Sub-Linear Lattice-Based Zero-Knowledge Arguments for Arithmetic Circuits

Bar-Ilan University

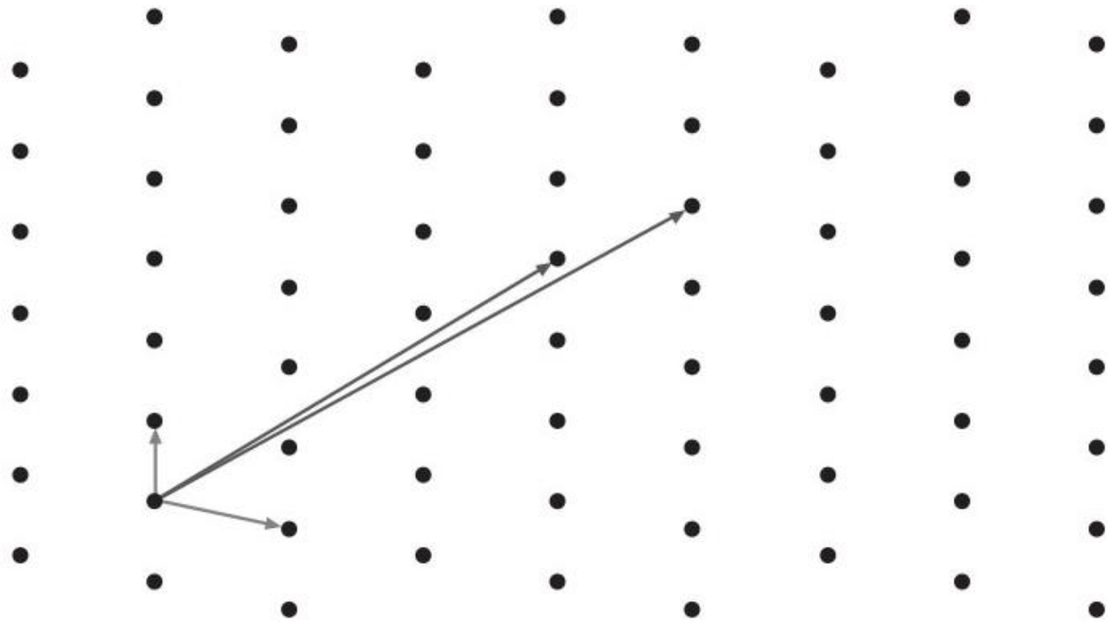
UCL IBM Research

Carsten Baum, <u>Jonathan Bootle</u>, Andrea Cerulli,
Rafael del Pino, Jens Groth and Vadim Lyubashevsky

Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits

An n-dimensional lattice $\mathcal{L}$ is

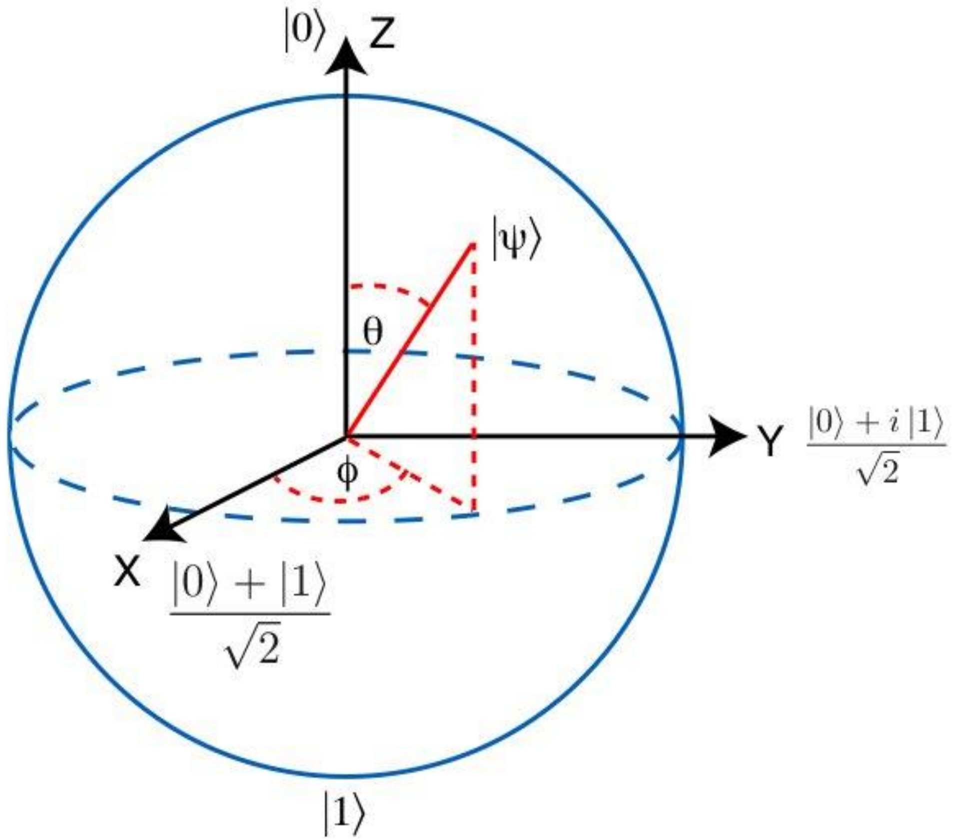
- A discrete additive subgroup of $\mathbb{R}^n$
- Generated by a basis $\mathcal{B} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$
- $\mathcal{L} = \sum_{i=1}^{n}(\mathbb{Z} \cdot \boldsymbol{b}_i)$



Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits

Why lattices?

- Quantum-resistant hard problems
- Worst-to-average case reductions
- Efficient operations



Lattice-Based
Zero-Knowledge Arguments for Arithmetic Circuits

Short Integer Solution (SIS) Problem

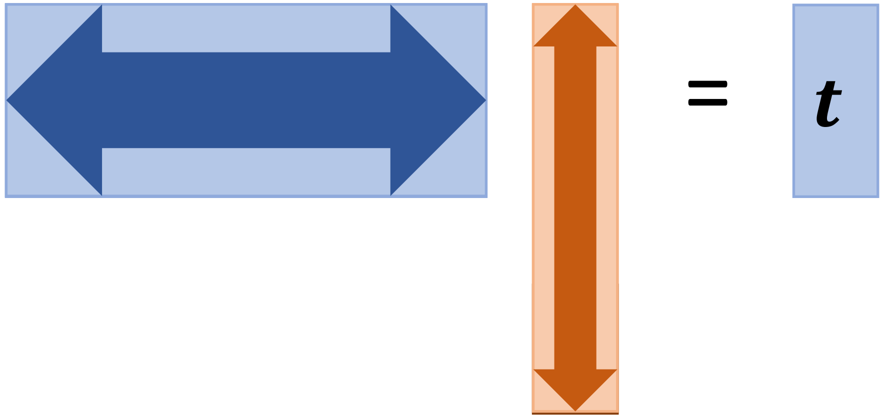
- Input: Random matrix $\mathrm{A} \in Z_q^{n\times m}$
- Goal: Find non-trivial $\boldsymbol{s} \in Z^m$ with $A\boldsymbol{s} = 0 \bmod q$ and $||s||_\infty < \beta$

$$A \, \boldsymbol{s} = \mathbf{0} \in Z_q^n$$

Lattice-Based
Zero-Knowledge Arguments for Arithmetic Circuits

Commitment/hashing from SIS:

- Binding/collision resistant by SIS
- Hiding by Leftover Hash Lemma
- Homomorphic
- Compressing

