Transparent SNARKs over Galois Rings

Yuanju Wei[1,2], Xinxuan Zhang[1,2] and Yi Deng[1,2]

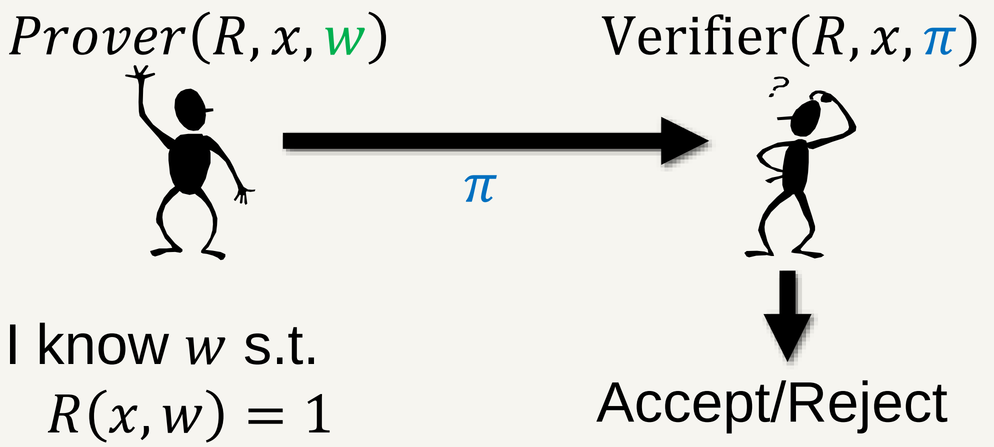
[1] Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, CAS

[2] School of Cyber Security, University of Chinese Academy of Sciences

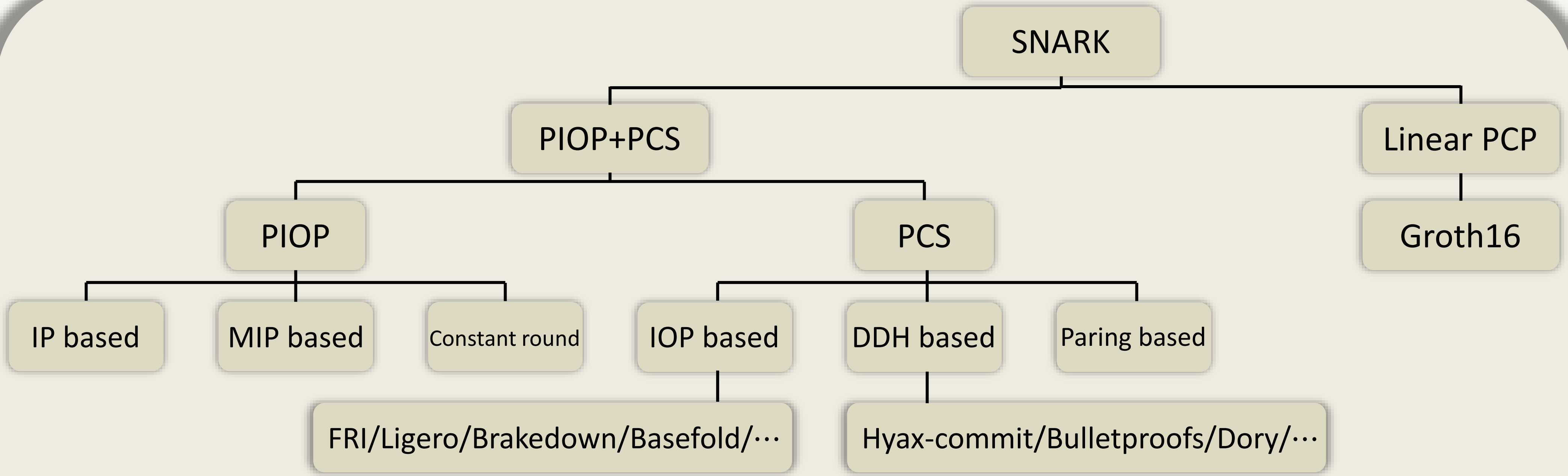
SNARK

- SNARK is a succinct non-interactive argument of knowledge.

- **Non-interactive.**
- **Succinctness:** sublinear proof sizes and sublinear verifier time.
- **Transparent:** It does not require a trusted setup.

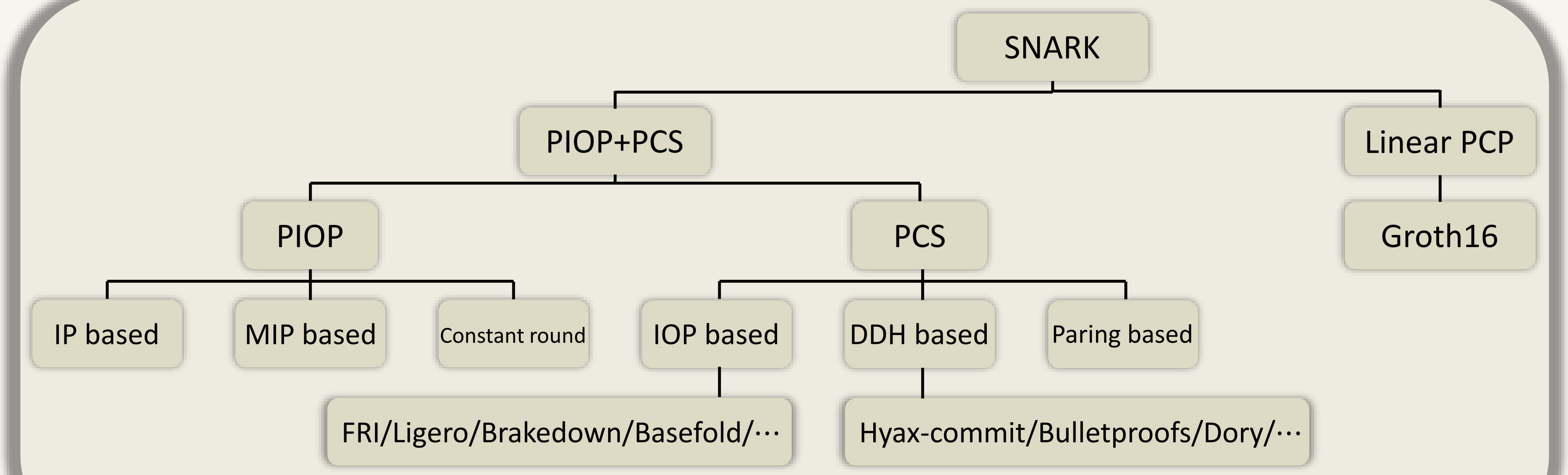


Why we need SNARKs over Galois Rings?



Source: [THA22]

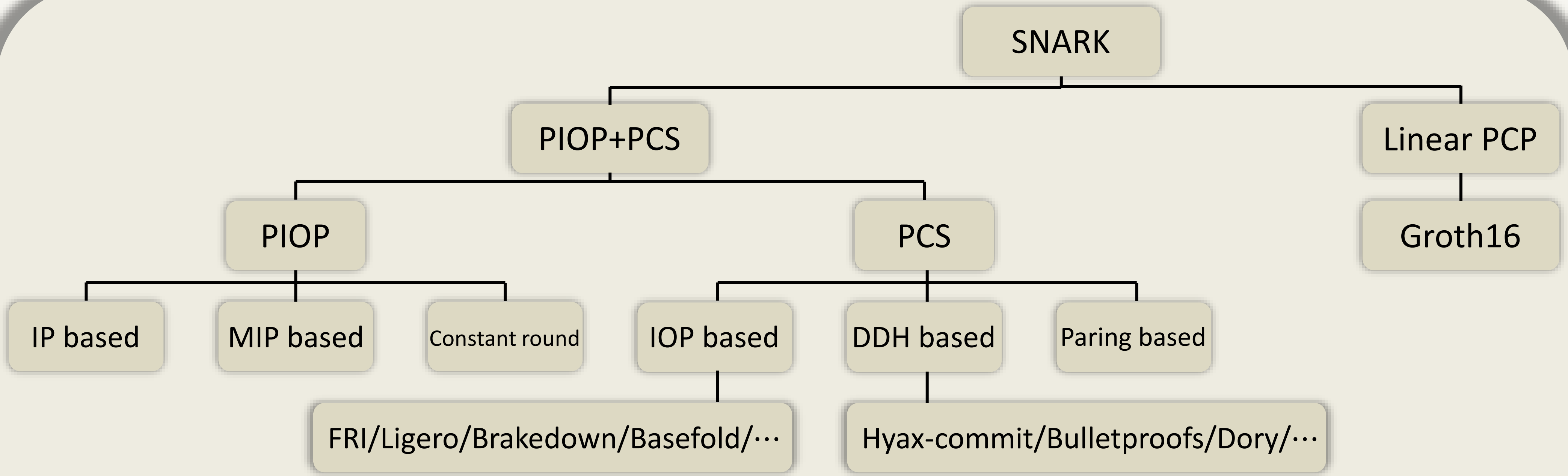
Why we need SNARKs over Galois Rings?



Source: [THA22]

- CPU computation over 2^{32} or 2^{64};
- Floating-point operation over 2^{k};
- FHE ciphertext in integer rings (can be mapped to Galois rings);
- ...

Why we need SNARKs over Galois Rings?

