Theory and Practice of Succinct Zero Knowledge Proofs

Lecture 06:
Multilinear PIOP for R1CS

Pratyush Mishra
UPenn
Fall 2023

Summary of last lecture

We constructed a succinct-verifier PIOP for R1CS with the following properties:

- Prover time: $O(n \log n)$
- Verifier time: $O(\log n)$
- Number of rounds: $O(1)$

This lecture: linear prover time

We will construct a succinct-verifier PIOP for R1CS with the following properties:

- Prover time: $O(n)$
- Verifier time: $O(\log n)$
- Number of rounds: $O(\log n)$

Key tool:
multilinear extensions

Key tool: Multilinear extensions

Multilinear Interpolation:

Given a function $f: \{0,1\}^\ell \to \mathbb{F}$, we can **extend** f to obtain a *multilinear* polynomial $p(X_1, \ldots, X_\ell)$ such that $p(x) = f(x)$ for all $x \in \{0,1\}^\ell$.

Multilinear means the polynomial has degree at most 1 in each variable.

Multilinear Lagrange Polynomial:

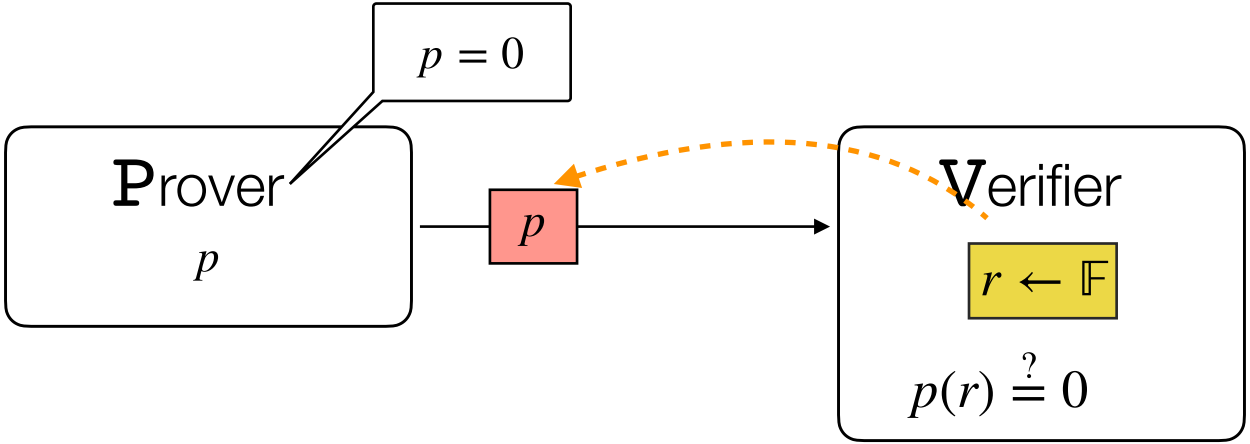
For each $i \in \{0,1\}^\ell$, $L^i(X)$ is 1 at i, and 0 for all $j \in \{0,1\}^\ell, j \neq i$.

Can write $L^i(X) := \prod_{j=1}^{\ell} (i_j \cdot X_j + (1 - i_j)(1 - X_j))$ => Can be evaluated in $O(\ell)$

Equiv, $L(i, X) := \prod_{j=1}^{\ell} (i_j \cdot X_j + (1 - i_j)(1 - X_j))$ is a multilinear poly over 2ℓ vars