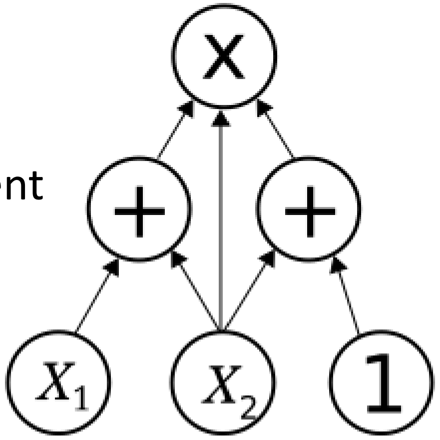


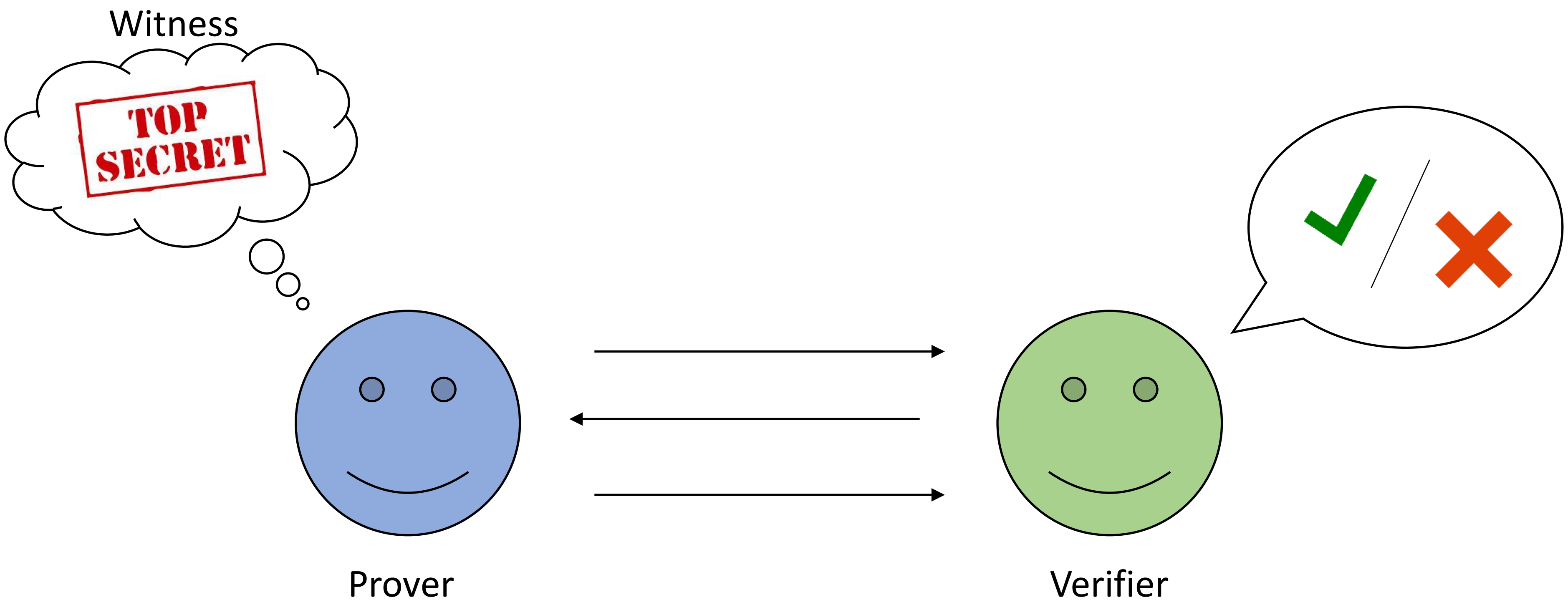
Lattice-Based

Zero-Knowledge Arguments

for Arithmetic Circuits

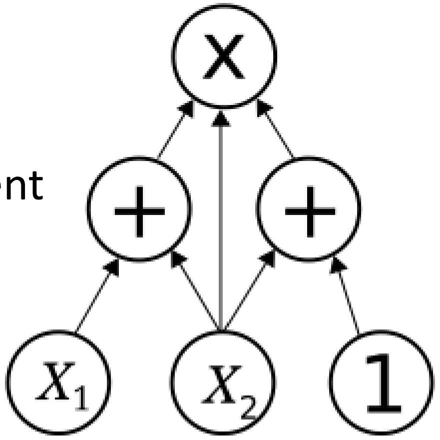
Statement



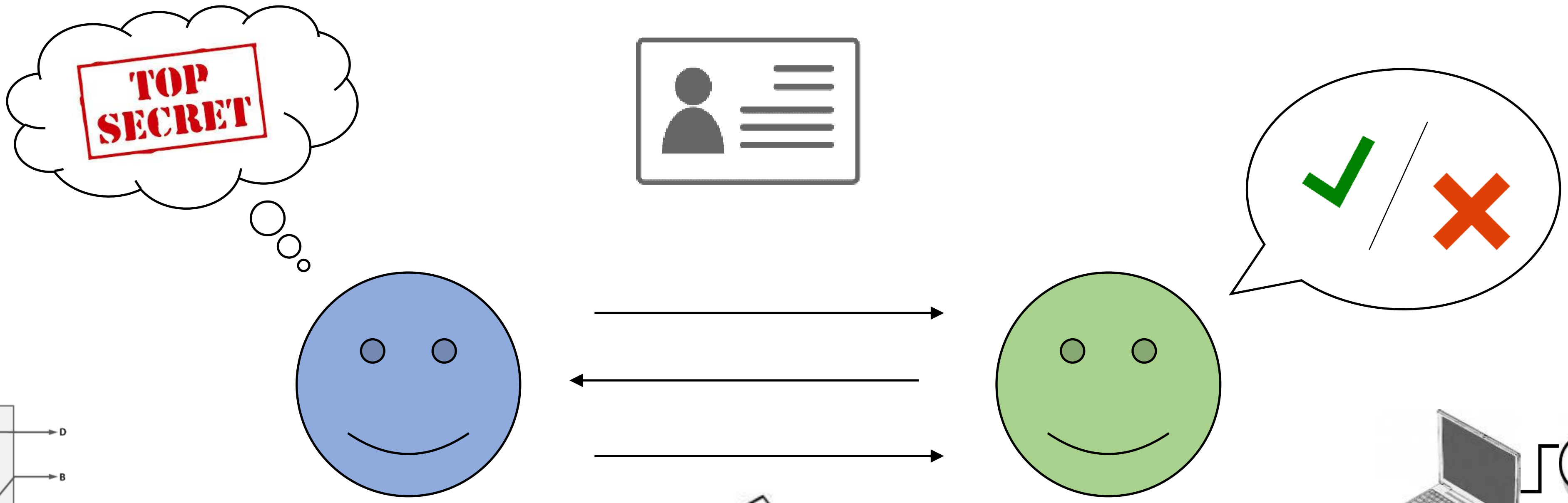


Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits

Statement

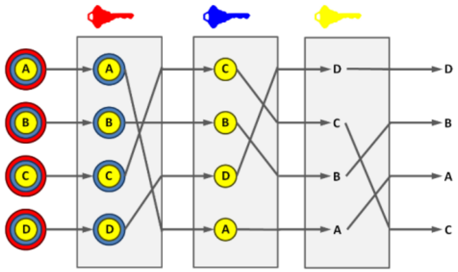


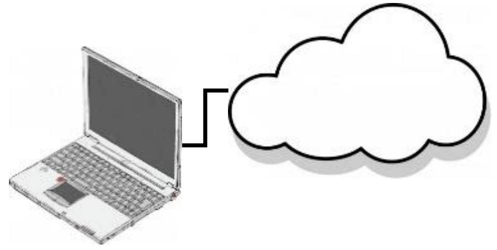
Witness



Prover

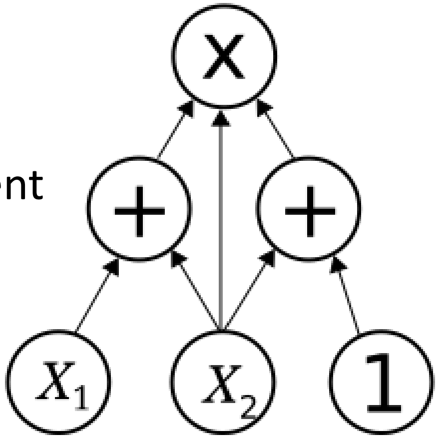
Verifier

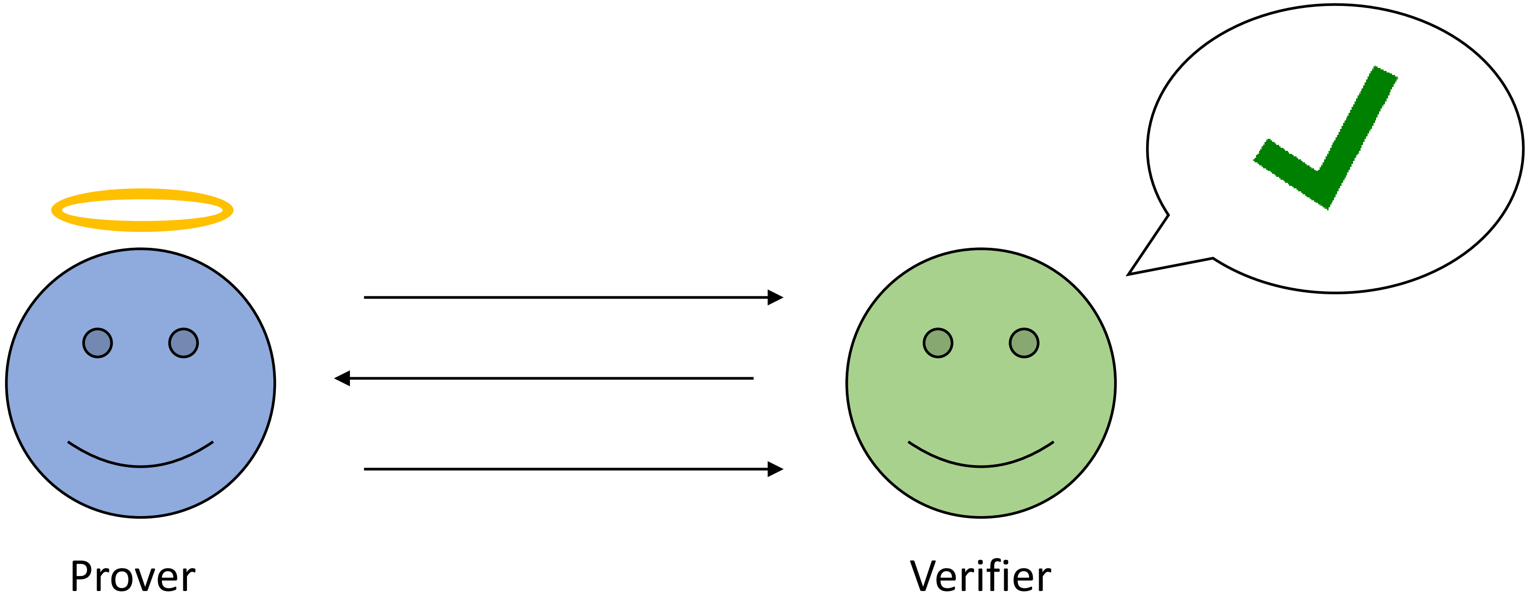




Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits

Statement

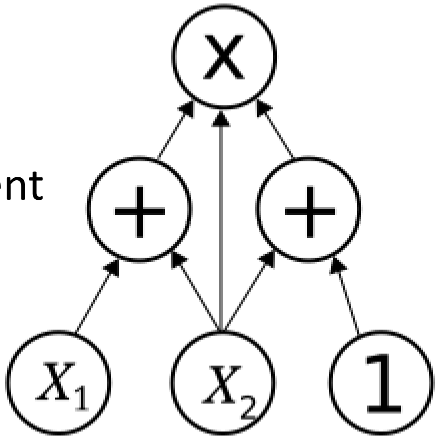


