Foundations of Probabilistic Proofs

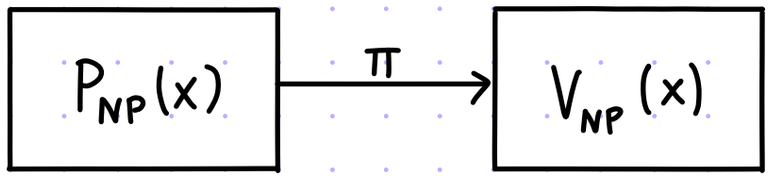
A course by **Alessandro Chiesa**

Lecture 13

Intro to IOPs

Interactive Oracle Proofs

NP captures proofs checkable via a deterministic polynomial-time verifier:



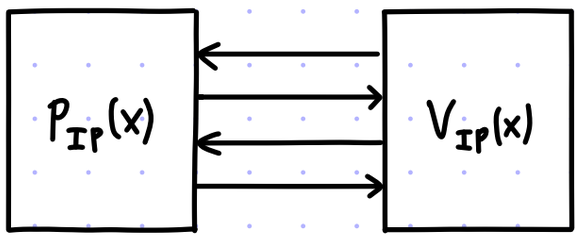
We studied two different extensions:

INTERACTIVE PROOFS

Polynomial-time verifier plus

① randomness

② interaction

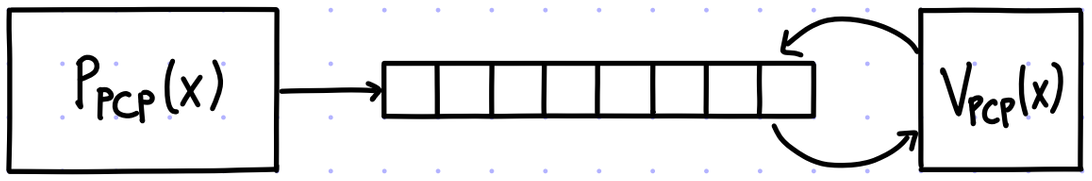


PROBABILISTICALLY-CHECKABLE PROOFS:

Polynomial-time verifier plus

① randomness

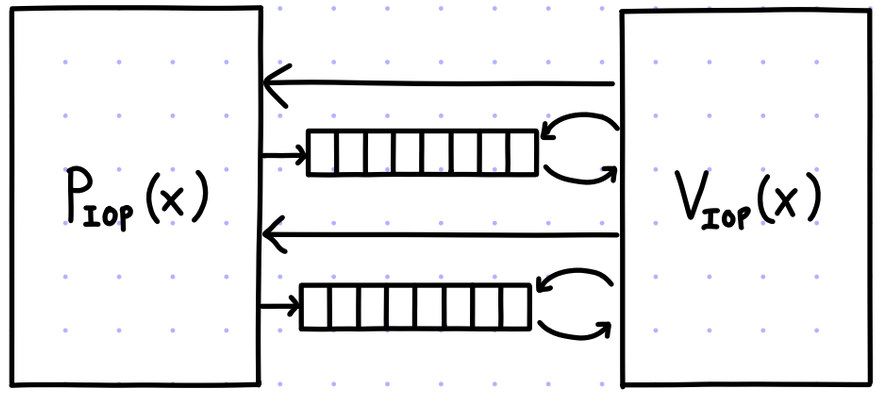
② oracle access to proof



Today we introduce the common extension:

INTERACTIVE ORACLE PROOFS (IOPs)

Polynomial-time verifier plus ① randomness ② interaction ③ oracle access to proof



Definition of IOP

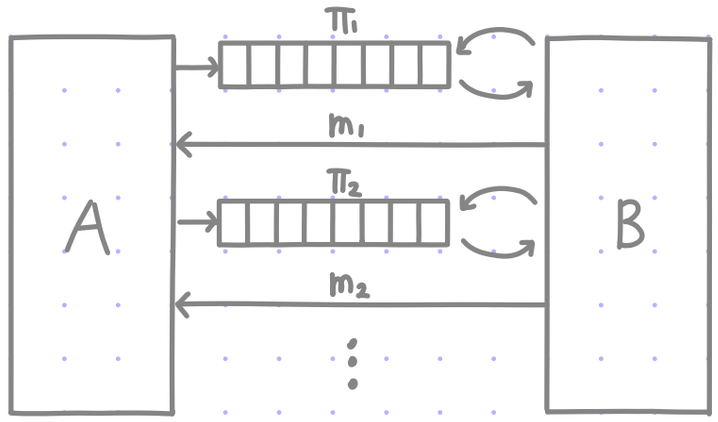
[The definition for a language L is a special case.]

We say that (P,V) is an **IOP system** for a relation R with completeness error ε_c and soundness error ε_s (with $1-\varepsilon_c > \varepsilon_s$) if the following holds:

① COMPLETENESS: $\forall (x,w) \in R \quad \Pr\left[\langle P(x,w), V(x) \rangle = 1 \right] \geq 1-\varepsilon_c$.