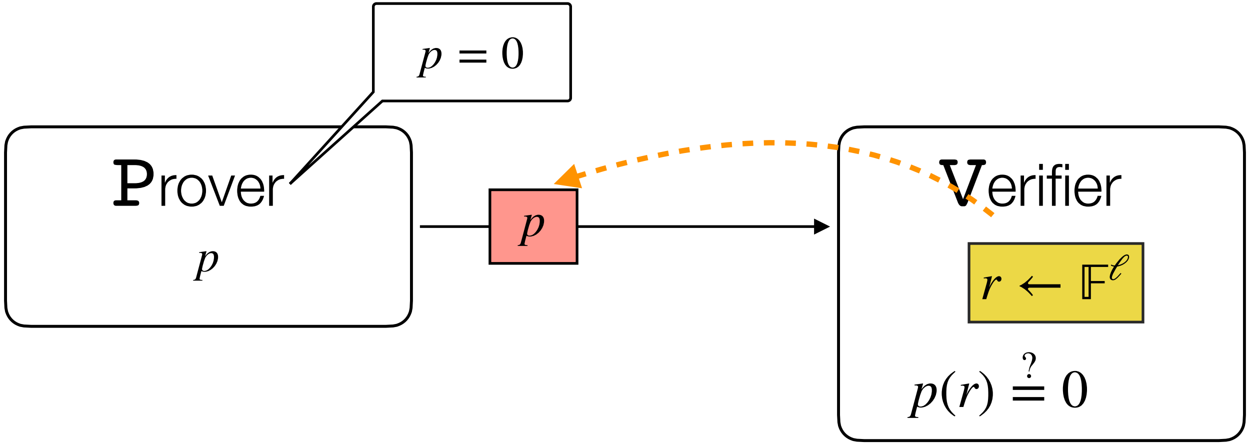
Common PIOPs

Recall: Univariate PIOP for Identity test



- **Completeness**: If $p = 0$, then definitely $p(r) = 0$.
- **Soundness**: If $p \neq 0$, then $p(r) = 0 \implies r$ is a root of p. But since r is random, this happens with probability $\dfrac{\deg(p)}{|\mathbb{F}|}$

Multilinear PIOP for Identity



- **Completeness**: If $p = 0$, then definitely $p(r) = 0$.
- **Soundness**: If $p \neq 0$, then $p(r) = 0 \implies r$ is a root of p.

How often does this happen?

Schwartz-Zippel-DeMillo-Lipton Lemma

Lemma: Let $p(X_1, \ldots, X_\ell) \in \mathbb{F}[X_1, \ldots, X_\ell]$ be an ℓ-variate degree d polynomial. Then $\Pr_{r_1,\ldots,r_\ell \leftarrow \mathbb{F}}[p(r_1, \ldots, r_\ell) = 0] = \dfrac{d}{|\mathbb{F}|}$

Proof: Via induction on number of variables ℓ

Base case: $\ell = 1$ follows from prior discussion

Hypothesis: Assume holds for $\ell - 1$ variables.

$\deg(p_i) \leq d - i$

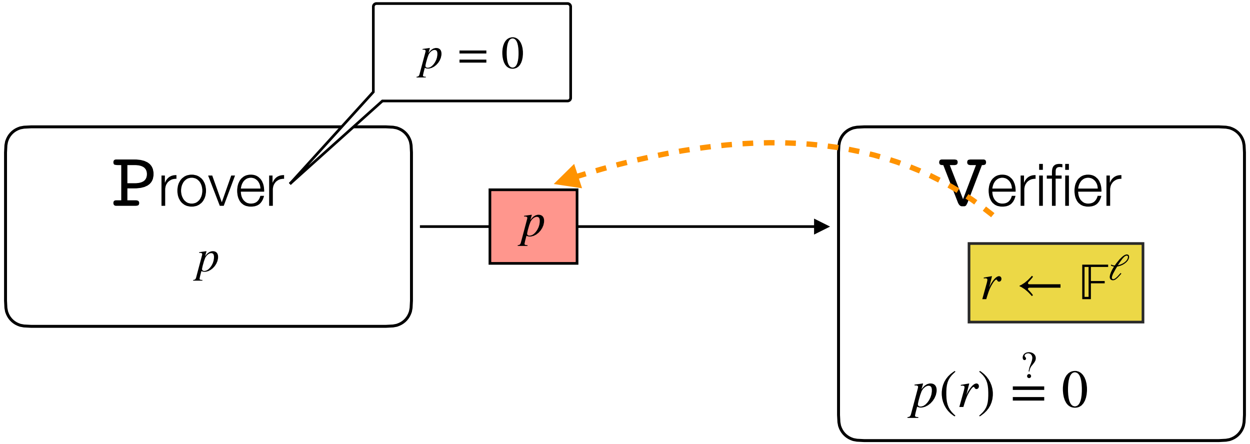
Then, we can write $p(X_1, \ldots, X_\ell) := \sum_{i=1}^{d} X_1^i p_i(X_2, \ldots, X_\ell)$

For random $r_2, \ldots, r_\ell$, $\Pr[p_i(r_2, \ldots, r_\ell) = 0] = (d - i)/|\mathbb{F}|$.