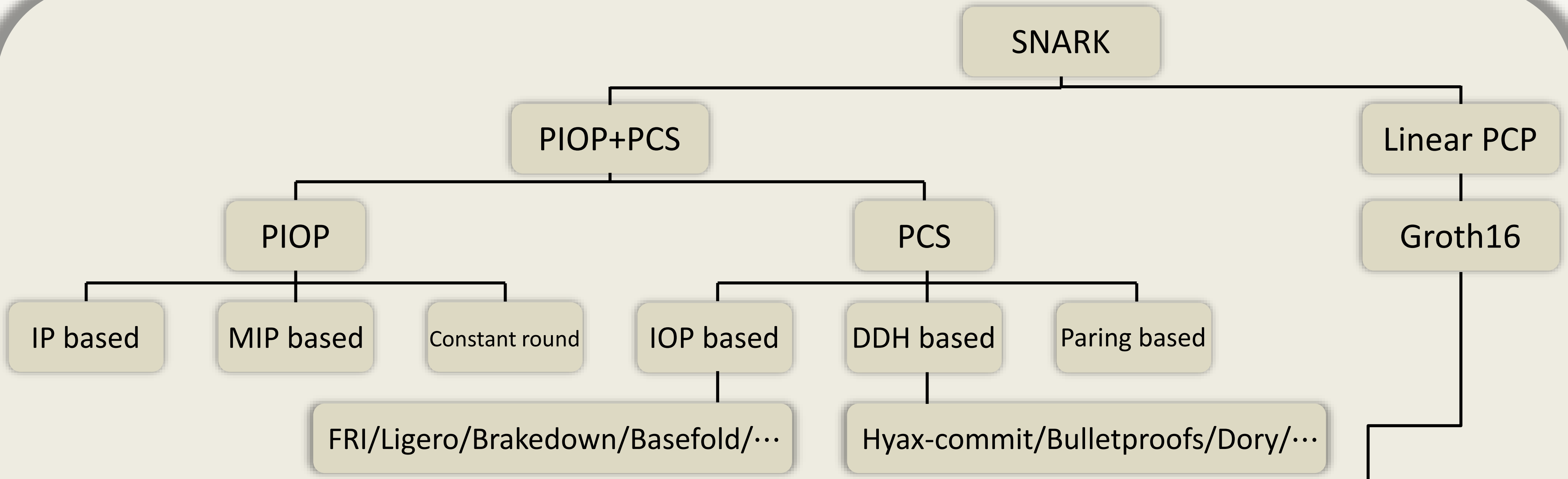


Source: [THA22]

- CPU computation over 2^{32} or 2^{64};
- Floating-point operation over 2^{k};
- FHE ciphertext in integer rings (can be mapped to Galois rings);
- …

- DDH based SNARK: Secp256k1 256bit
- Paring based SNARK: BN254 254bit
- FRI based SNARK: Goldilocks 64bit
- Expander code based SNARK: 128bit

Why we need SNARKs over Galois Rings?



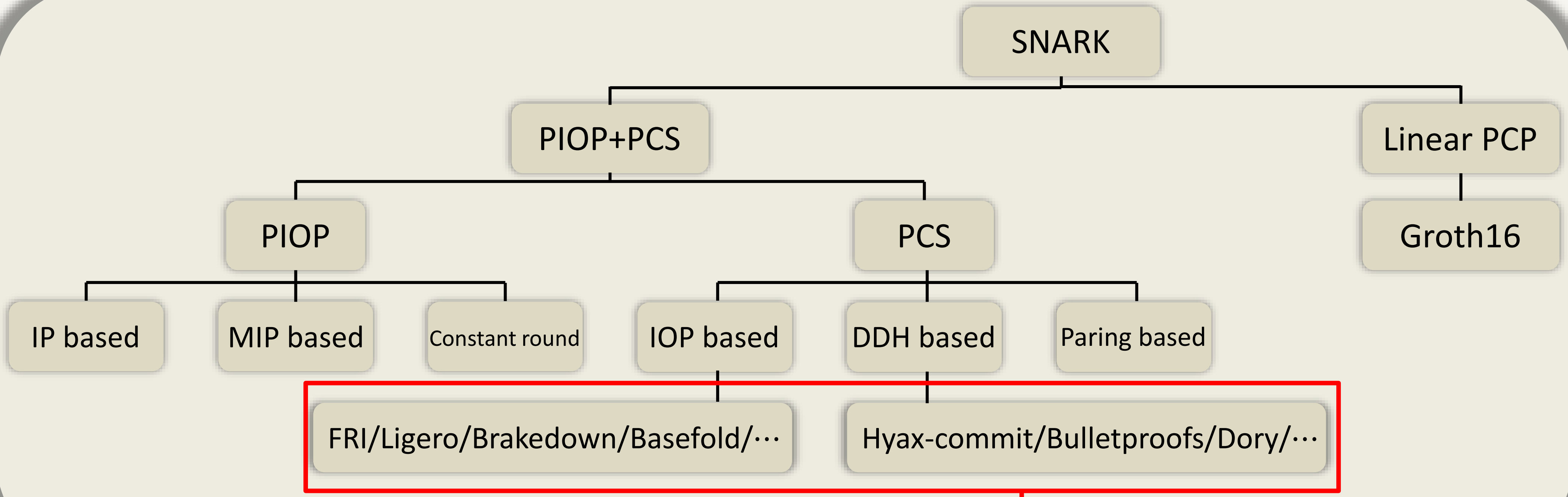
Source: [THA22]

Rinocchio [GNSV23]

- $O(l + m\log m)$ prover time;
- $O(1)$ proof size;
- $O(|x|)$ verifier time, where x is the statement;
- $O(l + m)$ length CRS.

l : wire number
m: multiplication numer
x: statement

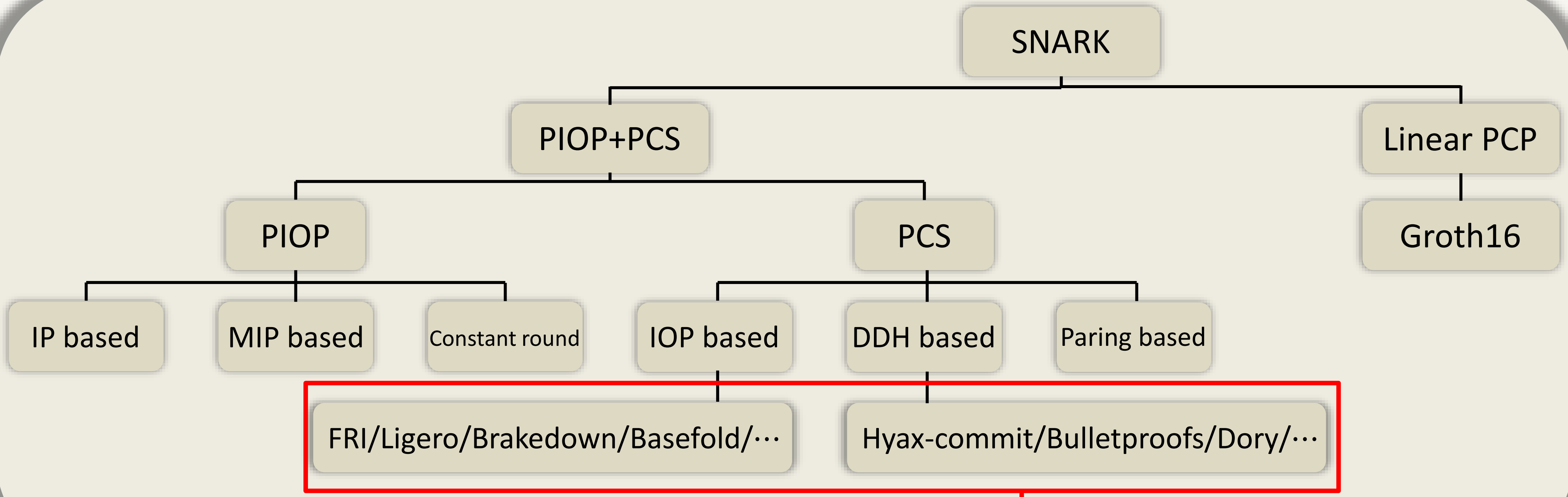
Why we need SNARKs over Galois Rings?



Source: [THA22]

Transparent but rely on finite field algebraic structure.

Why we need SNARKs over Galois Rings?

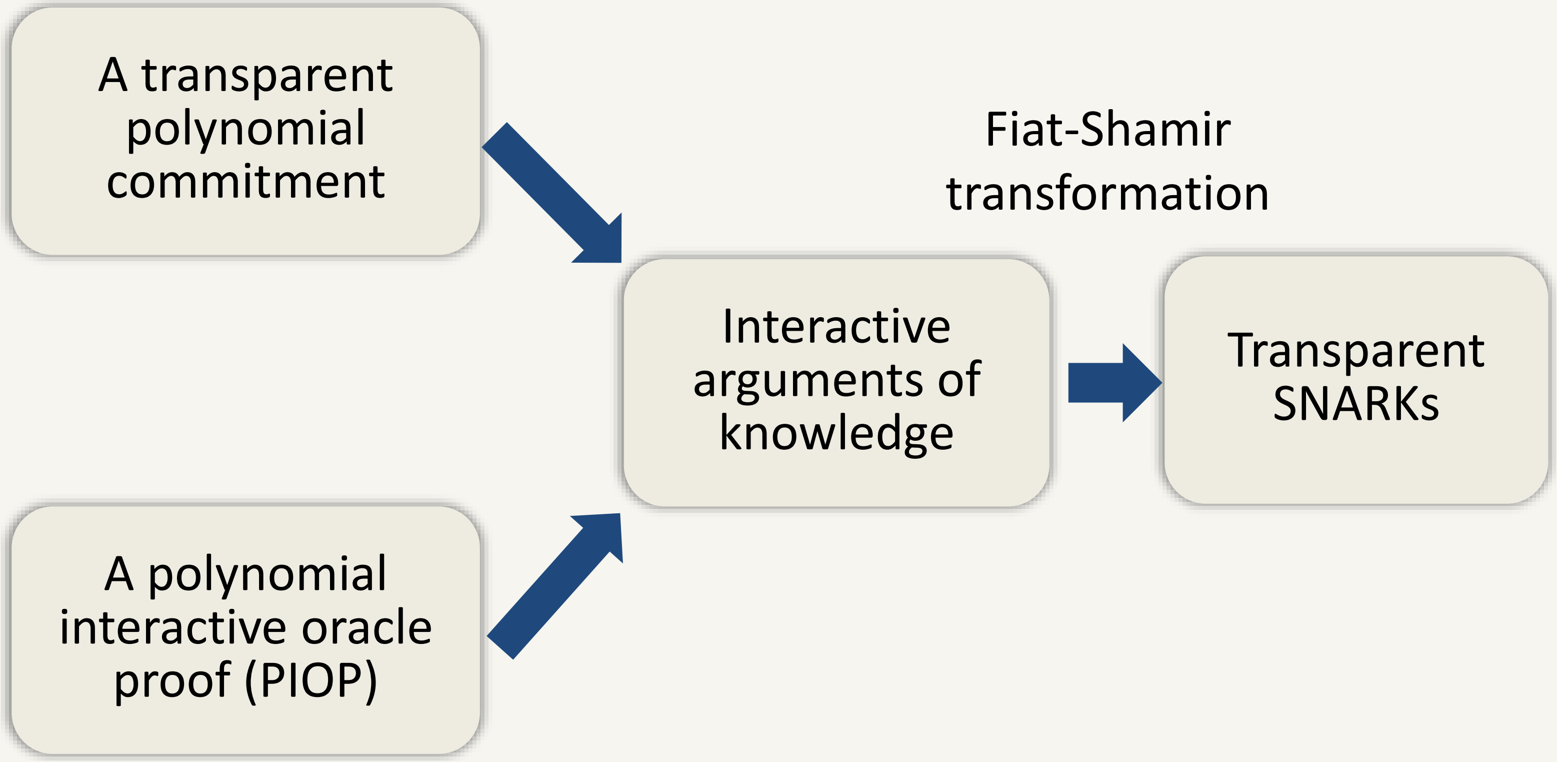


Source: [THA22]

