Counting Vampires: From Univariate Sumcheck to Updatable ZK-SNARK

Helger Lipmaa
simula

Janno Siim
simula

Michal Zajac
NETHERMIND

About this talk

Vampire

communication-efficient zkSNARK with updatable setup

How zkSNARKs are **built**?

What **trade-offs** are available?

Vampire

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bits	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bits	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bits	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bits	non-updatable	non-universal

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bits	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bits	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bits	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bits	non-updatable	non-universal

How?

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bits	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bits	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bits	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bits	non-updatable	non-universal

How?

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bits	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bits	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bits	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bits	non-updatable	non-universal

How?

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument **Count**
$-1\mathbb{G}$ vs Aurora sumcheck

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bits

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bits	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bits	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bits	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bits	non-updatable	non-universal

How?

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

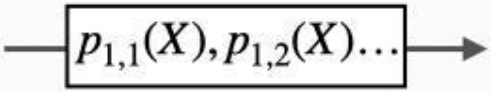
More **efficient** batching

How zkSNARKs are built?

$R(x, w)$

How zkSNARKs are built?

$R(x, w)$



$\mathcal{P}$

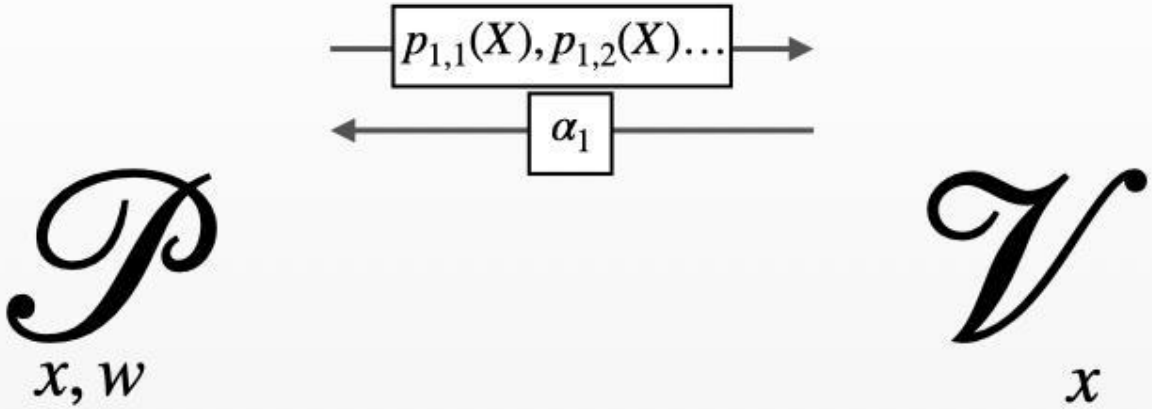
x, w

$\mathcal{V}$

x

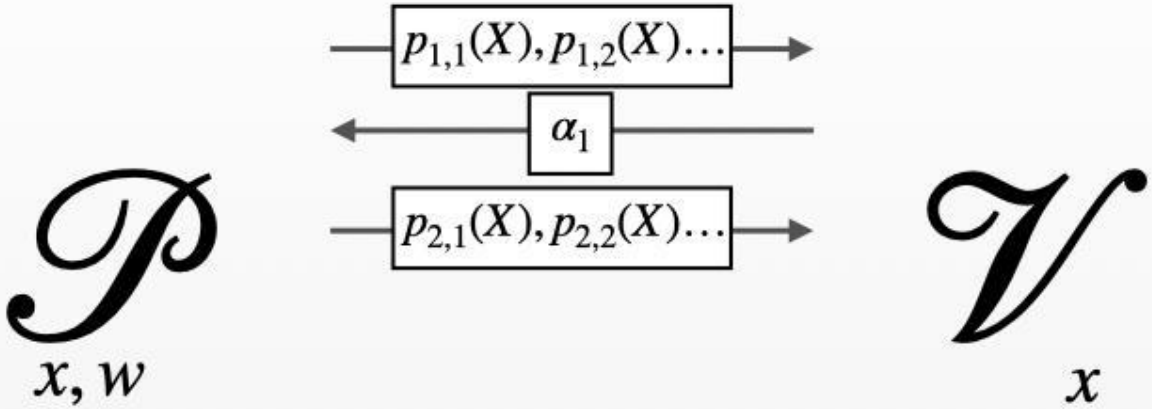
How zkSNARKs are built?

