Lecture 11

Foundations of Probabilistic Proofs
Fall 2020
Alessandro Chiesa

Polynomial-Size PCPs for NP

We have constructed exponential-size PCPs for NP:

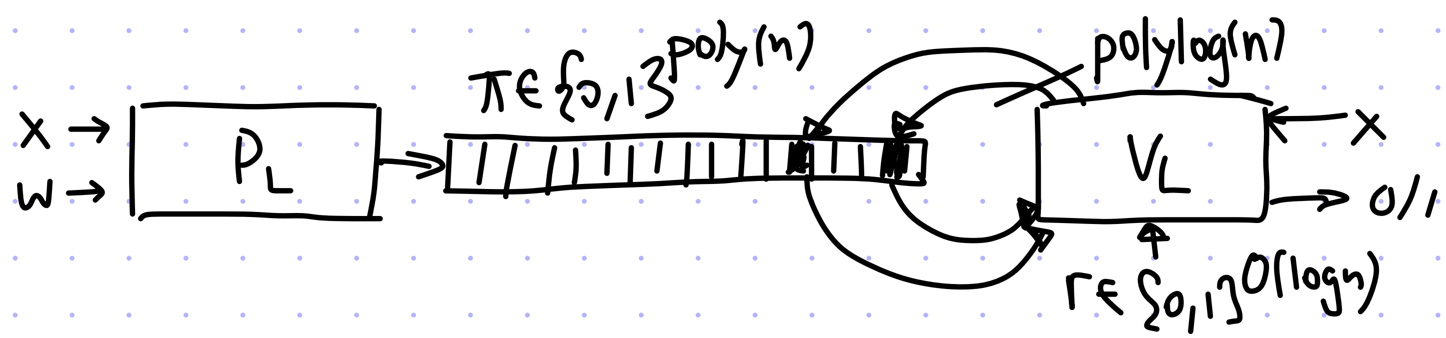
$$NP \subseteq PCP[\varepsilon_c = 0, \varepsilon_s = 0.5, \Sigma = \{0,1\}, \ell = \exp(n), q = O(1), r = \text{poly}(n)]$$

Our next goal is to reduce proof length to polynomial size:

<u>theorem:</u> $NP \subseteq PCP[\varepsilon_c = 0, \varepsilon_s = 0.5, \Sigma = \{0,1\}, \ell = \text{poly}(n), q = \text{poly}(\log n), r = O(\log n)]$

[We will see how to further reduce q to $O(1)$ towards the end of this course.]

That is, $\forall L \in NP$ $\exists$ PCP system (P_L, V_L) for L that looks like this:



<u>Proof strategy:</u>

① construct a low-degree PCP for NP ← today's lecture

② construct a low-degree test

③ low-degree PCP + low-degree test → polynomial-size PCP

Polynomial-Size PCP for Quadratic Equations

Recall the following NP-complete problem about quadratic equations over a field $\mathbb{F}$:

$$\mathrm{QESAT}(\mathbb{F}) = \{ (p_1, \dots, p_m) \mid \exists a_1, \dots, a_n \in \mathbb{F} \text{ s.t. } \forall i \in [m] \; p_i(a_1, \dots, a_n) = 0 \}.$$

We will construct a PCP for $\mathrm{QESAT}(\mathbb{F})$:

<u>theorem</u>: $\mathrm{QESAT}(\mathbb{F}) \subseteq \mathrm{PCP}[\varepsilon_c = 0, \varepsilon_s = 0.5, \Sigma = \mathbb{F}, \ell = |\mathbb{F}|^{O(\frac{\log n}{\log\log n})}, q = \mathrm{poly}(\log n), r = O(\log n)]$

We design the PCP in several steps: $\ell = \mathrm{poly}(n)$ if $|\mathbb{F}| = \mathrm{poly}(\log n)$