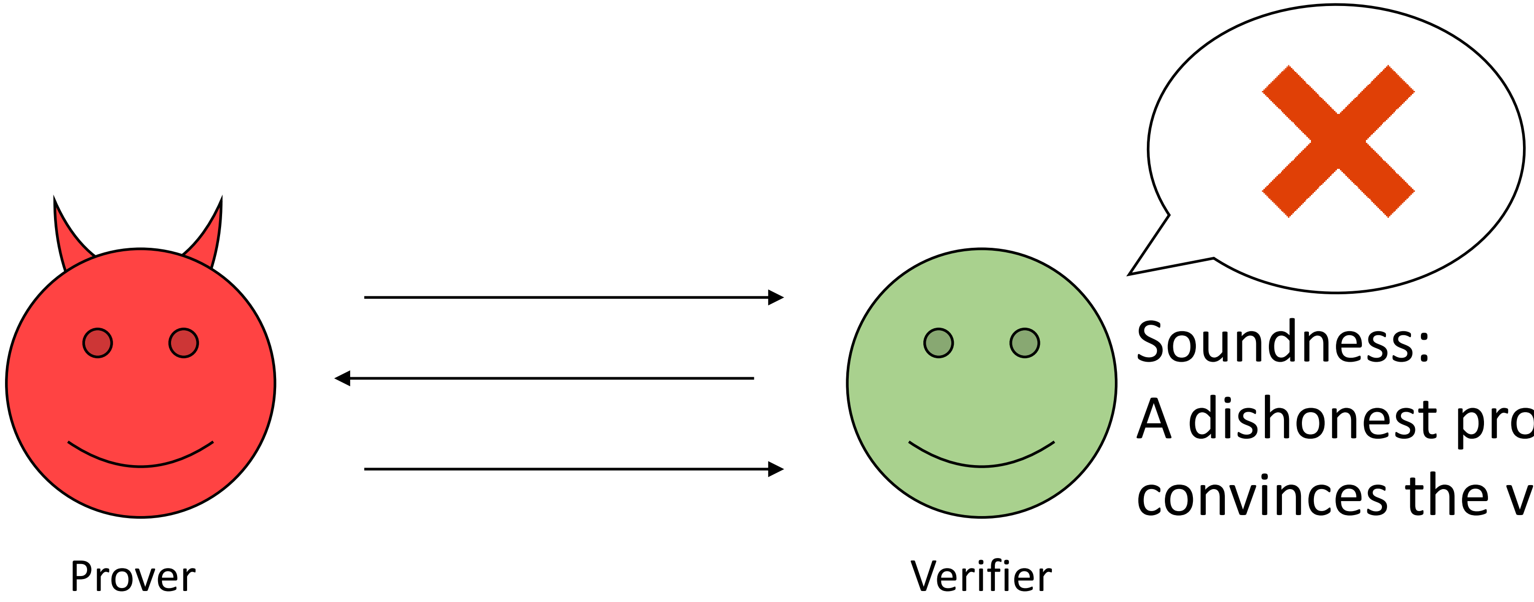


Completeness:
An honest prover convinces the verifier.

Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits

Statement





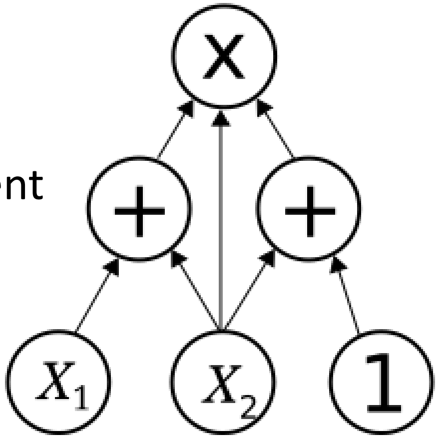
Completeness:
An honest prover convinces the verifier.

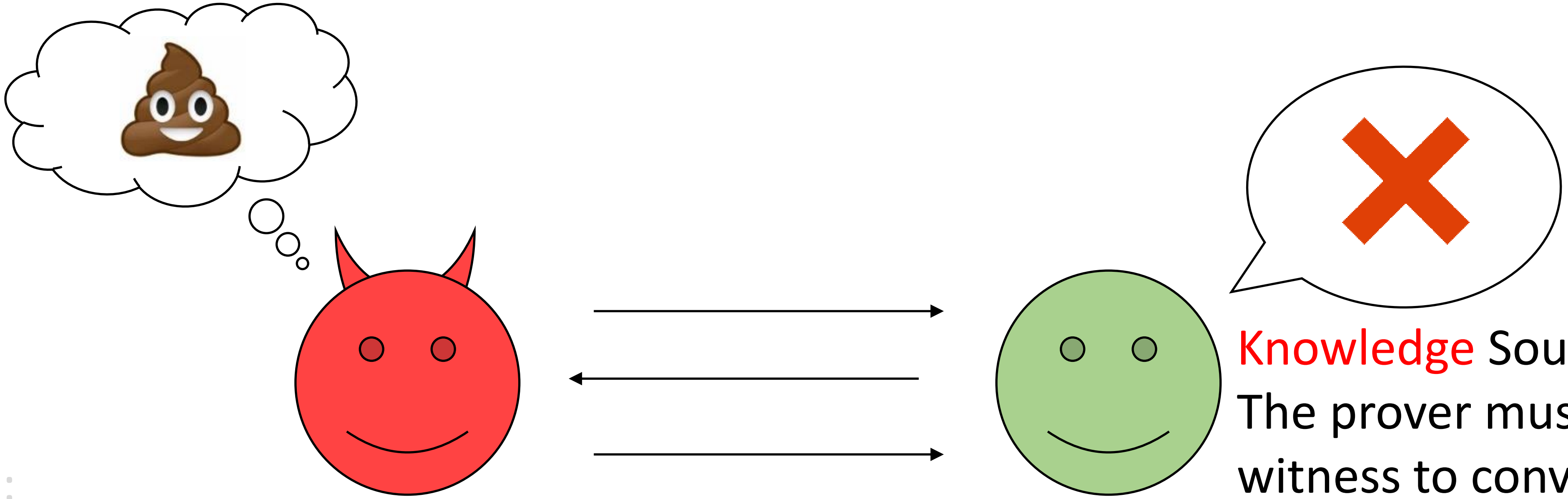
Soundness:
A dishonest prover never convinces the verifier.

Computational guarantee
-> argument

Lattice-Based Zero-Knowledge Arguments for Arithmetic Circuits

Statement





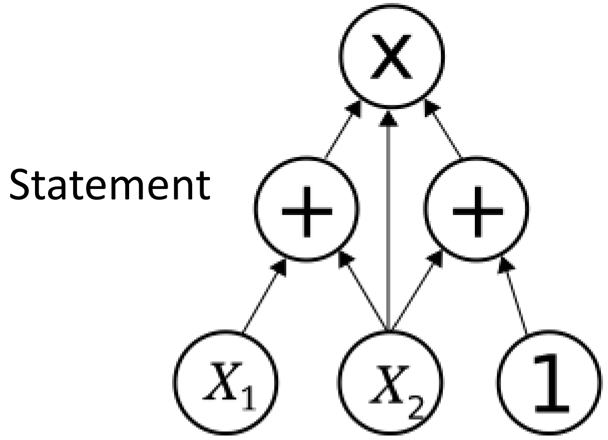
Prover

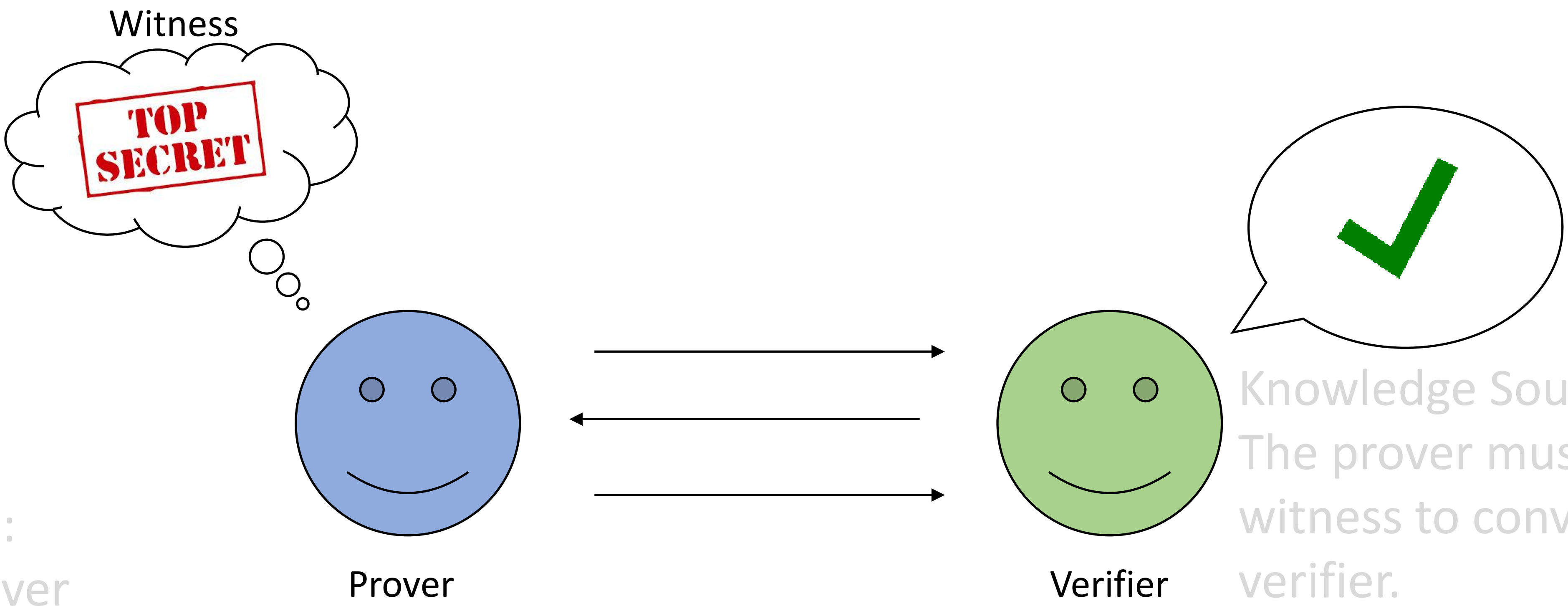
Verifier

Completeness:
An honest prover convinces the verifier.