$R(x, w)$



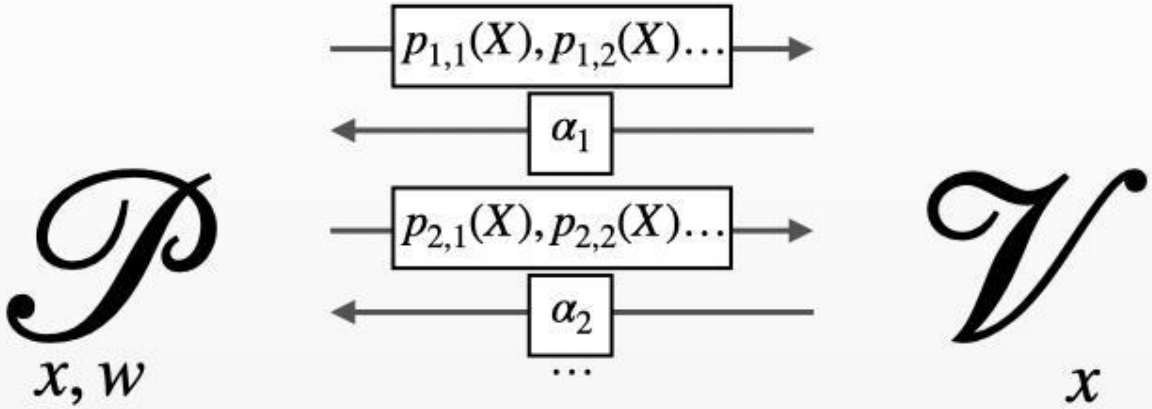
How zkSNARKs are built?

$R(x, w)$



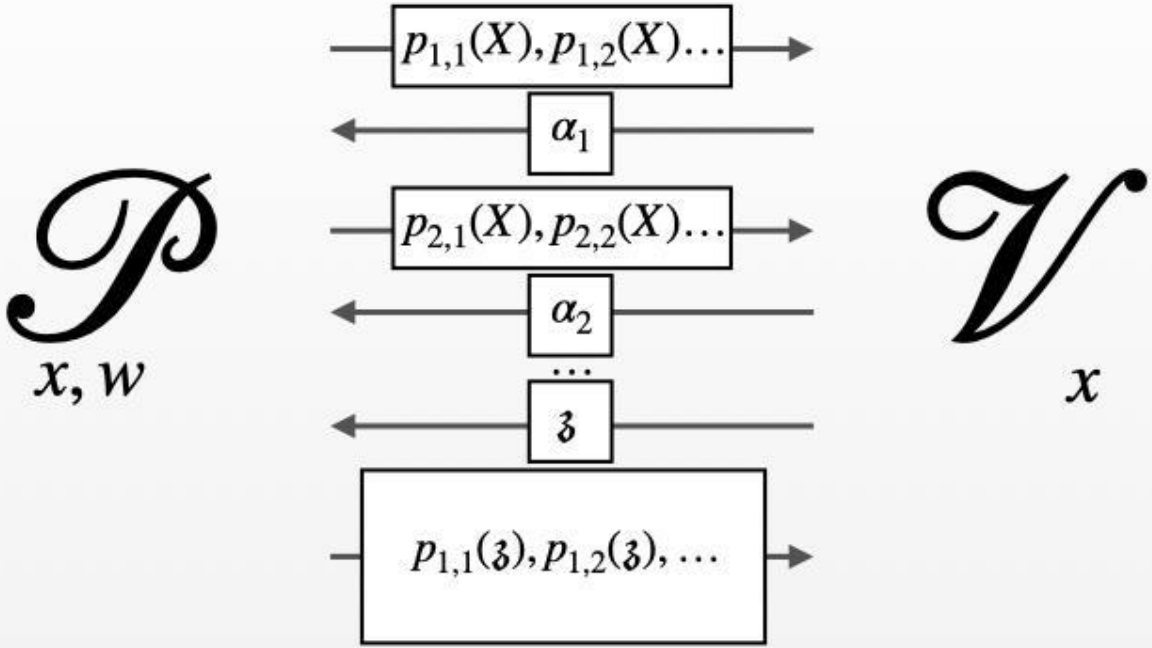
How zkSNARKs are built?

