LogRobin++: Optimizing Proofs of Disjunctive Statements in VOLE-Based ZK

Carmit Hazay, Bar-Ilan University and Ligero Inc.
David Heath, UIUC
Vladimir Kolesnikov, Georgia Tech
Muthuramakrishnan Venkitasubramaniam, Ligero Inc.
Yibin Yang, Georgia Tech

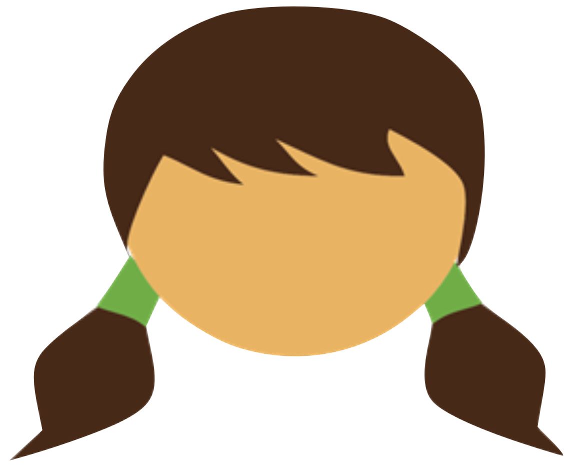
Zero-Knowledge Proof [GMR85]

Completeness: An honest P always succeeds

Soundness: A malicious P always fails

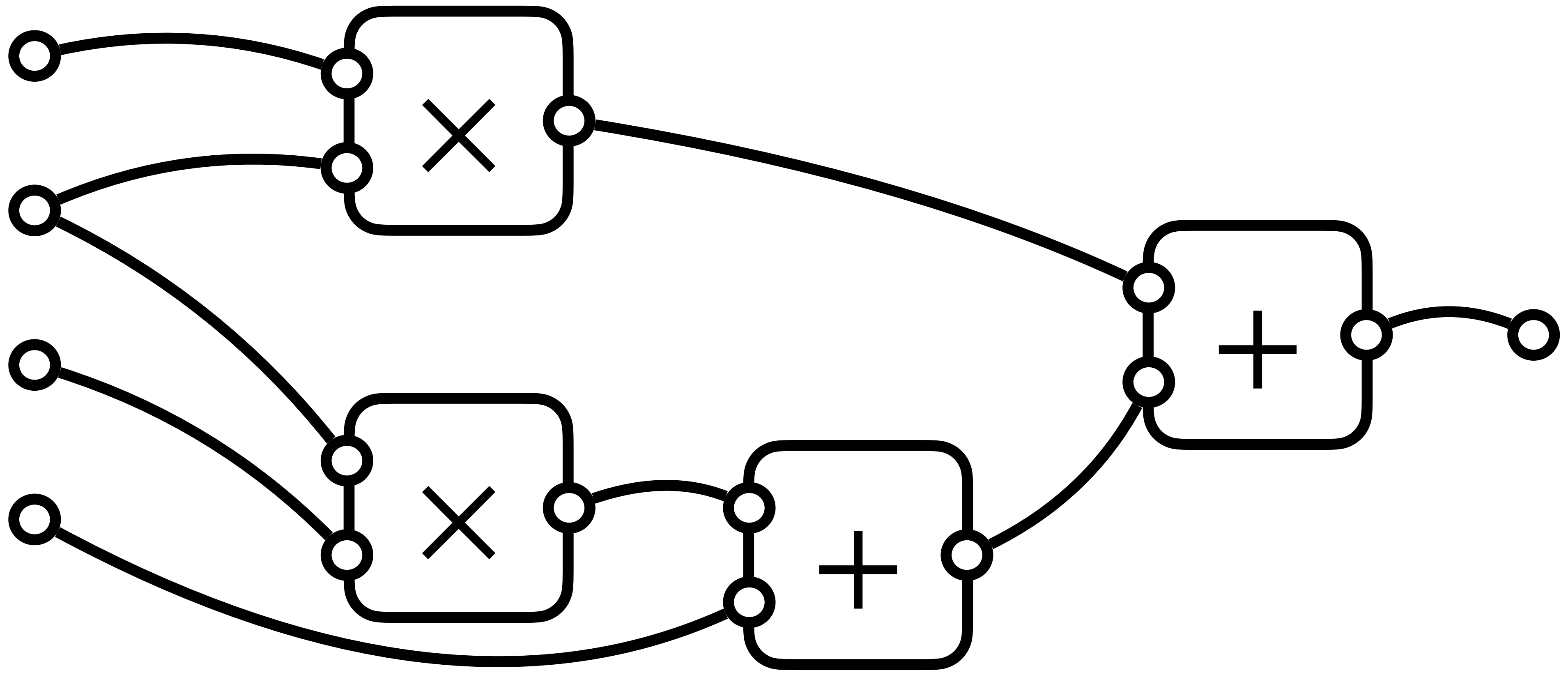
Zero Knowledge: A malicious V learns nothing more

Verifier

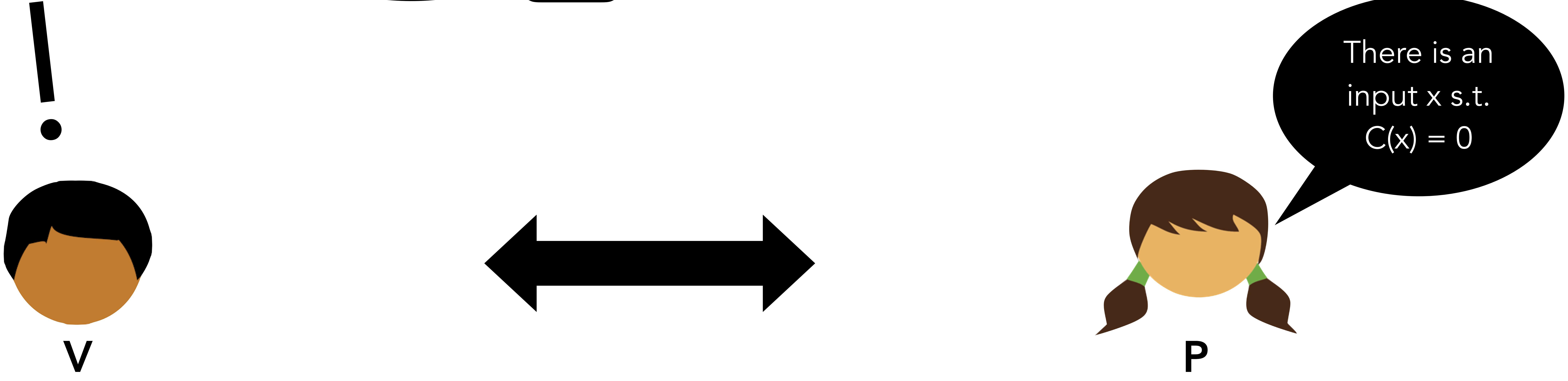


Prover

ZKP for a Circuit

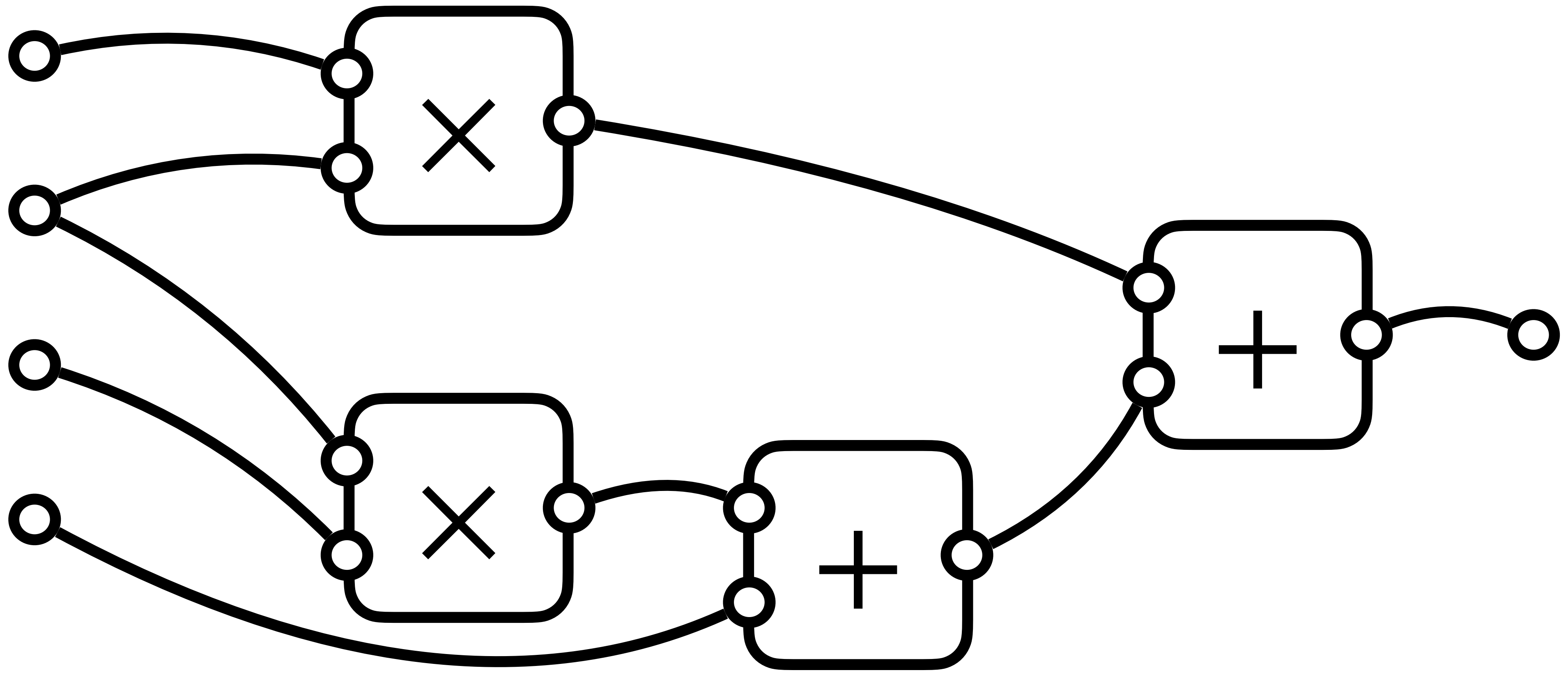


General-purpose ZK systems almost exclusively work with circuits or constraint systems

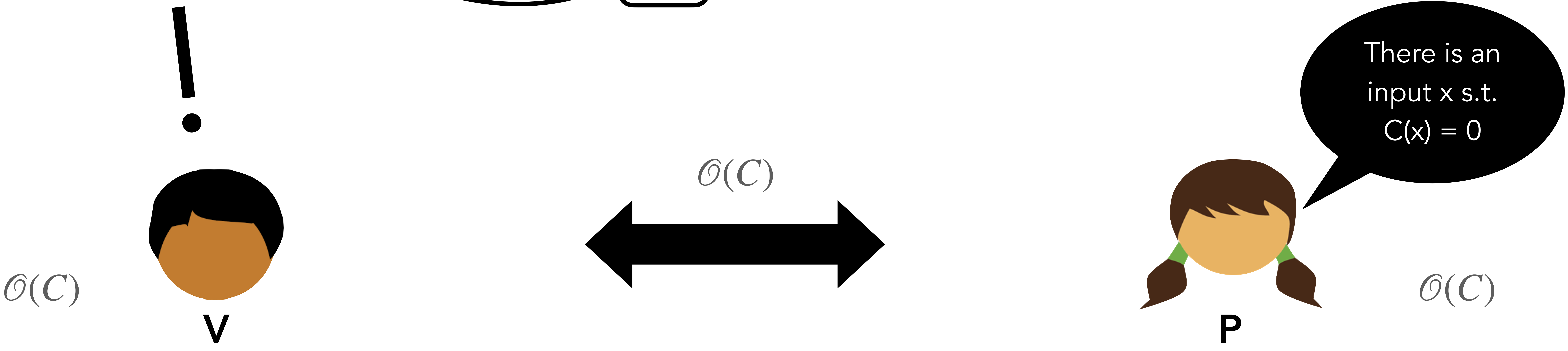


ZKP for a Circuit

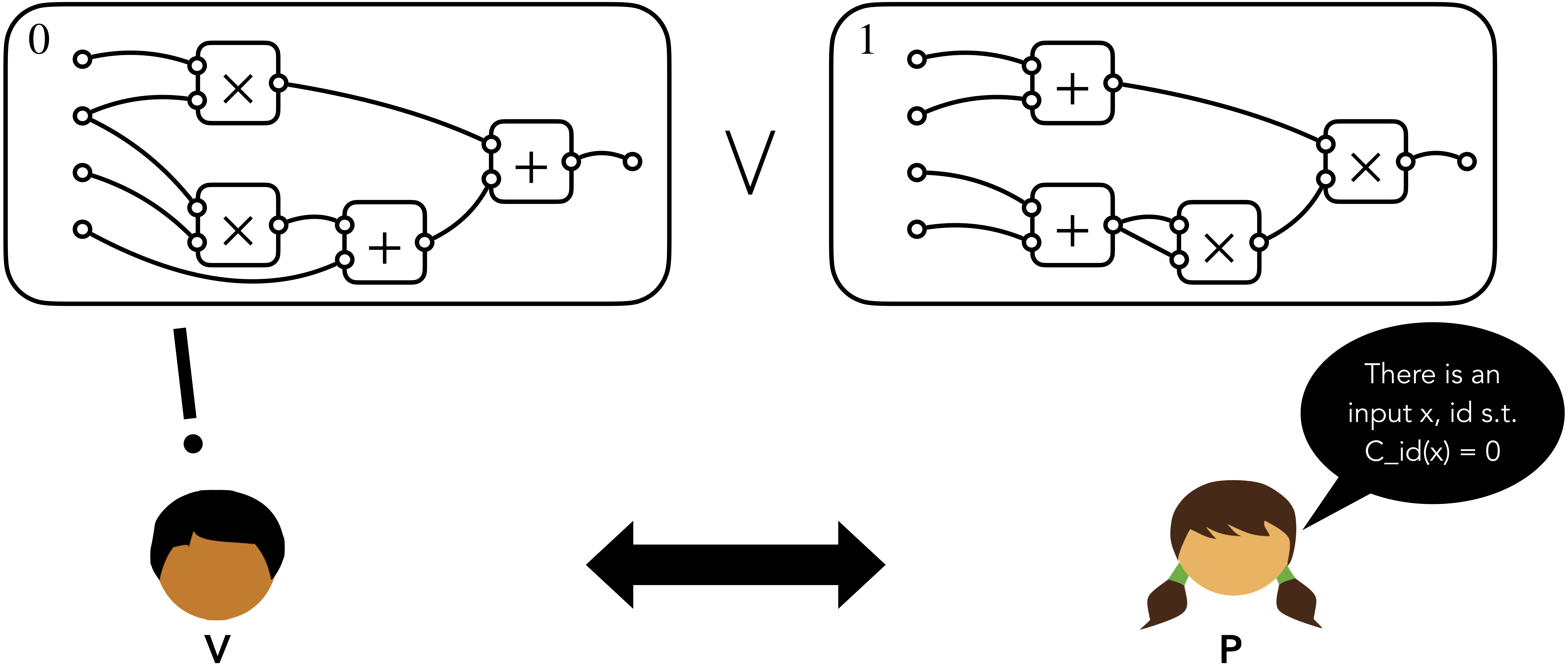
Instantiated over Vector Oblivious Linear Evaluation (VOLE) [DIO21, YSWW21]



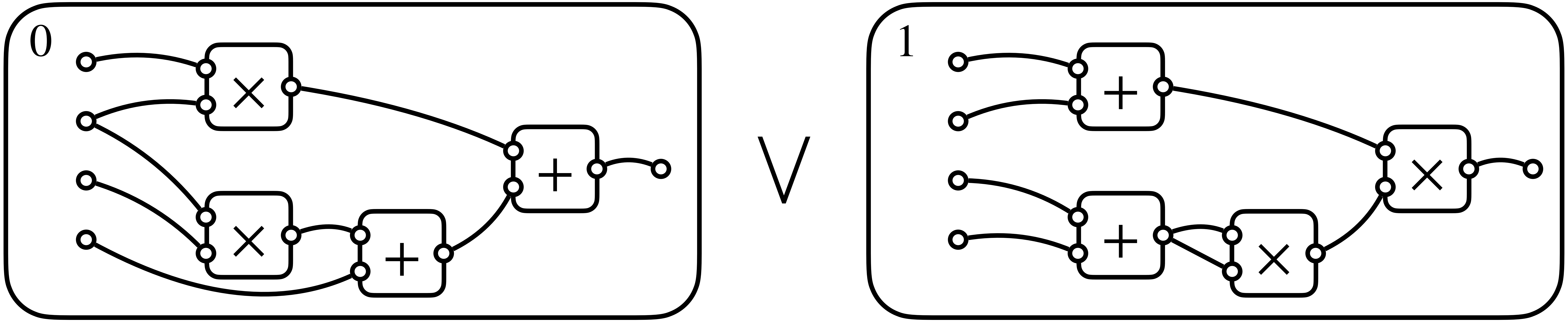
General-purpose ZK systems almost exclusively work with circuits or constraint systems

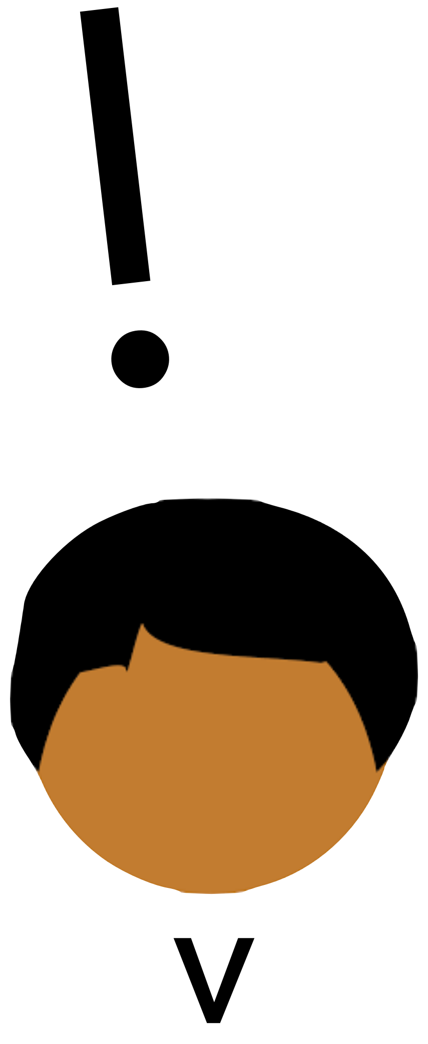


Disjunctive Statements as Circuits

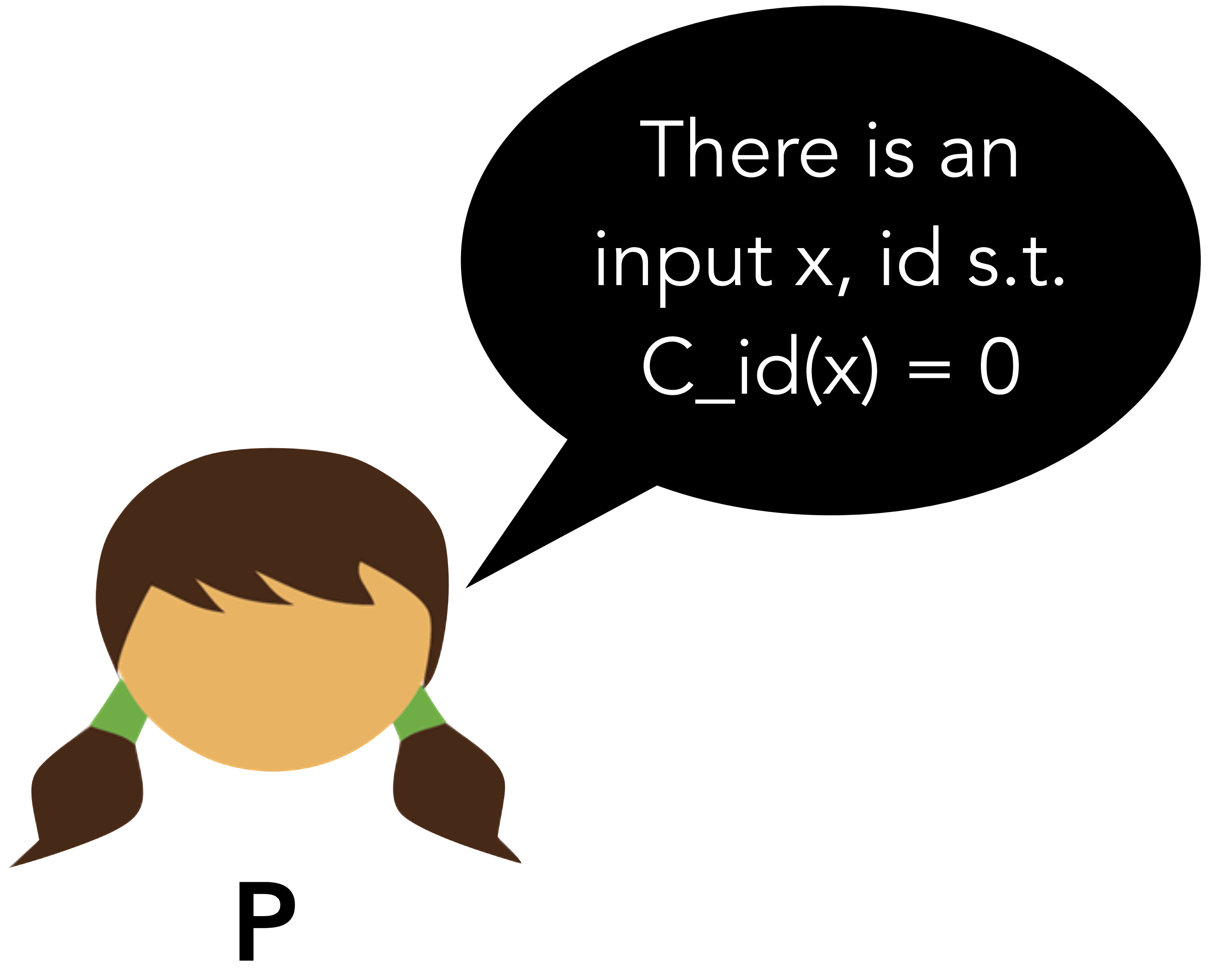


Disjunctive Statements as Circuits





The naïve solution: construct a multiplexed circuit of size $\mathcal{O}(C_0 + C_1)$



Why Disjunctive Statements?