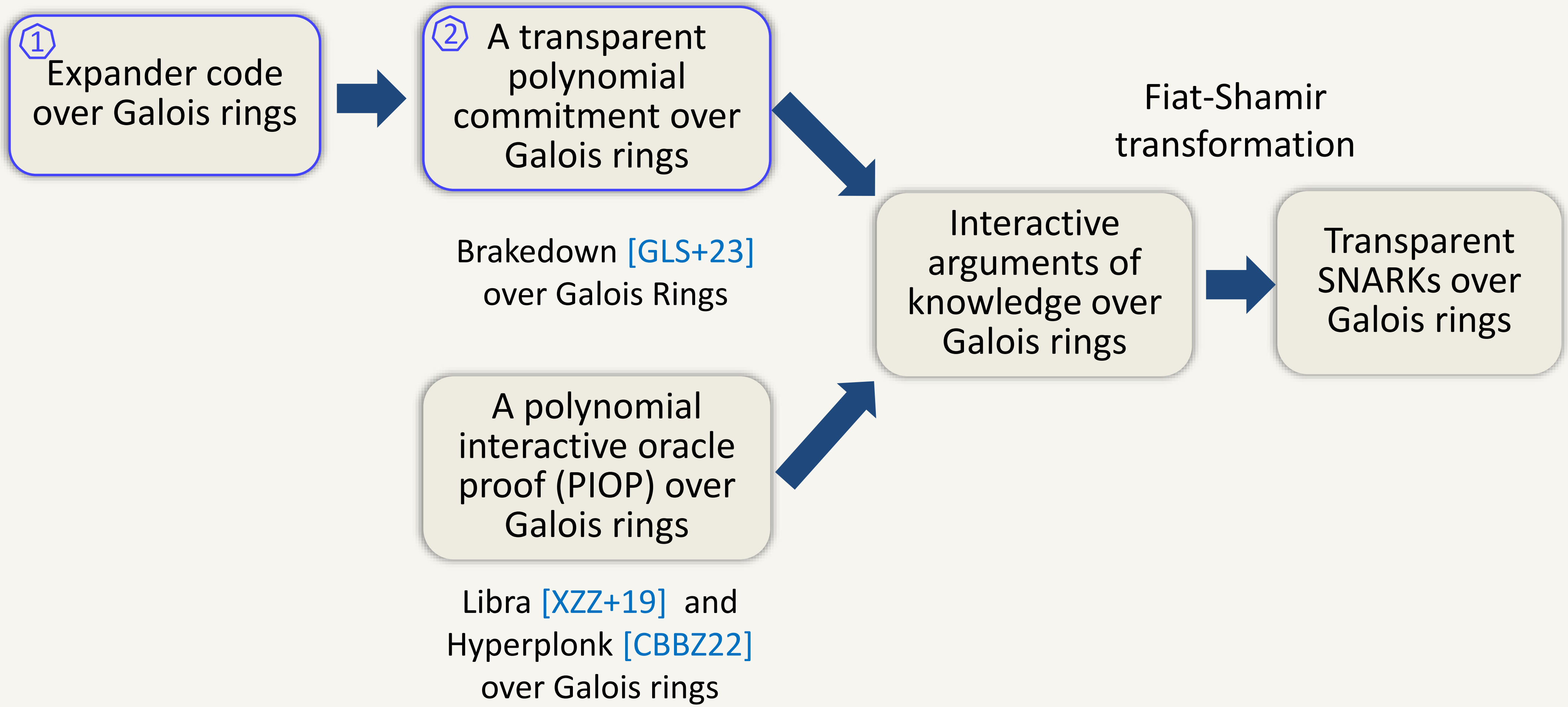
Transparent but rely on finite field algebraic structure.

Are Transparent Polynomial Commitments and SNARKs Possible Over Galois Rings?

PIOP+PCS


A transparent polynomial commitment
A polynomial interactive oracle proof (PIOP)
Fiat-Shamir transformation
Interactive arguments of knowledge
Transparent SNARKs

Our Contributions



Remainder of the talk

- **Expander code over Galois rings construction**
- **Brakedown commitment over Galois rings**
- **PIOP over Galois rings**

Galois Rings

- **Galois Rings :**

  $GR(p^s, r) \cong \mathbb{Z}_{p^s}[x]/f(x)$, where $f(x)$ is a monic polynomial of degree r which is irreducible modulo p^s.

- **Why Are SNARKs over Galois Rings So Challenging?**

  The presence of **zero divisors** in Galois rings invalidates the **Schwartz-Zippel lemma**, which is a fundamental component in proving the soundness of SNARKs.

  Schwartz-Zippel lemma over fields

  Let $P \in \mathbb{F}[x_1, \cdots, x_n]$ be a non-zero polynomial of total degree $d \geq 0$ over the field $\mathbb{F}$ and let $r_1, \cdots, r_n$ be selected at random independently and uniform from $\mathbb{F}$, then

$$\Pr[P(r_1, \cdots, r_n) = 0] \leq \frac{d}{|\mathbb{F}|}$$

Generalized Schwartz-Zippel lemma

- **Exceptional Set** [GNSV23]

  Let $A = \{a_1, \cdots, a_n\} \subset R$. We say that A is an exceptional set if $\forall i \neq j, a_i - a_j \in R^*$, where R^* is the set of all invertible elements in the ring R.

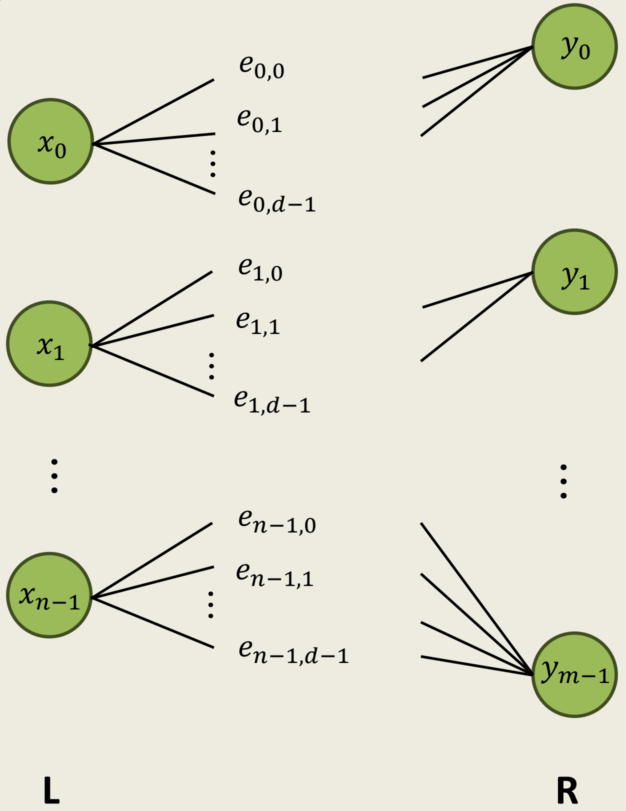
- **Generalized Schwartz-Zippel Lemma** [GNSV23]

  Let $f: R^n \to R$ be an n-variate nonzero polynomial. Let $A \subseteq R$ be a finite exceptional set. Let $\deg(f)$ denote the total degree of f. Then

$$\Pr_{a \in A^n}[f(a) = 0] \leq \frac{\deg(f)}{|A|}$$

The exceptional set of $GR(p^s, r)$ is $GF(p, r)$

Core Procedure in Expander Code



A bipartite graph $G = (L, R, E)$

$(x_0, \cdots, x_{n-1}) \rightarrow (y_0, \cdots, y_{m-1})$

$y_i = \sum_{x_j \in N(y_i)} e_{j,i} \cdot x_j$, $N(y_i)$ denote the neighbors of y_i.