② SOUNDNESS: $\forall x \notin L(R) \quad \forall \tilde{P} \quad \Pr\left[\langle \tilde{P}, V(x) \rangle = 1 \right] \leq \varepsilon_s$.

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:

- K : round complexity
- Σ : proof alphabet
- ℓ : proof length $(\ell_1 + \ell_2 + \cdots + \ell_K)$
- q : verifier query complexity $(q_1 + q_2 + \cdots + q_K)$
- r : verifier randomness complexity

We also care about private-coin vs. public-coin.

Each verifier message is random.
(So all queries can be at the end: interaction phase; then query phase.)

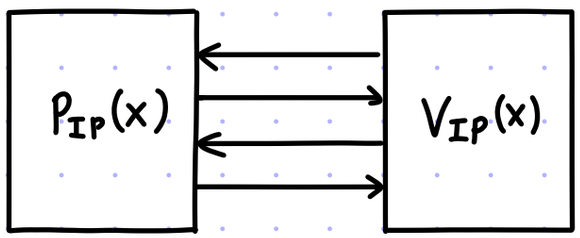
We denote by IOP the case with no restrictions (beyond V_{IOP} runs in polynomial time):

$$\mathrm{IOP} = \mathrm{IOP}\left[\varepsilon_c = 0,\ \varepsilon_s = \tfrac{1}{2},\ K = \mathrm{poly}(n),\ \Sigma = \exp(n),\ \ell = \exp(n),\ q = \mathrm{poly}(n),\ r = \mathrm{poly}(n) \right]$$

Two Lower Bounds

lemma: PSPACE ⊆ IOP

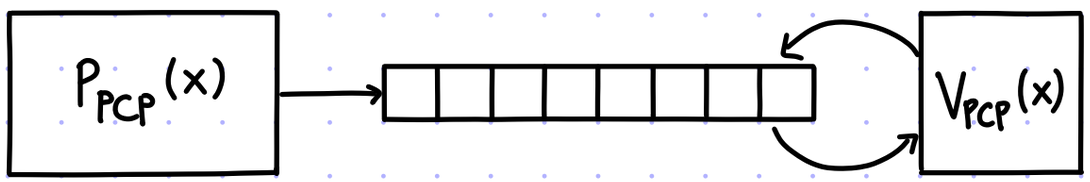
proof: An IP is (trivially) an IOP where in each round the prover sends a 1-symbol oracle and the verifier reads it.



Hence IP ⊆ IOP. Since IP = PSPACE, we get that PSPACE ⊆ IOP. ■

lemma: NEXP ⊆ IOP

proof: Any PCP is (trivially) an IOP where the prover sends a single message and the verifier probabilistically checks it.



Hence PCP ⊆ IOP. Since PCP = NEXP, we get that NEXP ⊆ IOP. ■

An Upper Bound

lemma: $\mathrm{IOP} \subseteq \mathrm{NEXP}$

proof:

We learned that any IP can be "unrolled" into a corresponding PCP, whose proof length equals the size of the IP's game tree.
(If the IP is public-coin, then the PCP is non-adaptive.)

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

$$\mathrm{IOP}\begin{bmatrix} \text{completeness error} & \varepsilon_c \\ \text{soundness error} & \varepsilon_s \\ \text{round complexity} & k \\ \text{alphabet} & \Sigma \\ \text{proof length} & \ell \\ \text{query complexity} & q \\ \text{randomness} & r \end{bmatrix} \subseteq \mathrm{PCP}\begin{bmatrix} \text{completeness error} & \varepsilon_c \\ \text{soundness error} & \varepsilon_s \\ & \\ \text{alphabet} & \Sigma \\ \text{proof length} & |\text{tree}| \\ \text{query complexity} & q \\ \text{randomness} & r \end{bmatrix}$$

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

The maximum PCP proof length is $\exp(n)^{\mathrm{poly}(n)} \cdot \exp(n) = \exp(n)$.

We have already proved $\mathrm{PCP} \subseteq \mathrm{NEXP}$. ■

We conclude that $\mathrm{IOP} = \mathrm{NEXP}$.

What are IOPs good for?

We have learned that IOPs do NOT give us more languages compared to PCPs.

This is OK: we aim for better parameters for languages in NEXP.

GOAL: leverage interaction to design IOPs that are "more efficient" (shorter proofs, fewer queries, ...) than known PCPs

But... PCPs are an awkward proof model and IOPs are only more awkward.

Why care about this goal?

Similarly to PCPs, there are two main applications:

① IOP — cryptographic transformation → succinct argument

② IOP — reduction → hardness of approximation result
↑ for stochastic constraint satisfaction problems

HOPE: designing efficient IOPs leads to more efficient succinct arguments (than what is possible with known PCPs)

Next few lectures: IOPs with parameters not achieved by known PCPs.

[Fundamental question: separations between PCPs and IOPs? Little is known.]