Also, $\Pr[p(r_1, r_2, \ldots, r_\ell) = 0 \ \mid \ p_i(r_2, \ldots, r_\ell) \neq 0] = i/|\mathbb{F}|$

Then,
$$\begin{aligned}\Pr[E_\ell] &= \Pr[E_\ell \cap E_{\ell-1}] + \Pr[E_\ell \cap \overline{E_{\ell-1}}] \\ &\leq \Pr[E_{\ell-1}] + i/|\mathbb{F}| \\ &= \frac{d}{|\mathbb{F}|}\end{aligned}$$

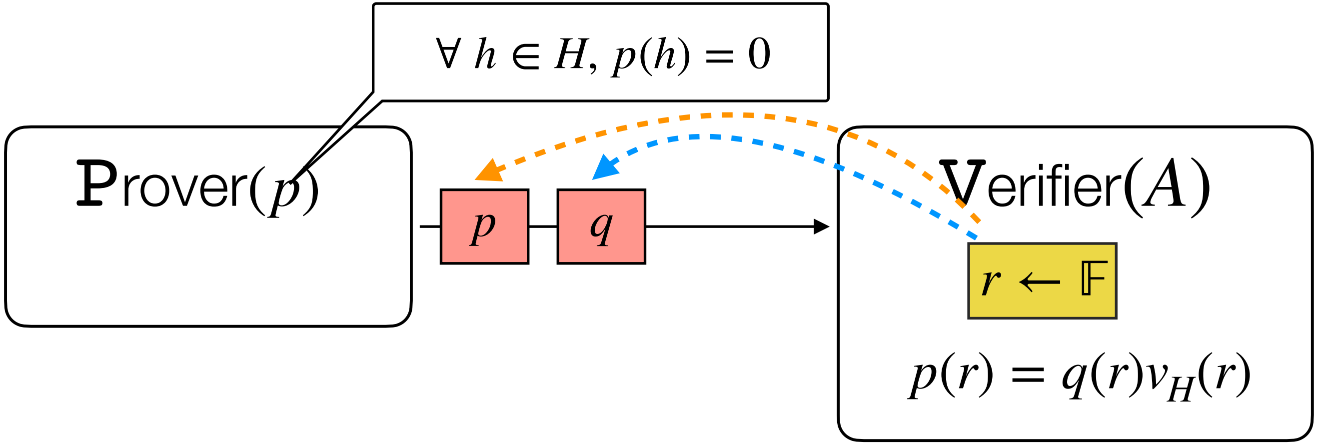
Multilinear PIOP for Identity



- **Completeness**: If $p = 0$, then definitely $p(r) = 0$.
- **Soundness**: If $p \neq 0$, then $p(r) = 0 \implies r$ is a root of p.