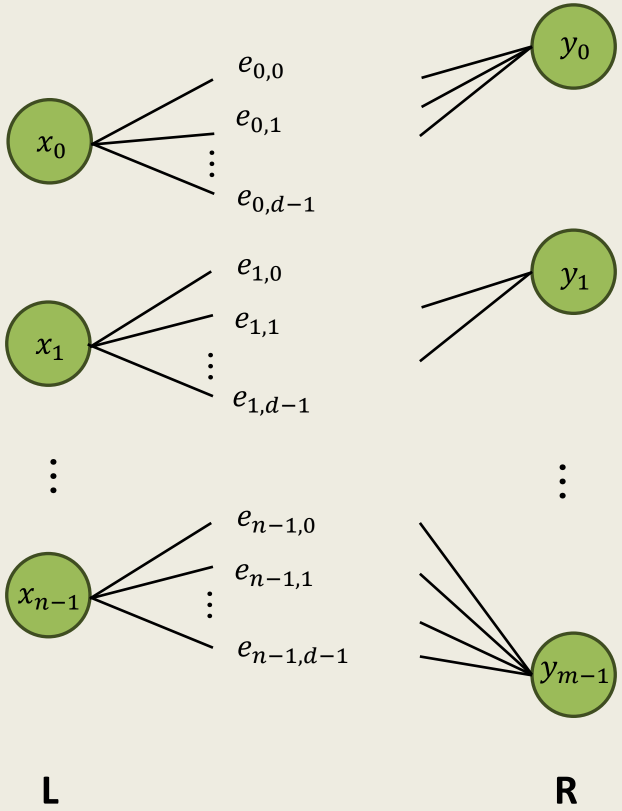
- **Expansion:**

  For every subset $S \subseteq L$ with $|S| = k$,$|N(S)| \geq b(k)$, where $b(k) = \max(k + 4, 1.28k)$

- **Nonzero:**

  For every subset $S \subseteq L$ satisfying the expansion and there is at least one nonzero element in S, the neighborhood $N(S)$ contains at least one non-zero element.

Core Procedure in Expander Code



A bipartite graph $G = (L, R, E)$

$(x_0, \cdots, x_{n-1}) \rightarrow (y_0, \cdots, y_{m-1})$

$y_i = \sum_{x_j \in N(y_i)} e_{j,i} \cdot x_j$, $N(y_i)$ denote the neighbors of y_i.

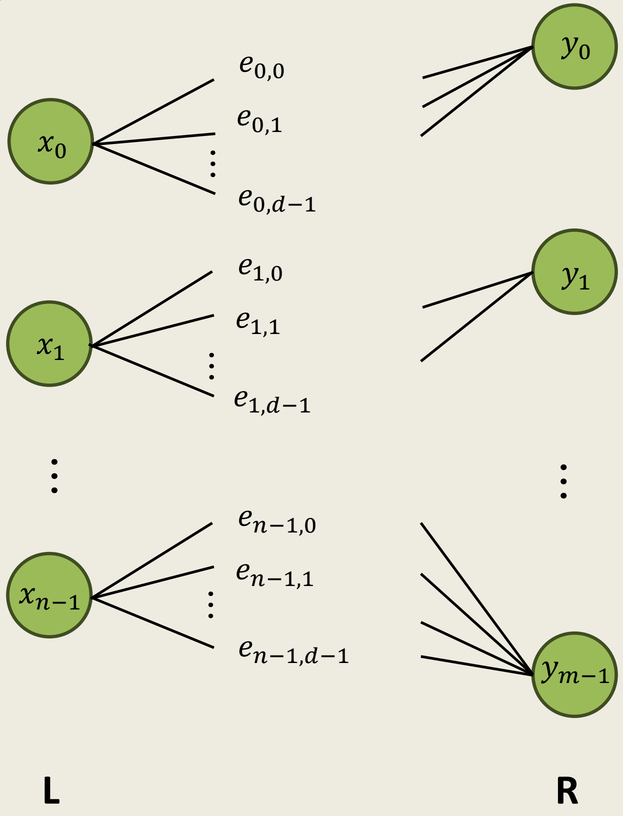
- **Expansion:**

  For every subset $S \subseteq L$ with $|S| = k, |N(S)| \geq b(k)$, where $b(k) = \max(k + 4, 1.28k)$

- **Nonzero:**

  For every subset $S \subseteq L$ satisfying the expansion and there is at least one nonzero element in S, the neighborhood $N(S)$ contains at least one non-zero element.

Core Procedure in Expander Code

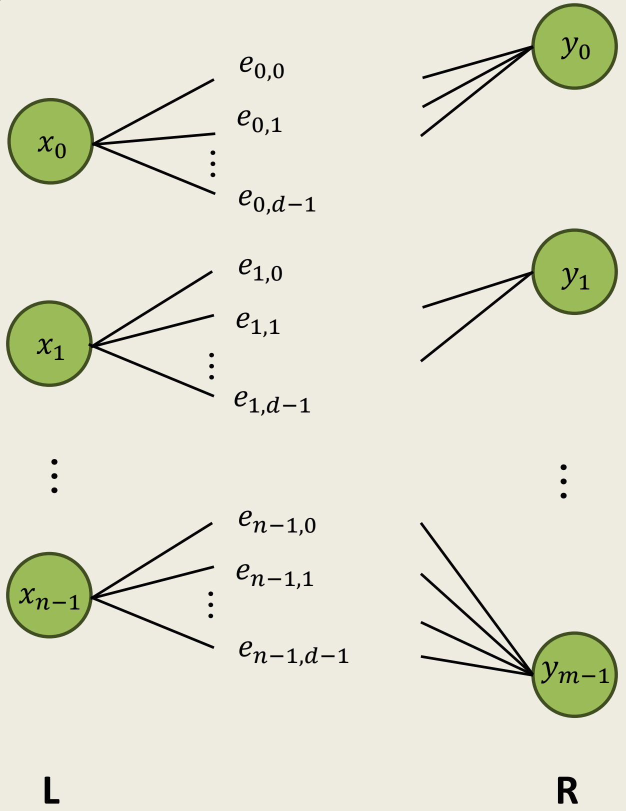


A bipartite graph $G = (L, R, E)$

$(x_0, \cdots, x_{n-1}) \rightarrow (y_0, \cdots, y_{m-1})$

$$y_i = \sum_{x_j \in N(y_i)} e_{j,i} \cdot x_j$$

Core Procedure in Expander Code



A bipartite graph $G = (L, R, E)$

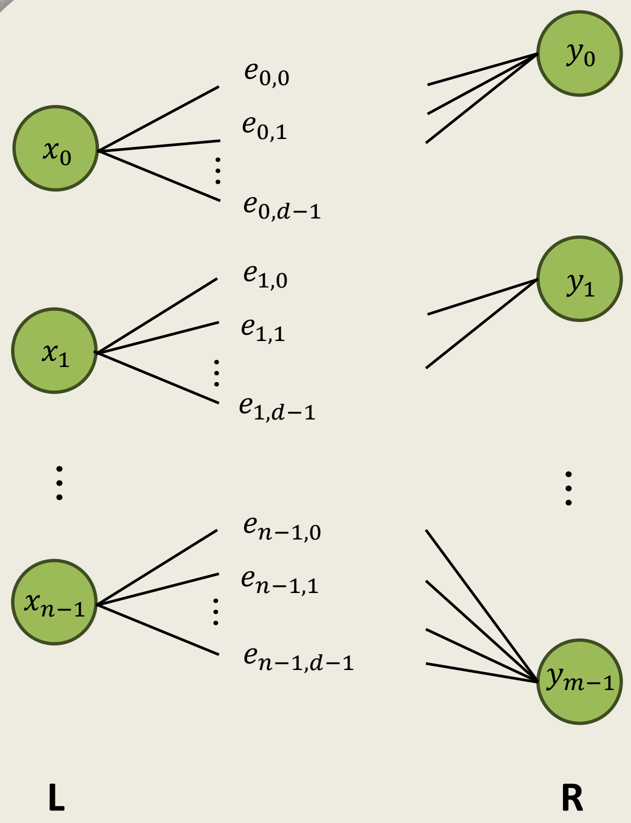
$(x_0, \cdots, x_{n-1}) \rightarrow (y_0, \cdots, y_{m-1})$

$$y_i = \sum_{x_j \in N(y_i)} e_{j,i} \cdot x_j$$

$$\downarrow$$

$$\Pr_{e_{i,j} \in GR(p^s, r)} [y_i(e_{j,i}) = 0]$$

Core Procedure in Expander Code



A bipartite graph $G = (L, R, E)$

$(x_0, \cdots, x_{n-1}) \rightarrow (y_0, \cdots, y_{m-1})$

$$y_i = \sum_{x_j \in N(y_i)} e_{j,i} \cdot x_j$$

$$\downarrow$$

$$\Pr_{e_{i,j} \in GR(p^s, r)} [y_i(e_{j,i}) = 0]$$

Generalized Schwartz-Zippel lemma fails to provide tight enough probability bounds.

Our Solution: Refined Parameter Analysis

- **Tightening the Bounds: Beyond Generalized Schwartz-Zippel**

  GCD over Galois rings:

  a is an element of ring $GR(p^s, r)$ and n is an integer. We define $\mathrm{GCD}(a, n)$ as $\mathrm{GCD}(a_0, \cdots, a_{r-1}, n)$. Where a is represented by $a_0 + a_1 x + \cdots + a_{r-1} x^{r-1}$.

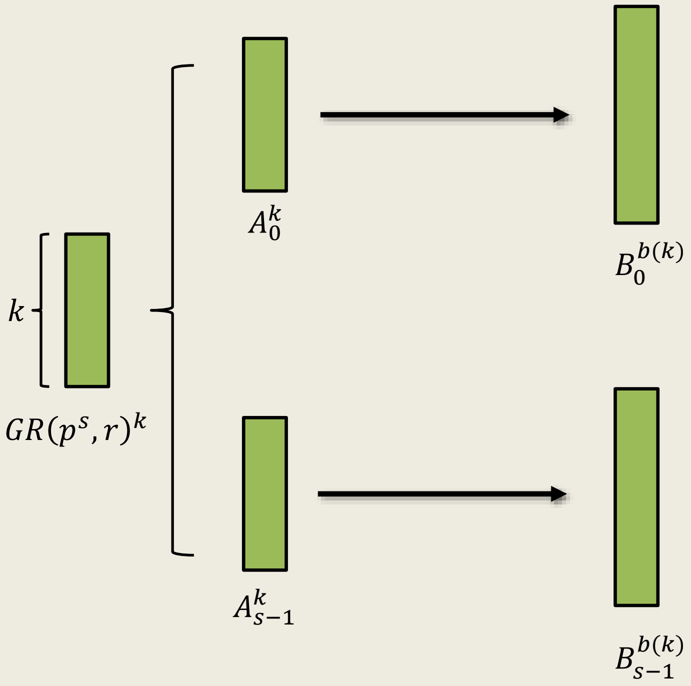
- **A key observation:**

  Consider elements $a, b \in GR(p^s, r)$. Let $d = \mathrm{GCD}(a, p^s)$. The linear equation $ax = b$ has at most d^r solutions.

$$\frac{d^r}{p^{sr}} \leq \frac{1}{p^r}$$

Equality is achieved when d attains its maximal value of p^{s-1}.