Why Disjunctive Statements?

ADD or MULT or MOD or ...

A crucial component to emulate the CPU execution (aka a RAM program) inside ZK

Prior Work: Robin [YHH+23]

Refined Oblivious Branching for INteractive zk

Prior Work: Robin [YHH+23]

Refined Oblivious Branching for INteractive zk

In VOLE-based ZK, we only need to pay for the largest clause in communication.

Prior Work: Robin [YHH+23]

Refined Oblivious Branching for INteractive zk

In VOLE-based ZK, we only need to pay for the largest clause in communication.

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \cdots \vee \boxed{\mathscr{C}_{B-2}} \vee \boxed{\mathscr{C}_{B-1}}$$

Defined over some field $\mathbb{F}$, each with n_{in} inputs and $n_\times$ multiplications.

★ For simplicity, we assume a large enough field.

Prior Work: Robin [YHH+23]

Refined Oblivious Branching for INteractive zk

In VOLE-based ZK, we only need to pay for the largest clause in communication.

$$\boxed{\mathscr{C}_0} \lor \boxed{\mathscr{C}_1} \lor \cdots \lor \boxed{\mathscr{C}_{B-2}} \lor \boxed{\mathscr{C}_{B-1}}$$

Defined over some field $\mathbb{F}$, each with n_{in} inputs and $n_\times$ multiplications.

★ For simplicity, we assume a large enough field.

Communication in the VOLE-hybrid model:

$\mathcal{O}(n_{in} + Bn_\times)$ field elements $\xrightarrow{\text{improve to}}$ $n_{in} + 3n_\times + \mathcal{O}(B)$ field elements

Prior Work: Robin [YHH+23]

Refined Oblivious Branching for INteractive zk

In VOLE-based ZK, we only need to pay for the largest clause in communication.

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \cdots \vee \boxed{\mathscr{C}_{B-2}} \vee \boxed{\mathscr{C}_{B-1}}$$

Defined over some field $\mathbb{F}$, each with n_{in} inputs and $n_\times$ multiplications.

★ For simplicity, we assume a large enough field.

Communication in the VOLE-hybrid model:

$\mathcal{O}(n_{in} + Bn_\times)$ field elements $\xrightarrow{\text{improve to}}$ $n_{in} + 3n_\times + \mathcal{O}(B)$ field elements

this work ↓ further improve

Our Results

Our Results

LogRobin++

Our Results

LogRobin++

Robin

$n_{in} + 3n_{\times} + \mathcal{O}(B)$

field elements

Our Results

LogRobin++

LogRobin

$n_{in} + 3n_{\times} + \mathcal{O}(\log B)$

field elements

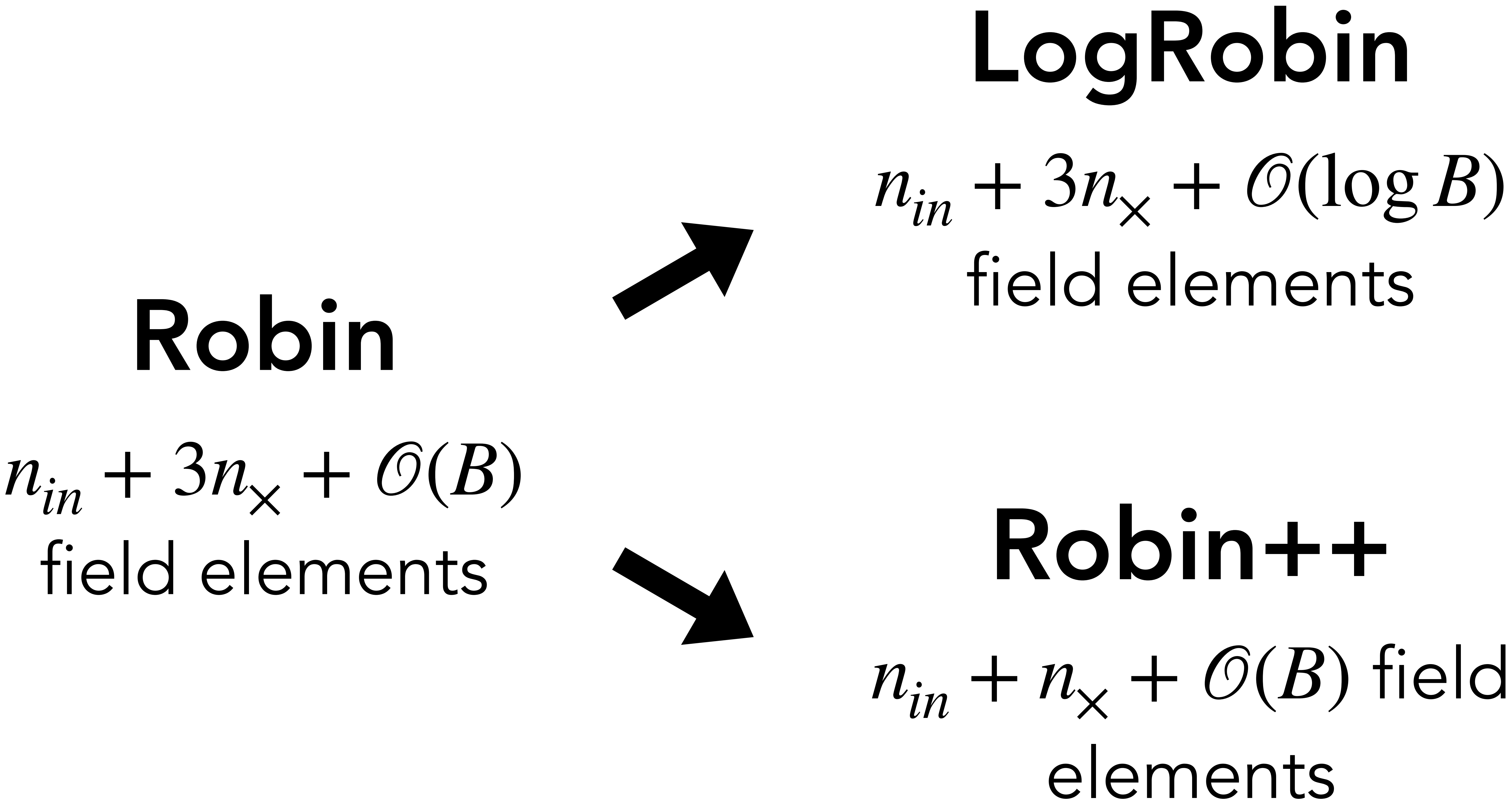
Robin

$n_{in} + 3n_{\times} + \mathcal{O}(B)$

field elements

Our Results

LogRobin++



LogRobin

$n_{in} + 3n_{\times} + \mathcal{O}(\log B)$ field elements

Robin

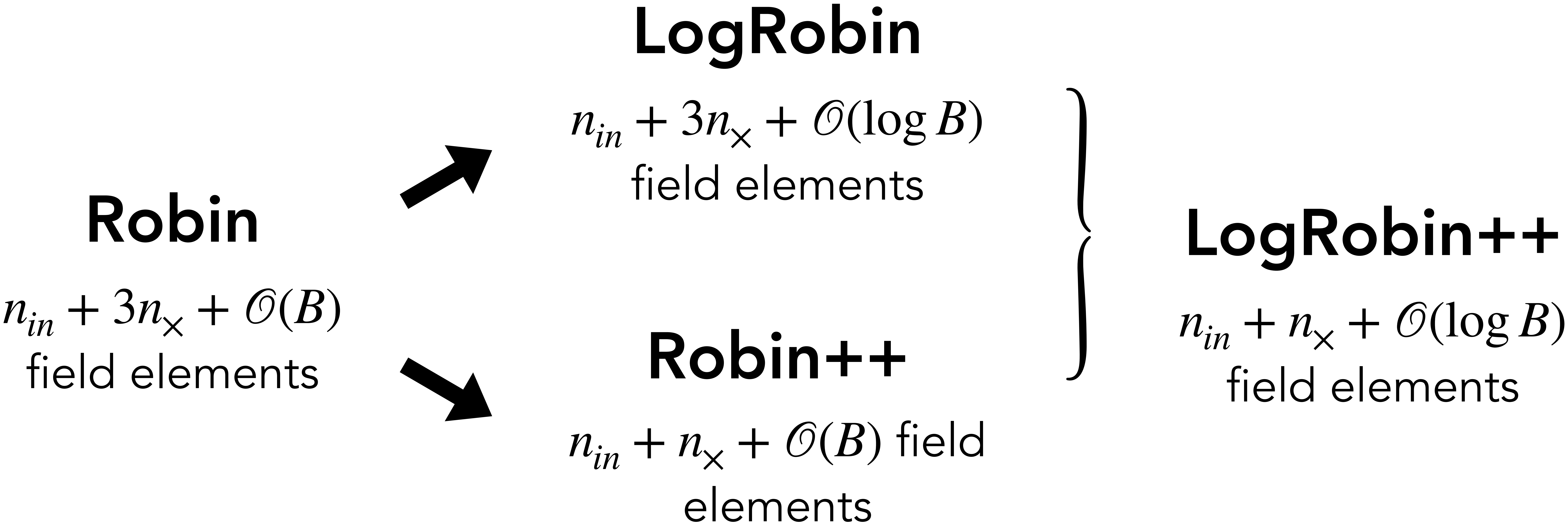
$n_{in} + 3n_{\times} + \mathcal{O}(B)$ field elements

Robin++

$n_{in} + n_{\times} + \mathcal{O}(B)$ field elements

Our Results

LogRobin++



Robin

$n_{in} + 3n_\times + \mathcal{O}(B)$ field elements

LogRobin

$n_{in} + 3n_\times + \mathcal{O}(\log B)$ field elements

Robin++

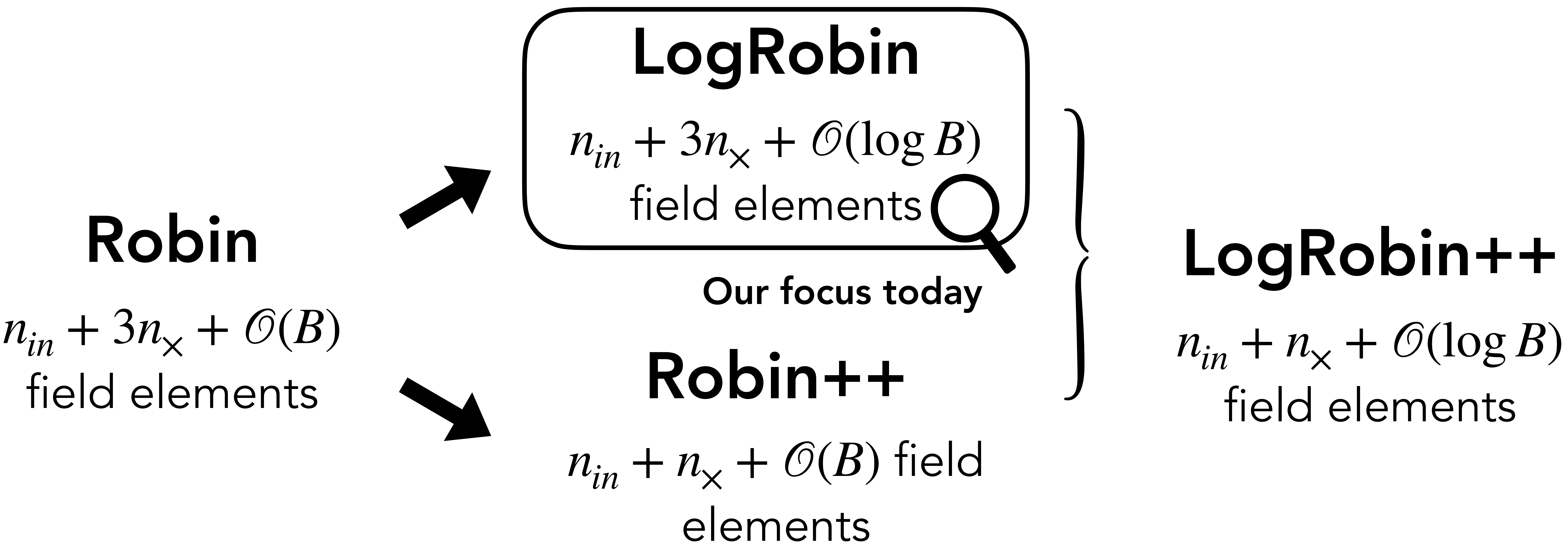
$n_{in} + n_\times + \mathcal{O}(B)$ field elements

LogRobin++

$n_{in} + n_\times + \mathcal{O}(\log B)$ field elements

Our Results

LogRobin++



Robin

$n_{in} + 3n_{\times} + \mathcal{O}(B)$
field elements

LogRobin

$n_{in} + 3n_{\times} + \mathcal{O}(\log B)$
field elements

Our focus today

Robin++

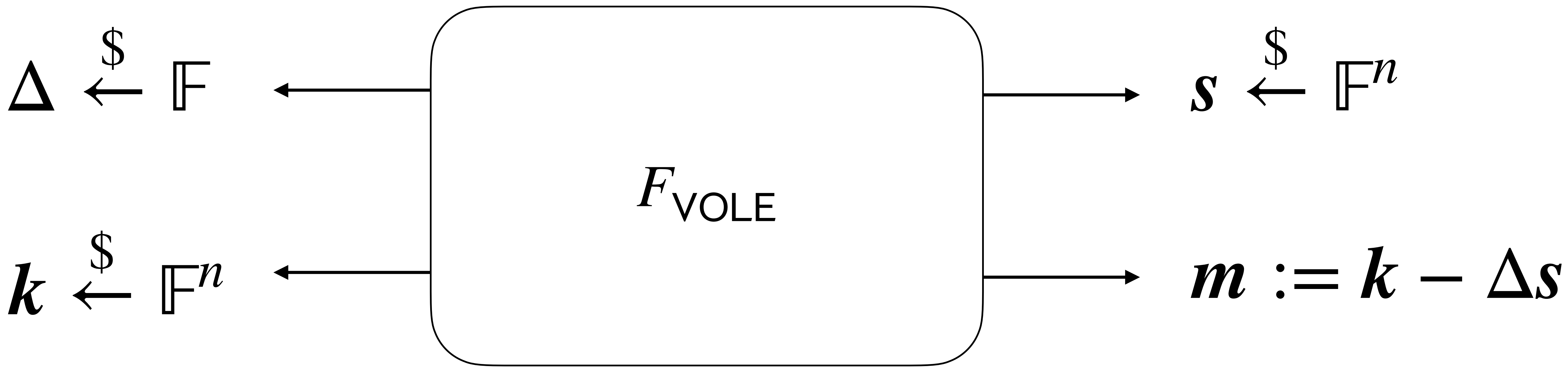
$n_{in} + n_{\times} + \mathcal{O}(B)$ field
elements

LogRobin++

$n_{in} + n_{\times} + \mathcal{O}(\log B)$
field elements

Preliminaries: Vector OLE Correlations

Preliminaries: Vector OLE Correlations



$\Delta \xleftarrow{\$} \mathbb{F}$

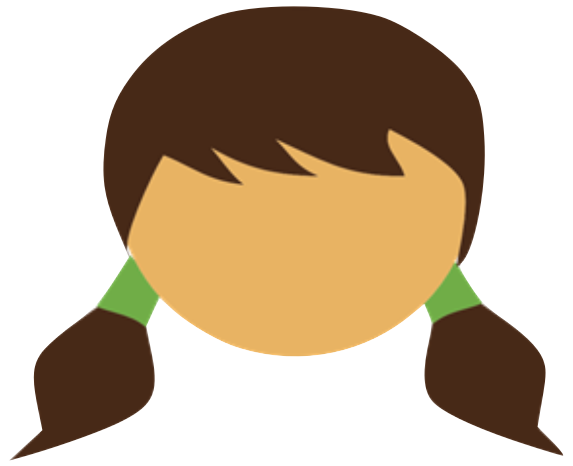
$\boldsymbol{k} \xleftarrow{\$} \mathbb{F}^n$

F_{VOLE}

$\boldsymbol{s} \xleftarrow{\$} \mathbb{F}^n$

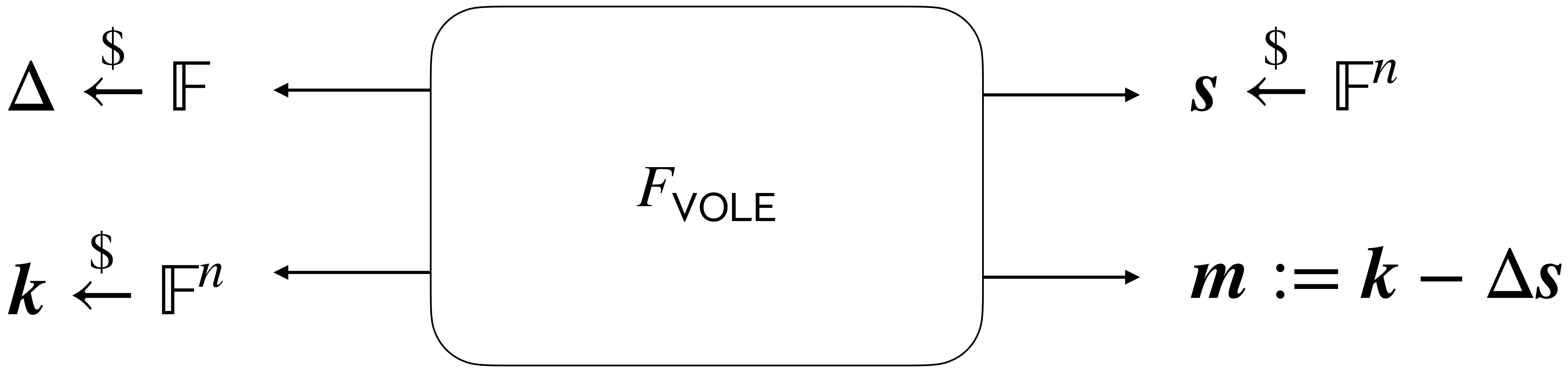
$\boldsymbol{m} := \boldsymbol{k} - \Delta \boldsymbol{s}$