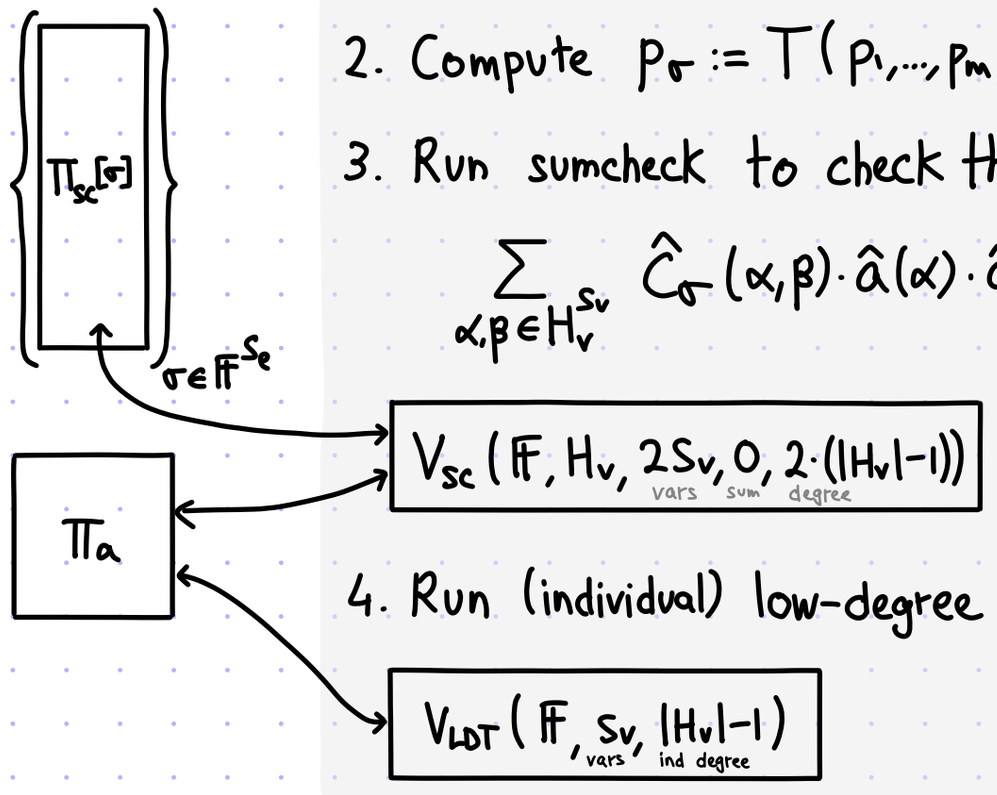
Recall: PCP for QESAT

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

theorem: $QESAT(\mathbb{F}) \subseteq PCP\left[\begin{array}{ll} \varepsilon_c = 0 & \Sigma = \mathbb{F} \\ \varepsilon_s = O(1) + O\left(\frac{\log^2 n}{\log\log n} \cdot \frac{1}{|\mathbb{F}|}\right) & \ell = |\mathbb{F}|^{O\left(\frac{\log n}{\log\log n}\right)} \\ & r = O\left(\frac{\log n}{\log\log n} \cdot \log|\mathbb{F}|\right) \end{array} \right]$

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

1. For every $\sigma \in \mathbb{F}^{S_e}$:
   - $p_\sigma := T(p_1,\dots,p_m;\sigma)$
   - $\Pi_{sc}[\sigma] :=$ eval table for sumcheck claim "$p_\sigma(a)=0$"
   - output $\Pi_{sc}[\sigma]$
2. Output $\hat{a}: \mathbb{F}^{S_v} \to \mathbb{F}$ as Π_a. (The LDE of $a:[n]\to\mathbb{F}$.)



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

1. Sample $\sigma \in \mathbb{F}^{S_e}$.
2. Compute $p_\sigma := T(p_1,\dots,p_m;\sigma)$.
3. Run sumcheck to check that $p_\sigma(a)=0$:

$$\sum_{\alpha,\beta \in H_v^{S_v}} \hat{C}_\sigma(\alpha,\beta)\cdot\hat{a}(\alpha)\cdot\hat{a}(\beta) = 0$$

4. Run (individual) low-degree test on Π_a:

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

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

at least cubic

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 #1: an IOP from the PCP for QESAT [1/2]

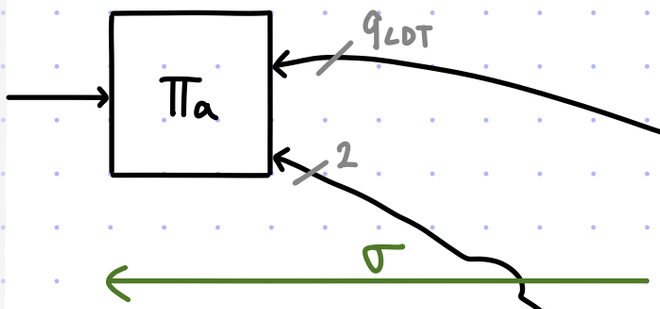
IDEA: reduce proof length by interacting when convenient.