Our Solution: Refined Parameter Analysis



Classification by GCD with p^s

$$GR(p^s,r) = A_0 \cup \cdots \cup A_{s-1} \cup \{0\}$$

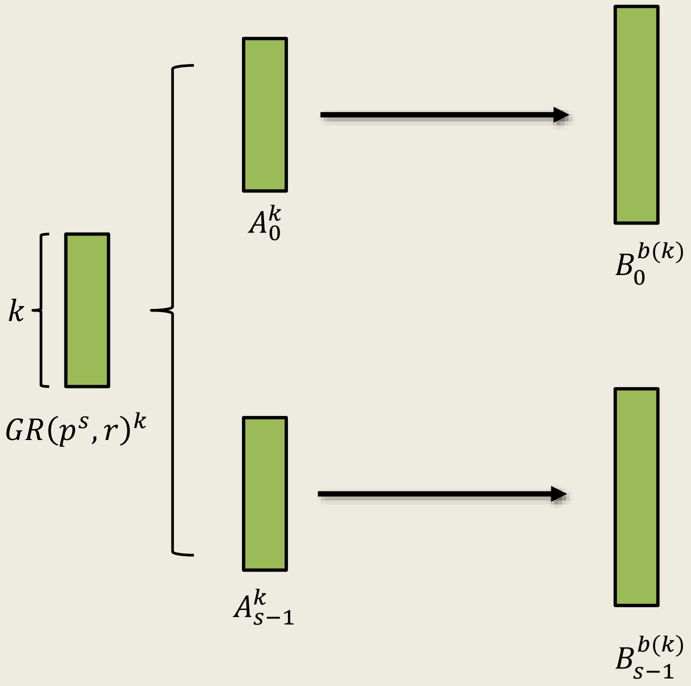
$$A_i = \{a | a \in GR(p^s,r) \cap \gcd(a,p^s) = p^i\},\ |A_i| = \left(\frac{p^s}{p^i}\right)^r$$

$$B_i = \{a | a \in GR(p^s,r) \cap \gcd(a,p^s) \geq p^i\}$$

Define the event E_i as A_i^k transformed to get $0^{b(k)}$:

$$\Pr[E_i] \leq |A_i|^k \frac{(p^i)^r}{p^{sr}} = \left(\frac{p^s}{p^i}\right)^{rk} \left(\frac{p^i}{p^r}\right)^{rb(k)} = \left(\left(\frac{p^i}{p^s}\right)^r\right)^{b(k)-k}$$

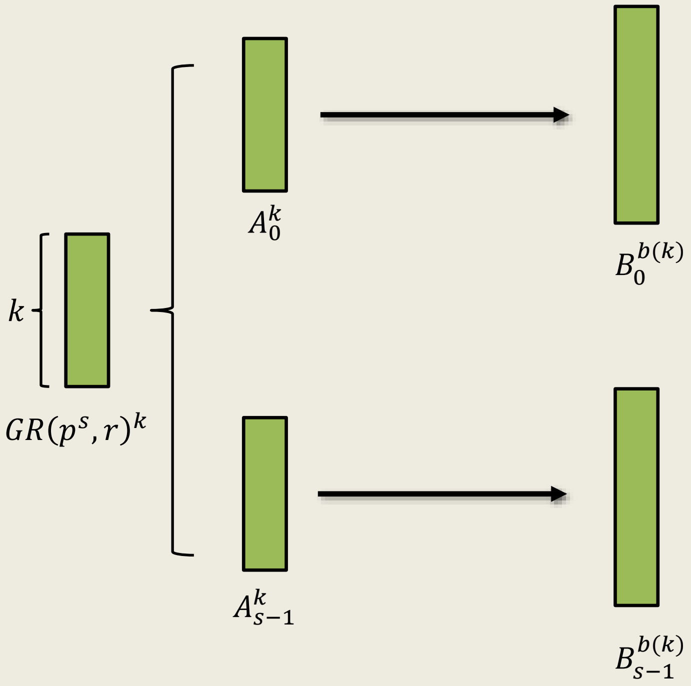
Our Solution: Refined Parameter Analysis



Classification by GCD with p^s

$$\begin{aligned}
\Pr[nonzero] &\le \Pr[E_0] + \Pr[E_1] + \cdots + \Pr[E_{s-1}] \\
&\le \sum_{i\in[0,s-1]} \left(\left(\frac{p^i}{p^s}\right)^r\right)^{b(k)-k} \\
&\le s\left(\left(\frac{p^{s-1}}{p^s}\right)^r\right)^{b(k)-k} \\
&= s\left(\frac{1}{p^r}\right)^{b(k)-k}
\end{aligned}$$

Our Solution: Refined Parameter Analysis



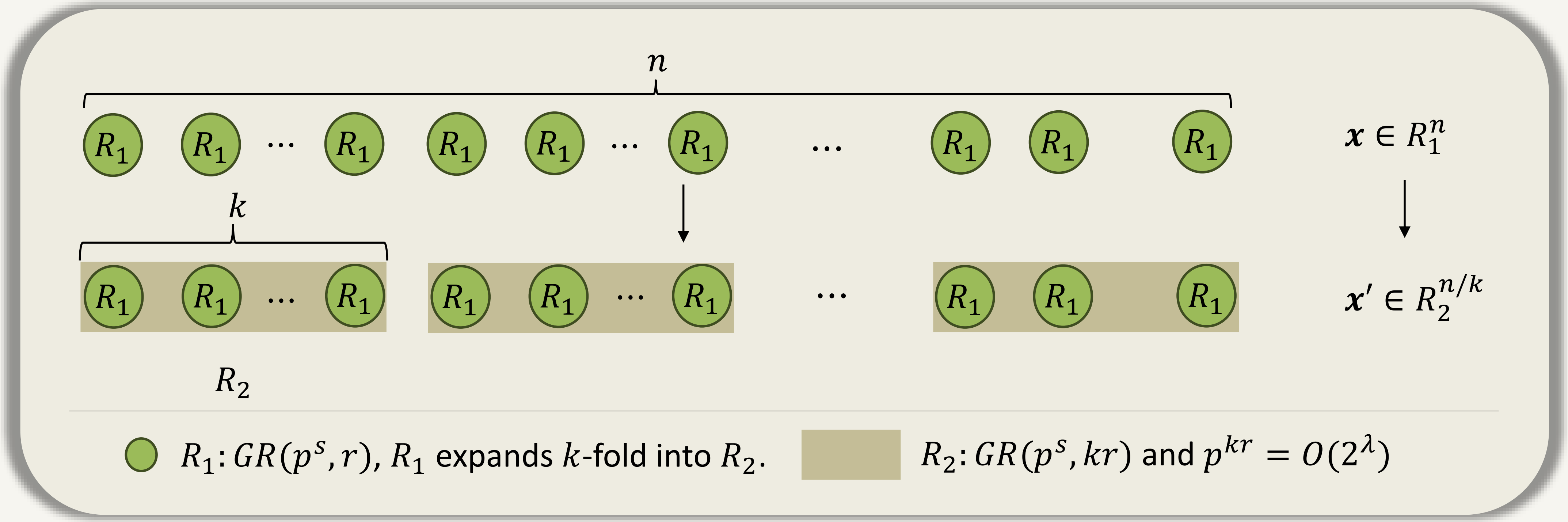
Classification by GCD with p^s

$$\Pr[nonzero] \le \Pr[E_0] + \Pr[E_1] + \cdots + \Pr[E_{s-1}]$$
$$\le \sum_{i\in[0,s-1]} \left(\left(\frac{p^i}{p^s}\right)^r\right)^{b(k)-k}$$
$$\le s\left(\left(\frac{p^{s-1}}{p^s}\right)^r\right)^{b(k)-k}$$
$$= s\left(\frac{1}{p^r}\right)^{b(k)-k}$$

We need $\left(\frac{1}{p^r}\right) = negl(\lambda)$, but What if p^r is not large enough?

Our Solution: Refined Parameter Analysis

- **Extend Binius [DP23] Block-level encoding to Galois rings**



Let Enc' is for linear encoding on R_2, then $Enc(\boldsymbol{x}) = Enc'(\boldsymbol{x}')$.

$\forall\, a \in R_1, a \cdot Enc(\boldsymbol{x}) = a \cdot Enc'(\boldsymbol{x}') = Enc'(a \cdot \boldsymbol{x}') = Enc(a \cdot \boldsymbol{x})$

Brakedown over Galois Rings

$$f(x_0, \cdots, x_{l-1}) = \sum_{b \in \{0,1\}^l} \prod_{i \in [0, l-1]} \big((1 - x_i)(1 - b_i) + x_i b_i\big) f(b)$$

$$U = \begin{bmatrix} f(0, \cdots, 0,0, \cdots 0) & f(0, \cdots, 0,0, \cdots 1) & \cdots & f(0, \cdots, 0,1, \cdots 1) \\ f(0, \cdots, 1,0, \cdots 0) & f(0, \cdots, 1,0, \cdots 1) & \cdots & f(0, \cdots, 1,1, \cdots 1) \\ \vdots & \vdots & \vdots & \vdots \\ f(1, \cdots, 1,0, \cdots 0) & f(1, \cdots, 1,0, \cdots 1) & \cdots & f(1, \cdots, 1,1, \cdots 1) \end{bmatrix}$$

$$\boldsymbol{s_1} = \big((1 - r_0, r_0) \otimes \cdots \otimes (1 - r_{l/2-1}, r_{l/2-1})\big)$$
$$\boldsymbol{s_2} = \big((1 - r_{l/2}, r_{l/2}) \otimes \cdots \otimes (1 - r_{l-1}, r_{l-1})\big)$$

$$f(r_0, \cdots, r_{l-1}) = \boldsymbol{s}_1^\top U \boldsymbol{s}_2$$

Brakedown over Galois Rings

$$f(x_0,\cdots,x_{l-1}) = \sum_{b\in\{0,1\}^l} \prod_{i\in[0,l-1]} \big((1-x_i)(1-b_i)+x_ib_i\big)f(b)$$

$$U = \begin{bmatrix} f(0,\cdots,0,0,\cdots 0) & f(0,\cdots,0,0,\cdots 1) & \cdots & f(0,\cdots,0,1,\cdots 1) \\ f(0,\cdots,1,0,\cdots 0) & f(0,\cdots,1,0,\cdots 1) & \cdots & f(0,\cdots,1,1,\cdots 1) \\ \vdots & \vdots & \vdots & \vdots \\ f(1,\cdots,1,0,\cdots 0) & f(1,\cdots,1,0,\cdots 1) & \cdots & f(1,\cdots,1,1,\cdots 1) \end{bmatrix}$$

$$\boldsymbol{s_1} = \Big((1-r_0,r_0)\otimes\cdots\otimes\big(1-r_{l/2-1},r_{l/2-1}\big)\Big)$$