From SZDL lemma, happens wp $\dfrac{\ell}{|\mathbb{F}|}$

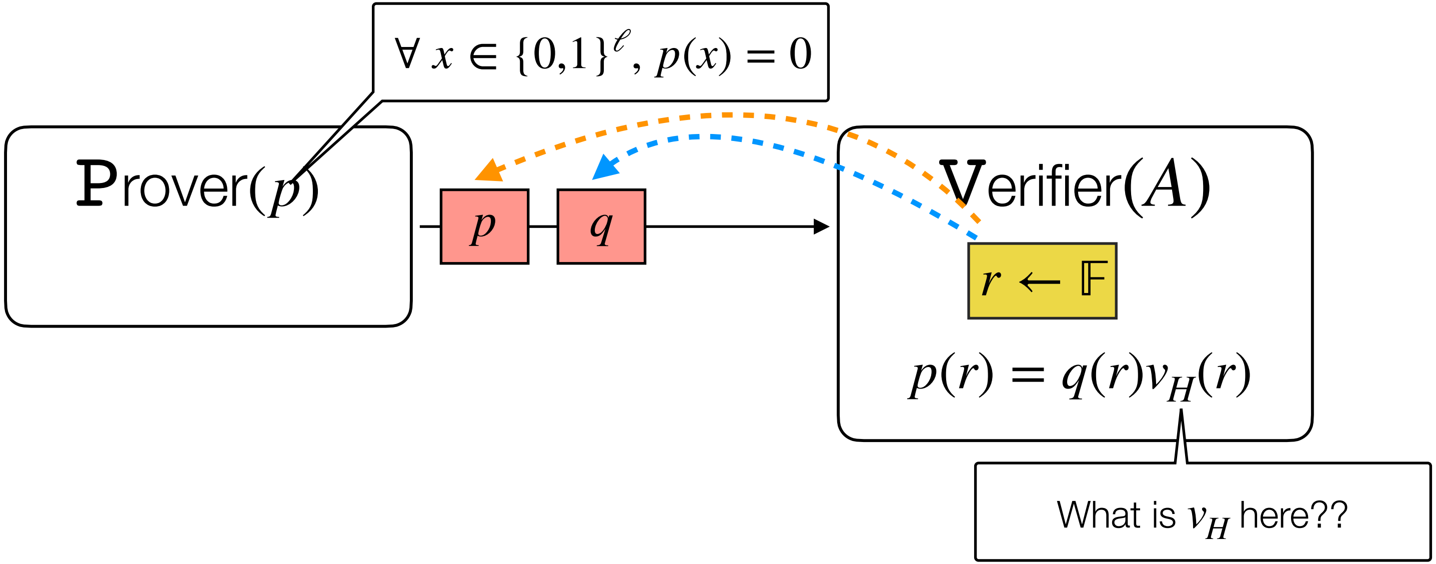
PIOP for ZeroCheck



Lemma: $\forall h \in H,\ \ p(h) = 0$ if and only if $\exists q$ such that $p = q \cdot v_H$.

- **Completeness**: Follows from lemma, and completeness of previous PIOP.
- **Soundness**: The lemma means that we have to check only equality of polynomials via the previous PIOP, and so soundness reduces to that of the previous PIOP.

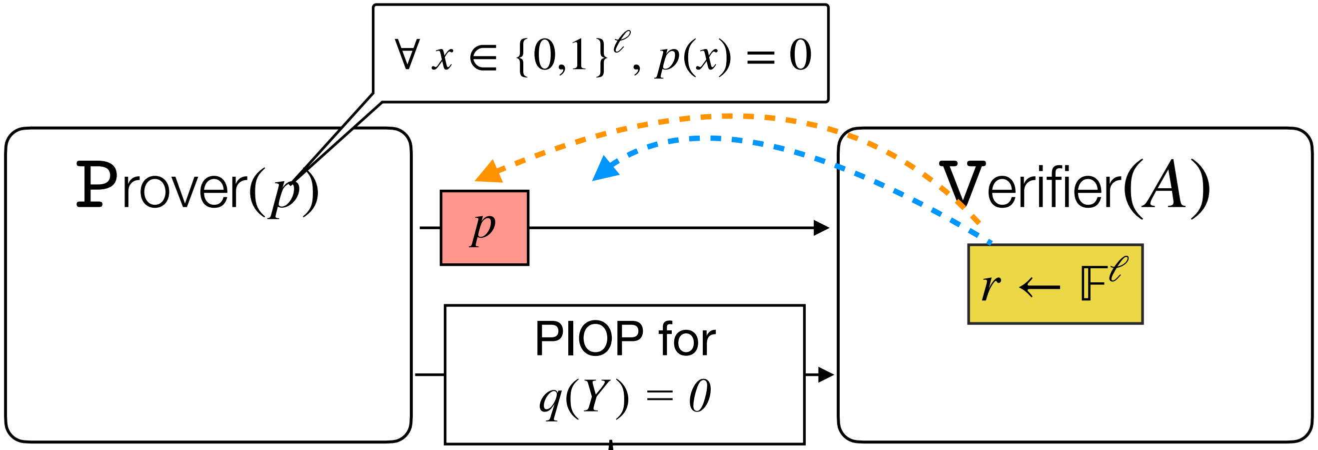
Multilinear PIOP for ZeroCheck



Lemma: $\forall\, x \in \{0,1\}^{\ell},\ p(x) = 0$ if and only if

$$q(Y) := \sum_{x \in \{0,1\}^{\ell}} p(x)L(x, Y) = 0$$

Multilinear PIOP for ZeroCheck



Internally requires

$$q(r) := \sum_{x \in \{0,1\}^\ell} p(x) L(x, r)$$

We need protocol for *multivariate sumcheck!*

Multivariate Sumcheck

(adapted from Justin Thaler’s slides)

Sumcheck Protocol [LFKN90]

- Input: V given oracle access to a ℓ-variate polynomial g over field $\mathbb{F}$.
- Goal: compute the quantity:

$$\sum_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_\ell \in \{0,1\}} g(b_1, \ldots, b_\ell) \,.$$

Sumcheck Protocol [LFKN90]

- **Start**: P sends claimed answer C_1. The protocol must check:

$$C = \sum_{b_1 \in \{0,1\}} \dots \sum_{b_\ell \in \{0,1\}} g(b_1, \dots, b_\ell)$$

- **Round 1**:
  - P sends **univariate** polynomial $s_1(X_1)$ claimed to equal:

$$H(X_1) := \sum_{b_2 \in \{0,1\}} \dots \sum_{b_\ell \in \{0,1\}} g(X_1, b_2, \dots, b_\ell)$$

  - V checks that $C_1 = s_1(0) + s_1(1)$.