- use a <u>small</u> amount of randomness to reduce m equations $p_1, \dots, p_m$ to 1 equation p, preserving satisfiability whp
- for every possible p, include a proof that p is satisfied by the low-degree extension of the candidate assignment
- add low-degree testing

Part 1: From m Equations to 1 Equation

<u>Lemma:</u> there is a probabilistic algorithm T s.t. for $|\mathbb{F}| = \text{polylog}(m)$

① $T(p_1,\dots,p_m)$ uses $O(\log m)$ random bits and outputs a quadratic equation $p(x_1,\dots,x_n)$

② if $\exists\, a$ s.t. $p_1(a) = \dots = p_m(a) = 0$ then $\Pr_r\left[T(p_1,\dots,p_m; r)(a) = 0\right] = 1$

③ if $p_1,\dots,p_m$ are unsatisfiable then $\Pr_r\left[\exists\, a \ T(p_1,\dots,p_m; r)(a) = 0\right] \le 1/2$.

<u>Idea #1:</u> T samples $j \in [m]$ and outputs p_j

This uses little randomness ($\log m$ bits) but the soundness error is large $\left(1 - \frac{1}{m}\right)$.

<u>Idea #2:</u> T samples $r_1,\dots,r_m \in \mathbb{F}$ and outputs $p = \sum_{j\in[m]} r_j p_j$

This has small soundness error $\left(\frac{1}{|\mathbb{F}|}\right)$ but uses too much randomness (n elts).

[This is essentially what we did inside the LPCP for QESAT($\mathbb{F}$).]

If we sample $r_1,\dots,r_m \in \mathbb{F}_2$ the soundess error is ok ($1/2$) but not randomness (n bits).

<u>Idea #3:</u> T samples $r \in \mathbb{F}$ and outputs $p = \sum_{j\in[m]} r^j p_j$

This uses little randomness (1 elt) but now requires the field to be large:

the soundness error is $\frac{m}{|\mathbb{F}|}$ so we need $|\mathbb{F}| \ge \Omega(m)$

Part 1: From m Equations to 1 Equation

Lemma: there is a probabilistic algorithm T s.t. for small-enough $\mathbb{F}$

① $T(p_1,\dots,p_m)$ uses $O(\log m)$ random bits and outputs a quadratic equation $p(x_1,\dots,x_n)$

② if $\exists\, a$ s.t. $p_1(a)=\dots=p_m(a)=0$ then $\Pr_r\left[T(p_1,\dots,p_m;r)(a)=0\right]=1$

③ if $p_1,\dots,p_m$ are unsatisfiable then $\Pr_r\left[\exists\, a\;\; T(p_1,\dots,p_m;r)(a)=0\right]\le 1/2$.

proof:

Identify $[m]$ with $H_e \subseteq \mathbb{F}$ with $|H_e|=O(\log m)$ and $s_e := \frac{\log m}{\log |H_e|}$.

The transformation T samples $r_1,\dots,r_{s_e}\in\mathbb{F}$ and outputs

$$p := \sum_{0\le j_1,\dots,j_{s_e}<|H_e|} r_1^{j_1}\cdots r_{s_e}^{j_{s_e}}\cdot p_{j_1\dots j_{s_e}}.$$

The soundness error is $\le \frac{s_e\cdot |H_e|}{|\mathbb{F}|} \le O\left(\frac{(\log m)^2}{|\mathbb{F}|}\right) \Rightarrow$ ok if $|\mathbb{F}|=\Omega((\log m)^2)$.

The amount of randomness is: $|\mathbb{F}|^{s_e} = O\left((\operatorname{poly}\log m)^{\frac{\log m}{\log O(\log m)}}\right) = 2^{O(\log m)} = \operatorname{poly}(m)$.

Part 2: Low-Degree PCP for 1 Equation

Consider this setting: $P(a \in \mathbb{F}^n) \rightarrow \boxed{\quad \pi \quad} \leftrightarrows V\begin{pmatrix}\text{quadratic poly} \\ p \in \mathbb{F}[X_1,\dots,X_n]\end{pmatrix}$ Is p satisfiable?