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

1. Output $\hat{a}: \mathbb{F}^{S_v} \to \mathbb{F}$ as π_a.
   (The LDE of $a:[n]\to\mathbb{F}$.)
2. Compute $p_\sigma := \sum_{j=1}^{m} \sigma_j \cdot p_j$.
3. Sumcheck IP for "$p_\sigma(a)=0$"

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

1. Run (individual) low-degree test on π_a:
   $V_{LDT}(\mathbb{F}, S_v, |H_v|-1)$
2. Send random $\sigma \in \mathbb{F}^m$. (i)
3. Compute $p_\sigma := \sum_{j=1}^{m} \sigma_j p_j$.
4. Sumcheck IP for "$p_\sigma(a)=0$"



$P_{SC}(\mathbb{F}, H_v, 2S_v, 0, 2\cdot(|H_v|-1), \hat{c}_\sigma \cdot \hat{a} \cdot \hat{a})$ ⋮ sumcheck IP for $\sum_{\alpha,\beta \in H_v^{S_v}} \hat{c}_\sigma(\alpha,\beta)\cdot\hat{a}(\alpha)\cdot\hat{a}(\beta)=0$ ⋮ $V_{SC}(\mathbb{F}_{field}, H_{v\,domain}, 2S_{v\,vars}, 0_{sum}, 2\cdot(|H_v|-1)_{degree})$ (ii)

(i) send randomness for reducing m equations to 1 equation
(in fact we can set $p_\sigma := \sum_{j=1}^{m} \sigma_j p_j$ instead of $p_j := \sum_{0\le j_1,\dots,j_{S_e}<|H_e|} \sigma_1^{j_1}\cdots\sigma_{S_e}^{j_{S_e}} p_{j_1,\dots,j_{S_e}}$)

(ii) engage in an interactive sumcheck instead of sending a sumcheck PCP string

The IOP is public-coin: all verifier messages are random (so all queries WLOG at the end).

Recycling #1: an IOP from the PCP for QESAT [2/2]

The soundness error is

$$\max\left\{\varepsilon_{LDT}(\delta), 2\delta + O\left(\frac{1+s_v\cdot|H_v|}{|\mathbb{F}|}\right)\right\}.$$