$R(x, w)$



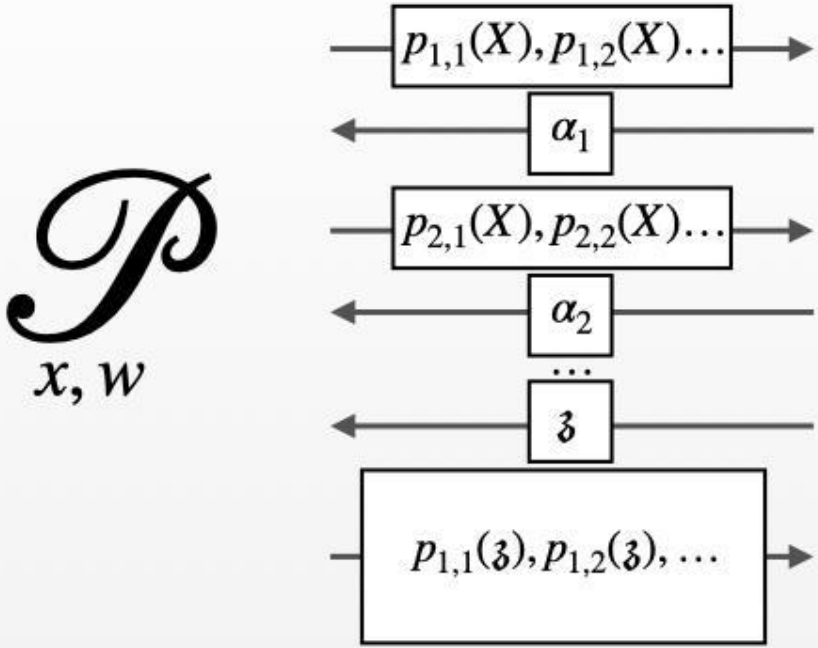
How zkSNARKs are built?

$R(x, w)$



How zkSNARKs are built?