The challenge is that the polynomial $p(x_1,\dots,x_n)$ may depend on <u>every</u> variable.

<u>Idea</u>: reduce to a sumcheck problem & use (unrolled) sumcheck

Step 1: arithmetize

- identify $[n]$ with $H_v^{S_v}$ for a subset $H_v \subseteq \mathbb{F}$ with $|H_v| = O(\log n)$ and $S_v := \dfrac{\log n}{\log |H_v|}$.
- satisfiability as a sum:

$$\forall a : [n] \to \mathbb{F}, \quad p(a) = \sum_{i,j \in [n]} c_{ij} a_i a_j = \sum_{\alpha,\beta \in H_v^{S_v}} \hat{c}(\alpha,\beta) \cdot \hat{a}(\alpha) \cdot \hat{a}(\beta)$$

where $\hat{a} : \mathbb{F}^{S_v} \to \mathbb{F}$ & $\hat{c} : \mathbb{F}^{2S_v} \to \mathbb{F}$ are the low-degree extensions of $a : [n] \to \mathbb{F}$ & $c : [n]^2 \to \mathbb{F}$.

The addend $q(y_1,\dots,y_{S_v}, z_1,\dots,z_{S_v}) := \hat{c}(y,z)\hat{a}(y)\hat{a}(z)$ has individual degree $\leq 2 \cdot (|H_v| - 1) \leq 2|H_v|$.

We have reduced the problem to $\displaystyle\sum_{\alpha,\beta \in H_v^{S_v}} q(\alpha,\beta) \stackrel{?}{=} 0$ for $\hat{c}(y,z)$ known by the verifier and $\hat{a}$ supplied by the prover.