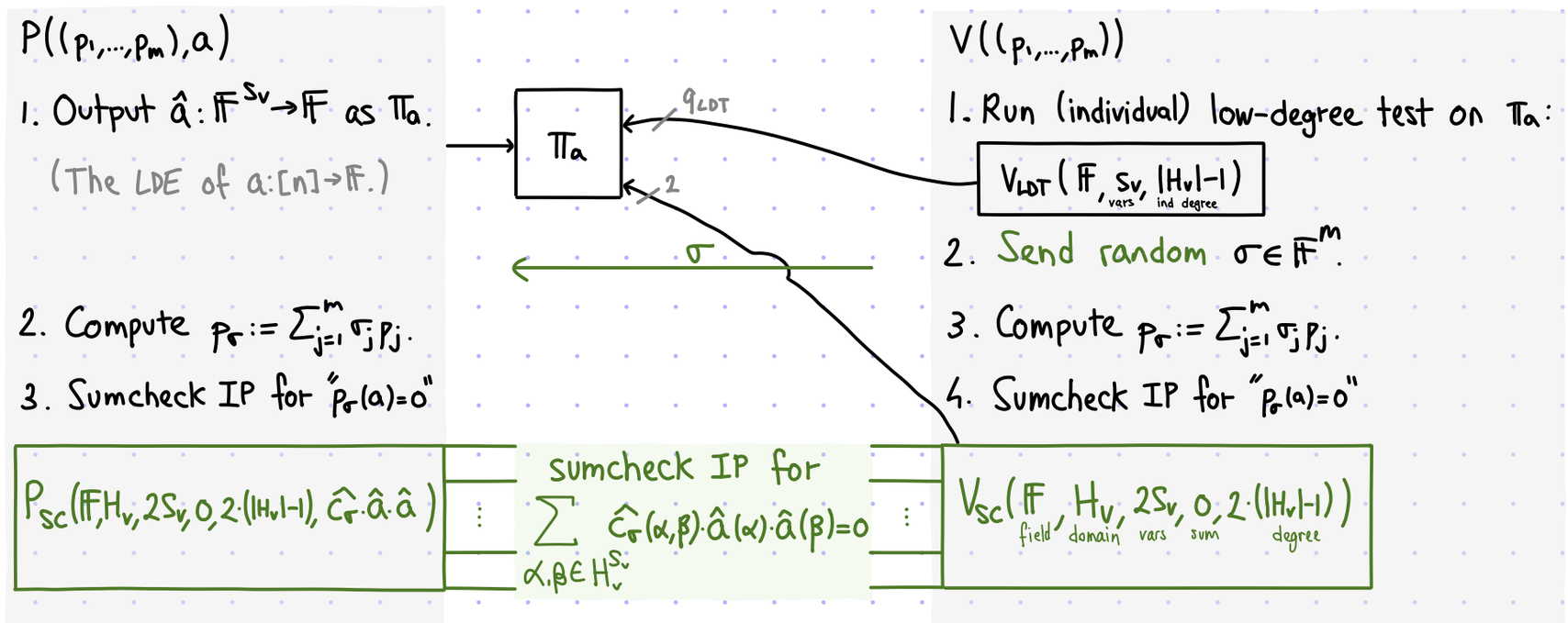
In the PCP it was

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

So we need

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

Take $|\mathbb{F}| = \Theta\left(\frac{\log n}{\log|H_v|}\cdot|H_v|\right)$.



The proof length is

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

take $|H_v| = O(\log^{1/\varepsilon} n)$

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

We proved the following theorem:

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

$$QESAT(\mathbb{F}) \in IOP\left[\varepsilon_c = 0, \varepsilon_s = 1/2, K = O\left(\varepsilon\cdot\frac{\log n}{\log\log n}\right), \Sigma = \{0,1\}, \ell = n^{1+\varepsilon}, q = \log^{O(1/\varepsilon)} n, r = \mathrm{poly}(m,n)\right]$$

almost linear

Recycling #2: an IOP from the PCP for IOSAT [1/3]

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

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

$$\mathrm{NTIME}(T)\subseteq \mathrm{IOP}\left[\begin{array}{l}\varepsilon_c=0,\\ \varepsilon_s=1/2,\end{array}\ k=O\left(\varepsilon\cdot\frac{\log T}{\log\log T}\right),\ \begin{array}{l}\Sigma=\{0,1\},\\ q=(\log T)^{O(1/\varepsilon)},\end{array}\ \ell=T^{1+\varepsilon},\ r=O(\log T\cdot(1+\varepsilon)),\ \begin{array}{l}pt=\mathrm{poly}(\ell)\\ vt=\mathrm{poly}(n,(\log T)^{1/\varepsilon})\end{array}\right]$$

We only see how to obtain the IOP for IOSAT:

theorem: For every $\varepsilon>0$,

$$\mathrm{IOSAT}\in \mathrm{IOP}\left[\begin{array}{l}\varepsilon_c=0,\\ \varepsilon_s=1/2,\end{array}\ k=O\left(\frac{\varepsilon\cdot n}{\log|\varphi|}\right),\ \begin{array}{l}\Sigma=\{0,1\},\\ q=|\varphi|^{O(1/\varepsilon)},\end{array}\ \ell=(|A|+|B|)^{1+O(\varepsilon)}\cdot\mathrm{poly}(|\varphi|),\ r=O(n\cdot(1+\varepsilon)),\ \begin{array}{l}pt=\mathrm{poly}(\ell)\\ vt=\mathrm{poly}(|z|,|\varphi|^{\frac{1}{\varepsilon}})\end{array}\right]$$

The missing ingredient for the IOP for NTIME(T) is a more efficient reduction that improves the size of the witness from $\Omega(T^3)$ to $T\cdot\mathrm{poly}(\log T)$.

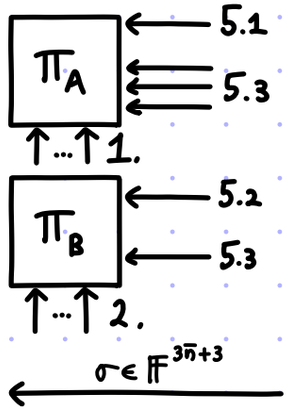
$$n=\cancel{3}\,1\cdot\log T+O(\log\log T),\quad m=\mathrm{poly}(\log T),\quad |\varphi|=\mathrm{poly}(\log T)$$

This yields: $\ell=T^{1+O(\varepsilon)}$, $q=(\log T)^{O(\frac{1}{\varepsilon})}$, $pt=\mathrm{poly}(T)$, $vt=\mathrm{poly}(|x|,(\log T)^{\frac{1}{\varepsilon}})$.

Recycling #2: an IOP from the PCP for IOSAT [2/3]

$P((m,n,\varphi,z),(A,B))$

1. Compute $C := T(\mathbb{F},H,(m,n,\varphi))$.
2. Output $\hat{A}: \mathbb{F}^{\bar{n}} \to \mathbb{F}$ as π_A.
   (The $(\mathbb{F},H,\bar{n})$-extension of $A:\{0,1\}^n \to \{0,1\}$.)
3. Output $\hat{B}: \mathbb{F}^{3\bar{n}+3} \to \mathbb{F}^m$ as π_B.
   (The $(\mathbb{F},H,3\bar{n}+3)$-extension of $B:\{0,1\}^{3n+3} \to \{0,1\}^m$.)
4. For $i \in [3\bar{n}+3]$, compute $\hat{\sigma}_i(x_i)$.
5. Do these sumcheck IPs in parallel:



$V((m,n,\varphi,z))$

1. Compute $C := T(\mathbb{F},H,(m,n,\emptyset))$.
2. $V_{LDT}^{\pi_A}(\mathbb{F}, \underset{\text{vars}}{\bar{n}}, \text{ind}, \underset{\text{degree}}{\leq |H|-1})$
3. $V_{LDT}^{\pi_B}(\mathbb{F}, \underset{\text{vars}}{3\bar{n}+3}, \underset{\text{outputs}}{m}, \text{ind}, \underset{\text{degree}}{\leq |H|-1})$
4. Sample $\sigma \in \mathbb{F}^{3\bar{n}+3}$.
5. Do these sumcheck IPs in parallel:

$$P_{sc}\left(\mathbb{F},H,\bar{n},0,\ \hat{A}(x)\cdot(\hat{A}(x)-1)\cdot\prod_{i\in[\bar{n}]}\hat{\sigma}_i(x_i)\right) \quad \vdots \quad \sum_{a\in H^{\bar{n}}} \hat{A}(a)\cdot(\hat{A}(a)-1)\cdot\prod_{i\in[\bar{n}]}\hat{\sigma}_i(a_i)=0 \quad \vdots \quad V_{sc}\left(\underset{\text{field}}{\mathbb{F}},\underset{\text{domain}}{H},\underset{\text{vars}}{\bar{n}},\underset{\text{sum}}{0},\underset{\text{degree}}{3\cdot(|H|-1)}\right)$$
$\to (\rho_1,\dots,\rho_{\bar{n}})$
$\leftarrow \pi_A(\rho)\cdot(\pi_A(\rho)-1)\cdot\prod_{i\in[\bar{n}]}\hat{\sigma}_i(\rho_i)$

$$P_{sc}\left(\mathbb{F},H,3\bar{n}+3,0^m,\ \hat{B}(x)\cdot(\hat{B}(x)-1)\cdot\prod_{i\in[3\bar{n}+3]}\hat{\sigma}_i(x_i)\right) \quad \vdots \quad \sum_{a\in H^{3\bar{n}+3}} \hat{B}(a)\cdot(\hat{B}(a)-1)\cdot\prod_{i\in[3\bar{n}+3]}\hat{\sigma}_i(a_i)=0^m \quad \vdots \quad V_{sc}\left(\underset{\text{field}}{\mathbb{F}},\underset{\text{domain}}{H},\underset{\text{vars}}{3\bar{n}+3},\underset{\text{sum}}{0^m},\underset{\text{degree}}{3\cdot(|H|-1)}\right)$$
$\to (\rho_1,\dots,\rho_{3\bar{n}+3})$
$\leftarrow \pi_B(\rho)\cdot(\pi_B(\rho)-1)\cdot\prod_{i\in[3\bar{n}+3]}\hat{\sigma}_i(\rho_i)$

$$P_{sc}\left(\mathbb{F},H,3\bar{n}+3,0,\ \cdot\prod_{i\in[3\bar{n}+3]}\hat{\sigma}_i(x_i)\right) \quad \vdots \quad \sum_{\substack{a=(v_1,v_2,v_3,c)\\ \in H^{3\bar{n}+3}}} C(a,\hat{A}(v_1),\hat{A}(v_2),\hat{A}(v_3),\hat{B}(a))\cdot\prod_{i\in[3\bar{n}+3]}\hat{\sigma}_i(a_i)=0 \quad \vdots \quad V_{sc}\left(\underset{\text{field}}{\mathbb{F}},\underset{\text{domain}}{H},\underset{\text{vars}}{3\bar{n}+3},\underset{\text{sum}}{0},\underset{\text{degree}}{(|\varphi|+1)\cdot(|H|-1)}\right)$$
where $X=(y_1,y_2,y_3,y_4)$

$(\rho_1,\dots,\rho_{3\bar{n}+3}) \Rightarrow$
- query π_A at $(\rho_1,\dots,\rho_{\bar{n}}),(\rho_{\bar{n}+1},\dots,\rho_{2\bar{n}}),(\rho_{2\bar{n}+1},\dots,\rho_{3\bar{n}})$
- query π_B at $(\rho_1,\dots,\rho_{3\bar{n}+3})$
- for every $i \in [3\bar{n}+3]$: eval $\hat{\sigma}_i$ at ρ_i
- eval C at $(\rho_1,\dots,\rho_{3\bar{n}+3},ans_1,ans_2,ans_3,ans_4)$

[Omitted is consistency between π_A and z.
This is another zero-on-subcube test.]