V



P

Preliminaries: Vector OLE Correlations



$\Delta \xleftarrow{\$} \mathbb{F}$

$\boldsymbol{k} \xleftarrow{\$} \mathbb{F}^n$

F_{VOLE}

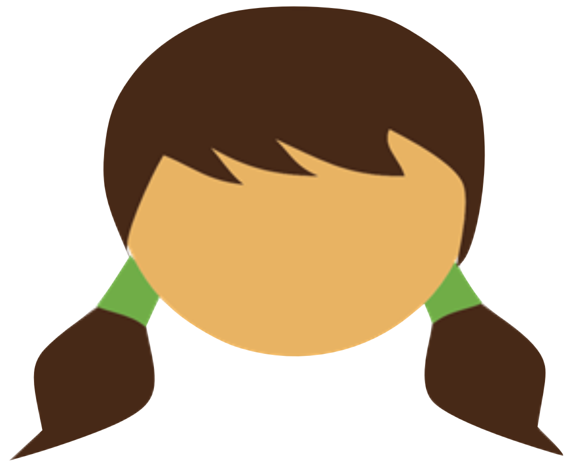
$\boldsymbol{s} \xleftarrow{\$} \mathbb{F}^n$

$\boldsymbol{m} := \boldsymbol{k} - \Delta \boldsymbol{s}$

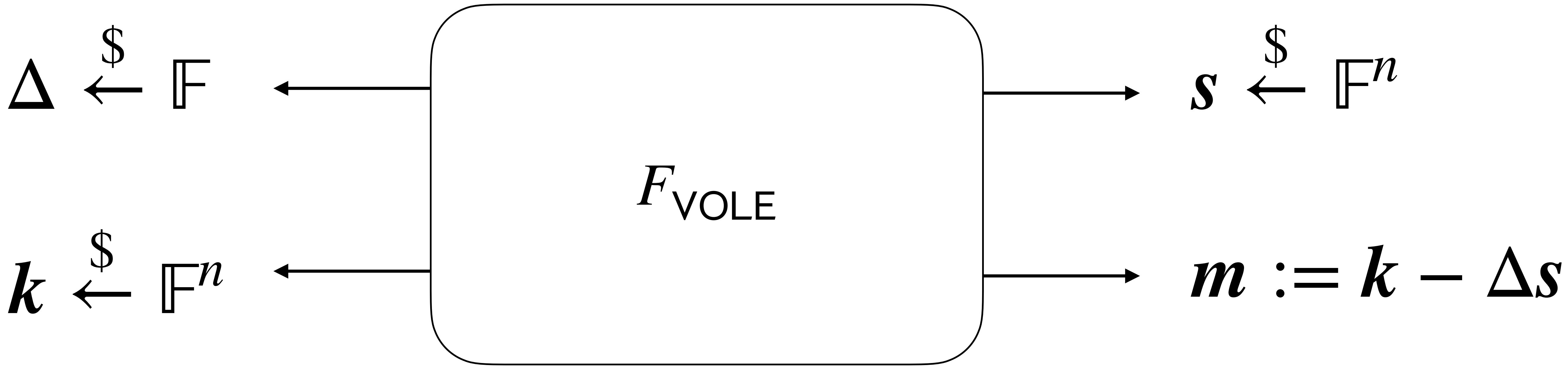
k_{s_0}, Δ

$[s_0]$

$s_0, m_{s_0} = k_{s_0} - s_0\Delta$



Preliminaries: Vector OLE Correlations



$\Delta \xleftarrow{\$} \mathbb{F}$

$\boldsymbol{k} \xleftarrow{\$} \mathbb{F}^n$

F_{VOLE}

$\boldsymbol{s} \xleftarrow{\$} \mathbb{F}^n$

$\boldsymbol{m} := \boldsymbol{k} - \Delta \boldsymbol{s}$

$[s_0]$ $[s_1]$

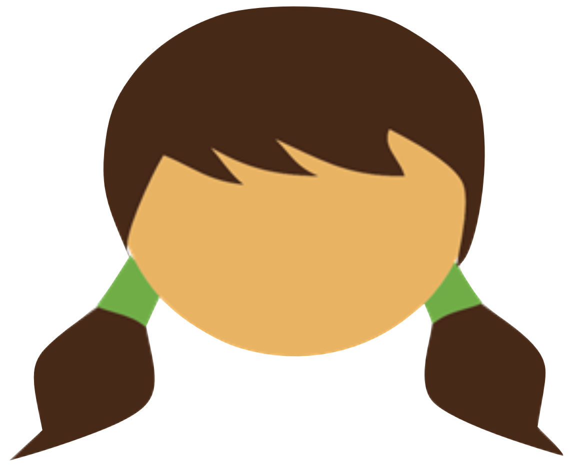
$a, b, c \in \mathbb{F}$

Linear Homomorphic

$a, b, c \in \mathbb{F}$

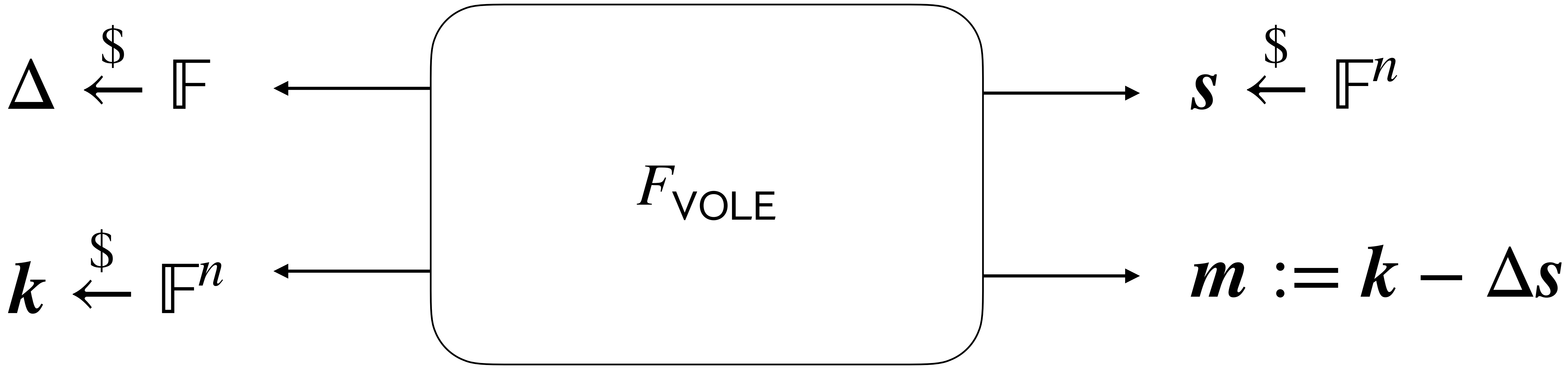
$[as_0 + bs_1 + c]$

V



P

Preliminaries: Vector OLE Correlations



$\Delta \xleftarrow{\$} \mathbb{F}$

$\boldsymbol{k} \xleftarrow{\$} \mathbb{F}^n$

F_{VOLE}

$\boldsymbol{s} \xleftarrow{\$} \mathbb{F}^n$

$\boldsymbol{m} := \boldsymbol{k} - \Delta\boldsymbol{s}$

$[s_0] \quad [s_1]$

$a, b, c \in \mathbb{F}$

Linear Homomorphic

$a, b, c \in \mathbb{F}$

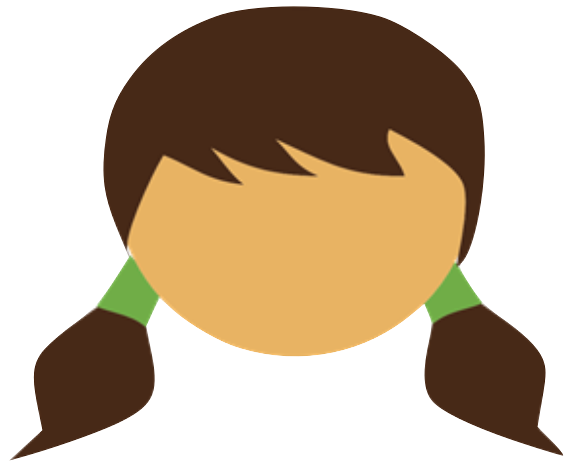
$x \in \mathbb{F}$

$[as_0 + bs_1 + c]$

V

$$\xleftarrow{\quad x - s_0 \quad}$$

$[x] = [s_0] + (x - s_0)$



P

Preliminaries: Multiplication ZK Check

Known as line-point zero-knowledge (LPZK) [DIO21, YSWW21]

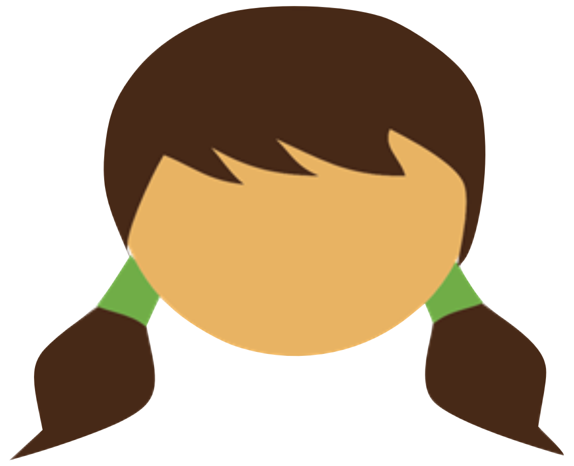
Preliminaries: Multiplication ZK Check

Known as line-point zero-knowledge (LPZK) [DIO21, YSWW21]

V

$[x]$ $[y]$ $[z]$

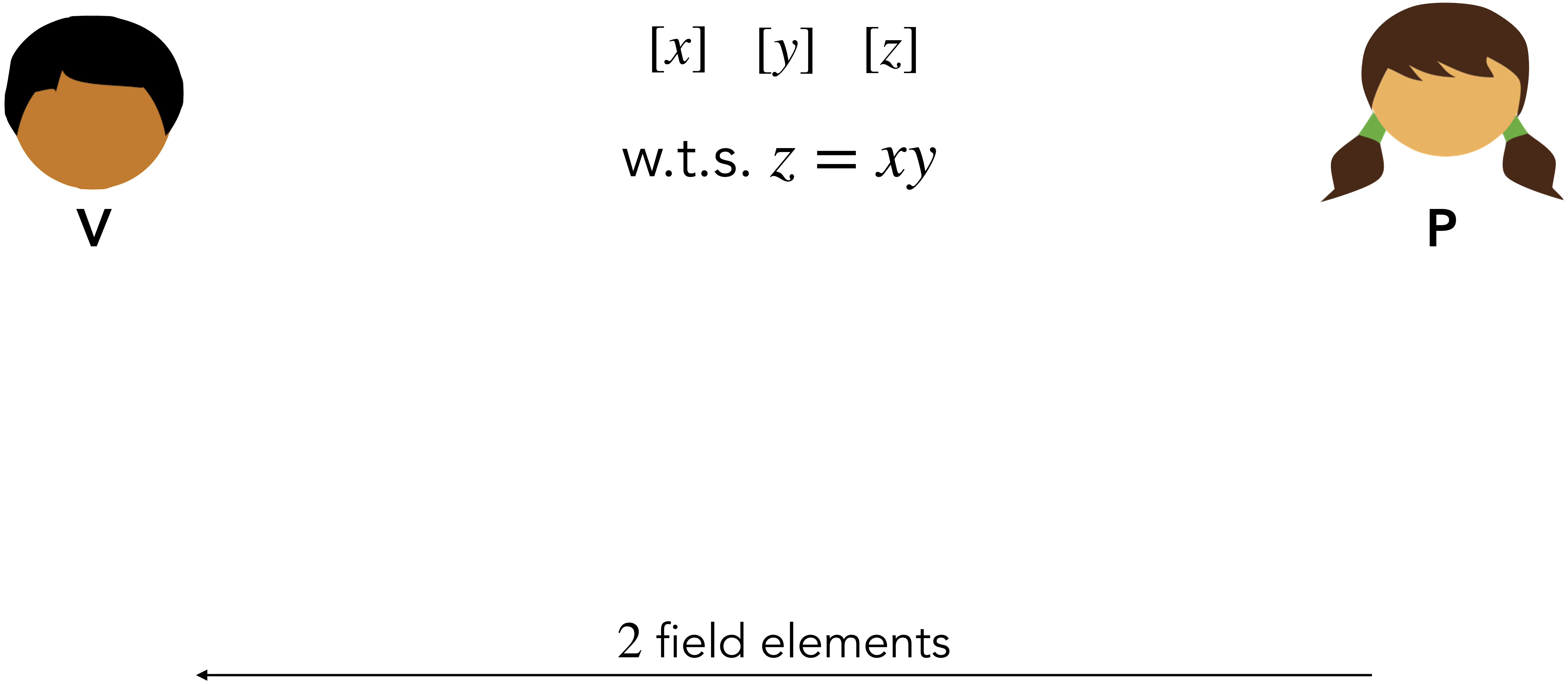
w.t.s. $z = xy$



P

Preliminaries: Multiplication ZK Check

Known as line-point zero-knowledge (LPZK) [DIO21, YSWW21]



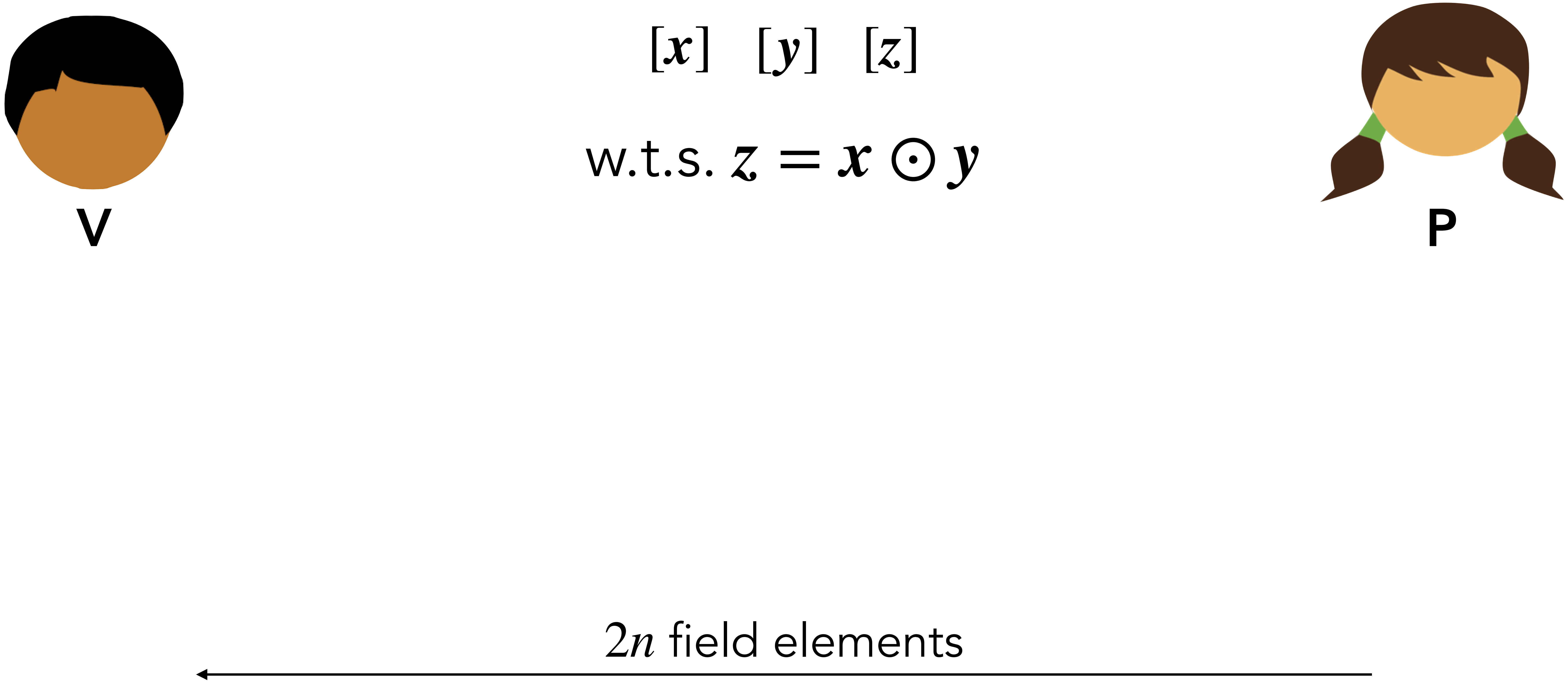
$[x]\quad [y]\quad [z]$

w.t.s. $z = xy$

2 field elements

Preliminaries: Multiplication ZK Check

Known as line-point zero-knowledge (LPZK) [DIO21, YSWW21]



V

P

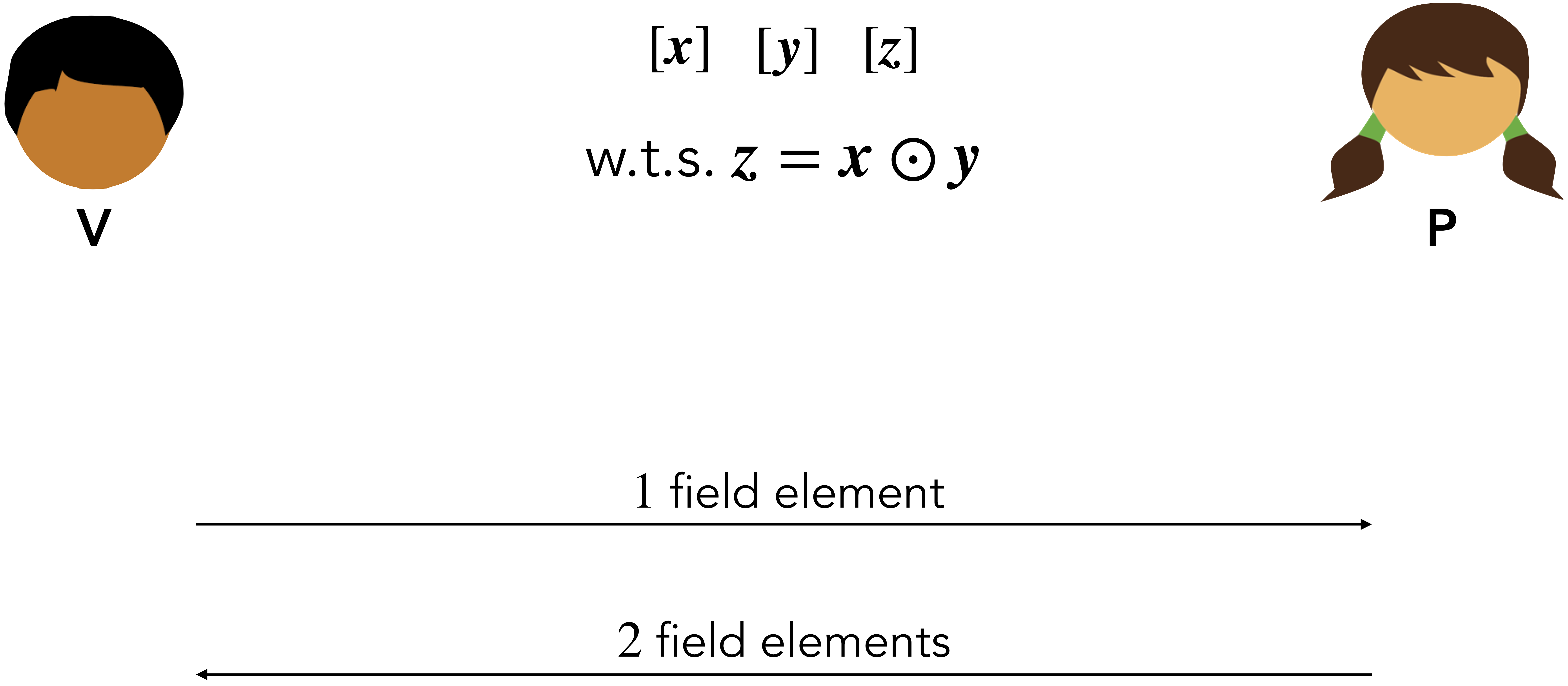
$[\boldsymbol{x}]$ $[\boldsymbol{y}]$ $[\boldsymbol{z}]$

w.t.s. $\boldsymbol{z} = \boldsymbol{x} \odot \boldsymbol{y}$

$2n$ field elements

Preliminaries: Multiplication ZK Check

Known as line-point zero-knowledge (LPZK) [DIO21, YSWW21]