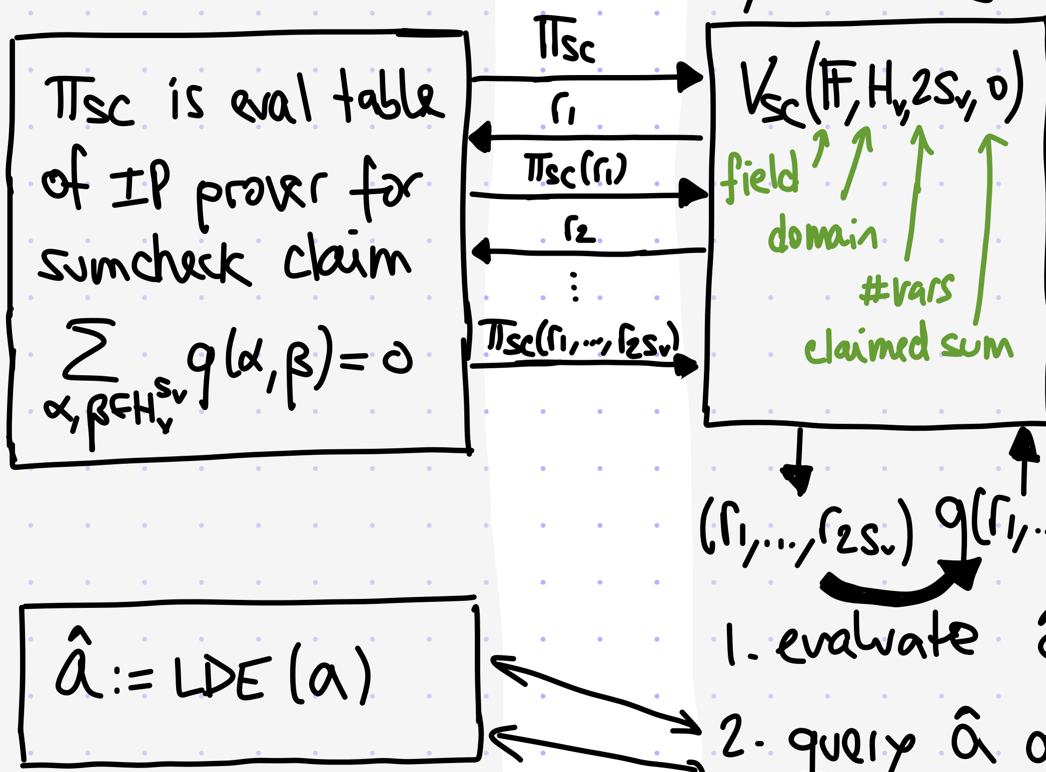
Part 2: Low-Degree PCP for 1 Equation

Step 1: $p(a)=0 \Leftrightarrow \sum_{\alpha,\beta\in H_v^{s_v}} q(\alpha,\beta)=0$ for $q(y,z):=\hat{c}(y,z)\cdot\hat{a}(y)\cdot\hat{a}(z)$

Step 2: probabilistically check the arithmetized statement

$\mathcal{P}(p,a)$ outputs $\pi:=(\hat{a},\pi_{sc})$

$V(p):=$ check that $\sum_{\alpha,\beta\in H_v^{s_v}} q(\alpha,\beta)=0$ by running sumcheck and querying $\hat{a}$



1. evaluate $\hat{c}(y,z)$ at $(r_1,\dots,r_{2s_v})$
2. query $\hat{a}$ at $(r_1,\dots,r_{s_v})$ & $(r_{s_v+1},\dots,r_{2s_v})$

proof length:
- $|\hat{a}|=|\mathbb{F}|^{s_v}$
- $|\pi_{sc}|=O(|\mathbb{F}|^{2s_v}|H_v|)$

$\Rightarrow |\mathbb{F}|^{O(s_v)}=\mathrm{poly}(n)$ if $|\mathbb{F}|=\mathrm{polylog}(n)$

query complexity:
- $O(s_v\cdot|H_v|)=O(\log^2 n)$ elements from π_{sc}
- 2 elements from $\hat{a}$

<u>Completeness</u>: if $p(a)$ then $\pi=(\mathrm{LDE}(a),\pi_{sc})$ always convinces the verifier

<u>Soundness</u>: if p unsatisfiable then $\forall\tilde{\pi}=(\tilde{a},\tilde{\pi}_{sc})$

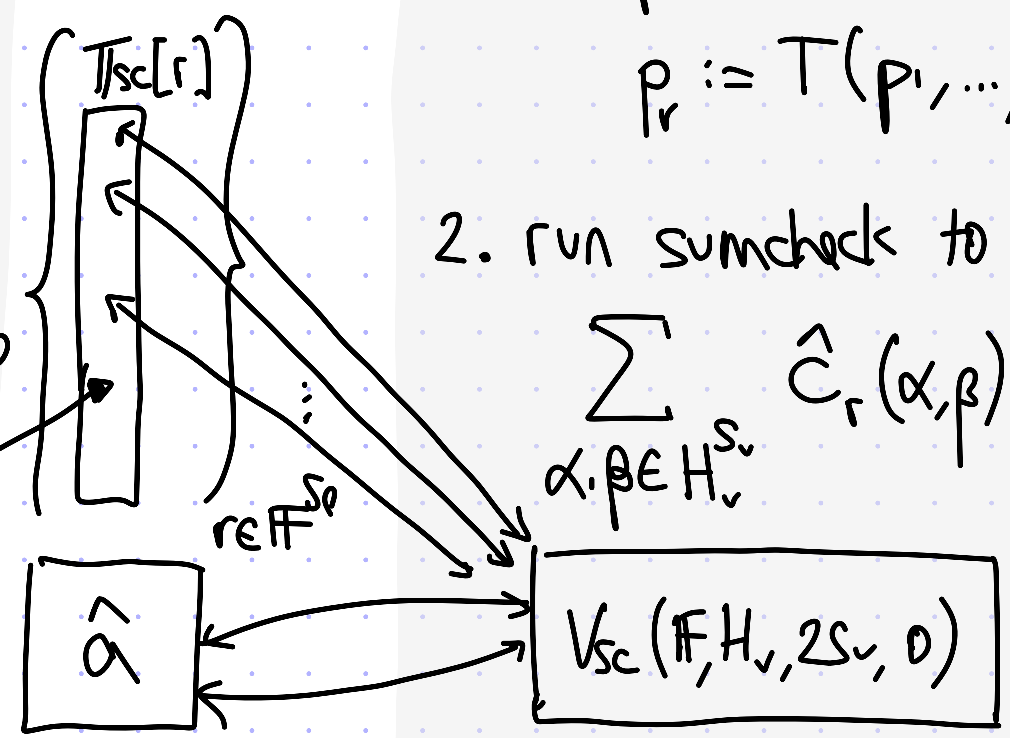
low-degree PCP condition → if $\tilde{a}$ is LDE of some a then $\varepsilon_s\le\frac{(2s_v)\cdot(2|H_v|)}{|\mathbb{F}|}\le O\left(\frac{(\log n)^2}{|\mathbb{F}|}\right)$