Knowledge Soundness:
The prover must know a witness to convince the verifier.
-> Proof/argument of knowledge

Lattice-Based
Zero-Knowledge Arguments
for Arithmetic Circuits





Completeness:
An honest prover convinces the verifier.

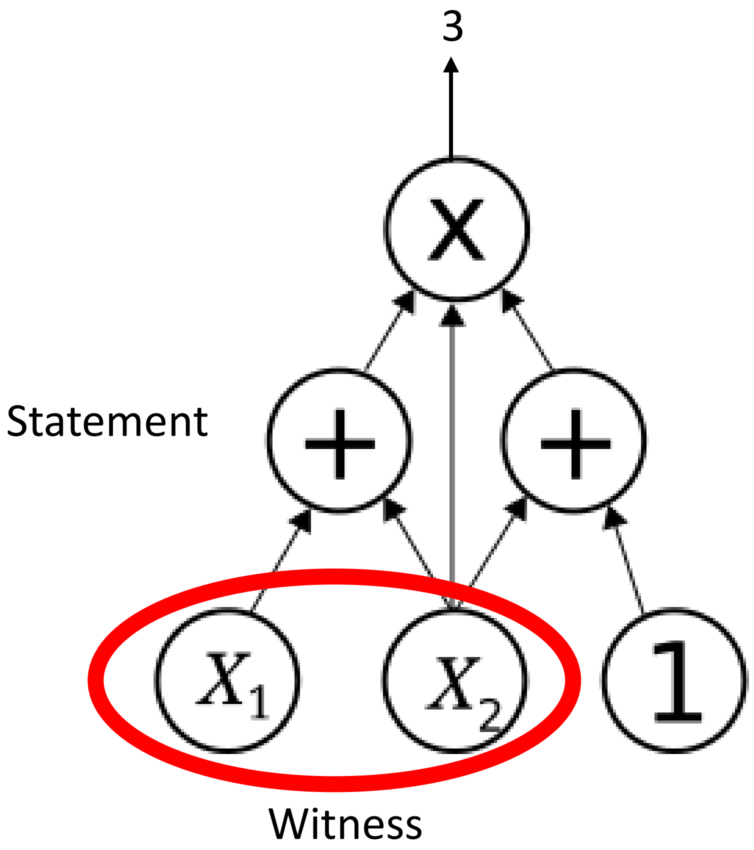
Knowledge Soundness:
The prover must know a witness to convince the verifier.
-> Proof/argument of knowledge

Zero-knowledge:
Nothing but the truth of the statement is revealed.

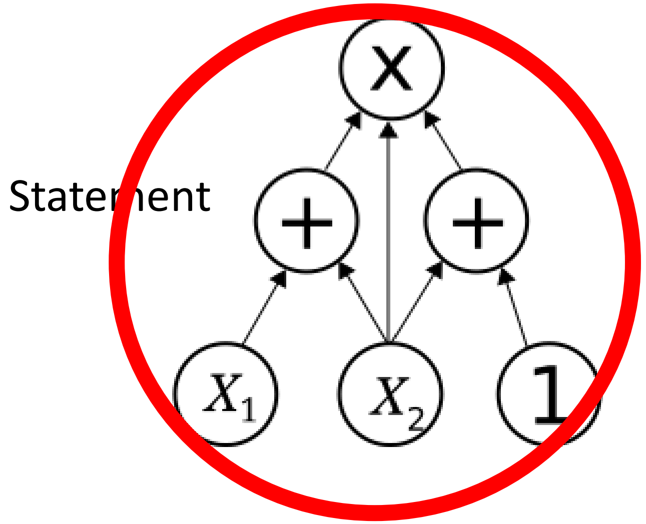
Lattice-Based Zero-Knowledge Arguments for Arithmetic Circuits

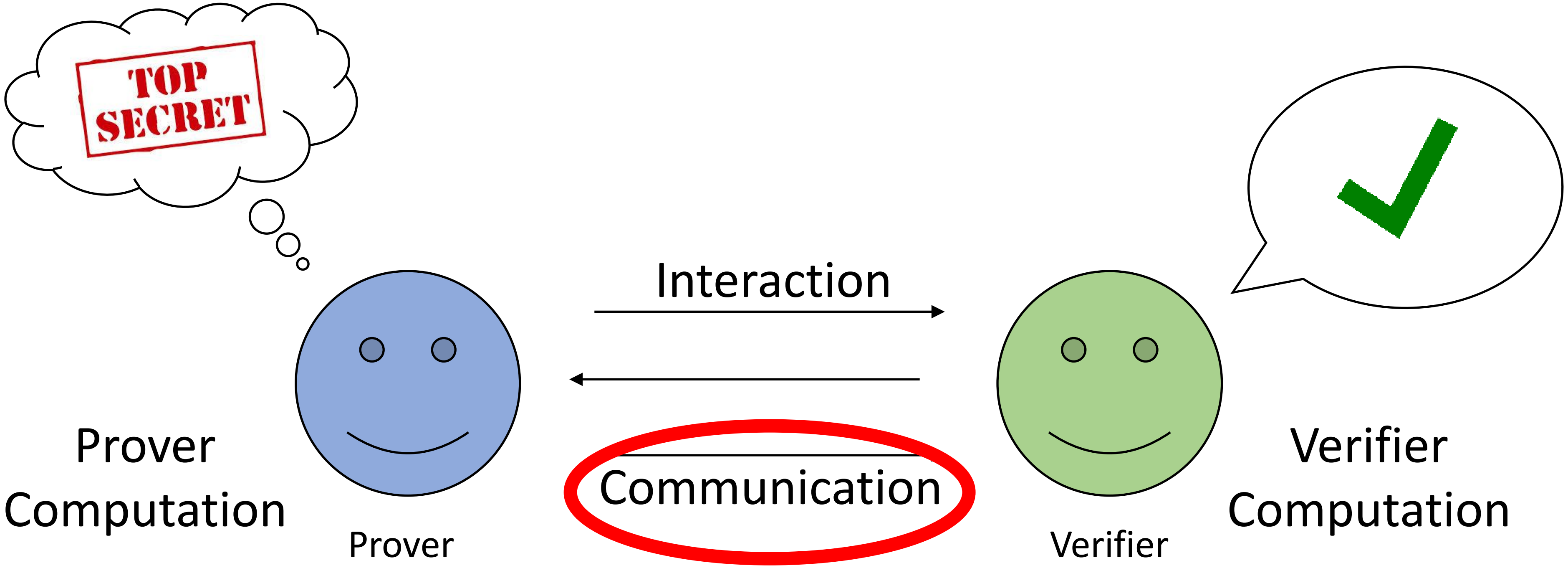
Why arithmetic circuits?

- C to circuit compilers
- Models cryptographic computations
- Witness existence? NP-Complete



Lattice-Based Zero-Knowledge Arguments for Arithmetic Circuits

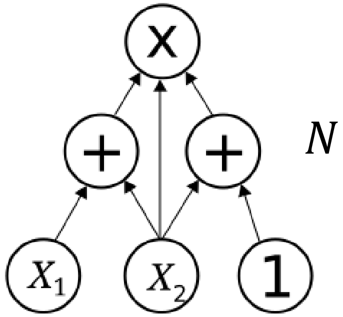




Cryptographic Assumption

Results Table

Expected # Moves	Communication	Prover Complexity	Verifier Complexity
$O(1)$	$O\left(\sqrt{N\lambda\log^3 N}\right)$	$O\left(N \log N\ (\log^2\lambda)\right)$	$O(N\log^3\lambda)$