Recycling #2: an IOP from the PCP for IOSAT [3/3]

If $|\mathbb{F}| = \Omega(|H| \cdot |\varphi| \cdot \bar{n})$ then the protocol is sound:

$$\varepsilon_S \leq \max\left\{\varepsilon_{LDT}(\delta_A), \varepsilon_{LDT}(\delta_B), \begin{matrix} O(\delta_A+\delta_B) + \frac{\bar{n}\cdot(|H|-1)}{|\mathbb{F}|} + \frac{(3\bar{n}+3)\cdot(|H|-1)}{|\mathbb{F}|} + \frac{(3\bar{n}+3)\cdot(|H|-1)}{|\mathbb{F}|} + \frac{\bar{n}\cdot(|H|-1)}{|\mathbb{F}|} \\ + \frac{\bar{n}\cdot 3(|H|-1)}{|\mathbb{F}|} + \frac{(3\bar{n}+3)\cdot 3(|H|-1)}{|\mathbb{F}|} + \frac{(3\bar{n}+3)\cdot(|\varphi|+1)(|H|-1)}{|\mathbb{F}|} + \frac{\bar{n}\cdot(|H|-1)}{|\mathbb{F}|} \end{matrix}\right\} \leq O(1).$$

If $|\mathbb{F}| = |H| \cdot \text{poly}(|\varphi|)$ and $|H| = |\varphi|^{1/\varepsilon}$ then the protocol is efficient:

- round complexity: $1 + \max\{\bar{n}, 3\bar{n}+3, 3\bar{n}+3\} = O(\bar{n}) = O\left(\frac{n}{\log|H|}\right) = O\left(\frac{\varepsilon \cdot n}{\log|\varphi|}\right)$.
- proof length: $|\pi_A| + |\pi_B| + |sc_1| + |sc_2| + |sc_3| + |sc_4|$

$$\begin{aligned} &= |\mathbb{F}|^{\bar{n}} + |\mathbb{F}|^{3\bar{n}+3} \cdot m + O(\bar{n}\cdot|H|) + O(\bar{n}\cdot|H|\cdot m) + O(\bar{n}\cdot|H|\cdot|\varphi|) + O(\bar{n}\cdot|H|) \\ &= |\mathbb{F}|^{\frac{n}{\log|H|}} + |\mathbb{F}|^{3\cdot\frac{n}{\log|H|}+3} + O\left(\frac{n}{\log|H|}\cdot|H|\cdot|\varphi|\right) \\ &= |A|^{\frac{\log|\mathbb{F}|}{\log|H|}} + |B|^{\frac{\log|\mathbb{F}|}{\log|H|}} \cdot |\mathbb{F}|^{3\left(1-\frac{1}{\log|H|}\right)} + O\left(\frac{n}{\log|H|}\cdot|H|\cdot|\varphi|\right) \\ &= |A|^{1+O\left(\frac{\log|\varphi|}{\log|H|}\right)} + |B|^{1+O\left(\frac{\log|\varphi|}{\log|H|}\right)} \text{poly}(|H|,|\varphi|) + O\left(\frac{n}{\log|H|}\cdot|H|\cdot|\varphi|\right) \\ &= |A|^{1+O(\varepsilon)} + |B|^{1+O(\varepsilon)} \cdot \text{poly}(|\varphi|) + \text{poly}(|\varphi|) = (|A|+|B|)^{1+O(\varepsilon)} \cdot \text{poly}(|\varphi|). \end{aligned}$$

- query complexity: $(m+1)\cdot q_{LDT} + O(1) + O(\bar{n}\cdot|H|) + O(\bar{n}\cdot|H|\cdot m) + O(\bar{n}\cdot|H|\cdot|\varphi|) + O(\bar{n}\cdot|H|) = O(\bar{n}\cdot|H|\cdot|\varphi|) = O\left(\frac{\varepsilon\cdot n}{\log|\varphi|}\cdot|\varphi|^{1+\frac{1}{\varepsilon}}\right) = |\varphi|^{O\left(\frac{1}{\varepsilon}\right)}$.
- randomness complexity: $r_{LDT} + O(\bar{n}\cdot\log|\mathbb{F}|) = O\left(\frac{n}{\log|H|}\cdot(\log|H|+\log|\varphi|)\right) = O(n\cdot(1+\varepsilon))$.
- verifier time: $t_{LDT} + \text{poly}(\bar{n}, |H|\cdot|\varphi|) + \text{poly}(n, |H|, |z|) = \text{poly}(|z|, |\varphi|^{1/\varepsilon})$.

Towards IOPs With Linear Proof Length

We reduced proof length significantly, by recycling PCP constructions.

Q: can we reduce proof length further? (E.g. to LINEAR?)

There is a SERIOUS OBSTACLE to improving proof length:
we encode assignments via low-degree multi-variate extensions (aka Reed-Muller code)

This encoding incurs an inherent polynomial blowup:

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

To overcome this barrier, we will switch to a DIFFERENT ENCODING for assignments.

Reason for optimism: we are severely underusing the IOP model.
The IOP provers of today send a proof oracle in the first round only.
(And they send messages in other rounds.)

→ We should leverage proof oracles in more rounds!