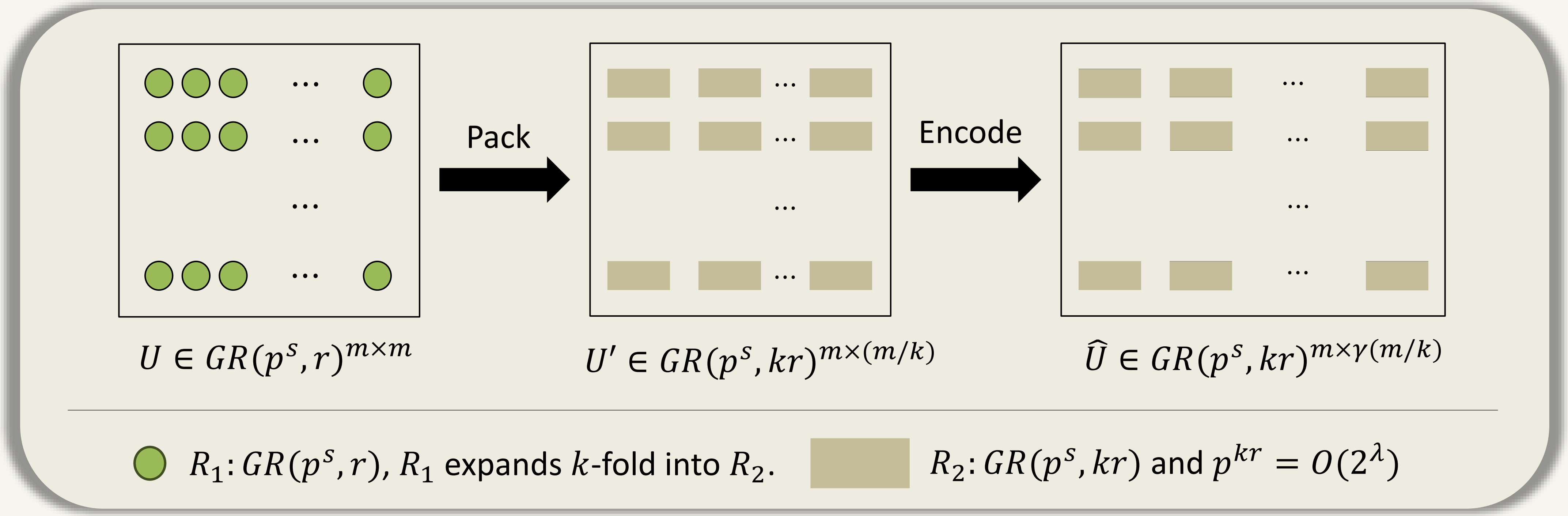
$$\boldsymbol{s_2} = \Big(\big(1-r_{l/2},r_{l/2}\big)\otimes\cdots\otimes(1-r_{l-1},r_{l-1})\Big)$$

$$f(r_0,\cdots,r_{l-1}) = \boxed{\boldsymbol{s}_1^{\top}U}\boldsymbol{s}_2$$

Brakedown over Galois Rings : Commit Phase

Let $\boldsymbol{U} \in GR(p^s, r)^{m \times m}$ be the coefficient matrix of the l-variable multilinear polynomial f to be committed, where $m = 2^{l/2}$ and $\boldsymbol{U} = (\boldsymbol{u}_0, \cdots, \boldsymbol{u}_{m-1})$.

Prover:

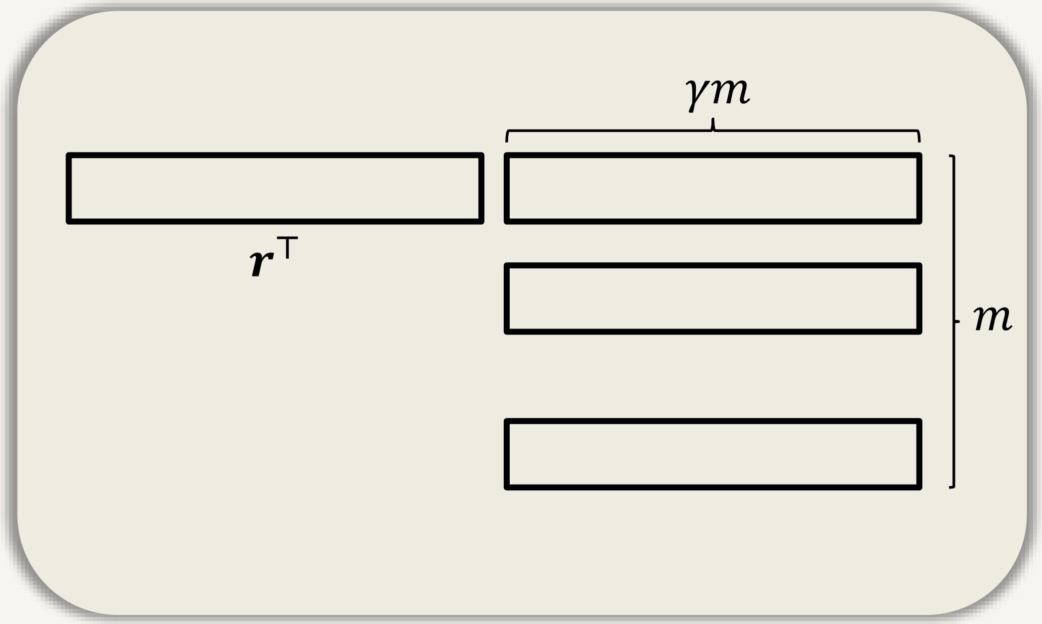


Where $\frac{1}{\gamma}$ is the code rate. The prover then constructs a Merkle tree from the $\hat{U}$ and sends its root hash to the verifier as the commitment.

Brakedown over Galois Rings: Testing Phase

Soundness Weakness

The verifier randomly checks a linear combination of matrix $\boldsymbol{U}$ by sampling an m-length vector $\boldsymbol{r}_1$. By the Schwartz-Zippel Lemma, the soundness error is at most $\frac{1}{p^r}$. However, the required soundness error must be $\frac{1}{p^{kr}}$ to meet security guarantees.

