\Pr_{q}\left[T_2[q] \neq q(a)\right] \leq .01$

3. and somehow $\forall j \quad P_j(a) = 0$

still $\forall$ !

[ Neither $\forall j \; T_2[P_j] = 0$ nor $\Pr_j\left[P_j(a) \neq 0\right] \leq \ldots$ will work! ]

Testing 3 : (Assuming 2) :

Idea 1 : Arithmetize AND :

Pick $\alpha_1 \ldots \alpha_m \in \mathbb{F}_2$ at random

& verify $\left(\sum \alpha_j P_j\right)(a) = 0$

if $\exists j$ s.t. $P_j(a) \neq 0$ Then $\Pr_{\alpha}\left[(\;) \neq 0\right] \geq \frac{1}{2}$

Idea 2 : to compute $\left(\sum \alpha_j P_j\right)(a)$.

Can't read $T_2\left[\sum \alpha_j P_j\right]$ ... (might be in the .01 fraction of errors)

... Instead Use "Worst-case to avg-case reduction"

Verify "$T_2[q + \sum \alpha_j P_j] = T_2[q]$" for random $q$.

Analysis: Assuming 2 :

- if $\exists j \; P_j(a) \neq 0$

  then $\Pr_{\alpha}\left[ (\sum \alpha_j P_j)(a) \neq 0 \right] \geq \frac{1}{2}$

- if $(\sum \alpha_j P_j)(a) \neq 0$

  then $\Pr_{q}\left[ T_2[q + \sum \alpha_j P_j](a) \neq (q + \sum \alpha_j P_j)(a) \right] \leq .01$

  $\Pr_{q}\left[ T_2[q] \neq q(a) \right] \leq .01$

  $$\Rightarrow \Pr_{q}\left[ T_2[q + \sum \alpha_j P_j] - T_2[q] \neq \underbrace{(q + \sum \alpha_j P_j)(a) - q(a)}_{= (\sum \alpha_j P_j)(a) \neq 0} \right] \leq .02$$

  $$\Rightarrow \Pr_{q}\left[ T_2[q + \sum \alpha_j P_j] = T_2[q] \right] \leq .02$$

(if $\exists j$ s.t. $P_j(a) \neq 0$ & assuming 2)

Combining $\Pr_{\alpha, q}\left[ T_2[q + \sum \alpha_j P_j] = T_2[q] \right] \leq .52$

Getting 1 & 2 : NON-TRIVIAL !!

~~e.~~ e.g. $2^{2^n}$ possibilities for $(T_1[\ell])_\ell$

$2^n$ possibilities for $a \in \{0,1\}^n$.

- ~~Idea~~ "Linearity Testing" [Blum-Luby-Rubinfeld]

Idea: if $T_1[\ell] = \ell(a)$ $\forall \ell$, then

$$T_1[\ell_1] + T_2[\ell_2] = T_1[\ell_1 + \ell_2] \quad \forall \ell_1, \ell_2$$

Check above for random $\ell_1, \ell_2$ !!

- Key Theorem: if $\forall a$, $\Pr_\ell [T_1[\ell] \neq \ell(a)] \geq \delta$

Then $\Pr_{\ell_1, \ell_2} [T_1[\ell_1] + T_2[\ell_2] \neq T_1[\ell_1 + \ell_2]]$

$$\geq \min \left\{ \frac{\delta}{2}, \frac{2}{9} \right\}.$$

Proof omitted. Not too hard. But very different...

Conclude: if $\neg$ 1 then reject w.p. .005.

## Getting ②.

**Idea 1:** Test $T_2[q_1] + T_2[q_2] = T_2[q_1 + q_2]$

for random $q_1, q_2$

— x —

$\left[ q(x) = \sum q_{ij} x_i \right.$

**Key Theorem** $\Rightarrow \exists (b_{ij})_{i,j \in [n]}$ s.t.

$$\Pr_q \left[ T_2[q\cancel{(x)}] \neq \sum q_{ij} b_{ij} \right] \leq .01$$

(or else ~~pro~~ test rejects w.p. $0.005$).

— x —

But also need $b_{ij} = a_i \cdot a_j \quad \forall i, j$!

$$\begin{bmatrix} \\ & b_{ij} & \\ \\ \end{bmatrix} = \begin{bmatrix} \\ a_i \\ \\ \end{bmatrix} \begin{bmatrix} & a_j & \end{bmatrix}$$

Idea. Pick random $c_1 \ldots c_n$, $d_1 \ldots d_n$ & verify

$$\begin{bmatrix} & c_i & \end{bmatrix} \begin{bmatrix} \\ & b_{ij} & \\ \\ \end{bmatrix} \begin{bmatrix} \\ d_j \\ \\ \end{bmatrix} = \begin{bmatrix} & c_i & \end{bmatrix} \begin{bmatrix} \\ a_i \\ \\ \end{bmatrix} \begin{bmatrix} & a_j & \end{bmatrix} \begin{bmatrix} \\ d_j \\ \\ \end{bmatrix}$$

