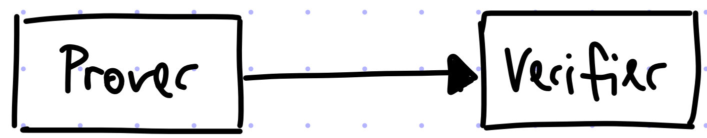
Lecture A.7

Intro to IOPs

Summer Graduate School on
Foundations and Frontiers of Probabilistic Proofs
2021.08.03

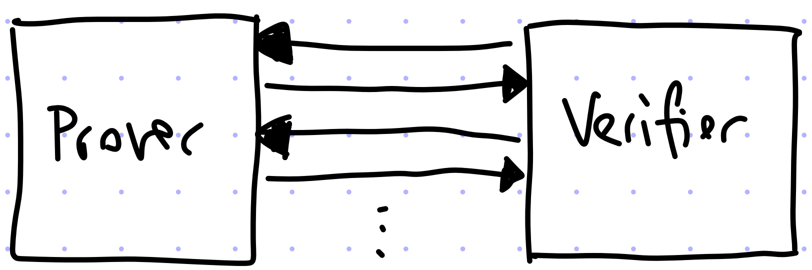
Interactive Oracle Proofs

Recall that NP is the model for traditional mathematical proofs:

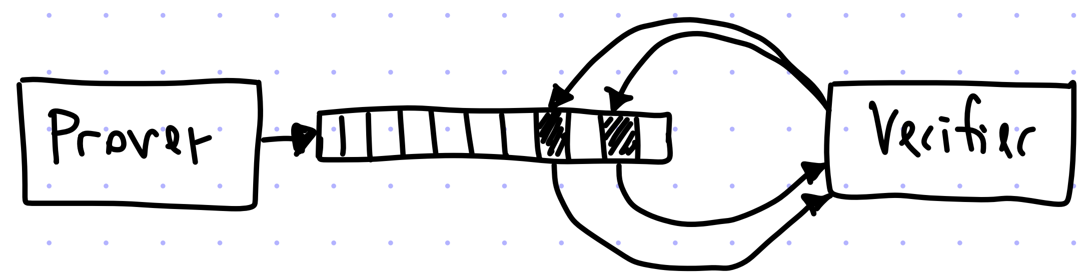


We have studied two different extensions:

IP: add randomness & interaction



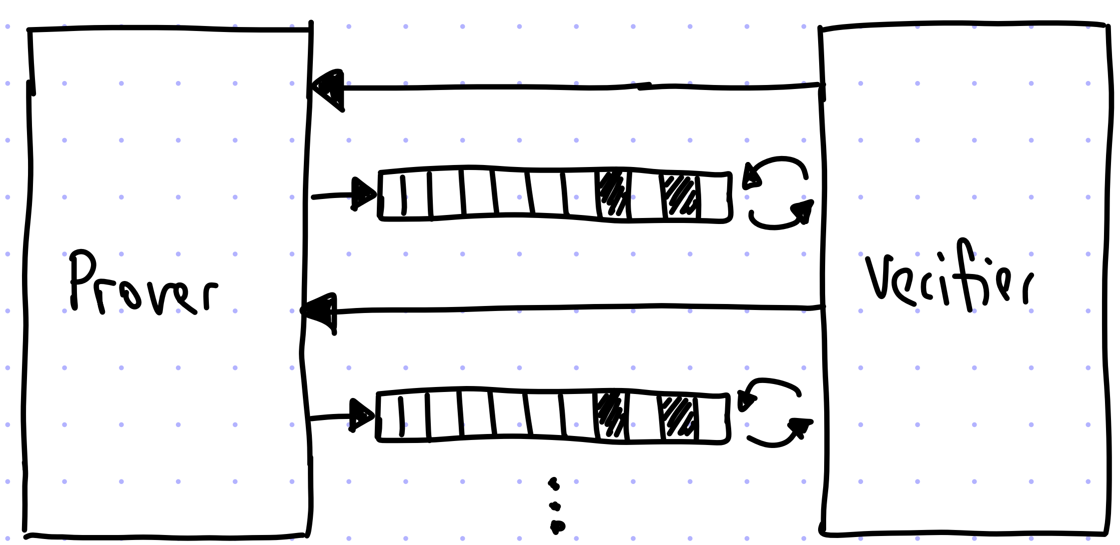
PCP: add randomness & oracle access to proof