N gates

Security parameter λ

Arithmetic Circuit Argument

The interesting parts

Extension Fields

Proof of Knowledge

Rejection Sampling

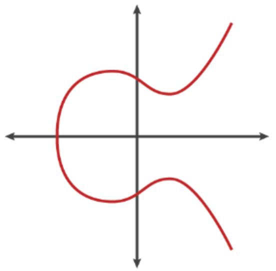
Arithmetic Circuits

Matrix Equations

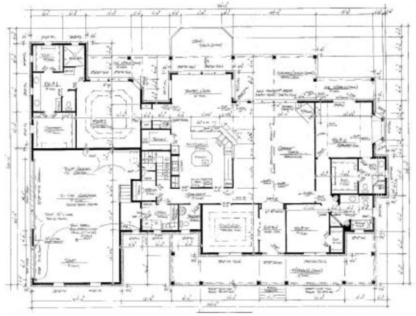
Polynomials

Commitments

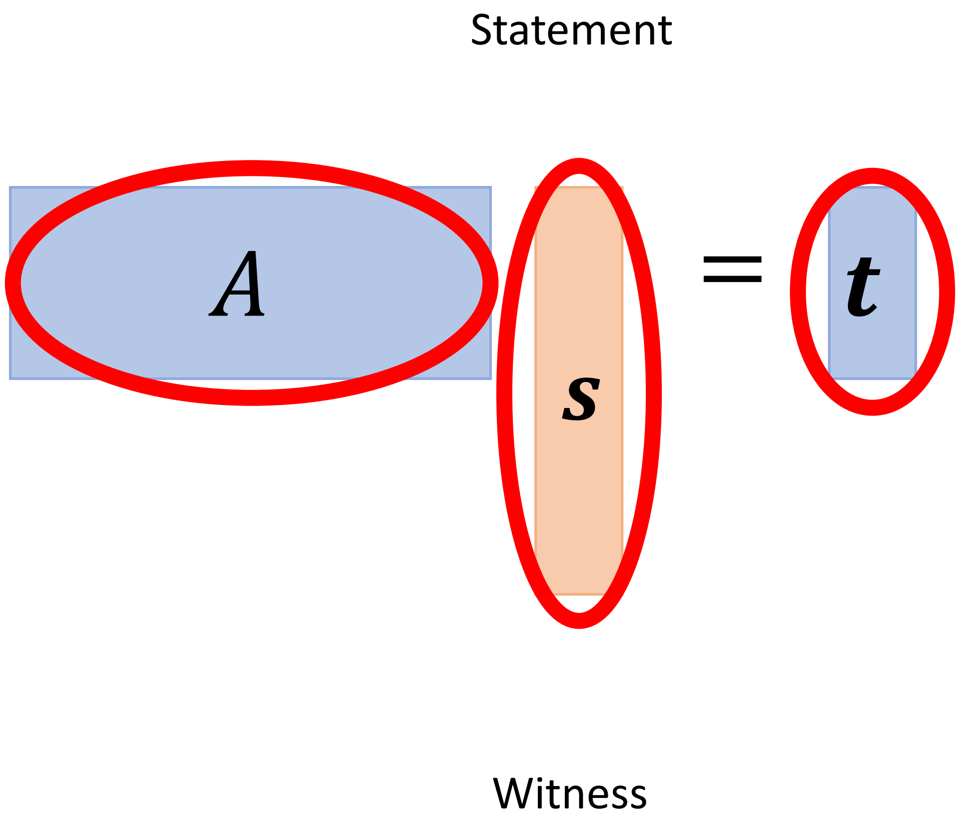
Protocol



Featured in prior works
DLOG Protocols
Information Theoretic Proofs



Proof of Knowledge



Proof of Knowledge

$$\underbrace{A\,\boldsymbol{s}_1 = \boldsymbol{t}_1 \qquad A\,\boldsymbol{s}_2 = \boldsymbol{t}_2 \qquad \cdots \qquad A\,\boldsymbol{s}_m = \boldsymbol{t}_m}_{m \approx \sqrt{N}}$$

$$\boldsymbol{s}_1 \;\Big]\approx \sqrt{N}$$

->Prover knows N small hashed integers

Proof of Knowledge

$$\underbrace{A\boldsymbol{s}_1 = \boldsymbol{t}_1 \qquad A\boldsymbol{s}_2 = \boldsymbol{t}_2 \qquad \cdots \qquad A\boldsymbol{s}_m = \boldsymbol{t}_m}_{\lambda \text{ preimages}}$$

Typical Proofs of Knowledge

Completeness: $A\boldsymbol{s} = \boldsymbol{t}$ $\qquad \|\boldsymbol{s}\|_\infty < \beta$

Knowledge Soundness: $A\boldsymbol{s} = 2\boldsymbol{t}$ $\qquad \|\boldsymbol{s}\|_\infty < K\beta$

None for us*

Soundness Slack

Simplistic Protocol

$$A\boldsymbol{s} = \boldsymbol{t}$$

P

V

$$A\boldsymbol{y} = \boldsymbol{w}$$

$$\boldsymbol{w}$$

$$c \in \{0,1\}$$

$$\boldsymbol{z} = \mathrm{c}\,\boldsymbol{s} + \boldsymbol{y}$$

Rejection Sampling

$$\boldsymbol{z}$$

Check: $||\boldsymbol{z}||_\infty < B$

$$A\boldsymbol{z} = \mathrm{c}\,\boldsymbol{t} + \boldsymbol{w}$$

Our Protocol

$$\boxed{\boldsymbol{z}} = \mathrm{c}\ \boxed{\boldsymbol{s}} + \boxed{\boldsymbol{y}} \qquad c \in \{0,1\}$$

Our Protocol

$$\boxed{\boldsymbol{z}} = \sum \boxed{\boldsymbol{s}_i}\, \mathrm{c_i} + \boxed{\boldsymbol{y}} \qquad c_i \in \{0,1\}$$

Our Protocol

$$\boldsymbol{z} = \boldsymbol{s}_1 + \boldsymbol{s}_2\, \mathrm{c}_2 \ldots + \boldsymbol{s}_m c_m + \boldsymbol{y}$$

$$\boldsymbol{z}' = \boldsymbol{s}_2\, \mathrm{c}_2 \ldots + \boldsymbol{s}_m c_m + \boldsymbol{y}$$

Extraction guaranteed by ‘heavy rows’ averaging argument

Our Protocol

$$\boxed{\mathbf{z}} = \sum \boxed{\mathbf{s}_i}\, c_i + \boxed{\mathbf{y}} \qquad c_i \in \{0,1\}$$

Parallel repetition for negligible soundness error

Our Protocol

$$\boxed{\boldsymbol{z}} = \sum \boxed{\boldsymbol{s}_i}\, \boldsymbol{c}_i^T + \boxed{\boldsymbol{y}} \qquad \boldsymbol{c}_i^T \in \{0,1\}^{O(\lambda)}$$

Parallel repetition for negligible soundness error

Proof-of-Knowledge Performance

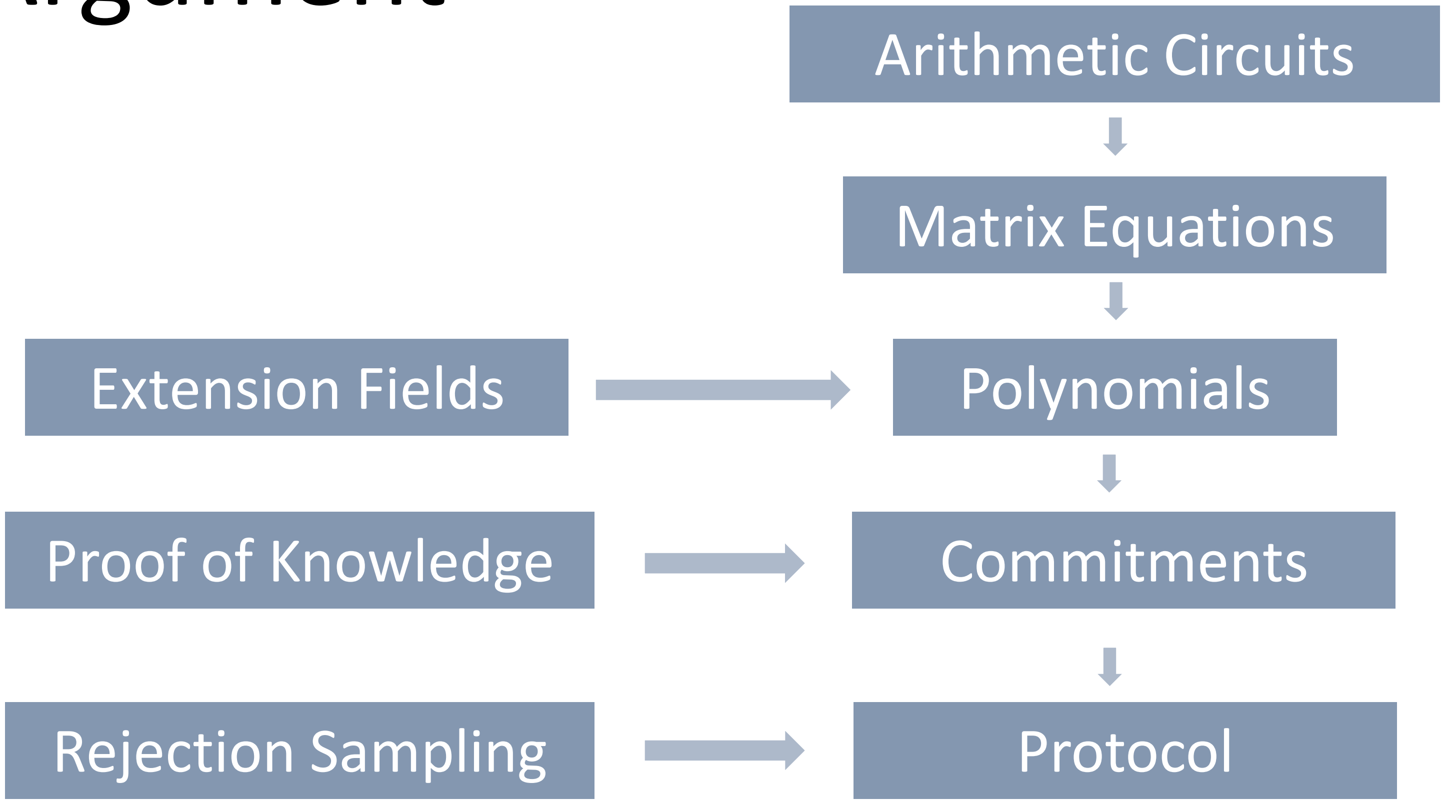
Expected # Moves	Communication	Prover Complexity	Verifier Complexity
$O(1)$	$O\left(\sqrt{N\lambda \log^3 N}\right)$	$O(N \log^3 \lambda)$	$O(\sqrt{N} \log^3 \lambda)$