Completeness: If $C_1 = \sum_{b_1 \in \{0,1\}} \dots \sum_{b_\ell \in \{0,1\}} g(b_1, \dots, b_\ell)$ then $C_1 = s_1(0) + s_1(1)$

Soundness: How can V check that $s_1 = H_1$?

Standard idea: Check that $s_1(r_1) = H_1(r_1)$ for random point r_1.
V can compute $s_1(r)$ directly from P's first message, but not $H_1(r_1)$.

Idea: Recursion!

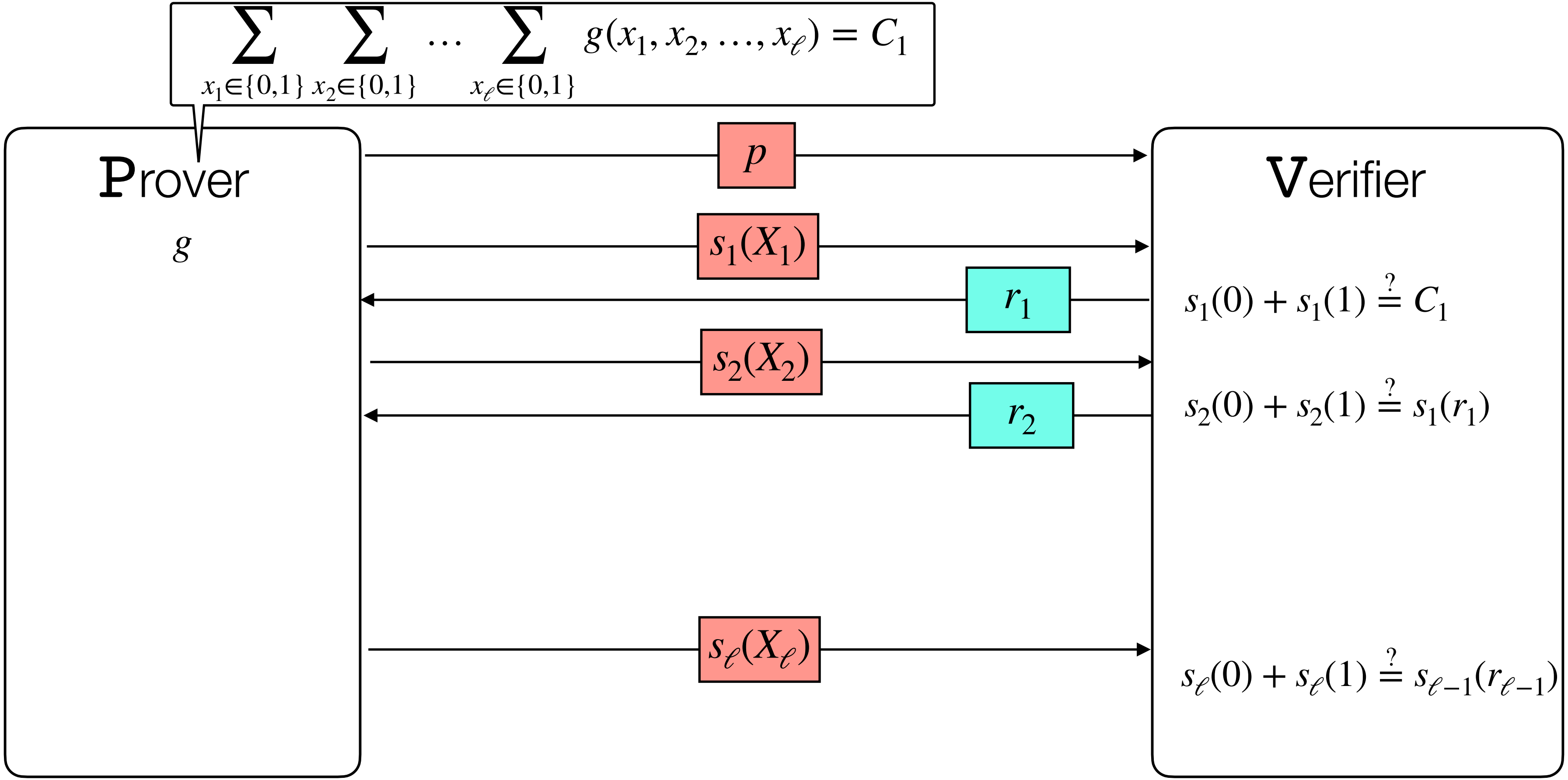
$$H(r_1) := \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_\ell \in \{0,1\}} g(r_1, b_2, \ldots, b_\ell)$$