Today we consider the common extension between the two:

Interactive Oracle Proof (IOP)

add randomness, interaction, and oracle access to proof



Definition of IOP

Let P be an all-powerful prover and V a ppt interactive oracle algorithm.

We say that (P,V) is an IOP system for a language L with completeness error ε_c and soundness error ε_s if the following holds:

① completeness: $\forall x \in L \ \Pr_{\rho}\left[\langle P(x), V(x;\rho)\rangle = 1\right] \geq 1 - \varepsilon_c$

② soundness: $\forall x \notin L \ \forall \tilde{P} \ \Pr_{\rho}\left[\langle \tilde{P}, V(x;\rho)\rangle = 1\right] \leq \varepsilon_s$

Above $\langle A, B\rangle$ denotes this process: $A \to \pi_1$, $m_1 \leftarrow B^{\pi_1}$, $A(m_1) \to \pi_2$, $m_2 \leftarrow B^{\pi_1,\pi_2}$, and so on until B decides to halt and output.

Efficiency measures:

- prover time
- verifier time
- round complexity
- randomness complexity
- alphabet size
- proof length $(|\pi_1| + |\pi_2| + \cdots)$
- query complexity $(q_1 + q_2 + \cdots)$
- public vs. private coins

each verifier message is random, so all queries can be at the end [interaction phase, then query phase]

Let IOP be the set of languages decidable via an interactive oracle proof.

An Upper Bound

Lemma: IOP $\subseteq$ NEXP

We saw how any IP can be "unrolled" into a corresponding PCP, whose size equals the size of the IP's game tree.
Completeness and soundness were unaffected.

Similarly, any IOP can be "unrolled" into a (very long) PCP:

$$\text{IOP}\left[\begin{array}{ll} \text{completeness error} & \varepsilon_c \\ \text{soundness error} & \varepsilon_s \\ \text{round complexity} & k \\ \text{alphabet} & \Sigma \\ \text{proof length} & \ell \\ \text{query complexity} & q \\ \text{randomness} & r \end{array}\right] \subseteq \text{PCP}\left[\begin{array}{ll} \text{completeness error} & \varepsilon_c \\ \text{soundness error} & \varepsilon_s \\ & \\ \text{alphabet} & \Sigma \\ \text{proof length} & |\text{tree}| \\ \text{query complexity} & q \\ \text{randomness} & r \end{array}\right]$$

Ex: if the verifier sends ℓ_v symbols in Σ_v across all rounds then $|\text{tree}| \leq |\Sigma_v|^{\ell_v} \cdot \ell$

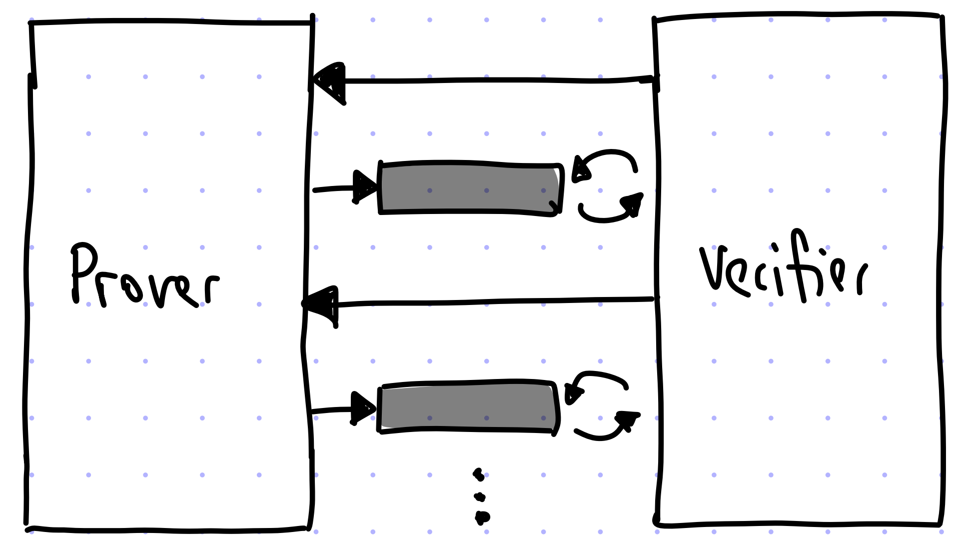
Note: the maximum PCP proof length is $\exp(n)^{\text{poly}(n)} \cdot \exp(n) = \exp(n)$.