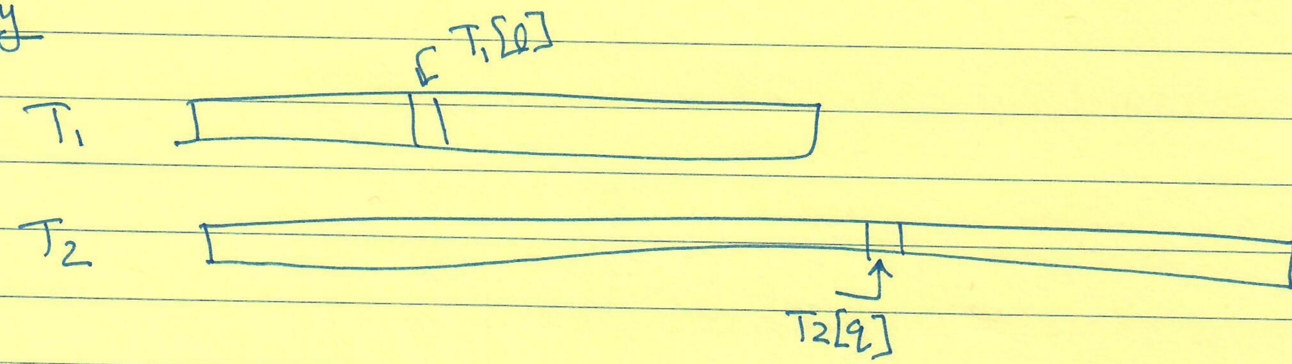
Equivalently: Pick random $l_1, l_2$ & verify

$$T_1(l_1)\cdot T_1(l_2) = T_2(l_1\cdot l_2 + q) - T_2(q).$$

Claim: $\underbrace{\exists i,j \text{ s.t. } b_{ij} \neq a_i \cdot a_j \Rightarrow \Pr[\downarrow \neq \downarrow]}_{\infty} \geq 1/4 - 0.0\cancel{2}4$

~~Con~~ Summary

Proof:




Verifier:

① $T_1[l_1] + T_1[l_2] = T_1[l_1 + l_2]$

② $T_2[q_1] + T_2[q_2] = T_2[q_1 + q_2]$

③ $T_1[l_1]\cdot T_1[l_2] = T_2[l_1 l_2 + q_1] - T_2[q_1]$

④ $T_2[q_1 + \sum \alpha_j P_j] = T_2[q_1]$.

10 queries!

Thm: if $\{P_j\}$ not sat. then $\Pr[\text{reject}] \geq .005$.

Proof: Cases: ① $T_1[\ ]$ not close to $l(a)$ ... $.005$

② $T_2[\ ]$ not close to $\sum \cancel{b_{ij}\, a_i a_j}\ b_{ij}\, q_{ij}$. $.005$

③ $b_{ij} \neq a_{ij}$ $\frac{1}{4} - .04$

④ Negation of all the above $.48$

What use is a PCP(poly(n), O(1)) for NP?

1. Non-trivial, insightful ...
2. Actually ... this is a PCPP
   of Proximity.

---

PCPP: $L$ has a PCPP with randomness $r()$, query $q()$, proximity $\rho$, soundness $s<1$ if the following hold.
$\exists$ Verifier $V$ that tosses $r(n)$ coins & prepares $q(n)$ queries to two oracles $X$
& proof $\pi$

& sat
$X \in L \Rightarrow \exists \pi$ s.t. $V$ accepts w.p. 1
if $X$ far from $L \Rightarrow \forall \pi$ $V$ accepts w.p. $\leq s$.

---

Our $L = \{L_{(P_1 \ldots P_m)}\} = \{ T_1 \mid \exists a$ s.t. $T_1[\ell] = \ell[a]\ \forall \ell$,
& $P_1(a) = \ldots P_m(a) = 0 \}$.

---

PCPP are useful to combine with other stuff to get better PCPs.
ALMSS $\Rightarrow$ Other stuff = high query PCP
Dinur $\Rightarrow$ Graph k-Col $\rightarrow$ Graph 3-coloring.

## Dinur's Proof

**Key Ingredient**: Generalized Graph K-Coloring.

- Input = Graph $G$ with functions $f_{uv} : [K] \times [K] \to \{0,1\}$ on edge $uv$.
- $G$ → "K-colorable" if $\exists\ \chi : V \to [K]$ s.t. $\forall$ $(u,v)$ $f_{uv}(\chi(u), \chi(v)) = 1$
- $G$ → $\epsilon$-unsat if $\Pr_{(u,v) \in E} [f_{uv}(\chi(u), \chi(v)) \neq 1] \geq \epsilon$.

---

**Key Lemma**: $\exists\ \epsilon_0 > 0$ $\exists$ reduction $G \longrightarrow H$

s.t. $|H| = O(|G|)$ & $\forall\ \epsilon$ if

$G$ $\cancel{K}$3-col $\Rightarrow$ $H$ is $\cancel{K}$3-col

$G$ $\epsilon$-unsat $\Rightarrow$ $H$ is $\min\{(2\epsilon), \epsilon_0\}$-unsat.

Start $G$ is $\cancel{K}$3-col or $\frac{1}{m}$-unsat.

Apply Key Lemma $(\log m)$ times → $G \to G_1 \cdots G_{\log m} = \overline{G}$

$\overline{G}$ is of size $c^{\log m} \cdot |G|$

$\overline{G}$ is $\epsilon_0$-unsat.

Key Lemma $\Leftarrow$ Lemma 1 + Lemma 2

Lemma 1 [Amplification]:

$\forall c \ \exists K, C'$ ~~s.t.~~ & transformation $G \to G_1$

s.t. $|G_1| \le C' \cdot |G|$ &

$G$ 3-col $\Rightarrow$ $G_1$ - K-col

$G$ $\epsilon$-unsat $\Rightarrow$ $G_1$ - $(c\epsilon)$-unsat.

—— x ——

Lemma 2: [Alphabet Reduction]:

$\exists \delta > 0$ s.t. $\forall K \ \exists C''$ & tran for $G_1 \to G_2$

s.t. $|G_2| \le C'' \cdot |G_1|$ &

$G_1$ K-col $\Rightarrow$ $G_2$ is 3-col

$G_1$ $\epsilon$-unsat $\Rightarrow$ $G_2$ is $(\epsilon \cdot \delta)$-unsat.

—— x ——

Next lecture : Proof of Lemma ~~1~~ & Lemma 2.