V

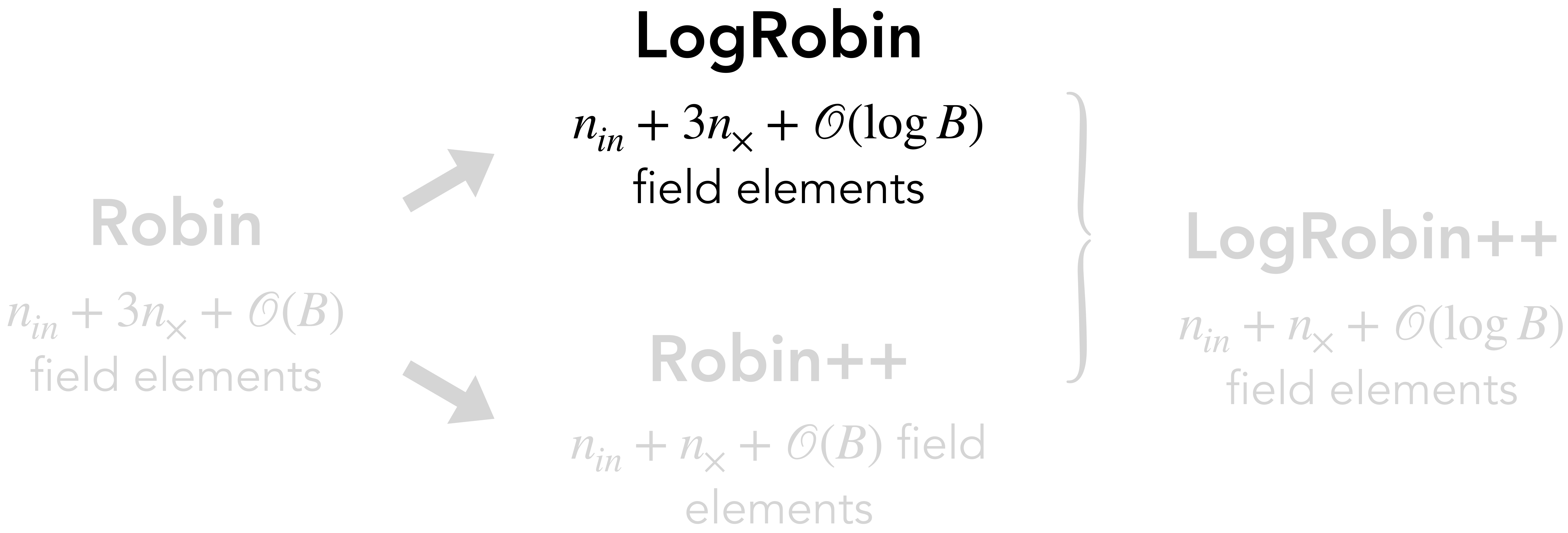
$[\boldsymbol{x}]$ $[\boldsymbol{y}]$ $[\boldsymbol{z}]$

w.t.s. $\boldsymbol{z} = \boldsymbol{x} \odot \boldsymbol{y}$

P

1 field element

2 field elements



Robin

$n_{in} + 3n_{\times} + \mathcal{O}(B)$
field elements

LogRobin

$n_{in} + 3n_{\times} + \mathcal{O}(\log B)$
field elements

Robin++

$n_{in} + n_{\times} + \mathcal{O}(B)$ field
elements

LogRobin++

$n_{in} + n_{\times} + \mathcal{O}(\log B)$
field elements

Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]$ $[in_2]$ $[in_3]$ $[in_4]$ $[\ell_1]$ $[r_1]$ $[o_1]$ $[\ell_2]$ $[r_2]$ $[o_2]$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\ \ [in_2]\ \ [in_3]\ \ [in_4]\ \ \underbrace{[\ell_1]\ \ [r_1]\ \ [o_1]}_{\text{mult.}}\ \ \underbrace{[\ell_2]\ \ [r_2]\ \ [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

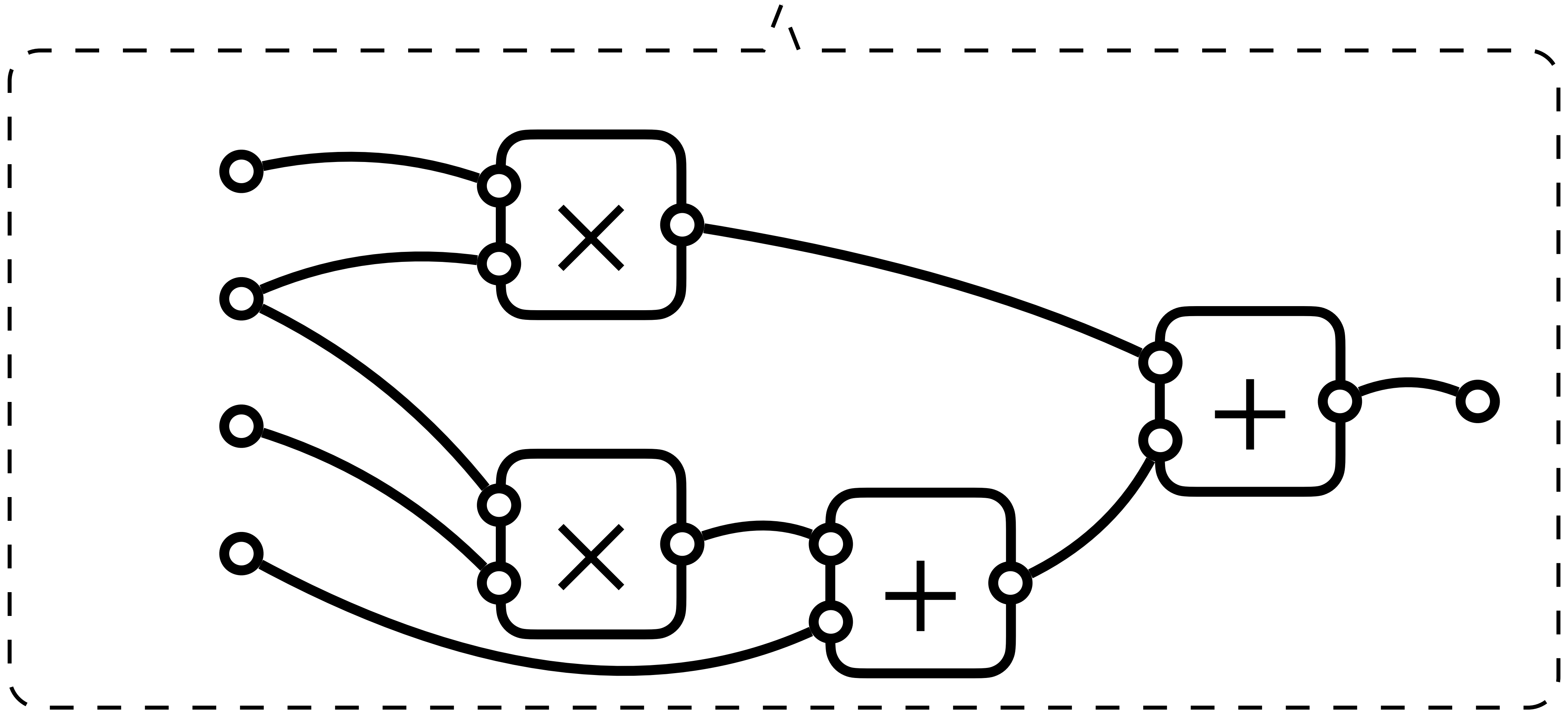
Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

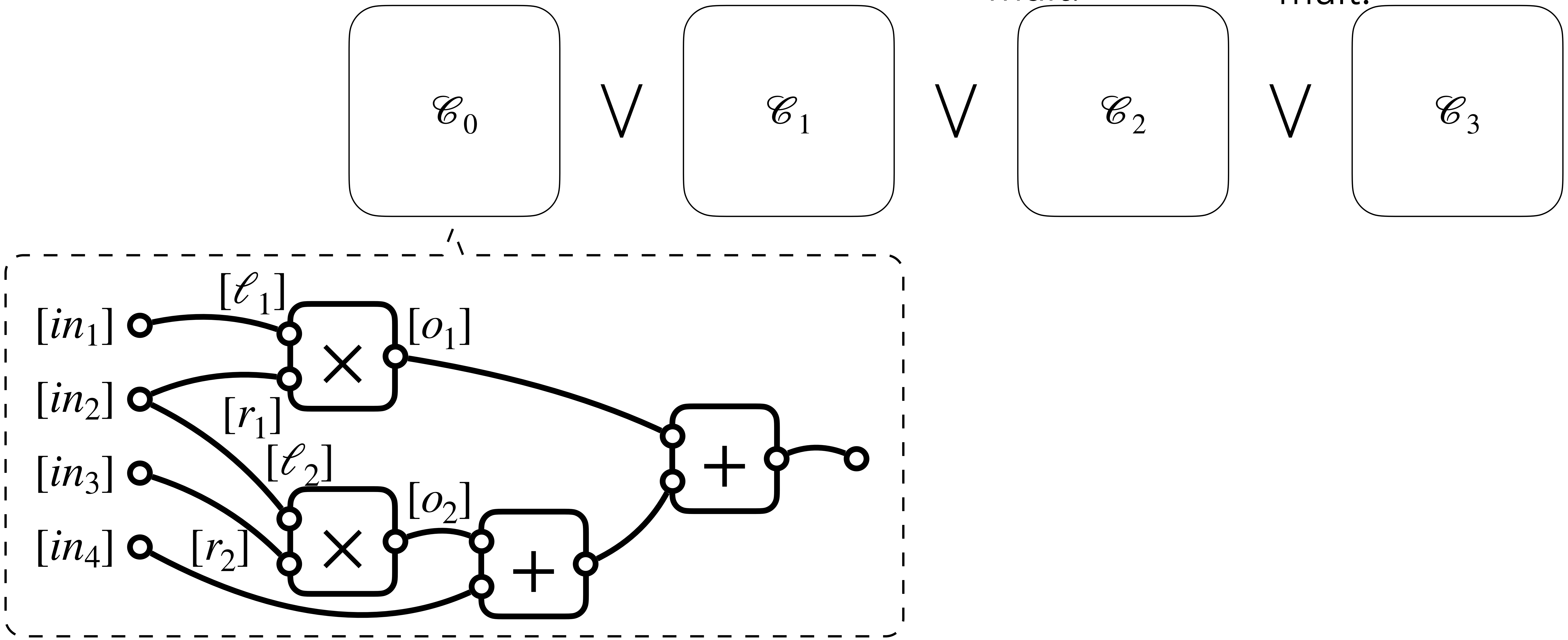


Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\ \ [in_2]\ \ [in_3]\ \ [in_4]\ \ \underbrace{[\ell_1]\ \ [r_1]\ \ [o_1]}_{\text{mult.}}\ \ \underbrace{[\ell_2]\ \ [r_2]\ \ [o_2]}_{\text{mult.}}$ of the "active" clause



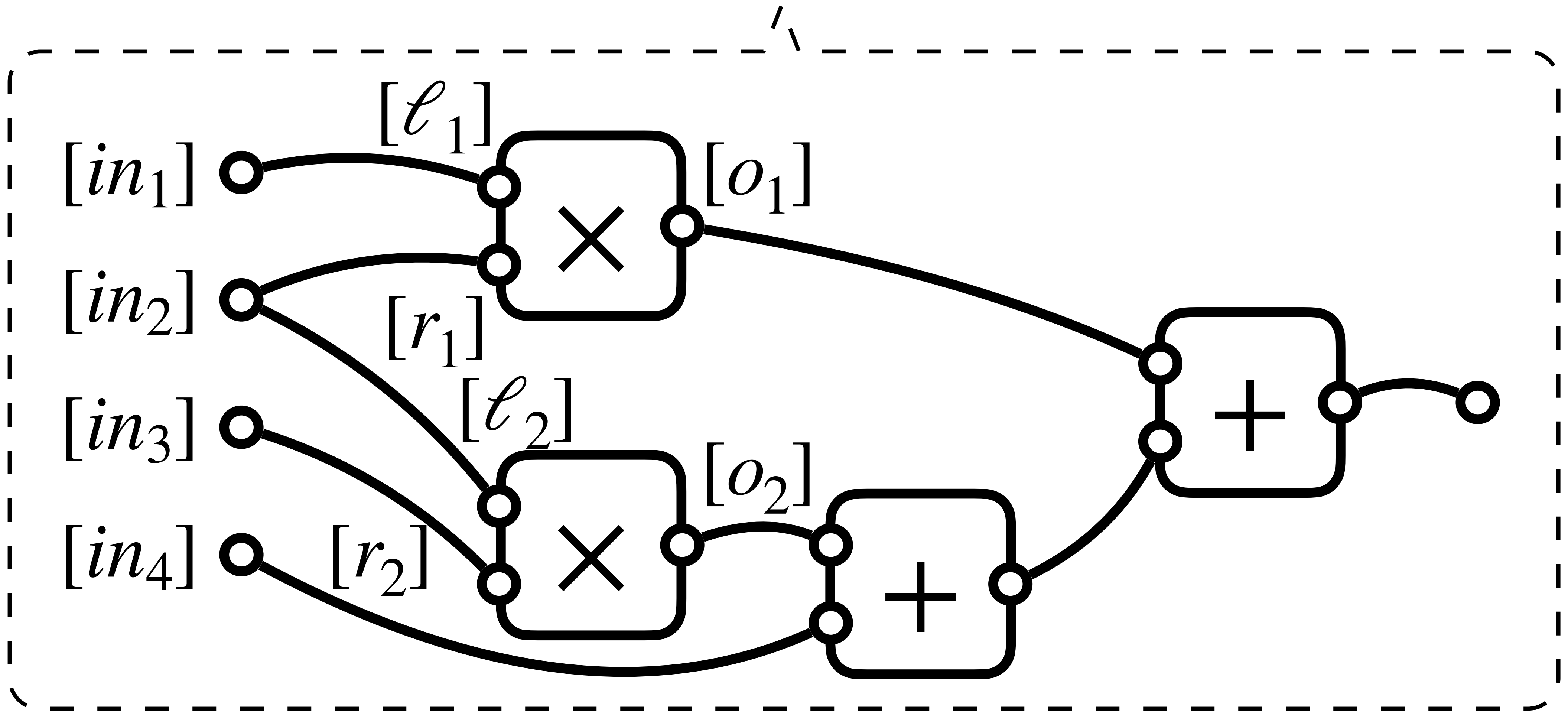
Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$



$$[\boldsymbol{v}_0] = \begin{bmatrix} in_1 - \ell_1 \\ in_2 - r_1 \\ in_2 - \ell_2 \\ in_3 - r_2 \\ o_1 + o_2 + in_4 \end{bmatrix}$$

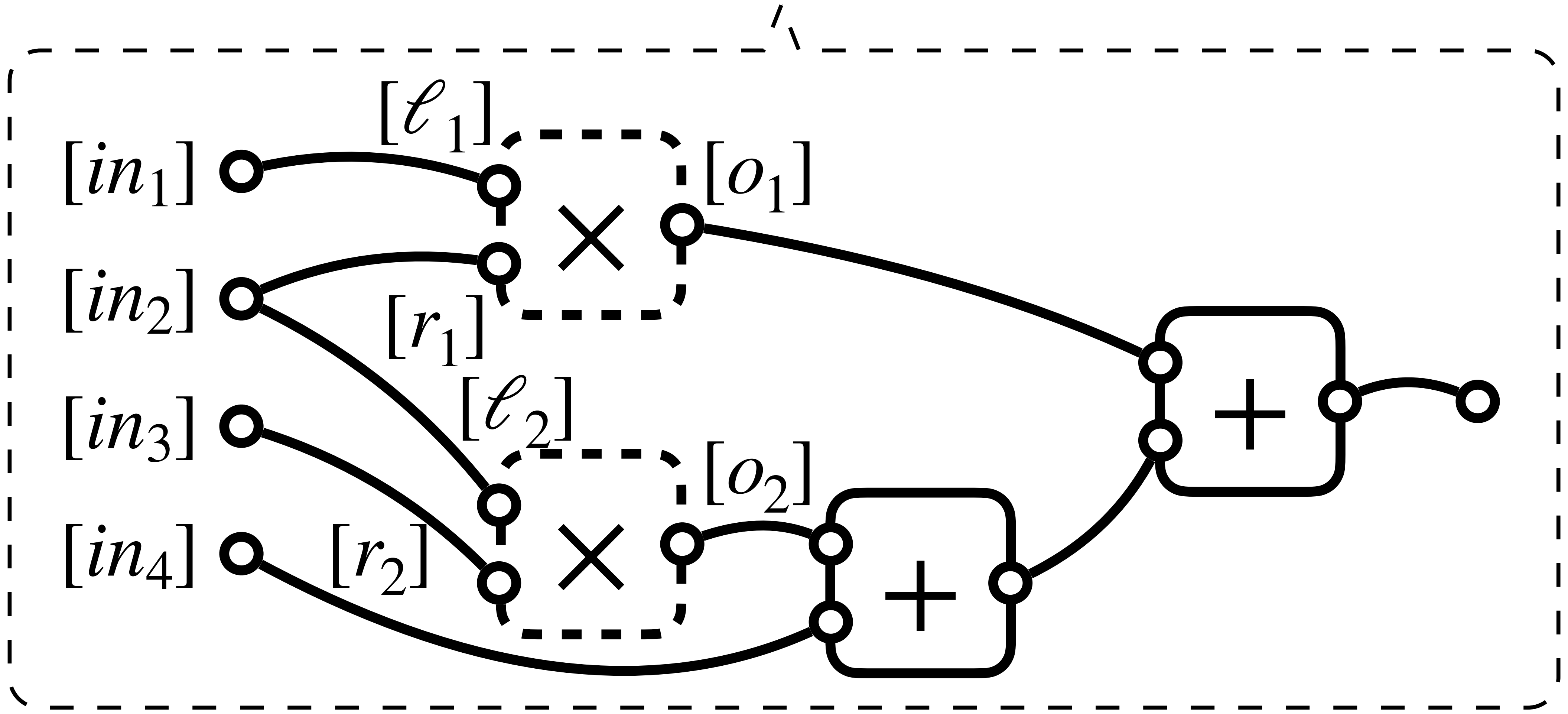
Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1] \quad [in_2] \quad [in_3] \quad [in_4] \quad \underbrace{[\ell_1] \quad [r_1] \quad [o_1]}_{\text{mult.}} \quad \underbrace{[\ell_2] \quad [r_2] \quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$



$$[\boldsymbol{v}_0] = \begin{bmatrix} in_1 - \ell_1 \\ in_2 - r_1 \\ in_2 - \ell_2 \\ in_3 - r_2 \\ o_1 + o_2 + in_4 \end{bmatrix}$$