$$\boldsymbol{A}\,\boldsymbol{s} = \boldsymbol{t}$$

N hashed integers

Security parameter λ

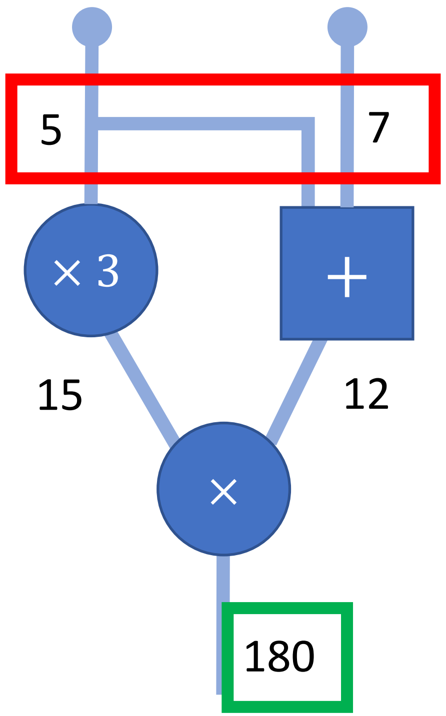
Arithmetic Circuit Argument



High Level Structure

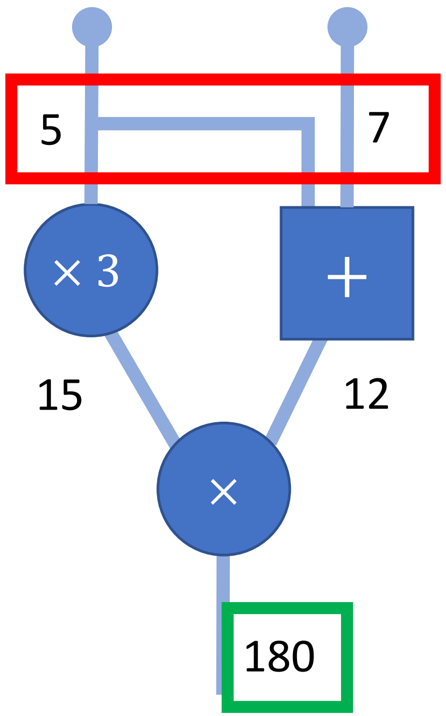
L		R		O
3	∘	5	=	15
15		12		180

5	+	7	=	12

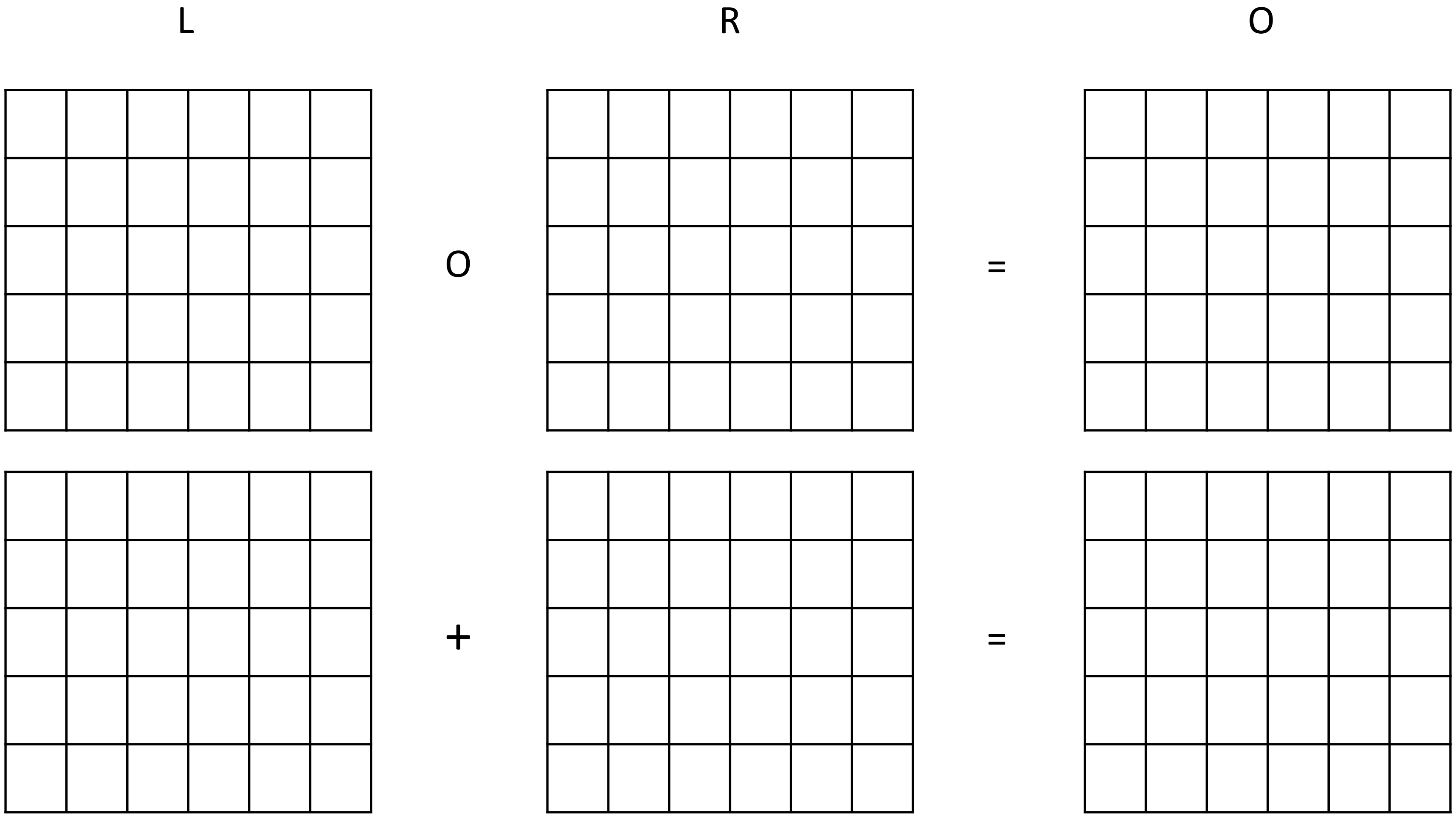


High Level Structure

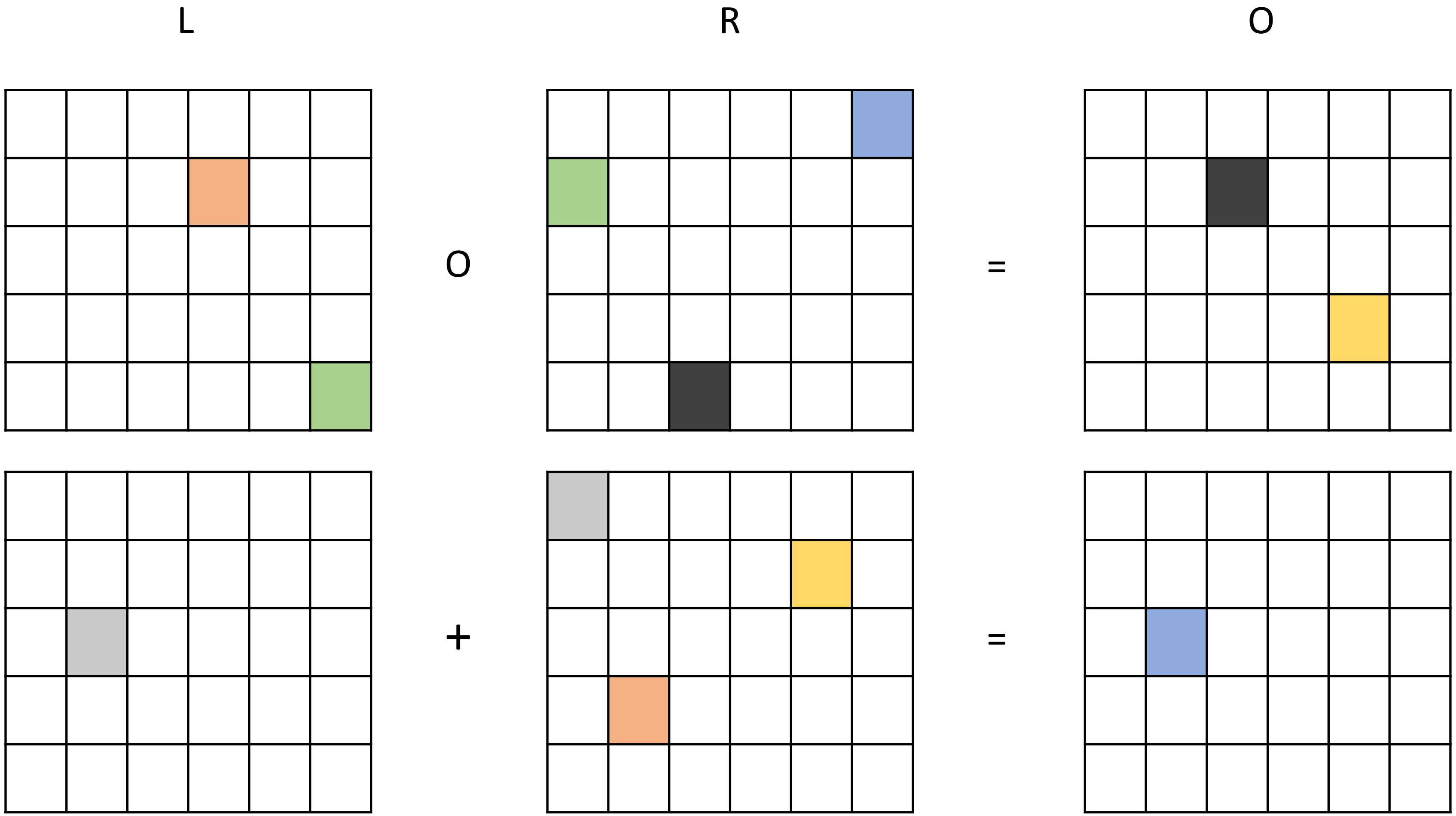
$$\begin{array}{ccccc} L & & R & & O \\ \begin{bmatrix}3\\15\end{bmatrix} & \circ & \begin{bmatrix}5\\12\end{bmatrix} & = & \begin{bmatrix}15\\180\end{bmatrix} \\ \\ 5 & + & 7 & = & 12 \end{array}$$