This is another sumcheck claim, over $\ell - 1$ variables!

Recursive Sumcheck [LFKN90]

- **Start**: P sends claimed answer C_1. The protocol must check:
$$C_1 = \sum_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_\ell \in \{0,1\}} g(b_1, \ldots, b_\ell)\,.$$
- **Round 1**:
  - P sends **univariate** polynomial $s_1(X_1)$ claimed to equal:
$$H_1(X_1) := \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_\ell \in \{0,1\}} g(X_1, b_2, \ldots, b_\ell)$$
  - V checks that $C_1 = s_1(0) + s_1(1)$ and sends $r_1 \xleftarrow{\$} \mathbb{F}$.
- **Round 2**:
  - P sends **univariate** polynomial $s_2(X_2)$ claimed to equal:
$$H_2(X_2) := \sum_{b_3 \in \{0,1\}} \cdots \sum_{b_\ell \in \{0,1\}} g(r_1, X_2, b_3, \ldots, b_\ell)$$
  - V checks that $s_1(r_1) = s_2(0) + s_2(1)$ and sends $r_2 \xleftarrow{\$} \mathbb{F}$.

Sumcheck protocol



Completeness

We already saw that if Prover is honest, then the checks in a given round will pass.

So if P is honest in all rounds, all checks will pass

Soundness

Claim:

If P does not send the prescribed messages,

then V rejects with probability at least $1 - \dfrac{\ell \cdot d}{|\mathbb{F}|}$

(d is the maximum degree of g)

Soundness

Proof is by induction on the number of variables ℓ.

Base case: $\ell = 1$. In this case, P sends a single message $s_1(X_1)$ claimed to equal $g(X_1)$. V picks r_1 at random, checks that $s_1(r_1) = g(r_1)$.

$$\text{If } s_1 \neq g, \text{ then } \Pr_{r_1 \in \mathbb{F}}[s_1(r_1) = g(r_1)] \leq \frac{d}{|\mathbb{F}|}.$$

Soundness

Inductive case: $\ell > 1$.

- Recall: P's first message $s_1(X_1)$ is claimed to equal

$$H_1(X_1) := \sum_{b_2 \in \{0,1\}} \ldots \sum_{b_\ell \in \{0,1\}} g(X_1, b_2, \ldots, b_\ell)\,.$$

- Then V picks a random r_1 and sends r_1 to P. They (recursively) invoke sumcheck to confirm that $s_1(r_1) = H_1(r_1)$.
- If $s_1 \neq H_1$, then $\Pr_{r_1 \in \mathbb{F}}[s_1(r_1) = H_1(r_1)] \leq \dfrac{d}{|\mathbb{F}|}$.
- If $s_1(r_1) \neq H_1(r_1)$, P must prove a *false* claim in the recursive call.
  - Claim is about $g(r_1, X_2, \ldots, X_\ell)$, which is ℓ-1 variate.
  - By induction, P convinces V in the recursive call with prob at most $\dfrac{d(\ell - 1)}{|\mathbb{F}|}$.

Soundness analysis: wrap-up

Summary: if $s_1 \neq H_1$, V accepts with probability at most:

$$\Pr_{r_1 \in \mathbb{F}}[s_1(r_1) = H(r_1)]$$

$$+$$

$$\Pr_{r_2,\ldots,r_\ell \in \mathbb{F}}[\text{V accepts} \mid s_1(r_1) \neq H(r_1)]$$

$$\leq \frac{d}{|\mathbb{F}|} + \frac{d(\ell - 1)}{|\mathbb{F}|} \leq \frac{d\ell}{|\mathbb{F}|}$$

Costs of the sumcheck protocol

- Total communication is $O(d\ell)$ field elements.
  - P sends ℓ univariate polynomials of degree at most d.
  - V sends $\ell - 1$ messages, each consisting of one field element.