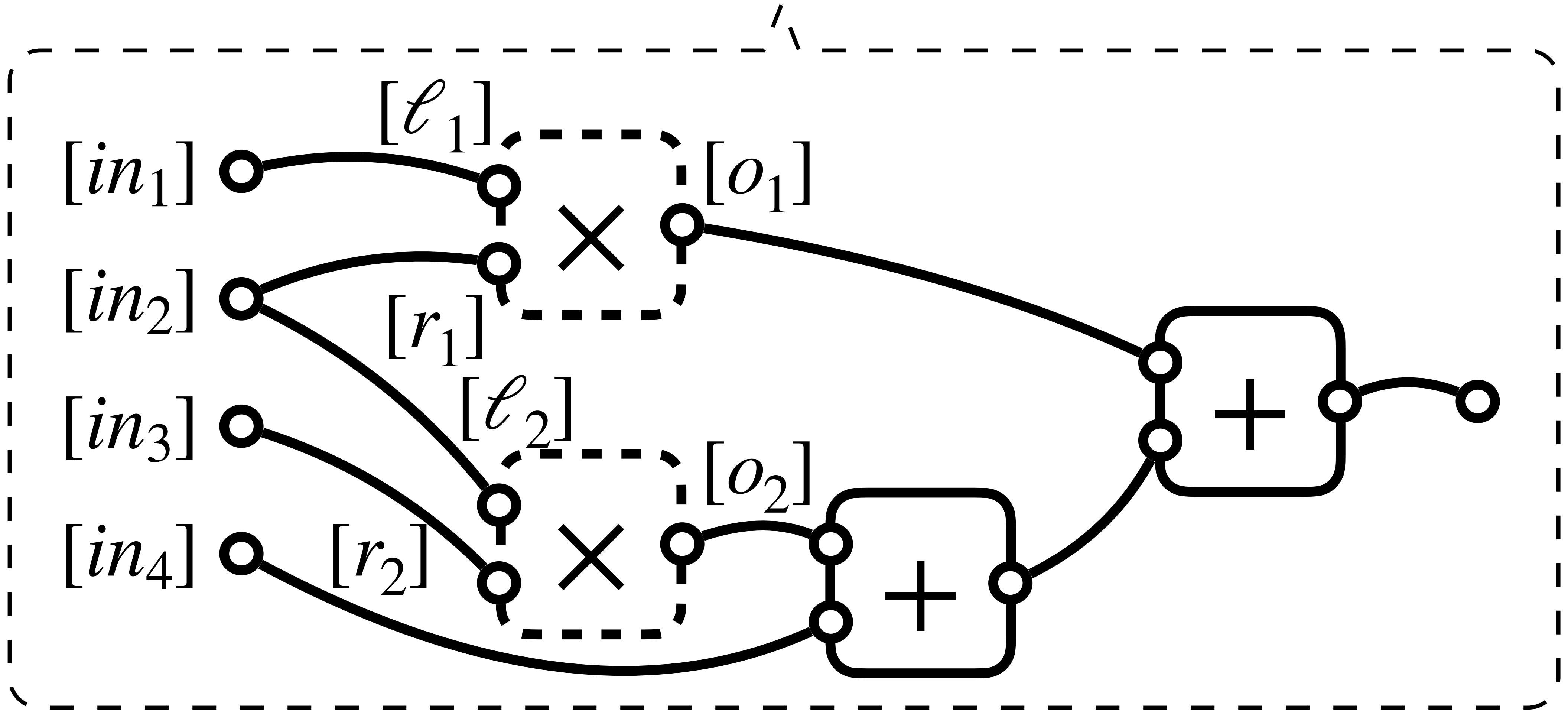
Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$



$$[\boldsymbol{v}_0] = \begin{bmatrix} in_1 - \ell_1 \\ in_2 - r_1 \\ in_2 - \ell_2 \\ in_3 - r_2 \\ o_1 + o_2 + in_4 \end{bmatrix}$$

$\mathscr{C}_0$ is the "active" one

i.f.f.

$\boldsymbol{v}_0 = 0$

Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

$[\boldsymbol{v}_0]\qquad [\boldsymbol{v}_1]\qquad [\boldsymbol{v}_2]\qquad [\boldsymbol{v}_3]$

Technical Overview: LogRobin

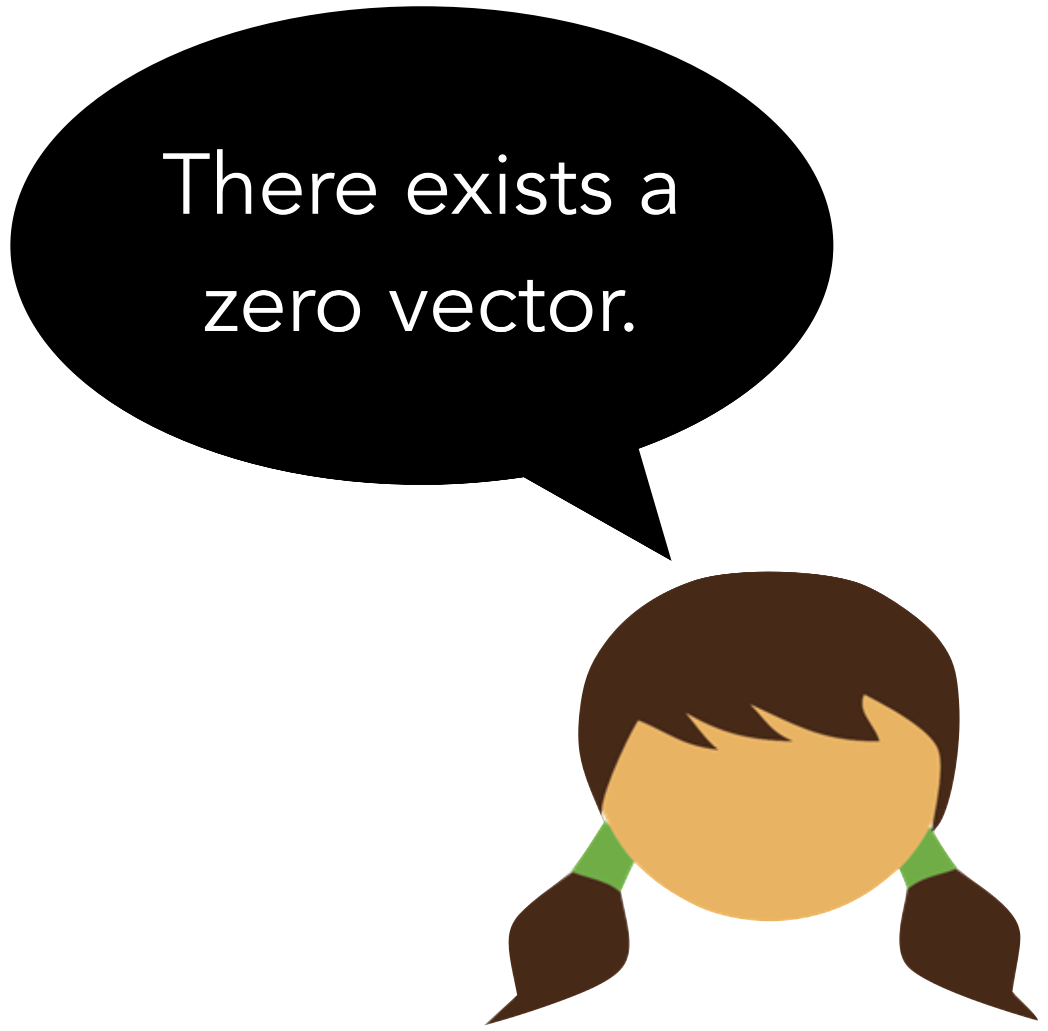
For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

$[\boldsymbol{v}_0]\qquad [\boldsymbol{v}_1]\qquad [\boldsymbol{v}_2]\qquad [\boldsymbol{v}_3]$



Technical Overview: LogRobin

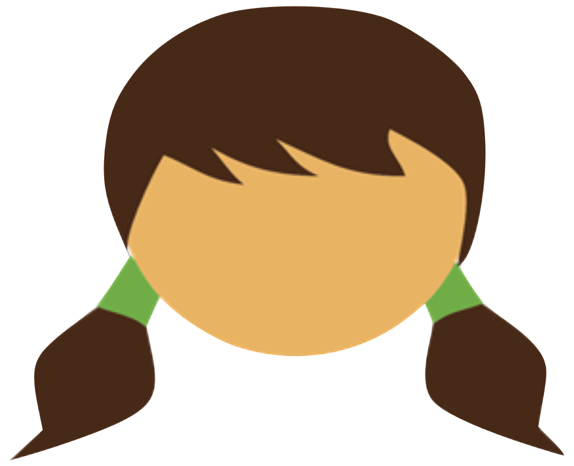
For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

★ For simplicity, we assume a large enough field.

$[in_1]\ \ [in_2]\ \ [in_3]\ \ [in_4]\ \ \underbrace{[\ell_1]\ \ [r_1]\ \ [o_1]}_{\text{mult.}}\ \ \underbrace{[\ell_2]\ \ [r_2]\ \ [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

$$\begin{array}{cccc} [\boldsymbol{v}_0] & [\boldsymbol{v}_1] & [\boldsymbol{v}_2] & [\boldsymbol{v}_3] \\ \times & \times & \times & \times \\ (1,\beta,\beta^2,\ldots) & (1,\beta,\beta^2,\ldots) & (1,\beta,\beta^2,\ldots) & (1,\beta,\beta^2,\ldots) \\ [e_0] & [e_1] & [e_2] & [e_3] \end{array}$$



Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

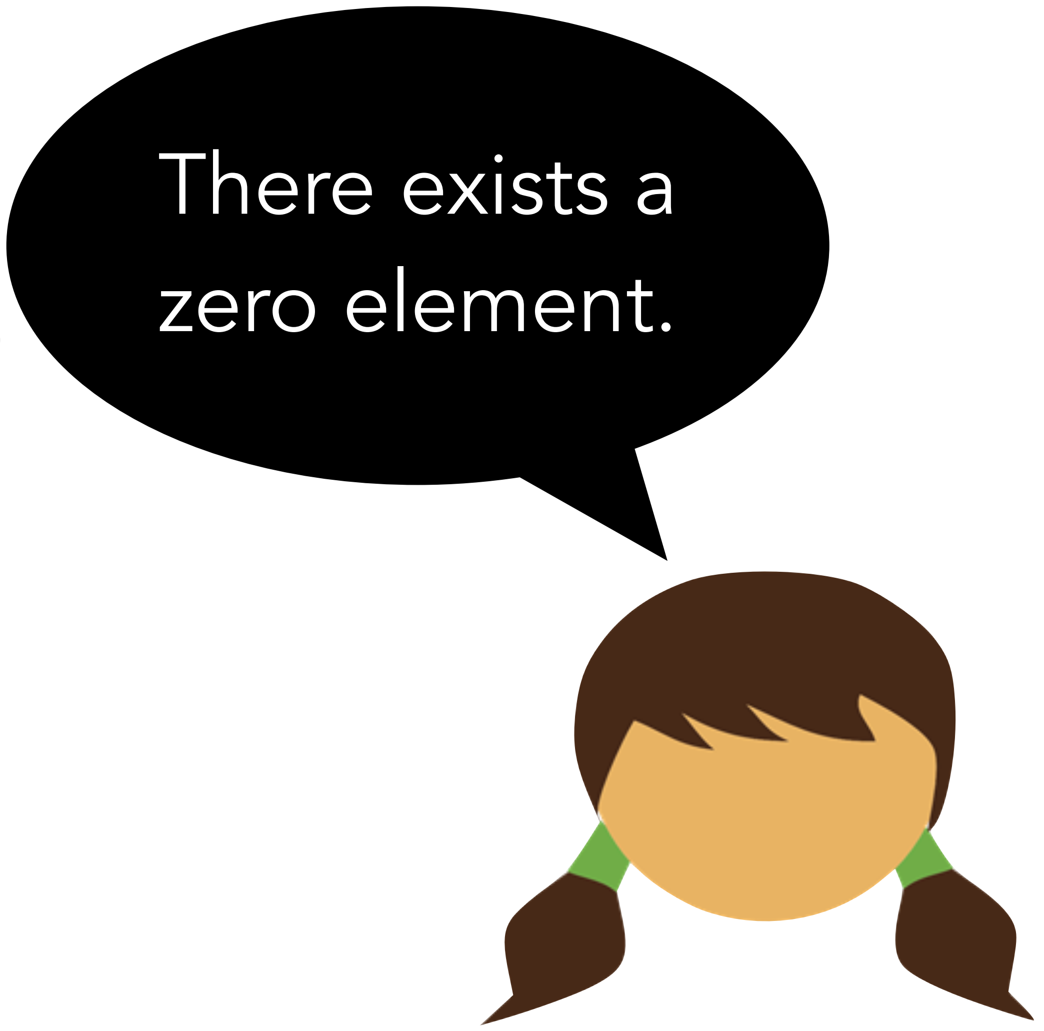
★ For simplicity, we assume a large enough field.

$[in_1]\quad [in_2]\quad [in_3]\quad [in_4]\quad \underbrace{[\ell_1]\quad [r_1]\quad [o_1]}_{\text{mult.}}\quad \underbrace{[\ell_2]\quad [r_2]\quad [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

$$[\boldsymbol{v}_0] \times (1, \beta, \beta^2, \ldots) \qquad [\boldsymbol{v}_1] \times (1, \beta, \beta^2, \ldots) \qquad [\boldsymbol{v}_2] \times (1, \beta, \beta^2, \ldots) \qquad [\boldsymbol{v}_3] \times (1, \beta, \beta^2, \ldots)$$

$$[e_0] \qquad [e_1] \qquad [e_2] \qquad [e_3]$$



Technical Overview: LogRobin

For example, consider the following 4-clause disjunctive statement, defined over some field $\mathbb{F}$, each with 4 inputs and 2 multiplications.

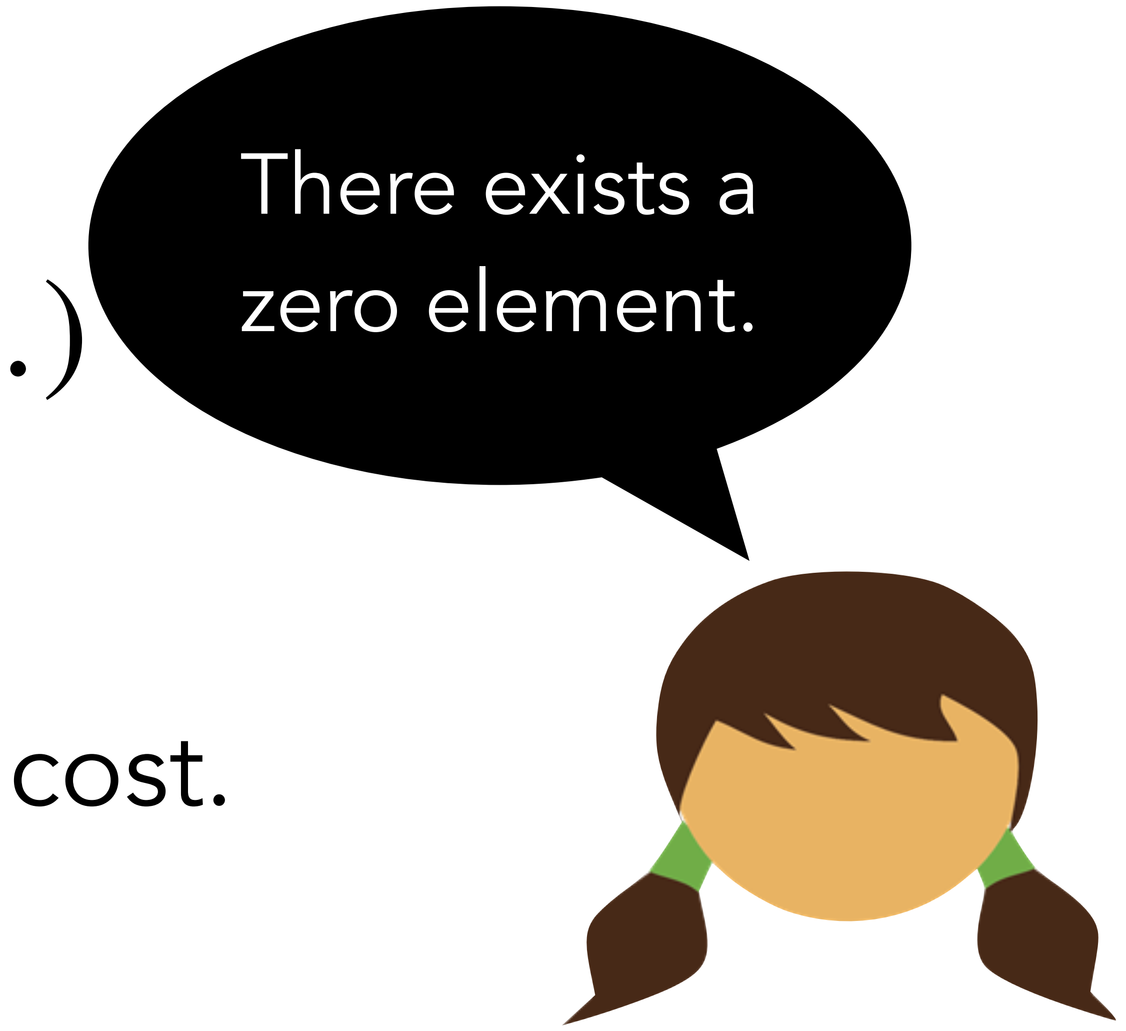
★ For simplicity, we assume a large enough field.

$n_{in} + 3n_{\times}$ field elements $[in_1]\ [in_2]\ [in_3]\ [in_4]\ \underbrace{[\ell_1]\ [r_1]\ [o_1]}_{\text{mult.}}\ \underbrace{[\ell_2]\ [r_2]\ [o_2]}_{\text{mult.}}$ of the "active" clause

$$\boxed{\mathscr{C}_0} \vee \boxed{\mathscr{C}_1} \vee \boxed{\mathscr{C}_2} \vee \boxed{\mathscr{C}_3}$$

$$[\boldsymbol{v}_0] \times (1, \beta, \beta^2, \ldots) \quad [\boldsymbol{v}_1] \times (1, \beta, \beta^2, \ldots) \quad [\boldsymbol{v}_2] \times (1, \beta, \beta^2, \ldots) \quad [\boldsymbol{v}_3] \times (1, \beta, \beta^2, \ldots)$$

$$[e_0] \quad [e_1] \quad [e_2] \quad [e_3]$$



In Robin, this is done by showing $e_0 e_1 e_2 e_3 = 0$, which needs a $\mathcal{O}(B)$ cost.