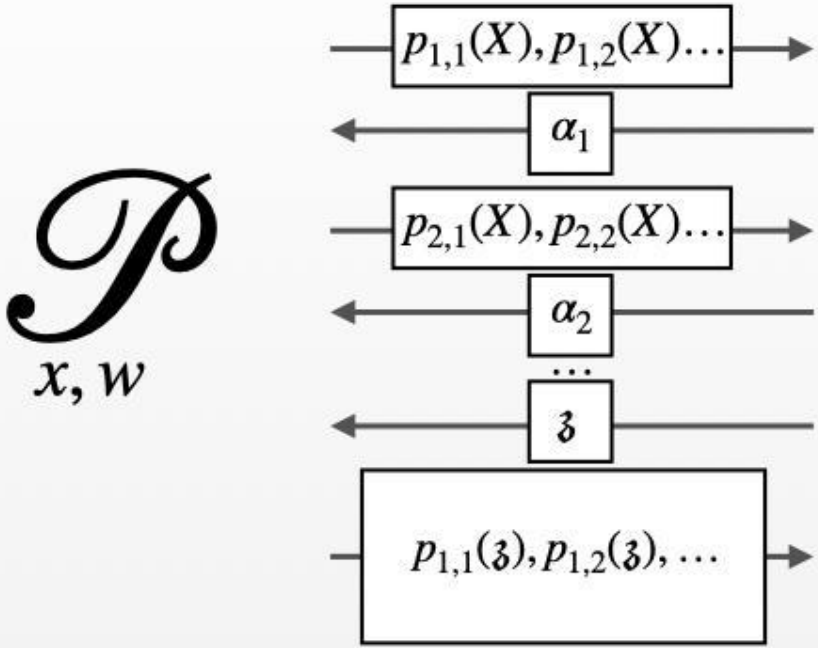
$R(x, w)$



$F(z, p_{1,1}(z), \ldots) \stackrel{?}{=} 0$

How zkSNARKs are built?

$R(x, w)$



(Idealized model)

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

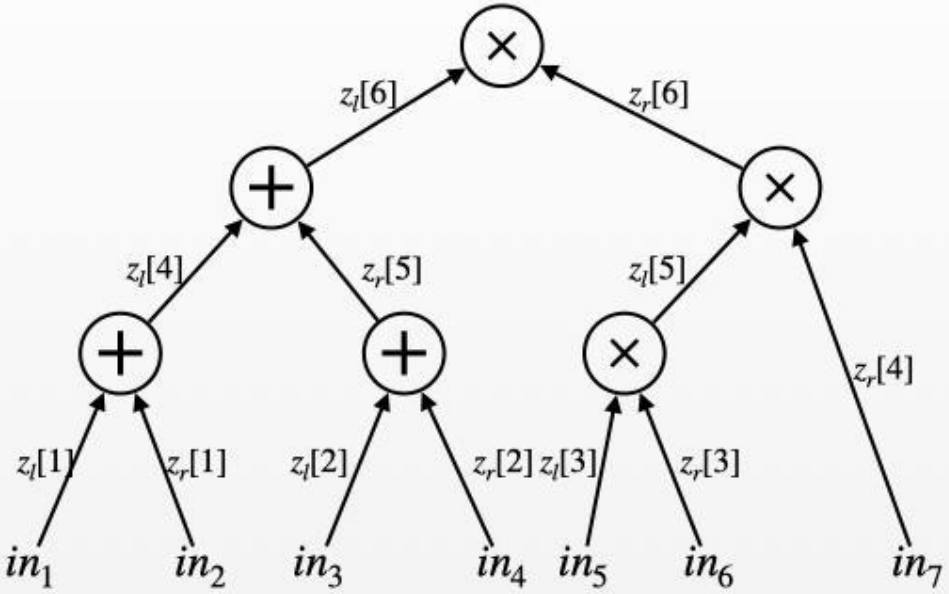
STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