We have already proved that PCP $\subseteq$ NEXP.

$\blacksquare$

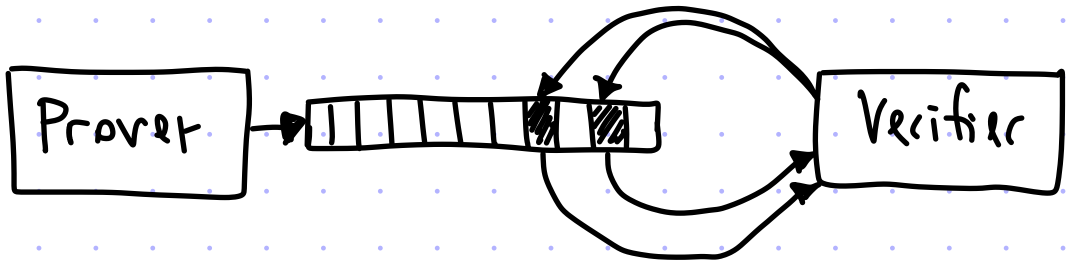
Two Lower Bounds

lemma: PSPACE $\subseteq$ IOP



proof: Any IP is (trivially) an IOP where in each round the prover sends a 1-symbol message and the verifier reads it. Hence $IP \subseteq IOP$.
We already proved that $IP = PSPACE$, so $PSPACE \subseteq IOP$. ■

lemma: NEXP $\subseteq$ IOP



proof: Any PCP is (trivially) an IOP where the prover sends a single message and the verifier probabilistically checks it.
Hence $PCP \subseteq IOP$.
We will prove that $PCP = NEXP$, so $NEXP \subseteq IOP$. ■

We conclude that $IOP = NEXP$.

What are IOPs good for?

We have learned that IOPs do not give us new languages over PCPs.
This is ok: we can try to achieve better parameters for languages in NEXP.

Our goal: leverage interaction to design IOPs that are "more efficient" (shorter proof length, fewer queries, etc.) than state-of-the-art PCPs

But... PCPs were an awkward proof model and IOPs are only more awkward.
So why care about the goal?

Similarly to PCPs, we can use cryptography to compile IOPs into cryptographic proofs (aka arguments). And if we can design efficient IOPs then we will get cryptographic proofs that are more efficient than from PCPs!

In the next few lectures we will learn how to construct IOPs that achieve parameter regimes that we do not know how to achieve with PCPs.

Curiously, despite this, to date we do not have strong separations between IOPs & PCPs.