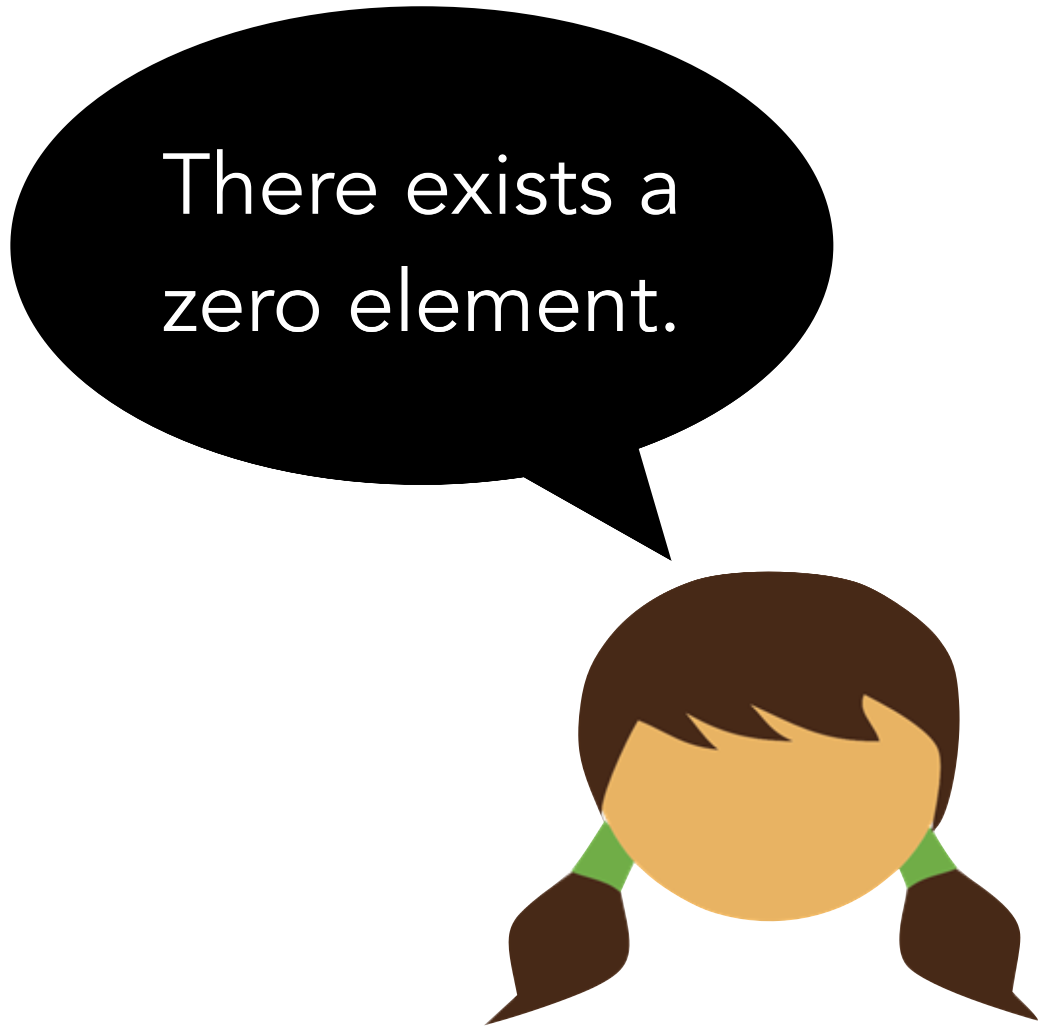
Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

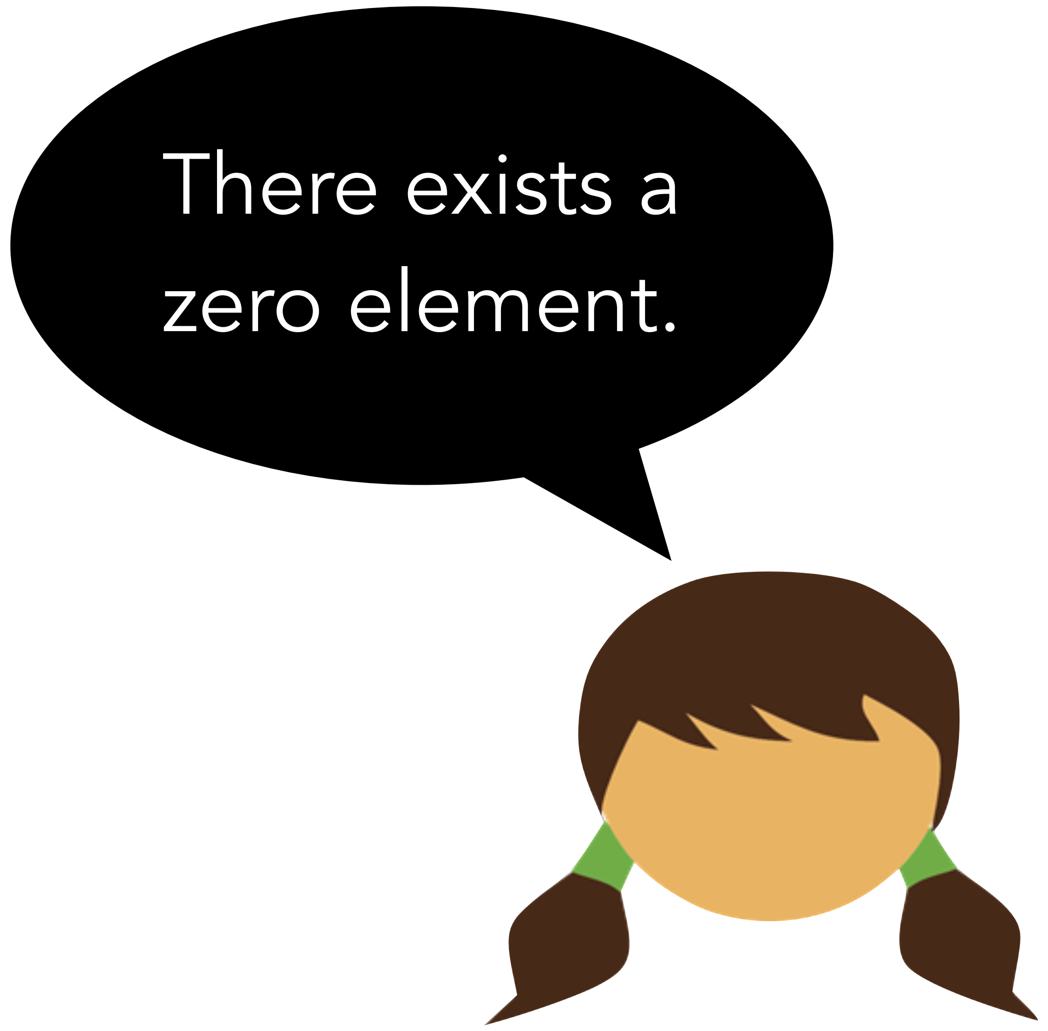
Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

Intuition: P not only knows that a zero exists but also its exact location.

↑

"active" clause index id



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

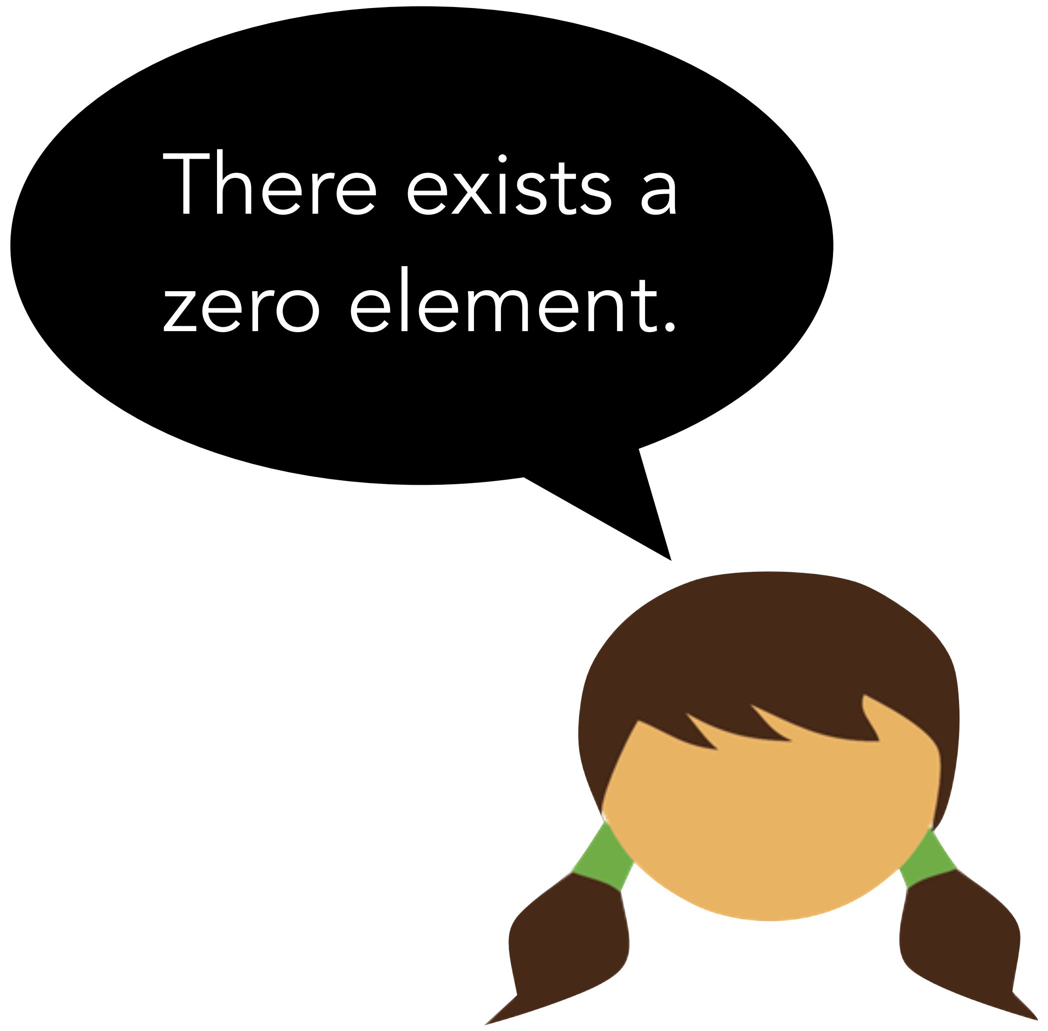
$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

Intuition: P not only knows that a zero exists but also its exact location.

"active" clause index id

$$id = \sum_{i=0}^{\log B-1} id_i \cdot 2^i$$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

Intuition: P not only knows that a zero exists but also its exact location.

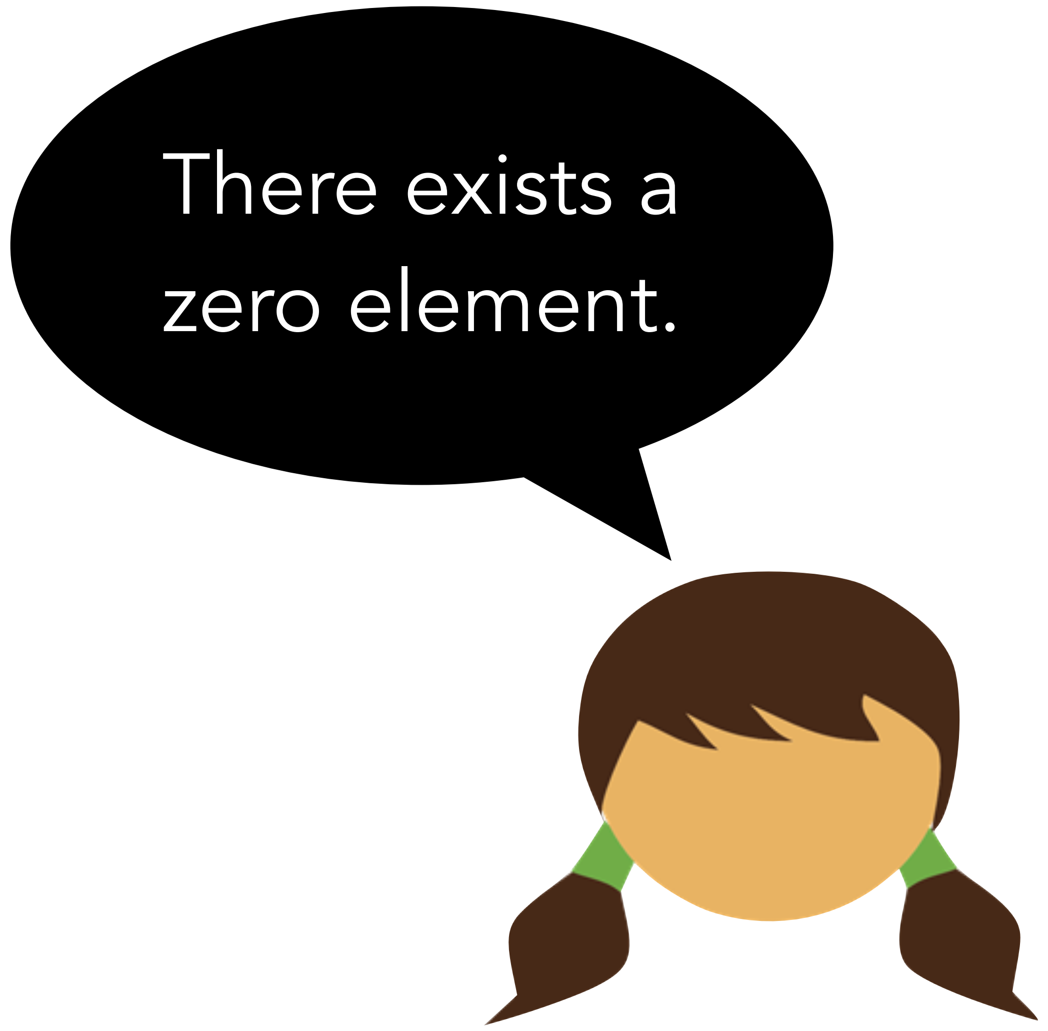
↑

"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

Intuition: P not only knows that a zero exists but also its exact location.

↑

"active" clause index id

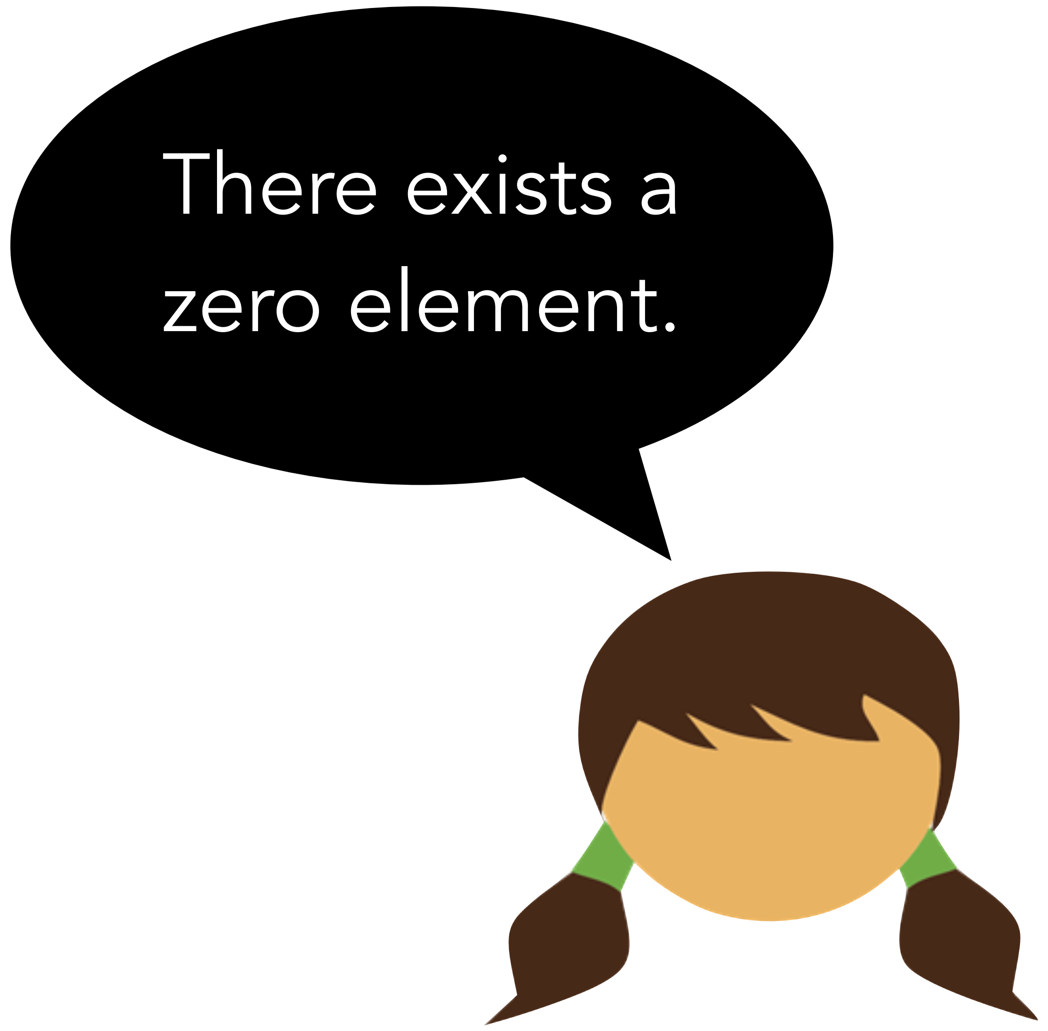
$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

$[\Lambda \cdot (1 - id_0) + \delta_0]$ $[\Lambda \cdot (1 - id_1) + \delta_1]$

$[\Lambda \cdot id_0 - \delta_0]$ $[\Lambda \cdot id_1 - \delta_1]$

$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

Intuition: P not only knows that a zero exists but also its exact location.

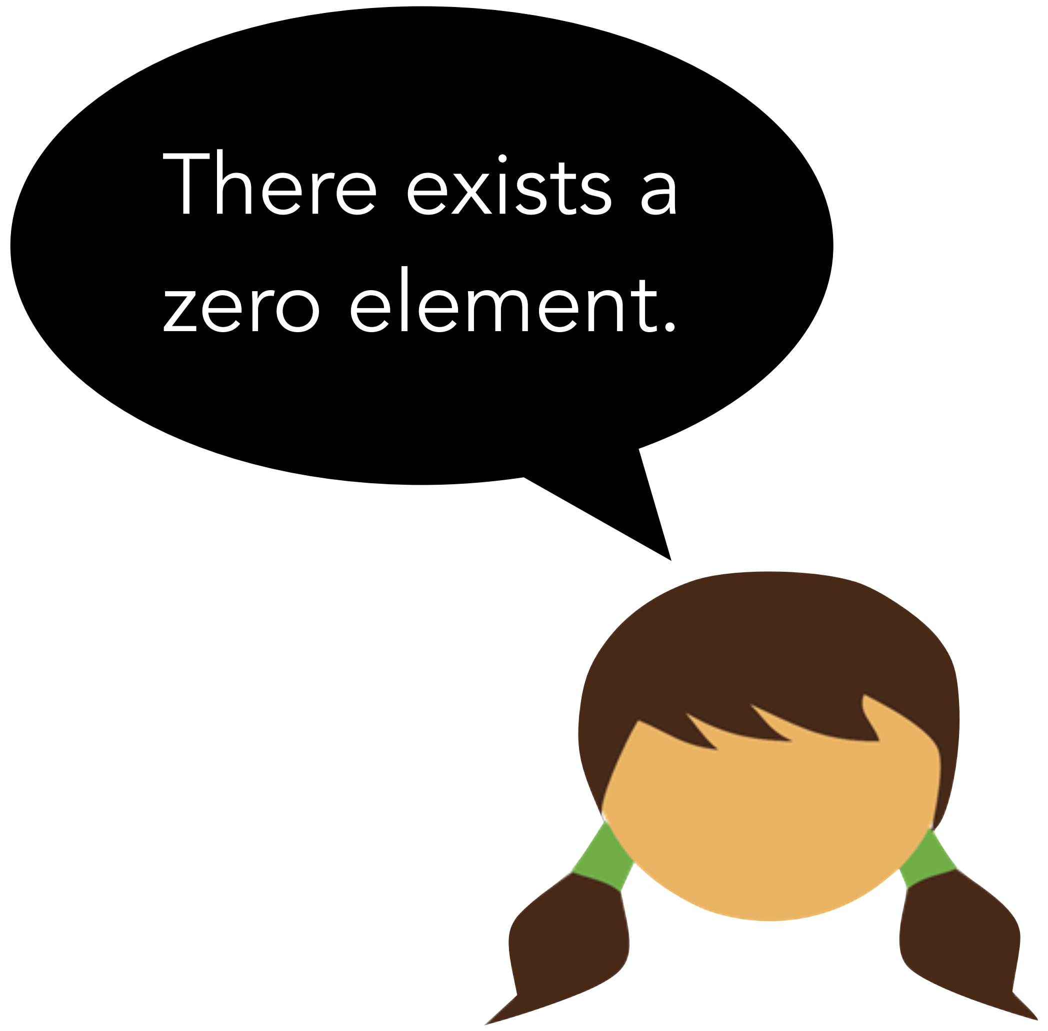
"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

$$M \quad \begin{matrix} \Lambda \cdot (1 - id_0) + \delta_0 & \Lambda \cdot (1 - id_1) + \delta_1 \\ \Lambda \cdot id_0 - \delta_0 & \Lambda \cdot id_1 - \delta_1 \end{matrix}$$

$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

η_0

Intuition: P not only knows that a zero exists but also its exact location.