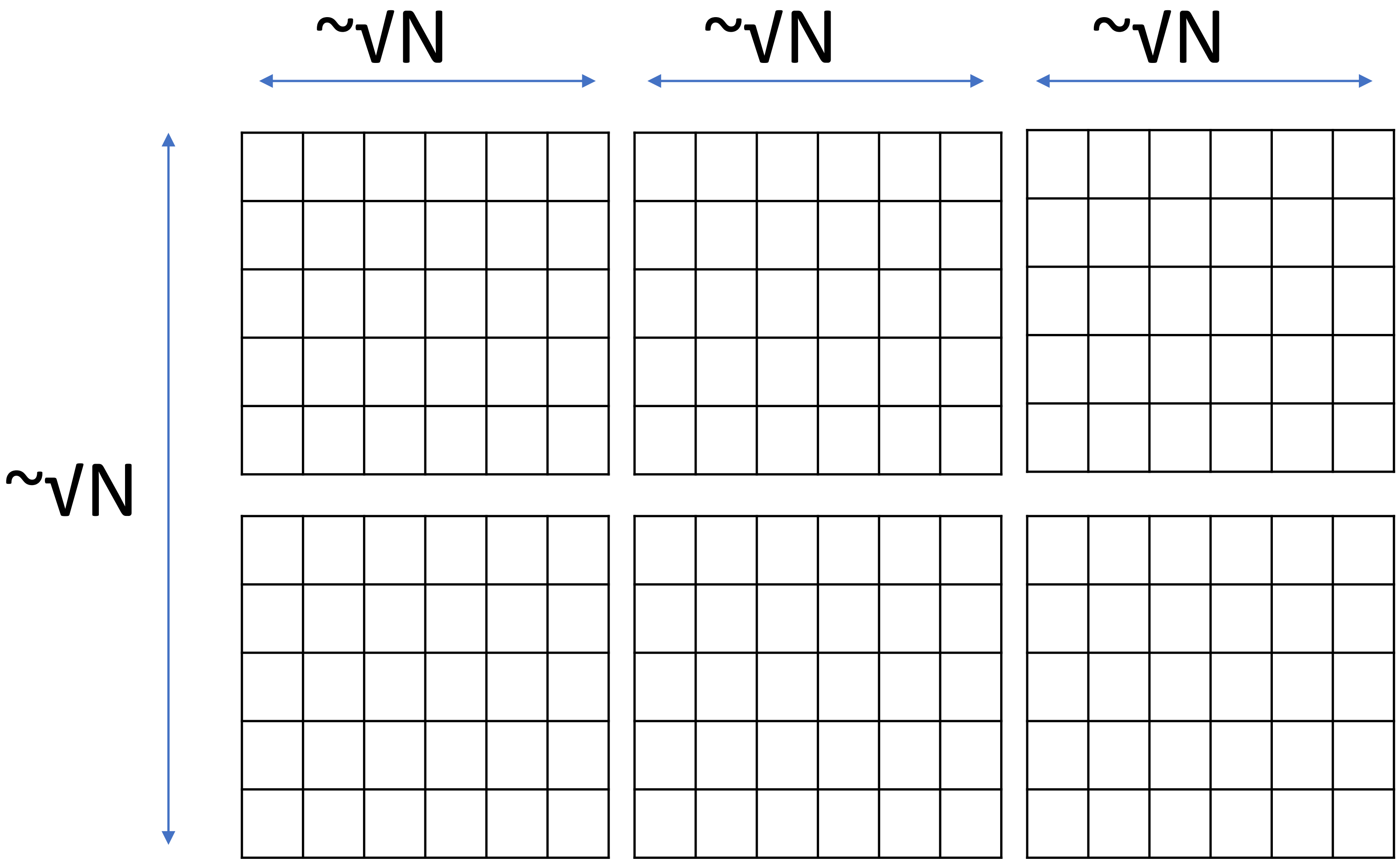
High Level Structure



High Level Structure



Matrix Dimensions

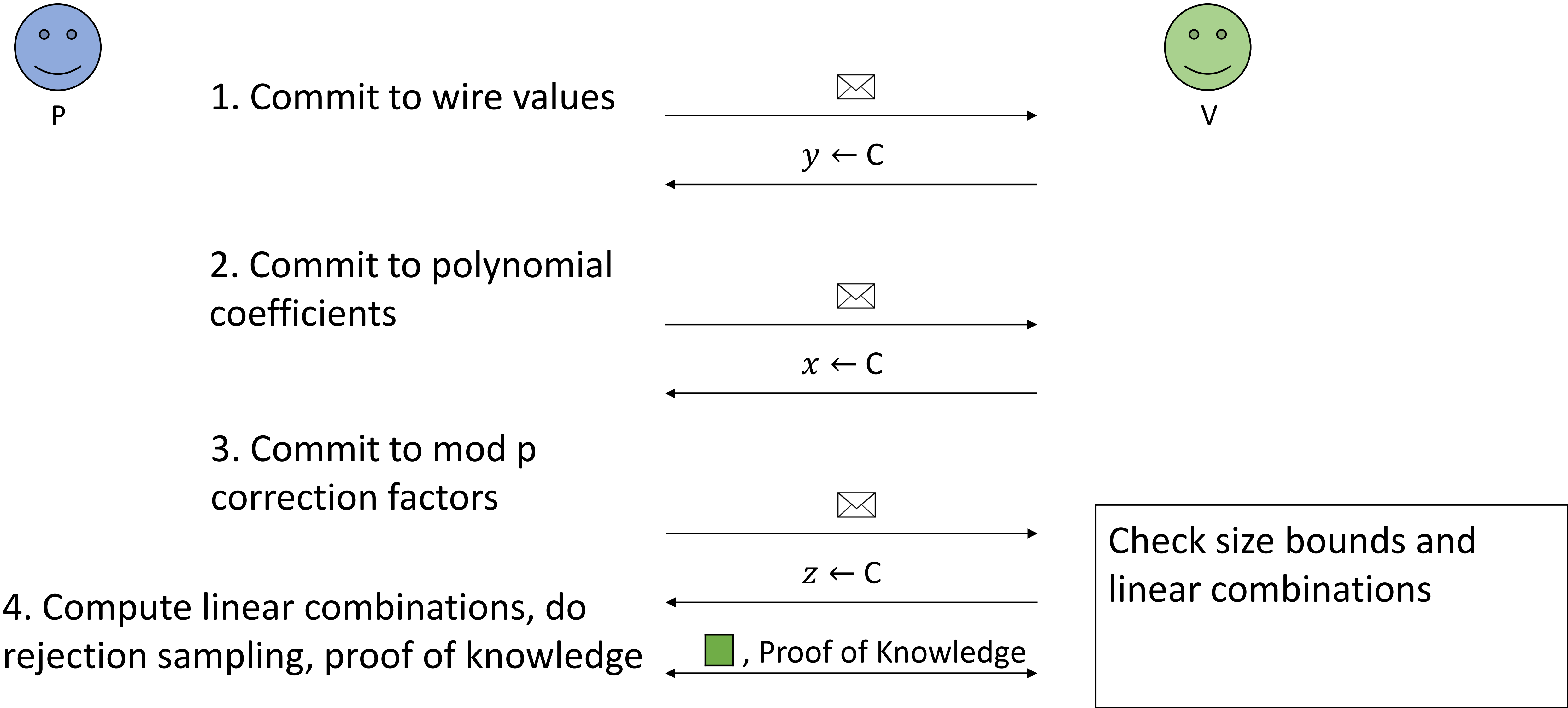


Paradigm from Previous Arguments

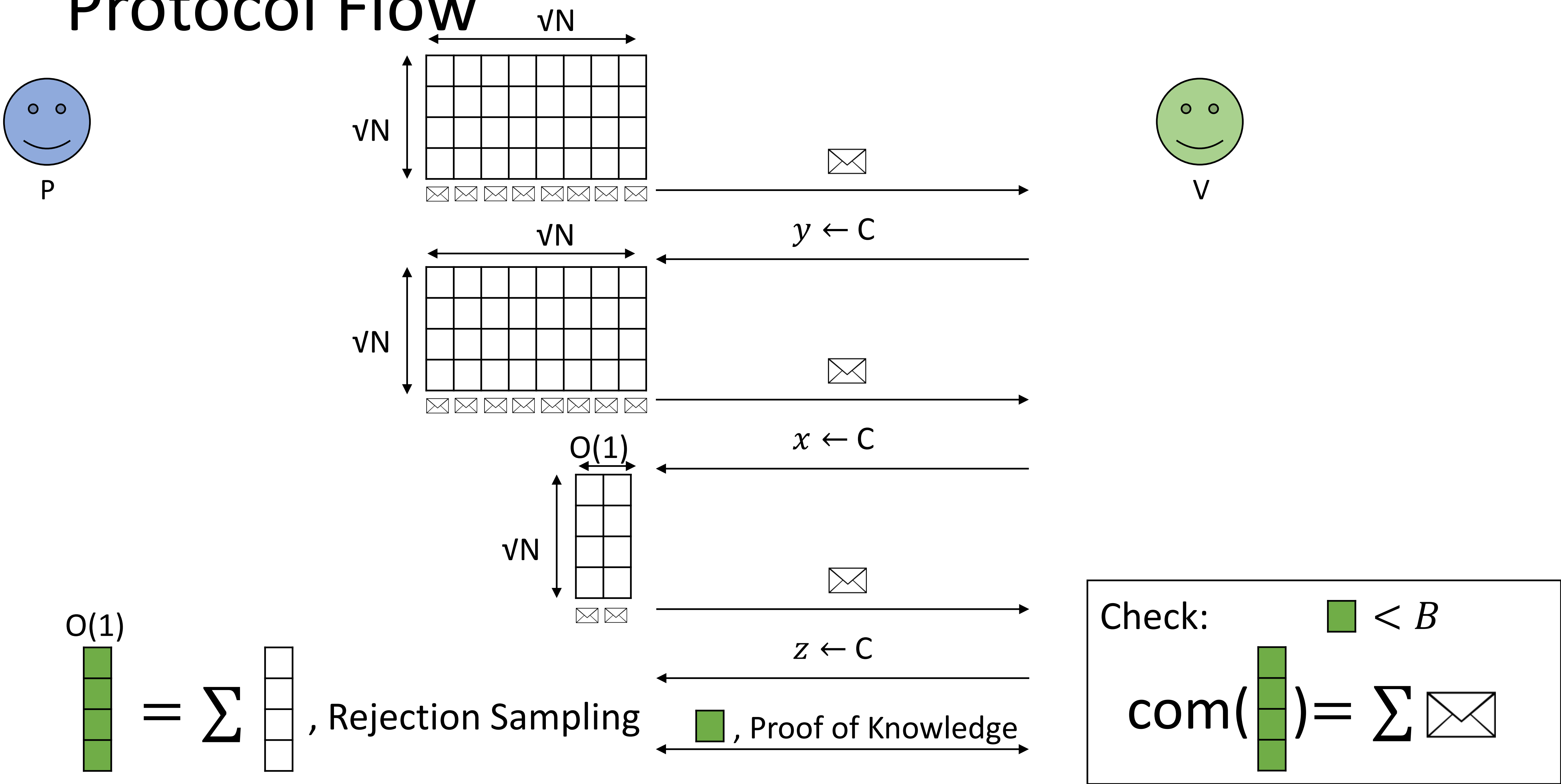
- Commit to vectors ([G09], [S09],[BCGGHJ17])
- Random challenge x
- Prover opens linear combinations
- Verifier conducts polynomial identity test
- AC-SAT in coefficients