Recall: PCP for QESAT

$\mathrm{QESAT}(\mathbb{F}) := \left\{ (p_1,\dots,p_m) \;\middle|\; \begin{array}{l} \exists\, a_1,\dots,a_n \in \mathbb{F} \text{ s.t.} \\ \forall j \in [m] \;\; p_j(a_1,\dots,a_n) = 0 \end{array} \right\}$

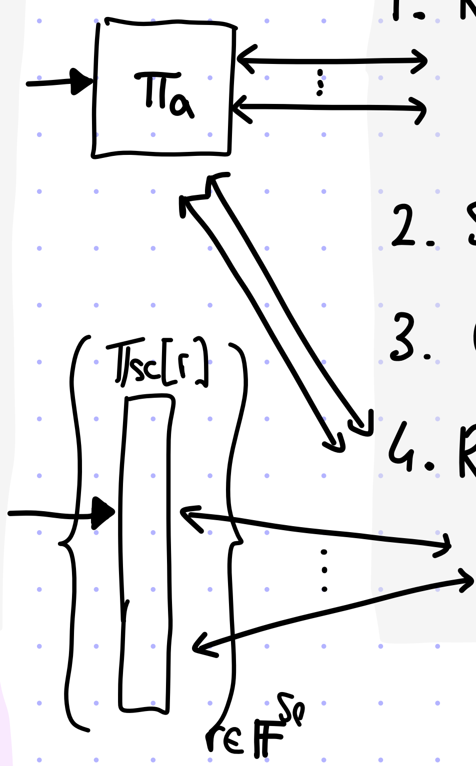
theorem: For every $\mathbb{F}$ with $|\mathbb{F}| = \Omega\left(\frac{\log^2 m}{\log\log m} + \frac{\log^2 n}{\log\log n}\right)$,

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

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

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

$$p_r(a) = \sum_{\alpha,\beta \in H_v^{S_v}} \widehat{\mathrm{coeff}(p_r)\Pi_a(\alpha)\Pi_a(\beta)}$$



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

1. Run (ind)$_{\circledast}$ low-degree test on Π_a

   $V_{LDT}(\underset{\text{field}}{\mathbb{F}}, \underset{\text{variables}}{S_v}, \underset{\text{degree}}{|H_v|})$
2. Sample $r \in \mathbb{F}^{S_e}$
3. Compute $p_r := T(p_1,\dots,p_m; r)$
4. Run sumcheck to check that $p_r(a) = 0$

   $V_{SC}(\underset{\text{field}}{\mathbb{F}}, \underset{\text{domain}}{H_v}, \underset{\text{variables}}{2S_v}, \underset{\text{sum}}{0}, \underset{\text{degree}}{2|H_v|})$

[$\circledast$ or total degree $\le S_v \cdot |H_v|$]

Notation:

- $H_v, H_e \subseteq \mathbb{F}$
- $S_v := \frac{\log n}{\log |H_v|}$

  so $[n] \leftrightarrow H_v^{S_v}$
- $S_e := \frac{\log m}{\log |H_e|}$

  so $[m] \leftrightarrow H_e^{S_e}$

(at least) cubic!

The proof length is $|\mathbb{F}|^{S_v} + |\mathbb{F}|^{S_e} \cdot O(|H_v| \cdot |\mathbb{F}|^{2S_v}) = O(|H_v| \cdot |\mathbb{F}|^{S_e + 2S_v})$.

If $|H_v| = O(\log n)$ and $|H_e| = O(\log m)$ then the length is $O\left(\log n \cdot |\mathbb{F}|^{\frac{\log m}{\log\log m + O(1)} + 2 \cdot \frac{\log n}{\log\log n + O(1)}}\right)$.

Recycling: IOP for NP from the PCP for NP (1/2)

Idea: reduce proof length by interacting when convenient.

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

1. Output $\Pi_a : \mathbb{F}^{S_v} \to \mathbb{F}$ that is LDE of $a : [n] \to \mathbb{F}$
2. Compute $p_r := \sum_{j=1}^{m} r_j p_j$
3. Sumcheck IP for "$p_r(a)=0$"

$P_{SC}(\mathbb{F}, H_v, 2S_v, 0, \hat{c}, \hat{a}\cdot\hat{a})$

Π_a

r

sumcheck IP to check that
$$\sum_{\alpha,\beta \in H_v^{S_v}} \widehat{\mathrm{coeff}}(p_r)\Pi_a(\alpha)\Pi_a(\beta) = 0$$