↑

"active" clause index id

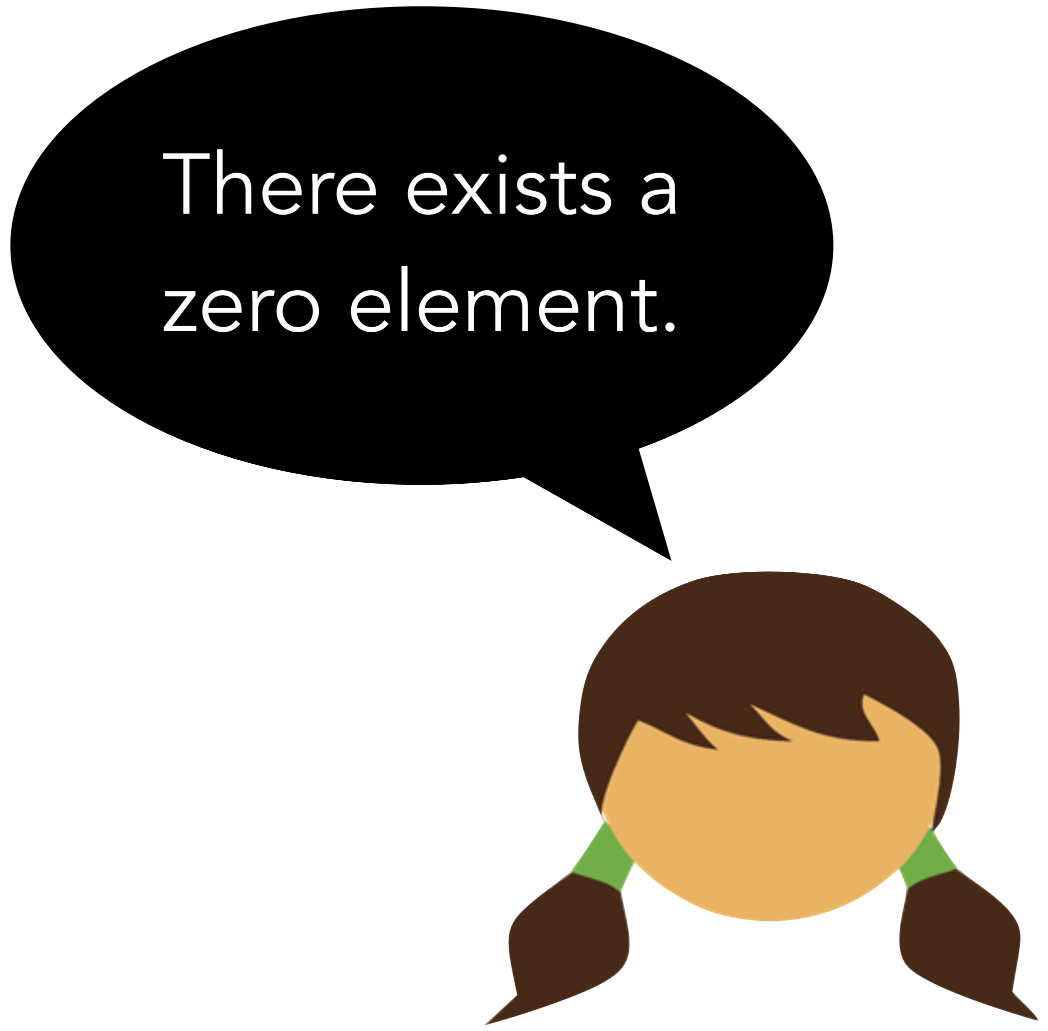
$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

M $\boxed{\Lambda \cdot (1 - id_0) + \delta_0}$ $\boxed{\Lambda \cdot (1 - id_1) + \delta_1}$

$\Lambda \cdot id_0 - \delta_0$ $\Lambda \cdot id_1 - \delta_1$

$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

η_0 η_1

Intuition: P not only knows that a zero exists but also its exact location.

↑

"active" clause index id

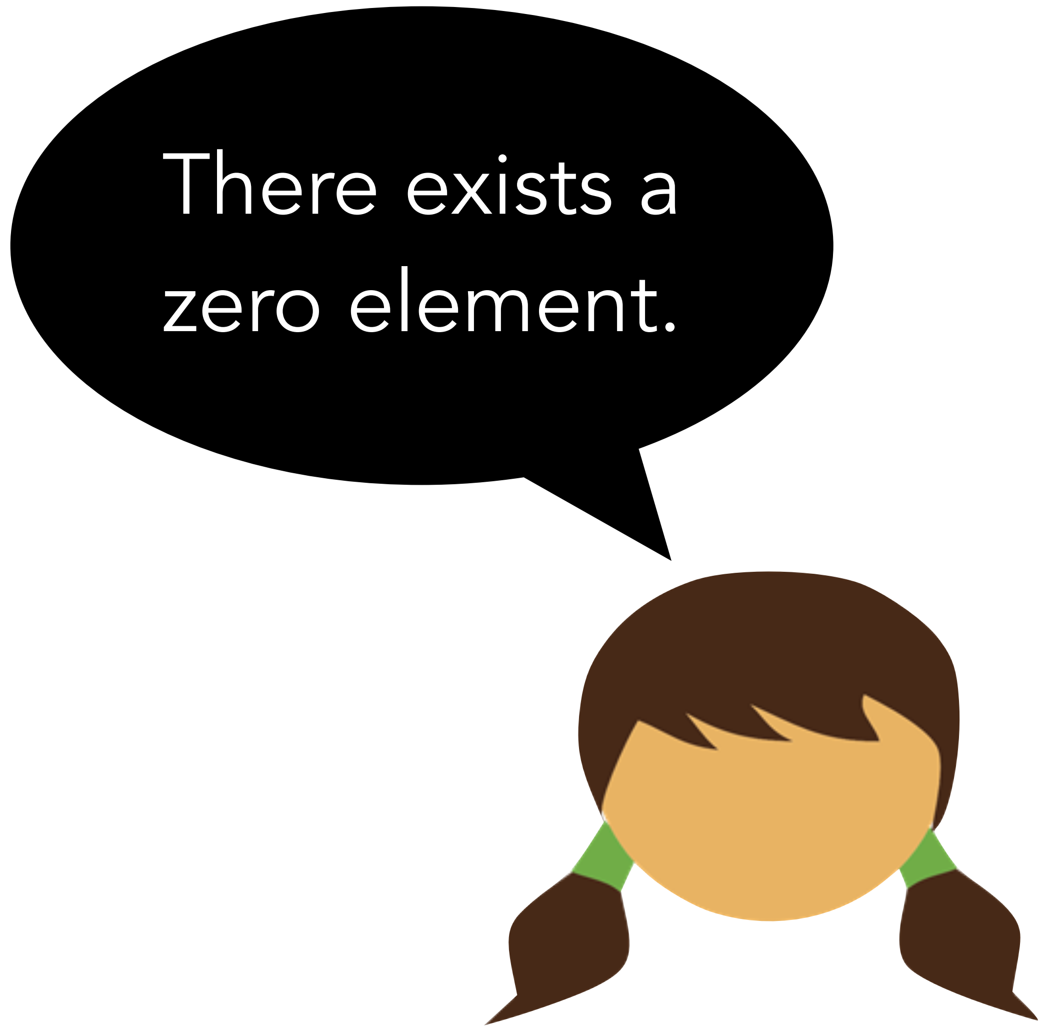
$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

M

$\Lambda \cdot (1 - id_0) + \delta_0$ $\boxed{\Lambda \cdot (1 - id_1) + \delta_1}$

$\boxed{\Lambda \cdot id_0 - \delta_0}$ $\Lambda \cdot id_1 - \delta_1$

$[\delta_0]$ $[\delta_1]$



$[id_0]$ $[id_1]$ Prove in ZK each is a bit

Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$$[e_0] \qquad [e_1] \qquad [e_2] \qquad [e_3]$$

$$\eta_0 \qquad \eta_1 \qquad \eta_2$$

$$M \quad \begin{array}{cc} \boxed{\Lambda \cdot (1 - id_0) + \delta_0} & \Lambda \cdot (1 - id_1) + \delta_1 \\ \Lambda \cdot id_0 - \delta_0 & \boxed{\Lambda \cdot id_1 - \delta_1} \end{array}$$

Intuition: P not only knows that a zero exists but also its exact location.

↑

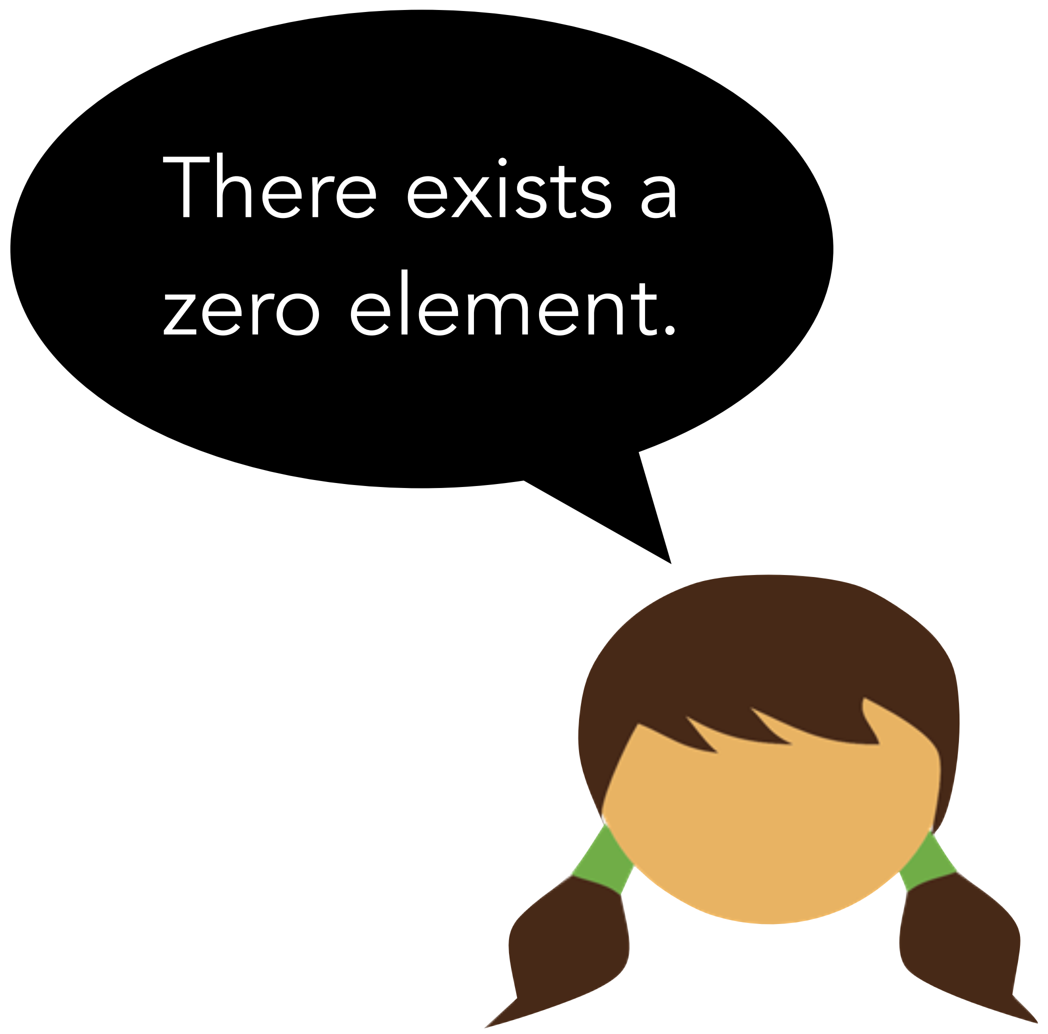
"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

$$[\delta_0] \qquad [\delta_1]$$

$$[id_0] \qquad [id_1]$$

Prove in ZK each is a bit



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

η_0 η_1 η_2 η_3

Intuition: P not only knows that a zero exists but also its exact location.

"active" clause index *id*

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

$$\Lambda \cdot (1 - id_0) + \delta_0 \qquad \Lambda \cdot (1 - id_1) + \delta_1$$

$$M \quad \boxed{\Lambda \cdot id_0 - \delta_0} \quad \boxed{\Lambda \cdot id_1 - \delta_1}$$

A random challenge Λ

There exists a zero element.

$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit

Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$[e_0]$ $[e_1]$ $[e_2]$ $[e_3]$

η_0 η_1 η_2 η_3

$$e_0\eta_0 + e_1\eta_1 + e_2\eta_2 + e_3\eta_3 = f(\Lambda)$$

M

$\Lambda \cdot (1 - id_0) + \delta_0$ $\Lambda \cdot (1 - id_1) + \delta_1$

$\Lambda \cdot id_0 - \delta_0$ $\Lambda \cdot id_1 - \delta_1$

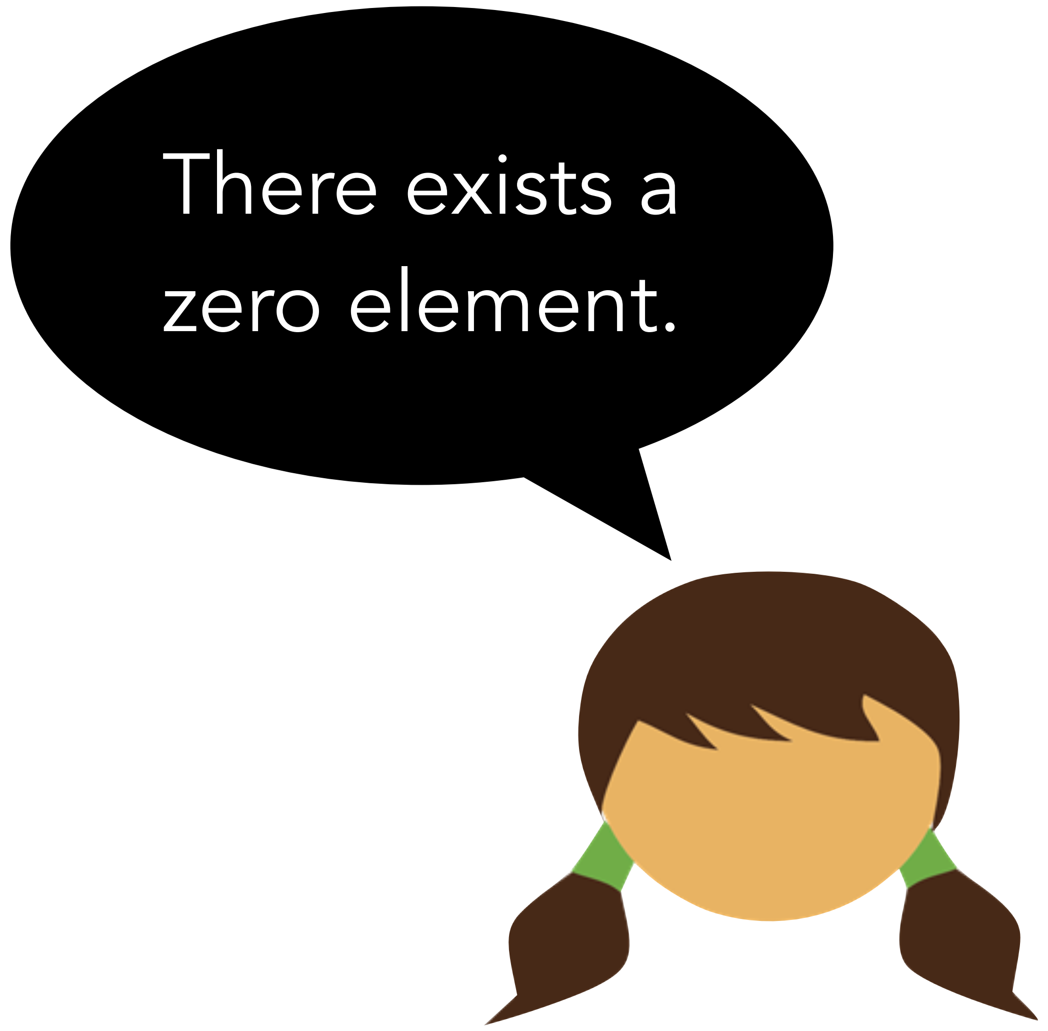
$[\delta_0]$ $[\delta_1]$

$[id_0]$ $[id_1]$ Prove in ZK each is a bit

Intuition: P not only knows that a zero exists but also its exact location.

"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$$[e_0] \qquad [e_1] \qquad [e_2] \qquad [e_3]$$

$$\eta_0 \qquad \eta_1 \qquad \eta_2 \qquad \eta_3$$

$$e_0\eta_0 + e_1\eta_1 + e_2\eta_2 + e_3\eta_3 = f(\Lambda)$$

$$M \quad \begin{matrix} \Lambda \cdot (1 - id_0) + \delta_0 & \Lambda \cdot (1 - id_1) + \delta_1 \\ \Lambda \cdot id_0 - \delta_0 & \Lambda \cdot id_1 - \delta_1 \end{matrix}$$

$$[\delta_0] \qquad [\delta_1]$$

$$[id_0] \qquad [id_1]$$

Prove in ZK each is a bit

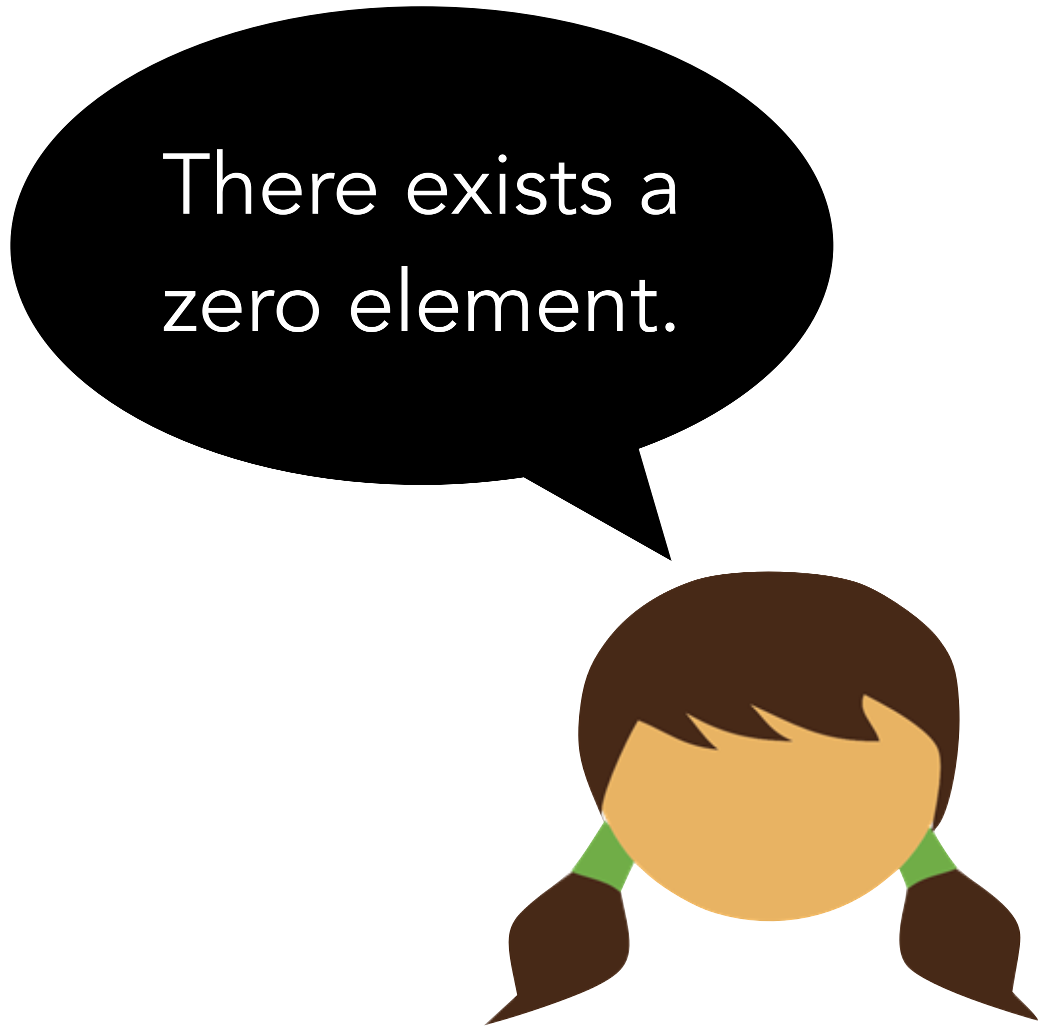
Intuition: P not only knows that a zero exists but also its exact location.