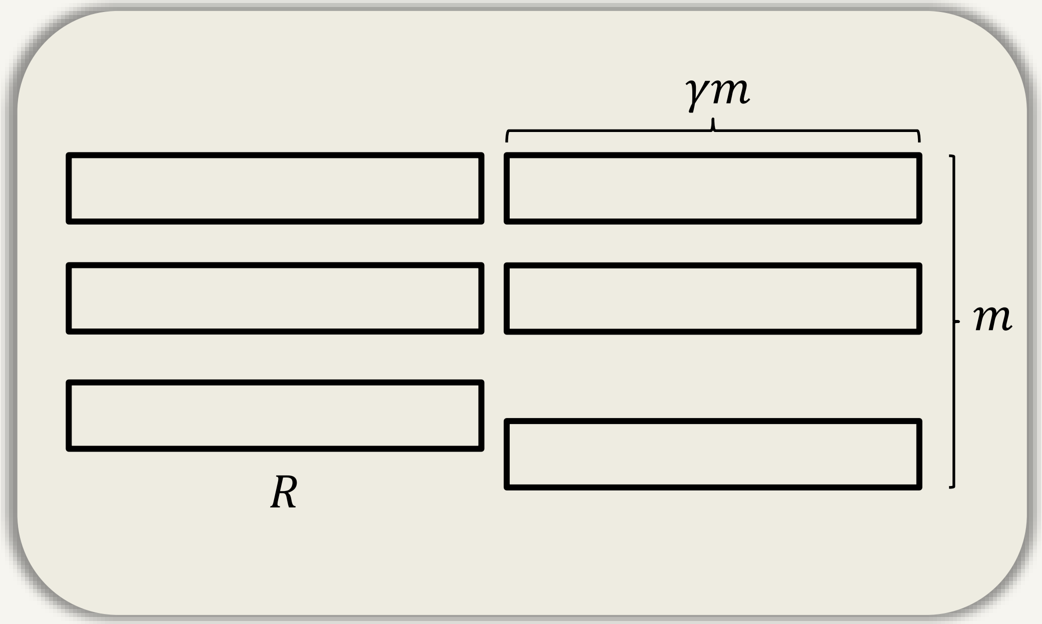


Our Solution: Repetition

[AHIV17] Repetition Version

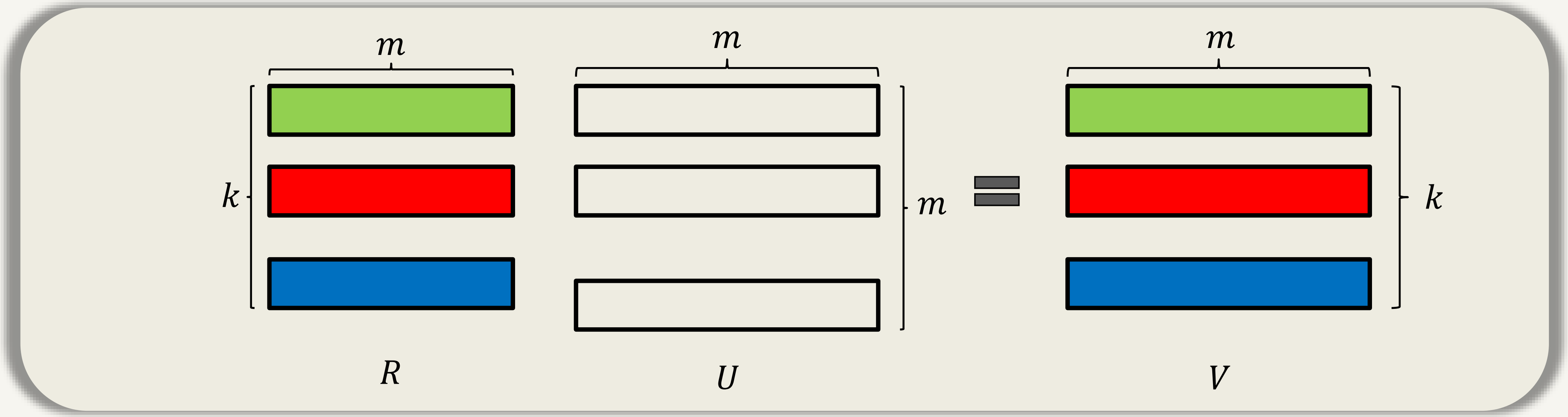
Fixed any $[l, n, d]$ code $C \subset R_2^l$ and a proximity parameter $e \in \{0, \cdots, \left\lfloor \frac{d-1}{3} \right\rfloor\}$. For a matrix $\widehat{U} \in R_2^{m \times l}$ with $d\left(\widehat{U}, C^m\right) > e$, and a matrix $R \in R_1^{k \times m}$ where each element of R is randomly chosen from R_1, let $W = RU$. Then:

$$\Pr\left[d\left(W, C^k\right) \leq e\right] \leq \frac{e+1}{p^{kr}}$$



where $d\left(W, C^k\right) := \frac{|\{j \in [l] | \exists i \; s.t. W_i[j] \neq c_i[j]\}|}{n}$ and c_i denote the closet codeword with row U_i in C.

Brakedown over Galois Rings: Testing Phase

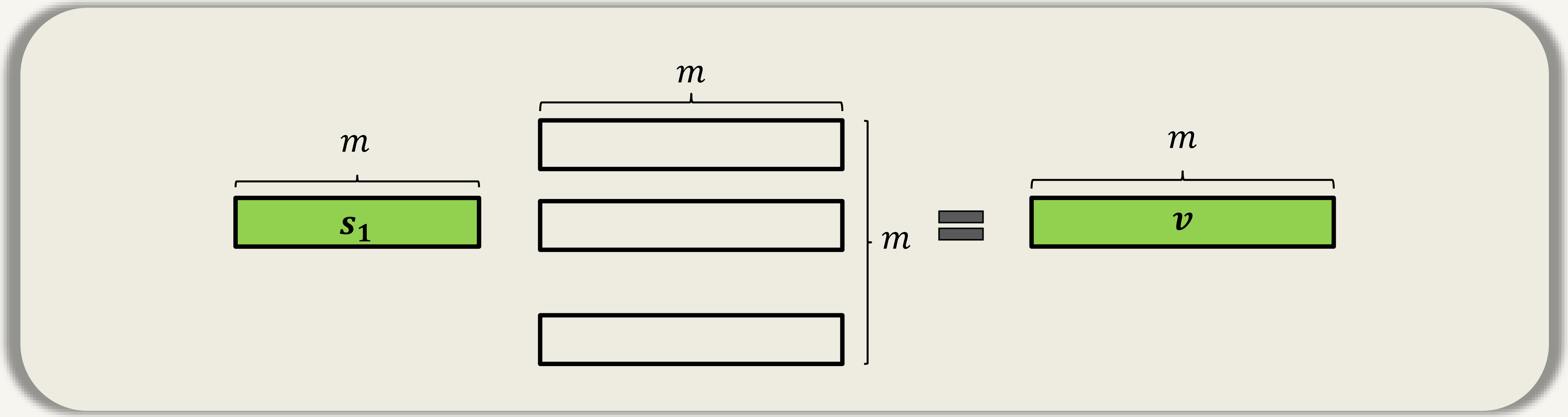


The procedure:

- V sends a random matrix $R \in \left(GR(p^s, r)\right)^{k\times m}$,
- P compute $V = RU$ and sends V,
- V picks $\Theta(\lambda)$ column indices and check $\forall\, i \in [0, k-1]$: $Enc(V_i)[j] = \sum_{s\in[0,m-1]} R_i[s] \cdot \widehat{U}_s[j]$

PS: The fundamental verification unit after encoding is the $GR(p^s, kr)$.

Brakedown over Galois Rings: Evaluation Phase



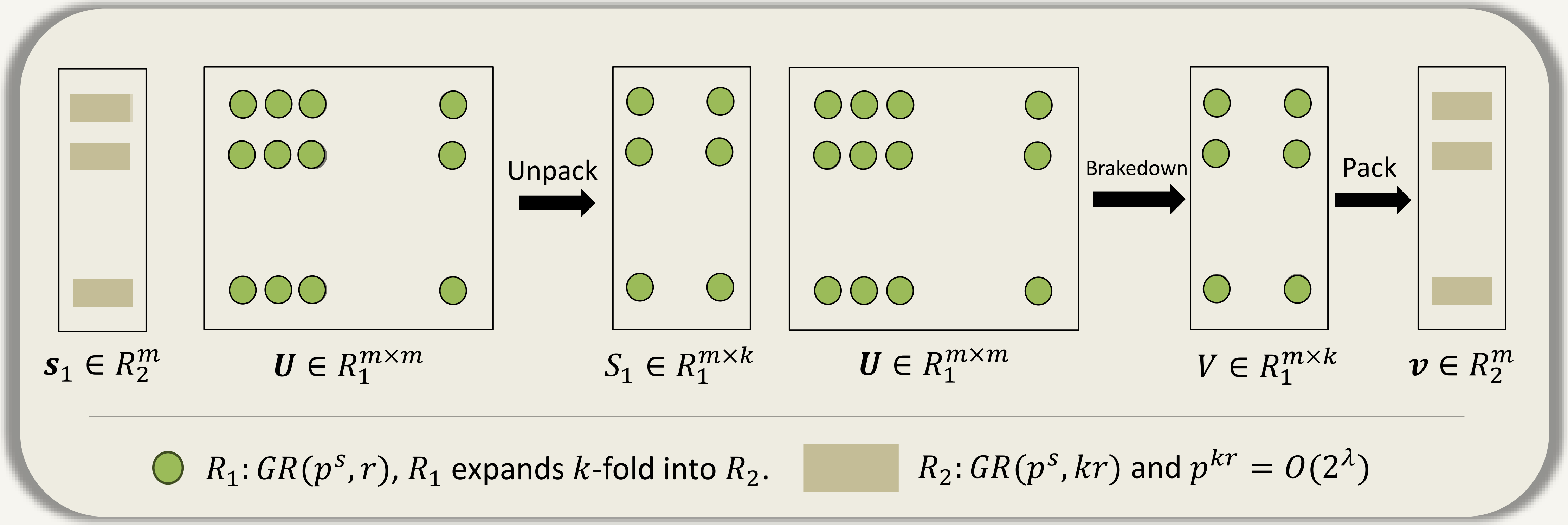
The procedure follows the testing phase exactly, except:

- The verifier substitutes the matrix R with the vector $\boldsymbol{s}_1^\top$, and
- Computes the evaluation $\boldsymbol{v}\top\boldsymbol{s}_2$ upon receiving vector v from the prover and successfully verifying it.

Efficient Computation in Galois Ring Extensions

Polynomial Commitments with Coefficients over R_1 and Evaluations over R_2

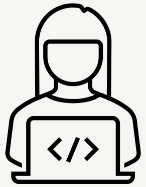
The arithmetic circuit C performs all computations over R_1, ensuring the polynomial f's coefficients lie in R_1, while the evaluation of f is opened over R_2 for verifiable safety.



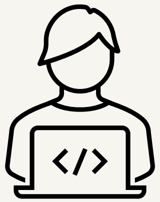
Sumcheck over Galois Rings

statement $\sum_{x_1}\sum_{x}\cdots\sum_{x} p(x_0,\cdots,x_{l-1}) = H$ Let $H_0 = H$, and $r_{<0} \coloneqq \emptyset$

Prover



Verifier



Round i, $i \in [0, l-2]$

- Compute $p_i(X_i) = \sum_{(x_{i+1},\cdots,x_{l-1})} p(r_0,\cdots,r_{i-1},X_i,x_{i+1},\cdots,x_{l-1})$

$p_i(X_i)$ →

← r

- Check if $p_i(0) + p_i(1) = H_i$
- Randomly chose r from the **exceptional set** from R
- Compute $H_{i+1} = p_i(r)$

(The size of exceptional set is $O(2^\lambda)$)