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

1. Run (ind) low-degree test on Π_a
   $V_{LDT}(\mathbb{F}, S_v, |H_v|)$ — field, variables, degree
2. Send random $r \in \mathbb{F}^m$ ⓘ
3. Compute $p_r := \sum_{j=1}^{m} r_j p_j$
4. Sumcheck IP for "$p_r(a)=0$"

$V_{SC}(\mathbb{F}, H_v, 2S_v, 0, 2|H_v|)$ — field, domain, variables, sum, degree ⓘⓘ

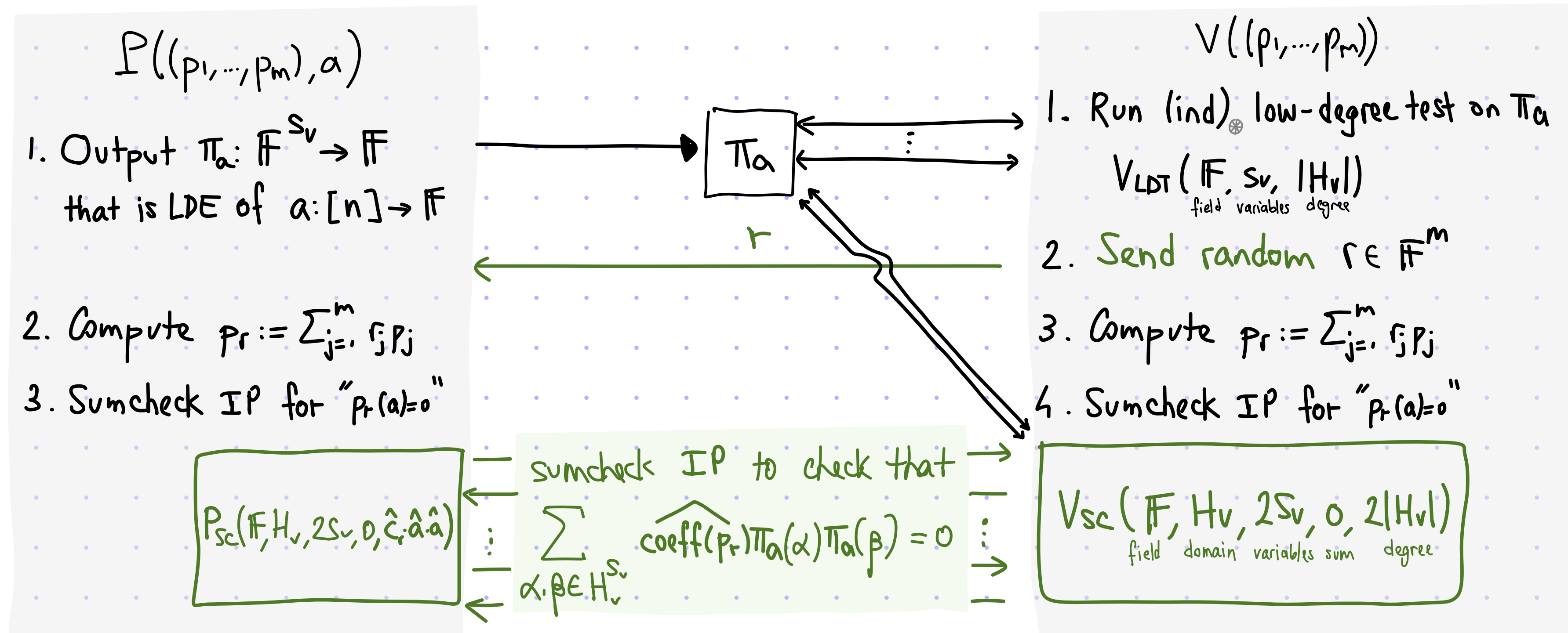
ⓘ send randomness for reducing m equations to 1 equation
(in fact we can set $p_r := \sum_{j=1}^{m} r_j p_j$ instead of $p_j := \sum_{0 \le j_1,\dots,j_{S_e} < |H_e|} r_1^{j_1} \cdots r_{S_e}^{j_{S_e}} p_{j_1 \dots j_{S_e}}$)