↑

"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

Key Observation: If P is honest, this must be a degree-1 polynomial in Λ. Moreover, P knows all $\mathcal{O}(\log B)$ coefficients before Λ is sampled.



Technique: $\mathcal{O}(\log B)$ Zero Membership Proof

Inspired by [Groth and Kohlweiss, Eurocrypt'15]

$$\begin{array}{cccc} [e_0] & [e_1] & [e_2] & [e_3] \\ \eta_0 & \eta_1 & \eta_2 & \eta_3 \end{array}$$

$$e_0\eta_0 + e_1\eta_1 + e_2\eta_2 + e_3\eta_3 = f(\Lambda)$$

$$M \quad \begin{array}{cc} \Lambda \cdot (1 - id_0) + \delta_0 & \Lambda \cdot (1 - id_1) + \delta_1 \\ \Lambda \cdot id_0 - \delta_0 & \Lambda \cdot id_1 - \delta_1 \end{array}$$

$$\begin{array}{cc} [\delta_0] & [\delta_1] \\ [id_0] & [id_1] \end{array}$$

Intuition: P not only knows that a zero exists but also its exact location.

↑

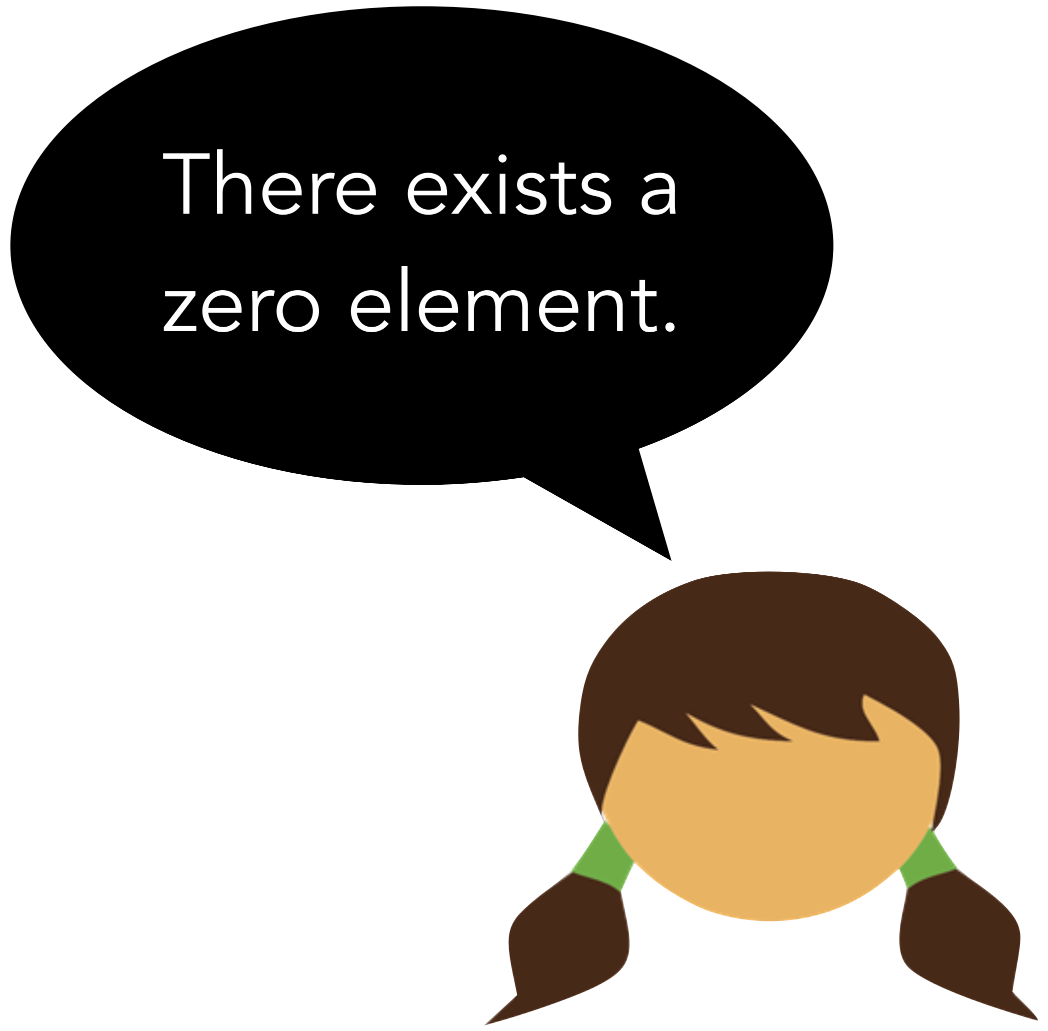
"active" clause index id

$$id = \sum_{i=0}^{\log B - 1} id_i \cdot 2^i$$

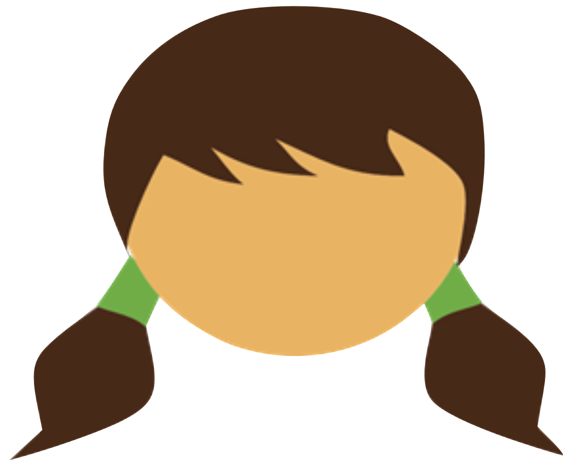
Key Observation: If P is honest, this must be a degree-1 polynomial in Λ. Moreover, P knows all $\mathcal{O}(\log B)$ coefficients before Λ is sampled.

$\Rightarrow$ P can commit to the coefficients initially and show two different ways to evaluate the same $f(\Lambda)$!

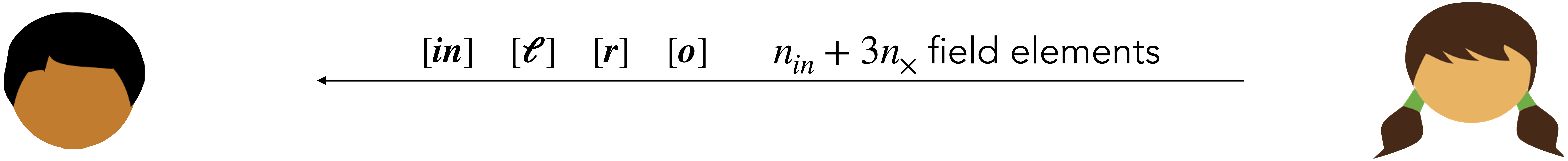
Prove in ZK each is a bit



LogRobin: Full Diagram

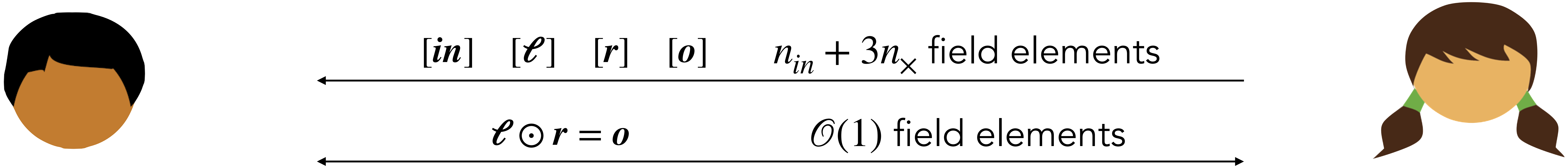


LogRobin: Full Diagram



$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

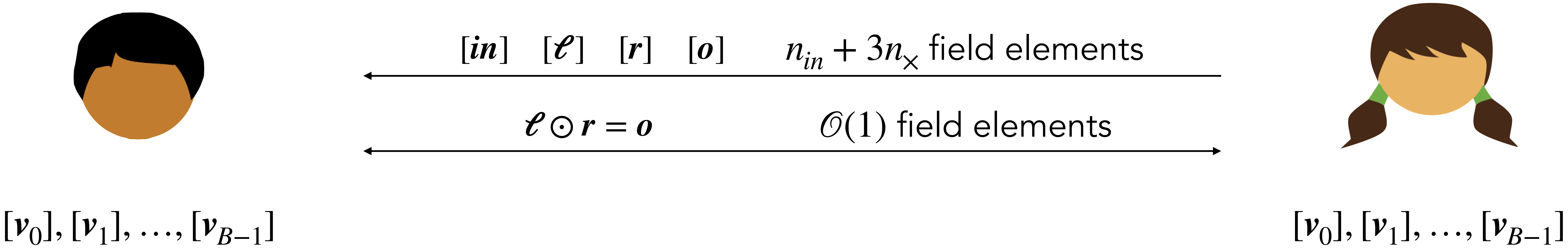
LogRobin: Full Diagram



$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

LogRobin: Full Diagram



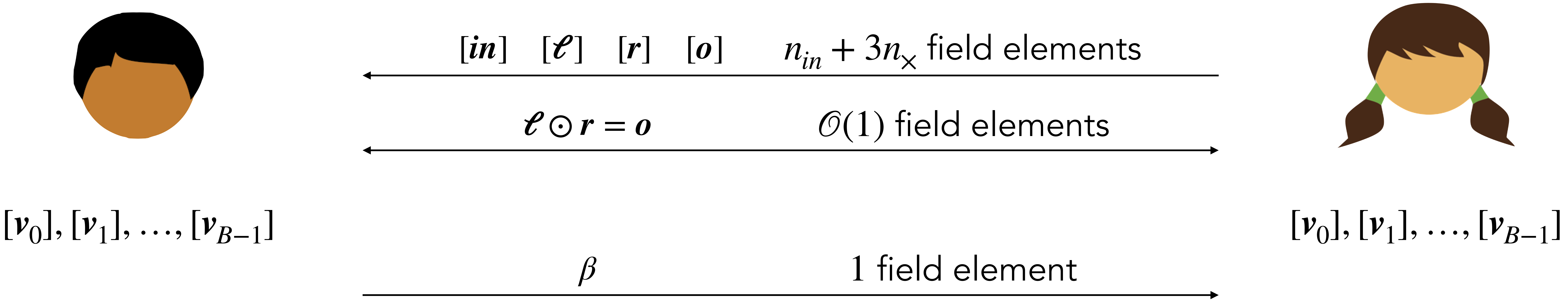
$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

LogRobin: Full Diagram



$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_\times$ field elements

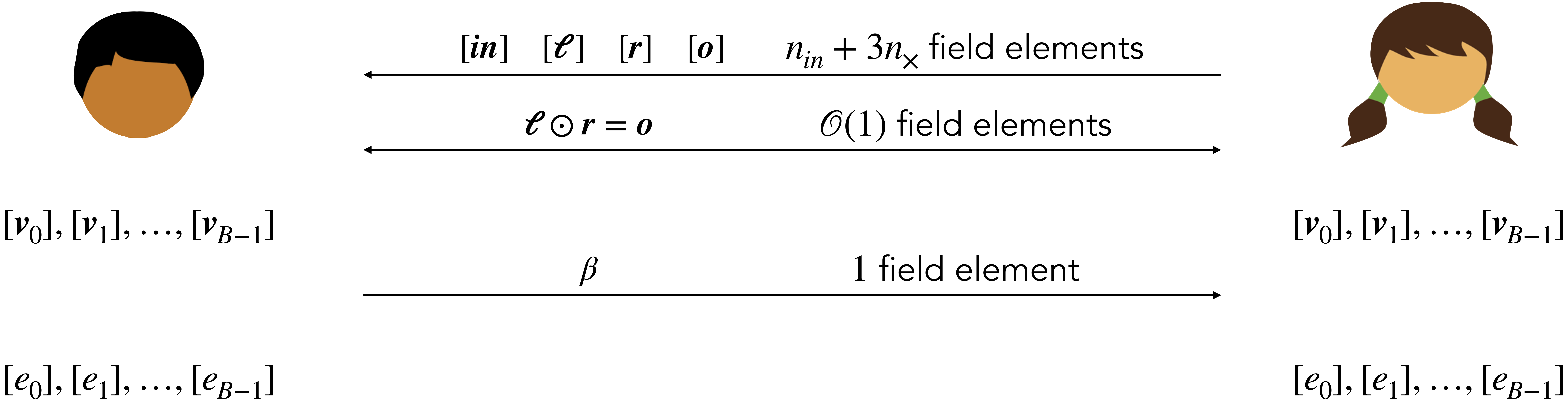
$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

LogRobin: Full Diagram



$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

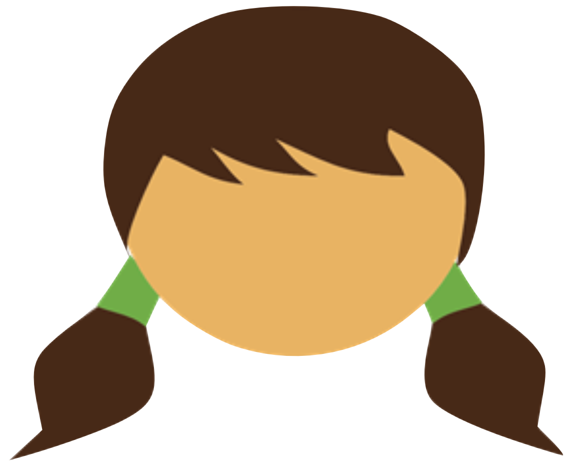
$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[e_0], [e_1], \ldots, [e_{B-1}]$

LogRobin: Full Diagram



$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_\times$ field elements

$\longleftarrow$

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$\longleftrightarrow$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$\longrightarrow$

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$[id_0], [id_1], \ldots, [id_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[\boldsymbol{e}_0], [\boldsymbol{e}_1], \ldots, [\boldsymbol{e}_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$\boldsymbol{id} \odot (\boldsymbol{id} - \boldsymbol{1}) = \boldsymbol{1}$ $\mathcal{O}(1)$ field elements

$[id_0], [id_1], \ldots, [id_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements

$[\boldsymbol{e}_0], [\boldsymbol{e}_1], \ldots, [\boldsymbol{e}_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$\boldsymbol{id} \odot (\boldsymbol{id} - \boldsymbol{1}) = \boldsymbol{1}$ $\mathcal{O}(1)$ field elements

$[id_0], [id_1], \ldots, [id_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

Λ 1 field element

M $\mathcal{O}(\log B)$ field elements

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements ←

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements ↔

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$ $[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element →

$[e_0], [e_1], \ldots, [e_{B-1}]$ $\boldsymbol{id} \odot (\boldsymbol{id} - \mathbf{1}) = \mathbf{1}$ $\mathcal{O}(1)$ field elements $[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$ $[id_0], [id_1], \ldots, [id_{\log B-1}]$ $[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements ←

$\sum_{i=0}^{B-1} \eta_i [e_i]$ $\sum_{i=0}^{B-1} \eta_i [e_i]$

Λ 1 field element →

M $\mathcal{O}(\log B)$ field elements ←

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$ $[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[e_0], [e_1], \ldots, [e_{B-1}]$ $\boldsymbol{id} \odot (\boldsymbol{id} - \boldsymbol{1}) = \boldsymbol{1}$ $\mathcal{O}(1)$ field elements $[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$ $[id_0], [id_1], \ldots, [id_{\log B-1}]$ $[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements

$\sum_{i=0}^{B-1} \eta_i [e_i]$ $\sum_{i=0}^{B-1} \eta_i [e_i]$

Λ 1 field element

M $\mathcal{O}(\log B)$ field elements

$\sum_{i=0}^{B-1} \Lambda^i [c_i]$ $\sum_{i=0}^{B-1} \Lambda^i [c_i]$

LogRobin: Full Diagram

$[\boldsymbol{in}]$ $[\boldsymbol{\ell}]$ $[\boldsymbol{r}]$ $[\boldsymbol{o}]$ $n_{in} + 3n_{\times}$ field elements

$\boldsymbol{\ell} \odot \boldsymbol{r} = \boldsymbol{o}$ $\mathcal{O}(1)$ field elements

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

$[\boldsymbol{v}_0], [\boldsymbol{v}_1], \ldots, [\boldsymbol{v}_{B-1}]$

β 1 field element

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

$\boldsymbol{id} \odot (\boldsymbol{id} - \boldsymbol{1}) = \boldsymbol{1}$ $\mathcal{O}(1)$ field elements

$[id_0], [id_1], \ldots, [id_{\log B-1}]$

$[c_0], [c_1], \ldots, [c_{\log B-1}]$ $\mathcal{O}(\log B)$ field elements

$[e_0], [e_1], \ldots, [e_{B-1}]$

$[\delta_0], [\delta_1], \ldots, [\delta_{\log B-1}]$

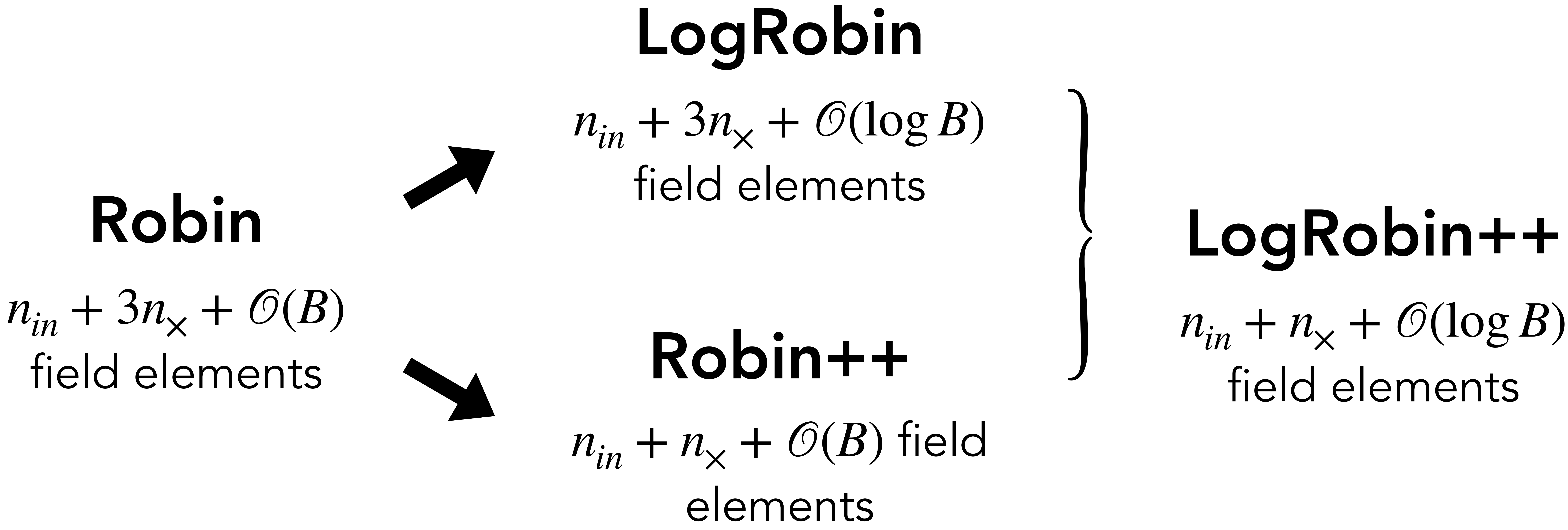
$$\sum_{i=0}^{B-1} \eta_i[e_i] = \sum_{i=0}^{B-1} \Lambda^i[c_i]$$

Λ 1 field element

M $\mathcal{O}(\log B)$ field elements

$$\sum_{i=0}^{B-1} \eta_i[e_i] = \sum_{i=0}^{B-1} \Lambda^i[c_i]$$

Summary of Our Results



Robin

$n_{in} + 3n_{\times} + \mathcal{O}(B)$
field elements

LogRobin

$n_{in} + 3n_{\times} + \mathcal{O}(\log B)$
field elements

Robin++

$n_{in} + n_{\times} + \mathcal{O}(B)$ field
elements

LogRobin++

$n_{in} + n_{\times} + \mathcal{O}(\log B)$
field elements

Q/A

ePrint

GitHub

Email: yyang811@gatech.edu