Round l

- Compute $p_{l-1}(X_{l-1}) = p(r_0,\cdots,r_{l-2},X_{l-1})$

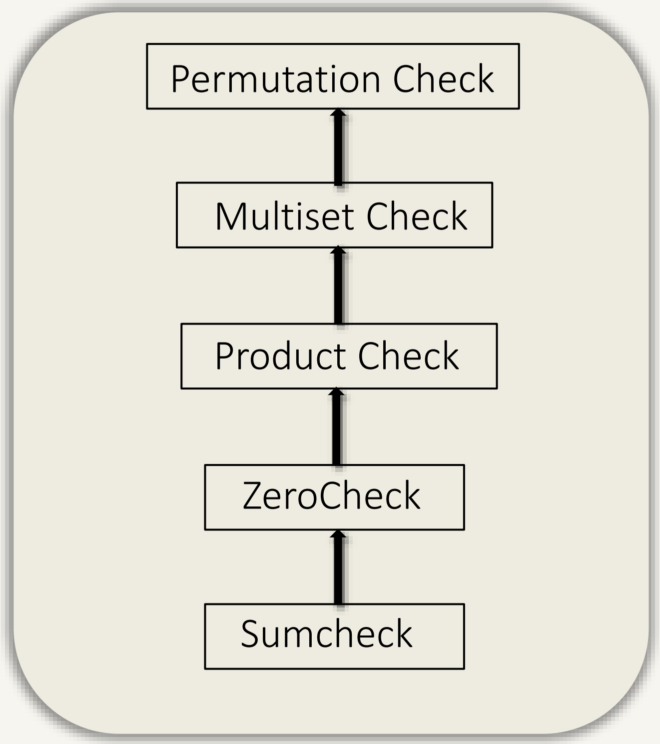
$p_{l-1}(X_{l-1})$ →

- Check if $p_{l-1}(0) + p_{l-1}(1) = H_{l-1}$
- Randomly chose r from the **exceptional set** from R
- Compute $p(r_0,\cdots,r_{l-1}) = p_{l-1}(r)$

HyperPlonk over Galois Rings

HyperPlonk **[CBBZ22]**

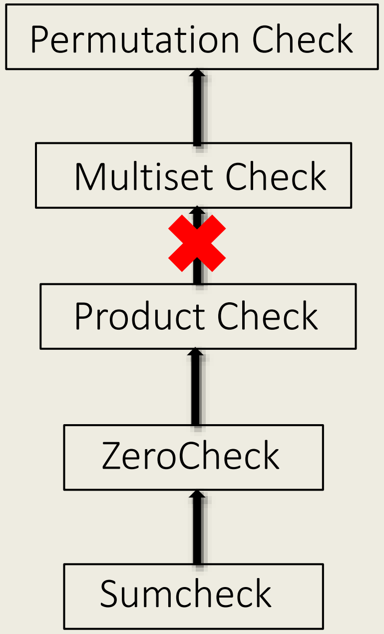
- **Gate Constraints** SumCheck
- **Permutation Constraints**



HyperPlonk over Galois Rings

HyperPlonk [CBBZ22]

- **Gate Constraints** SumCheck
- **Permutation Constraints**



Finite fields:

$S \subset \mathbb{F}_p, \phi: S \to \mathbb{F}[x], \quad \phi(S) = f_s: f_s(x) = \prod_{a \in S}(x - a)$

By the Schwartz-Zippel lemma, the function ϕ is guaranteed to be injective.

Galois rings:

The zero divisors interfere with the Schwartz-Zippel Lemma, causing ϕ not to be injective, e.g., under mod 8, both sets $\{3,5\}, \{1,7\}$ get the polynomial $x^2 - 1$.

HyperPlonk over Galois Rings

HyperPlonk [CBBZ22]

- **Gate Constraints** ZeroCheck
- **Permutation Constraints**

$$\boxed{\text{Permutation Check}} \uparrow \boxed{\text{ZeroCheck}} \uparrow \boxed{\text{Sumcheck}}$$

Permutation $\sigma: \{0,1\}^l \to \{0,1\}^l$, $\tilde{\sigma} = \left(\sigma_0(\boldsymbol{x}), \cdots, \sigma_{l-1}(\boldsymbol{x})\right)$, where σ_i denotes the i-th bit of the permutation.

$$f\left(\tilde{\sigma}(\boldsymbol{x})\right) - g(\boldsymbol{x}) = 0, \forall\, \boldsymbol{x} \in \{0,1\}^l$$

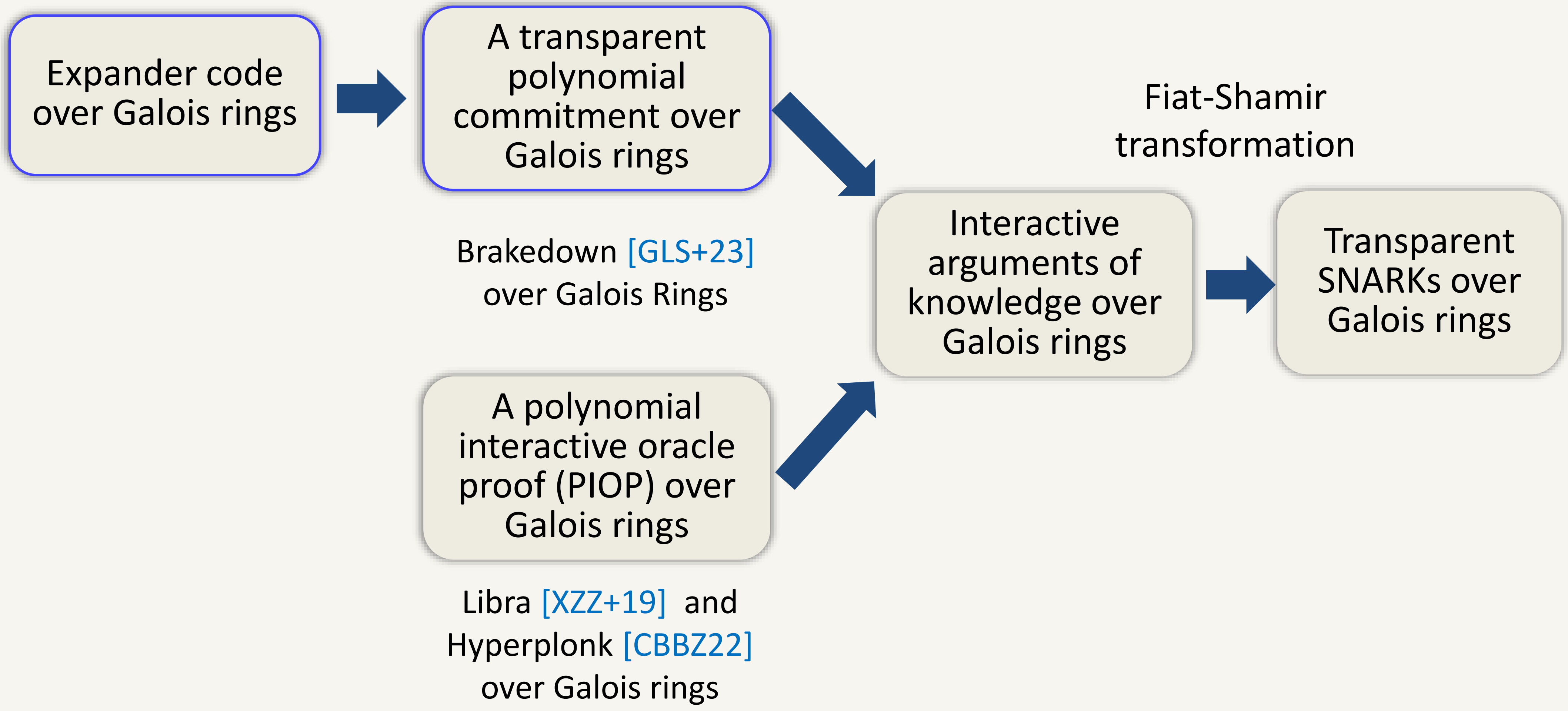
$$\Downarrow$$

$$\sum_{y \in \{0,1\}^l} \left(f(\boldsymbol{y}) \cdot eq(\tilde{\sigma}(\boldsymbol{x}), \boldsymbol{y}) - g(\boldsymbol{y}) \cdot eq(\boldsymbol{x}, \boldsymbol{y})\right) = 0\,, \forall \boldsymbol{x} \in \{0,1\}^l$$

$$\Downarrow$$

$$\sum_{x \in \{0,1\}^l} eq(\boldsymbol{x}, \boldsymbol{y}) \sum_{y \in \{0,1\}^l} \left(f(\boldsymbol{y}) \cdot eq(\tilde{\sigma}(\boldsymbol{x}), \boldsymbol{y}) - g(\boldsymbol{y}) \cdot eq(\boldsymbol{x}, \boldsymbol{y})\right) = 0$$

Transparent SNARK over Galois Rings



Thank you for your attention

Reference

- [THA22] Justin Thaler. Proofs, arguments, and zero-knowledge. Found. Trends Priv. Secur., 4(2-4):117–660, 2022. url:https://people.cs.georgetown.edu/jthaler/ProofsArgsAndZK.html
- [GNSV23] Chaya Ganesh, Anca Nitulescu, and Eduardo Soria-Vazquez. Rinocchio: Snarks for ring arithmetic. Journal of Cryptology, 36(4):41, 2023.
- [GLS+23] Alexander Golovnev, Jonathan Lee, Srinath T. V. Setty, Justin Thaler, and Riad S. Wahby. Brakedown: linear-time and field-agnostic SNARKs for R1CS. In Helena Handschuh and Anna Lysyanskaya, editors, CRYPTO 2023, Part II, volume 14082 of LNCS, pages 193–226. Springer, Cham, August 2023. doi: 10.1007/978-3-031-38545-2_7.
- [DP23] Benjamin E. Diamond and Jim Posen. Succinct arguments over towers of binary fields. Cryptology ePrint Archive, Paper 2023/1784, 2023. URL: https://eprint.iacr.org/2023/1784.

Reference

- [XZZ+19] Tiancheng Xie, Jiaheng Zhang, Yupeng Zhang, Charalampos Papamanthou, and Dawn Song. Libra: Succinct zero-knowledge proofs with optimal prover computation. In Alexandra Boldyreva and Daniele Micciancio, editors, Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part III, volume 11694 of Lecture Notes in Computer Science, pages 733–764. Springer, 2019.
- [CBBZ22] Binyi Chen, Benedikt Bünz, Dan Boneh, and Zhenfei Zhang. Hyperplonk: Plonk with linear-time prover and high-degree custom gates. Cryptology ePrint Archive, Paper 2022/1355, 2022. URL: https://eprint.iacr.org/2022/1355.
- [AHIV17] Scott Ames, Carmit Hazay, Yuval Ishai, and Muthuramakrishnan Venkitasubramaniam. Ligero: Lightweight sublinear arguments without a trusted setup. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017, pages 2087–2104. ACM, 2017.