$3x$	2	6	6	2	0	1	9	2	7	4
$+4x^2$	5	3	7	2	8	3	6	1	6	9
$+8x^3$	5	7	6	7	1	4	2	6	8	3
$+7x^4$	6	3	7	2	7	5	3	2	4	7
$=$	5	2	8	7	3	1	0	4	7	3

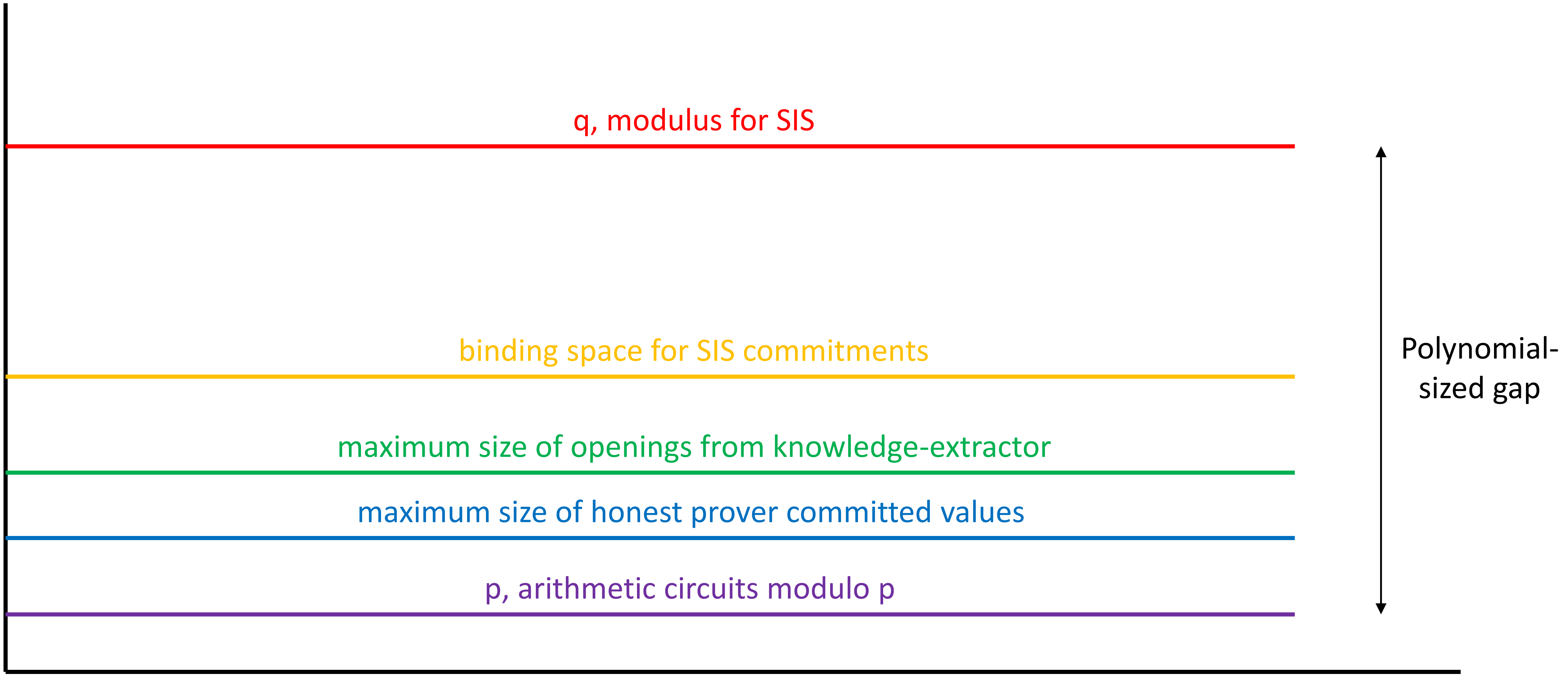
Protocol Flow



Protocol Flow



Parameter Choice



Small Modulus Issues

- Schwarz-Zippel Lemma over Z_p
- Multivariate polynomial $p(x_1, x_2, \dots, x_n)$, total degree d
- Choose random evaluation points $r_1, r_2, \dots, r_n$

$$\underbrace{\Pr[p(r_1, r_2, \dots, r_n) = 0] \le \frac{d}{p}}$$

Not negligible!

- DLOG: $p \approx 2^\lambda$
- SIS: modulus usually $\mathrm{poly}(\lambda)$

Extension Fields

- $GF(p^k)$ a vector space over $GF(p)$
- $GF(p^k)$-multiplications are linear maps on $GF(p)$
- Homomorphic commitments

$$\underbrace{\Pr[p(r_1, r_2, \ldots, r_n) = 0] \le \frac{d}{p}}_{\text{Not negligible!}}$$

Extension Fields

- $GF(p^k)$ a vector space over $GF(p)$
- $GF(p^k)$-multiplications are linear maps on $GF(p)$
- Homomorphic commitments
- View k commitments as a homomorphic commitment to a $GF(p^k)$ element!
- Run protocol over $GF(p^k)$ (extends [CDK14])

$$\underbrace{\Pr[p(r_1, r_2, \ldots, r_n) = 0] \le \frac{d}{p^k}}_{\text{Negligible!}}$$

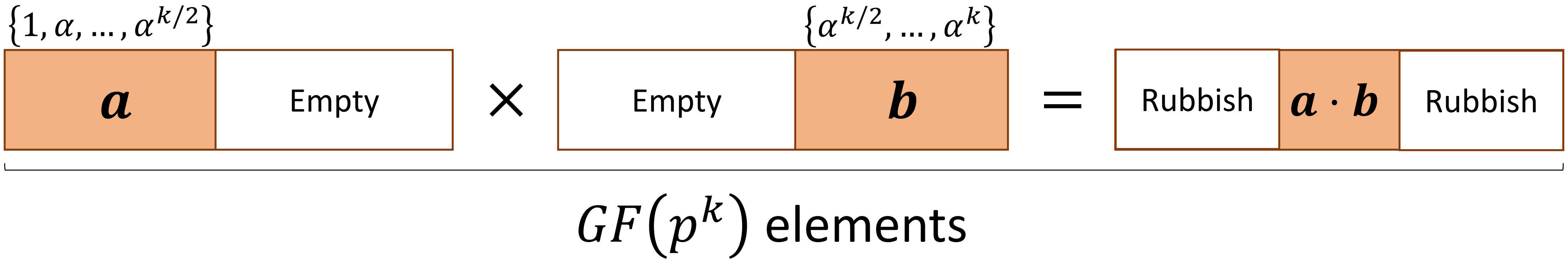
Embedding Base Field Operations

- $GF(p^k) = GF(p)[\alpha]$ basis: $\{1, \alpha, \alpha^2, \dots, \alpha^k\}$

$$\underbrace{\boxed{\quad a \quad} \times \boxed{\quad b \quad} = \boxed{\quad c \quad}}_{GF(p^k)\ \text{elements}}$$

Embedding Base Field Operations

- $GF(p^k) = GF(p)[\alpha]$ basis: $\{1, \alpha, \alpha^2, \dots, \alpha^k\}$



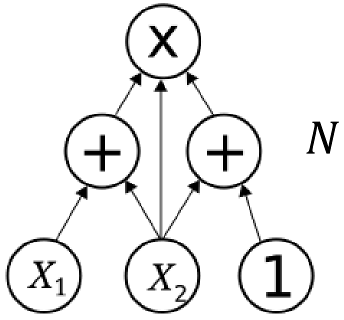
Future Work:
Can we match the $O(\log N)$ proof sizes of DLOG protocols?

Thanks!

Expected # Moves	Communication	Prover Complexity	Verifier Complexity
$O(1)$	$O\left(\sqrt{N\lambda\log^3 N}\right)$	$O\left(N \log N \, (\log^2 \lambda)\right)$	$O(N\log^3 \lambda)$

https://eprint.iacr.org/2018/560.pdf

- General Statements
- Sub-linear proofs
- Relies on SIS



N gates

Security parameter λ