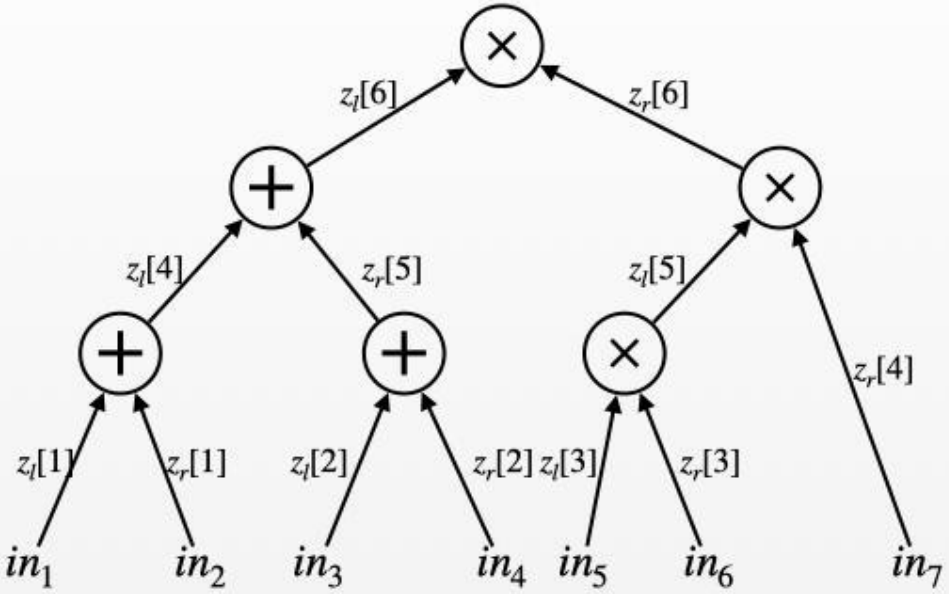
STEP 3

More **efficient** batching

RiCS Lite

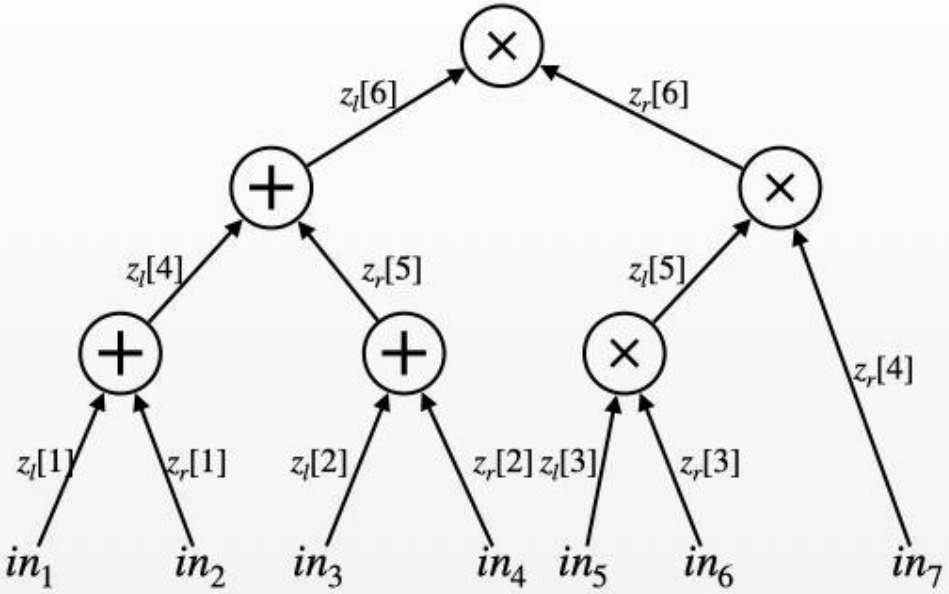

$z_l[6]$
$z_r[6]$
$z_l[4]$
$z_r[5]$
$z_l[5]$
$z_r[4]$
$z_l[1]$
$z_r[1]$
$z_l[2]$
$z_r[2]$
$z_l[3]$
$z_r[3]$
in_1
in_2
in_3
in_4
in_5
in_6
in_7