Low-Degree PCP for Quadratic Equations

We put Part 1 and Part 2 together:

$P((p_1,\dots,p_m),a) :=$

1. For every $r \in \mathbb{F}^{S_e}$:
   - $p_r = T(p_1,\dots,p_m; r)$
   - $\Pi_{sc}[r] :=$ eval table for sumcheck to show $p_r(a)=0$
   - output $\Pi_{sc}[r]$
2. output $\hat{a}: \mathbb{F}^{S_v} \to \mathbb{F}$ [LDE of $a:[n]\to\mathbb{F}$]



$V((p_1,\dots,p_m)) :=$

1. sample $r \in \mathbb{F}^{S_e}$ and compute $p_r := T(p_1,\dots,p_m; r)$
2. run sumcheck to check that $$\sum_{\alpha,\beta \in H_v^{S_v}} \hat{C}_r(\alpha,\beta)\,\hat{a}(\alpha)\,\hat{a}(\beta) = 0$$

Completeness: if $p_1(a)=\dots=p_m(a)$ then $\forall r \in \mathbb{F}^{S_e}$ $p_r(a)=0$ and so $\sum_{\alpha,\beta\in H_v^{S_v}} \hat{C}_r(\alpha,\beta)\,\hat{a}(\alpha)\,\hat{a}(\beta)=0$

Soundness: if $(p_1,\dots,p_m)$ is unsatisfiable then, except w.p. $\le O\left(\frac{S_e|H_e|}{|\mathbb{F}|}\right) = O\left(\frac{\log^2 m}{|\mathbb{F}|}\right)$, so is p_r.

Hence, $\forall\, \hat{a}$ that is LDE, $\sum_{\alpha,\beta\in H_v^{S_v}} \hat{C}_r(\alpha,\beta)\,\hat{a}(\alpha)\,\hat{a}(\beta) \neq 0$. So, $\forall\, \tilde{\Pi}_{sc}$, the sumcheck accepts w.p. at most $O\left(\frac{S_v|H_v|}{|\mathbb{F}|}\right) \le O\left(\frac{\log^2 n}{|\mathbb{F}|}\right)$. So $|\mathbb{F}| = \Omega(\mathrm{poly}(\log m, \log n))$ suffices.

Low-Degree Testing [statement only]

lemma [to be proved in the next lecture]

there exists a ppt oracle machine V_{LDT} s.t. $\forall f: \mathbb{F}^n \to \mathbb{F}$

① **completeness:** if f has total degree at most d then $\Pr[V_{LDT}^f(\mathbb{F}, n, d) = 1] = 1$

② **soundness:** if f is $\frac{1}{10}$-far from all functions of total degree at most d then $\Pr[V_{LDT}^f(\mathbb{F}, n, d) = 1] \leq \frac{1}{2}$.