ⓘⓘ engage in an interactive sumcheck instead of sending a sumcheck PCP

Recycling: IOP for NP from the PCP for NP (2/2)

The new proof length is

$$|\mathbb{F}|^{s_v} + O(s_v \cdot |H_v|) = O(|\mathbb{F}|^{s_v}) = O\left(|\mathbb{F}|^{\frac{\log n}{\log|H_v|}}\right).$$



The soundness error is

$$\max\left\{ \varepsilon_{LDT}(\delta),\ 2\delta + O\left(\frac{s_v \cdot |H_v|}{|\mathbb{F}|}\right)\right\}$$

so we need $|\mathbb{F}| = \Omega(s_v \cdot |H_v|) = \Omega\left(\frac{\log n}{\log|H_v|} \cdot |H_v|\right)$. So let's take $|\mathbb{F}| = O\left(\frac{\log n}{\log|H_v|} \cdot |H_v|\right)$.

The proof length becomes

$$O\left(|\mathbb{F}|^{\frac{\log n}{\log|H_v|}}\right) = O\left(\left(\frac{\log n}{\log|H_v|} \cdot |H_v|\right)^{\frac{\log n}{\log|H_v|}}\right) = O\left(n^{\frac{\log|H_v| + \log\log n - \log\log|H_v|}{\log|H_v|}}\right) = O\left(n^{1+\frac{\log\log n - \log\log|H_v|}{\log|H_v|}}\right) = O(n^{1+\varepsilon})$$

if we take $|H_v| = O(\log^{1/\varepsilon} n)$. We proved the following theorem:

<u>theorem</u>: For every $\varepsilon > 0$ and $\mathbb{F}$ with $|\mathbb{F}| = \Theta\left(\frac{\log^{O(\frac{1}{\varepsilon})} n}{\log\log n}\right)$,

$$QESAT(\mathbb{F}) \in PCP[\varepsilon_c = 0, \varepsilon_s = 0.5, \Sigma = \{0,1\}, \ell = n^{1+\varepsilon}, q = \log^{O(\frac{1}{\varepsilon})} n, r = \mathrm{poly}(m,n)]$$

almost linear !

Towards Efficient IOPs

A similar modification can be done to the PCP for NTIME(T) to get:

<u>theorem</u>: For every time function $T:\mathbb{N}\to\mathbb{N}$ with $T(n)=\Omega(n)$ and $\forall \varepsilon>0$

$$\mathrm{NTIME}(T) \subseteq \mathrm{IOP}\left[\begin{array}{c}\varepsilon_c=0,\ \varepsilon_s=0.5,\ \Sigma=\{0,1\},\ pt=\mathrm{poly}_{\varepsilon}(T),\ vt=\mathrm{poly}_{1/\varepsilon}(n,\log T) \\ \ell=T^{1+O(\varepsilon)},\ q=(\log T)^{O(\frac{1}{\varepsilon})},\ r=\mathrm{poly}_{1/\varepsilon}(\log T)\end{array}\right]$$

Without much effort, we reduced proof length significantly!

Q: can we reduce proof length even further (e.g. to linear)?

A serious obstacle to improving proof length is that we are encoding assignments via the multi-variate low-degree extension (also known as the Reed-Muller code), which inherently incurs a polynomial blowup:

$$|\mathbb{F}|^{s} \geq (s\cdot|H|)^{s} = \left(\frac{\log N}{\log|H|}\cdot|H|\right)^{\frac{\log N}{\log|H|}} = N^{\frac{\log|H|+\log\log N-\log\log|H|}{\log|H|}} = N^{\left(1+\frac{\log\log N-\log\log|H|}{\log|H|}\right)} = N^{1+O(\varepsilon)}$$

To do better, we will change how we encode assignments.

Reason for optimism: we are severely underusing the IOP model, as the prover sends a proof oracle in the first round only. We should send oracles in more rounds!