RiCS Lite



$$W = \begin{bmatrix} I & 0 & L \\ 0 & I & R \\ 0 & 0 & 0 \end{bmatrix}$$

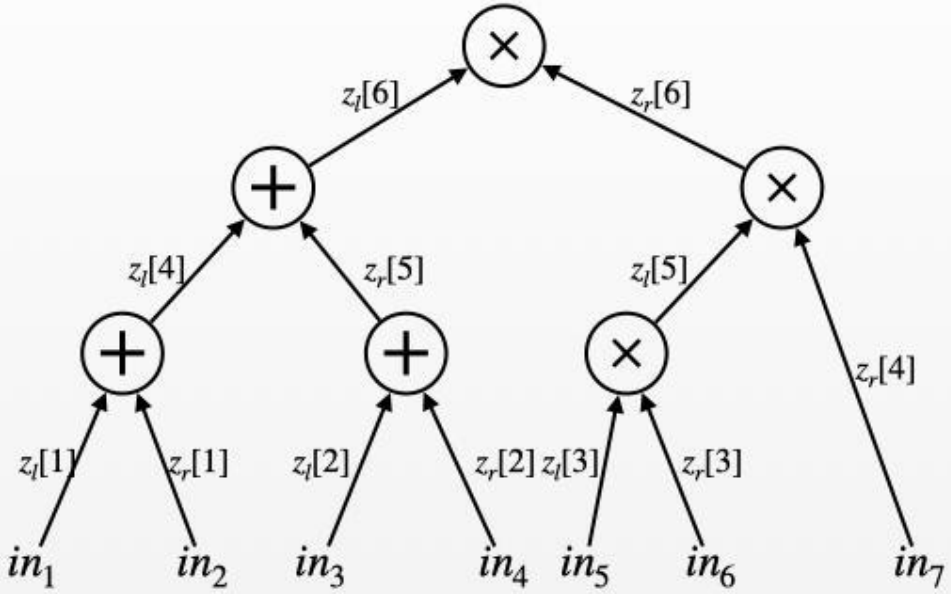
RiCS Lite



$$W = \begin{bmatrix} I & 0 & L \\ 0 & I & R \\ 0 & 0 & 0 \end{bmatrix}$$

$$z = \begin{bmatrix} z_l \\ z_R \\ z_l \circ z_R \end{bmatrix}$$

RiCS Lite

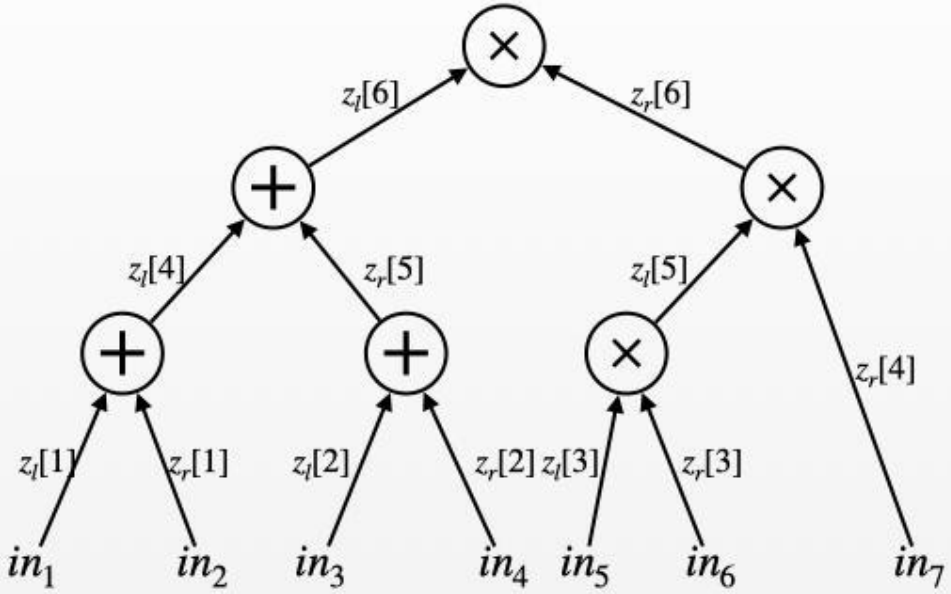


$$W = \begin{bmatrix} I & 0 & L \\ 0 & I & R \\ 0 & 0 & 0 \end{bmatrix}$$

$$z = \begin{bmatrix} z_l \\ z_R \\ z_l \circ z_R \end{bmatrix}$$

$$Wz = 0$$

RiCS Lite



$$W = \begin{bmatrix} I & 0 & L \\ 0 & I & R \\ 0 & 0 & 0 \end{bmatrix}$$

$$z = \begin{bmatrix} z_l \\ z_R \\ z_l \circ z_R \end{bmatrix}$$

$$Wz = 0$$

RICS vs RICSLite

RiCS vs RiCSLite

$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

RICS vs RICSLite