- V's runtime is: $O(d\ell + \text{[time to evaluate } g \text{ at random point]})$

- P's runtime is at most: $O(d2^{\ell} + \text{[time to evaluate } g \text{ at random point]})$

Multilinear PIOP
For R1CS

What checks do we need?

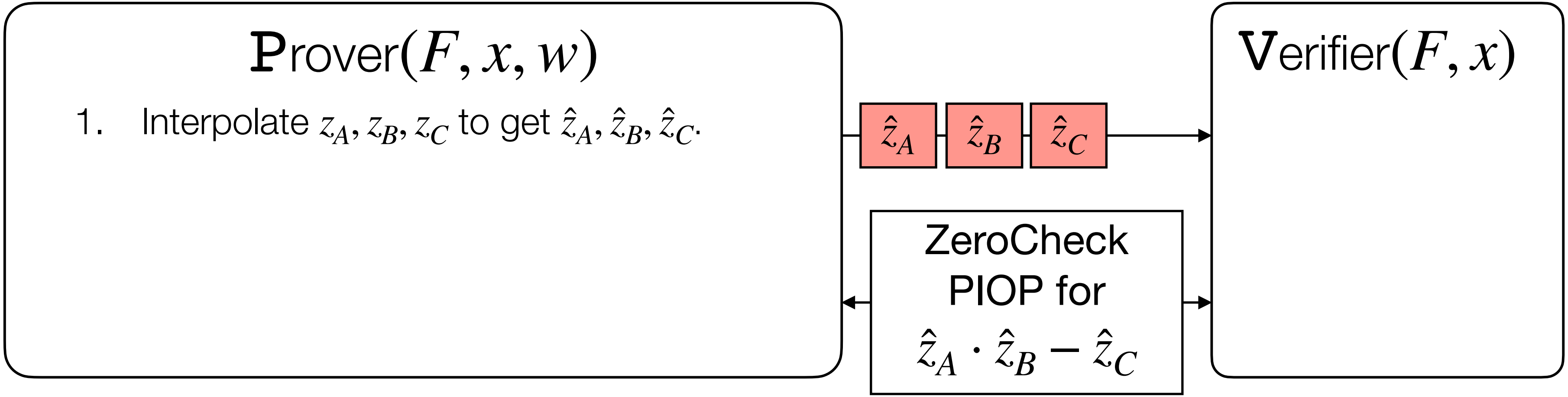
Step 1: Correct Hadamard product

check that for each i, $z_A[i] \cdot z_B[i] = z_C[i]$

Step 2: Correct matrix-vector multiplication

check that $Mz = z_M \quad \forall M \in \{A, B, C\}$

Multilinear PIOP for Rowcheck



Costs of Rowcheck PIOP

- Think of $\ell = \log_2 n$
- Total communication is $O(\log_2 n)$ field elements.

- V's runtime is: $O(\log_2 n + [\text{time to evaluate } g \text{ at random point}])$

- P's runtime is at most: $O(n + [\text{time to evaluate } g \text{ at random point}])$

Multilinear PIOP for Lincheck

Prover(M, z)

1. Compute $z_M := Mz$
2. Interpolate z, z_M to get $\hat{z}, \hat{z}_M$
3. Interpolate $(\vec{r}, \vec{r}^\top M)$ to get $(\hat{r}, \hat{r}_M)$

$\hat{z}$ $\hat{z}_M$

r

Sumcheck for $\hat{r} \cdot \hat{z}_M - \hat{r}_M \cdot \hat{z}$

Verifier(M)

1. $r \xleftarrow{\$} \mathbb{F}$
2. $\vec{r} := (1, r, \ldots, r^{n-1})$
3. Interpolate $(\vec{r}, \vec{r}^\top M)$ to get $(\hat{r}, \hat{r}_M)$

To prove $\displaystyle\sum_{b_1 \in \{0,1\}} \ldots \sum_{b_\ell \in \{0,1\}} \hat{r} \cdot \hat{z}_M - \hat{r}_M \cdot \hat{z} = 0$

Costs of Lincheck PIOP

- Think of $\ell = \log_2 n$
- Total communication is $O(\log_2 n)$ field elements.

- V's runtime is: $O(\log_2 n + \text{[time to evaluate } g \text{ at random point]})$

- P's runtime is at most: $O(n + \text{[time to evaluate } g \text{ at random point]})$