From IOP to Succinct Interactive Argument [1/2]

Similarly to PCPs, IOPs lead (with the help of cryptography) to SUCCINCT INTERACTIVE ARGUMENTS.

An interactive argument (IA) in an interactive proof (IP) where soundness is relaxed to:

COMPUTATIONAL SOUNDNESS: $\forall x \notin L \;\; \forall$ efficient $\tilde{P} \quad \Pr_{r_V}\left[\langle \tilde{P}(1^\lambda), V(1^\lambda, x; r_V)\rangle = 1\right] \leq \varepsilon_s(\lambda, x)$.

theorem: Suppose that $L \in \mathrm{IOP}\left[\begin{array}{lll} \text{public-coin} & \text{proof alphabet } \Sigma & \text{prover time } pt \\ \text{round complexity } k & \text{proof length } \ell & \\ & \text{query complexity } q & \text{verifier time } vt \end{array}\right]$.

Then we can use crypto to construct a public-coin interactive argument for L with:

round complexity $k+1$ prover time $O_\lambda(pt)$

communication complexity $O_\lambda(k + q \cdot \log|\Sigma| \cdot \log \ell)$ verifier time $O_\lambda(vt)$

The (interactive) BCS protocol builds on Kilian's protocol (based on PCPs): commit to each proof oracle and then locally open the queried locations.

Q: what if the IOP is private-coin? An extension of the (interactive) BCS protocol yields a private-coin (succinct) interactive argument if the IOP is "public-query" (a necessary condition) and the IOP has an efficient transcript continuation sampler (unclear if necessary).

From IOP to Succinct Interactive Argument [2/2]

The (interactive) **BCS protocol** is described below.

$P_{IA}(1^\lambda, x, w)$		$V_{IA}(1^\lambda, x)$
	$\xleftarrow{h}$	Sample CRH: $h \leftarrow H_\lambda$
Produce IOP string: $\pi_i := P_{IOP}(x, w, \rho_1, \dots, \rho_{i-1})$.	For $i = 1, \dots, k$:	
Commit to it: $(rt_i, aux_i) := MT[h].Commit(\pi_i)$.	$\xrightarrow{rt_i}$	
	$\xleftarrow{\rho_i}$	Sample IOP randomness: $\rho_i \leftarrow \{0,1\}^{r_i}$.
Deduce query sets $\{Q_i \subseteq [\ell_i]\}_{i \in [k]}$ for $V_{IOP}^{(\pi_i)_{i\in[k]}}(x; (\rho_i)_{i\in[k]})$.		
$\forall\, i \in [k]$: Set answers $a_i := \pi_i[Q_i] \in \Sigma^{Q_i}$.		
$\forall\, i \in [k]$: Authenticate answers $pf_i := MT[h].Open(aux_i, Q_i)$.	$\xrightarrow{((Q_i, a_i, pf_i))_{i\in[k]}}$	$V_{IOP}^{[Q_i, a_i]_{i\in[k]}}(x; (\rho_i)_{i\in[k]}) \stackrel{?}{=} 1$
		$\bigwedge_{i\in[k]} MT[h].Check(rt_i, Q_i, a_i, pf_i) \stackrel{?}{=} 1$

- round complexity : $k+1$
- communication complexity: $poly(\lambda) + \lambda \cdot k + r + q \cdot (\log \ell + \log|\Sigma| + \lambda \cdot \log \ell) \approx poly(\lambda) + \lambda \cdot k + r + q \cdot (\log|\Sigma| + \lambda \cdot \log \ell)$.
- prover time: $time(P_{IOP}) + time(V_{IOP}) + O_\lambda(\ell \cdot \log|\Sigma|) \approx pt + O_\lambda(\ell \cdot \log|\Sigma|)$.
- verifier time: $poly(\lambda) + r + time(V_{IOP}) + O_\lambda(\log \ell \cdot \log|\Sigma|) \approx vt + O_\lambda(\log \ell \cdot \log|\Sigma|)$.

Bibliography

Intro to IOPs

- [BCS 2016]: Interactive oracle proofs, by Eli Ben-Sasson, Alessandro Chiesa, Nick Spooner.
- [KR 2008]: Interactive PCP, by Yael Kalai, Ran Raz.
- [RRR 2016]: Constant-round interactive proofs for delegating computation, by Omer Reingold, Guy Rothblum, Ron Rothblum. (▸Video1, Video2), (▸Video3).
- How computer scientists learned to reinvent the proof. Quanta 2022.

Succinct Arguments from IOPs

- [BCS 2016]: Interactive oracle proofs, by Eli Ben-Sasson, Alessandro Chiesa, Nick Spooner.
- [CY 2024]: Building cryptographic proofs from hash functions, by Alessandro Chiesa, Eylon Yogev. (▸Video)
- [CDGS 2023]: On the security of succinct interactive arguments from vector commitments, by Alessandro Chiesa, Marcel Dall'Agnol, Ziyi Guan, Nicholas Spooner.