$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$

RiCS vs RiCSLite

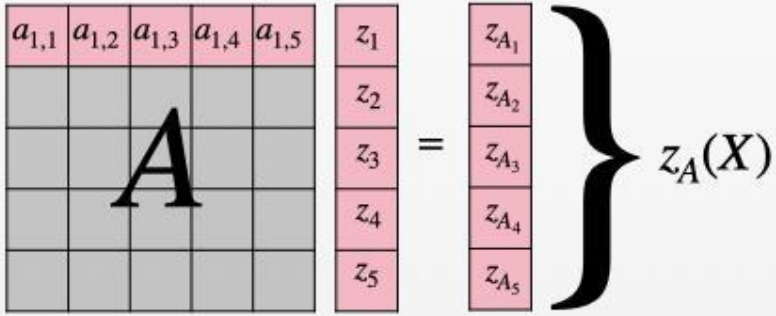
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



RiCS vs RiCSLite

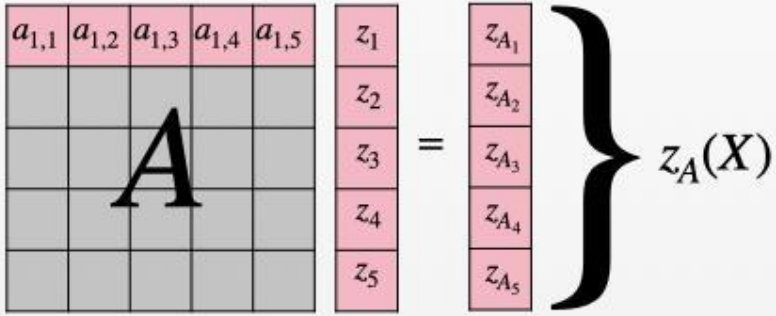
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



$$z(X), z_A(X), z_B(X), z_C(X)$$

RiCS vs RiCSLite

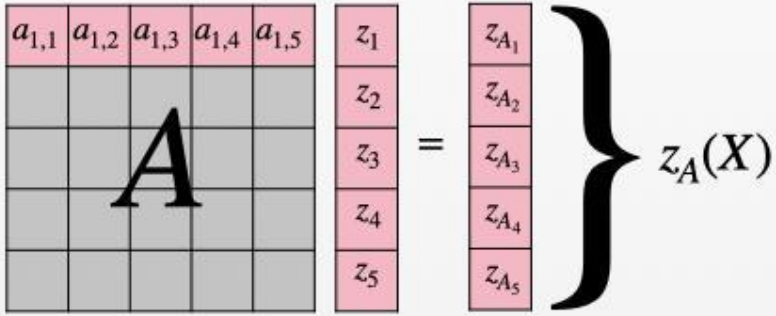
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^{n}$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



$$z(X), z_A(X), z_B(X), z_C(X)$$

$W \in \mathbb{F}^{3n \times 3n}$

$z \in \mathbb{F}^{3n}$

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

RiCS vs RiCSLite

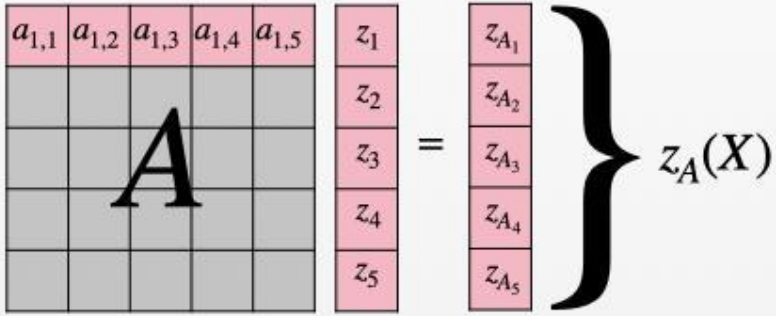
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