③ **efficiency:** $V_{LDT}(\mathbb{F}, n, d)$ makes $\mathrm{poly}(|\mathbb{F}|, n, d)$ queries

Why total degree test?

It is simpler and we can make do with it [see next slide].

Also there is a generic way to "lift" a total degree test to an individual degree test.

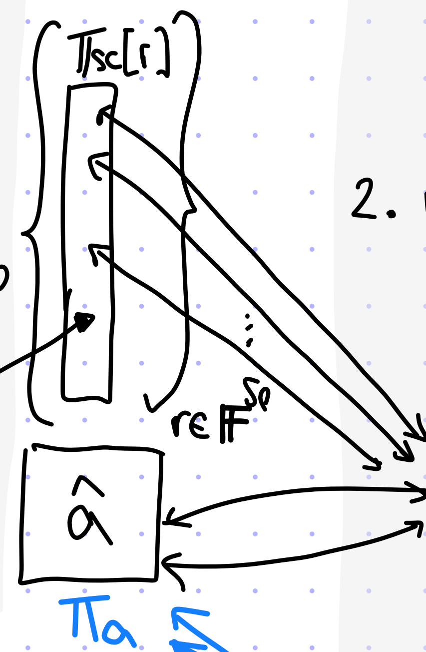
Remark: the requirement that f is defined on $\mathbb{F}^n$ rather than D^n for $D \subseteq \mathbb{F}$ comes from the LDT [this can be relaxed somewhat but is not easy]

At Last: PCP for Quadratic Equations

theorem: $\mathrm{QESAT}(\mathbb{F}) \leq \mathrm{PCP}\left[\varepsilon_c = 0, \varepsilon_s = 0.5, \Sigma = \mathbb{F}, \ell = |\mathbb{F}|^{O\left(\frac{\log n}{\log\log n}\right)}, q = \mathrm{poly}(\log n), r = O(\log n)\right]$

$P((p_1,\dots,p_m), a) :=$

1. For every $r \in \mathbb{F}^{S_e}$:
   - $p_r = T(p_1,\dots,p_m; r)$
   - $\Pi_{sc}[r] :=$ eval table for sumcheck to show $p_r(a) = 0$
   - output $\Pi_{sc}[r]$
2. output $\hat{a} : \mathbb{F}^{S_v} \to \mathbb{F}$
   [LDE of $a : [n] \to \mathbb{F}$]



$V((p_1,\dots,p_m)) :=$

1. Sample $r \in \mathbb{F}^{S_e}$ and compute
   $p_r := T(p_1,\dots,p_m; r)$
2. run sumcheck to check that
   $$\sum_{\alpha,\beta \in H_v^{S_v}} \hat{C}_r(\alpha,\beta)\,\hat{a}(\alpha)\,\hat{a}(\beta) = 0$$
   $V_{sc}(\mathbb{F}, H_v, 2S_v, 0)$
3. run low-degree test on Π_a
   $V_{LDT}(\mathbb{F}, S_v, S_v|H_v|)$

① If we can only ensure that total degree of $\hat{a}$ is $S_v|H_v|$ then the soundness error of the term $O\left(\frac{S_v|H_v|}{|\mathbb{F}|}\right)$ increases to $O\left(\frac{S_v^2|H_v|}{|\mathbb{F}|}\right)$. That's ok.

② If $\tilde{\Pi}_a$ is $\frac{1}{10}$-far from LD then V_{LDT} accepts w.p. $\leq 1/2$. If $\tilde{\Pi}_a$ is $\frac{1}{10}$-close to some $\hat{a}$, then ... we don't need self-correction! Vsc's 2 queries are random, so pay $2 \cdot \frac{1}{10}$ in error.