$$z(X), z_A(X), z_B(X), z_C(X)$$

$W \in \mathbb{F}^{3n \times 3n}$

$z \in \mathbb{F}^{3n}$

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$$Wz = 0$$

RiCS vs RiCSLite

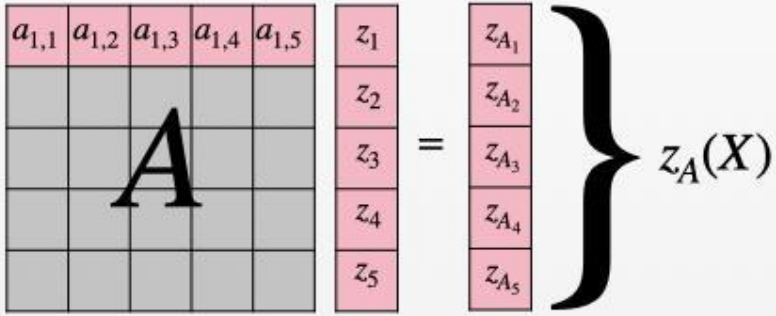
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^{n}$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



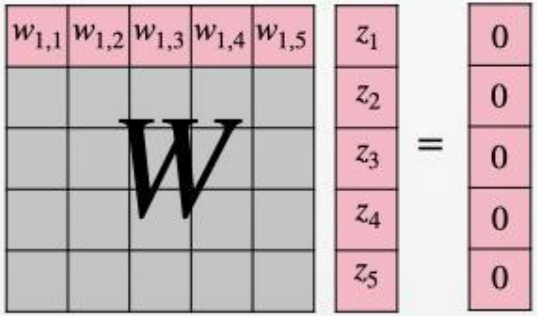
$$z(X), z_A(X), z_B(X), z_C(X)$$

$W \in \mathbb{F}^{3n \times 3n}$

$z \in \mathbb{F}^{3n}$

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$$Wz = 0$$



RiCS vs RiCSLite

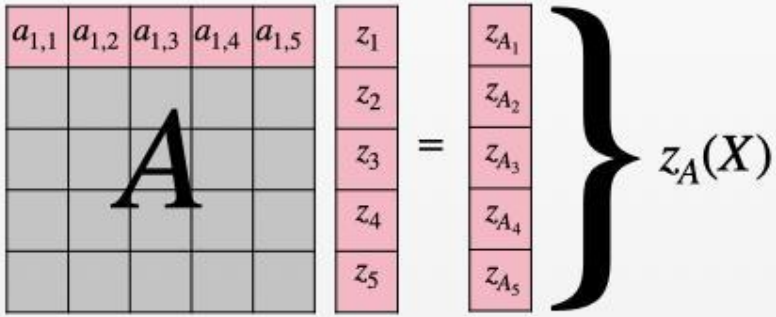
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^n$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



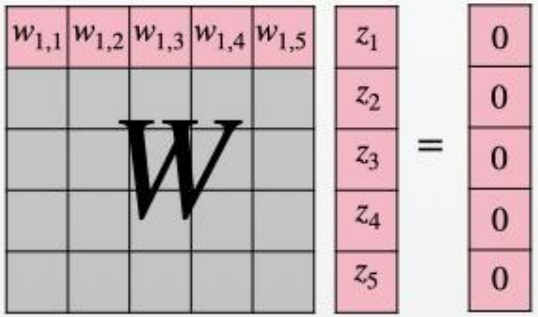
$$z(X), z_A(X), z_B(X), z_C(X)$$

$W \in \mathbb{F}^{3n \times 3n}$

$z \in \mathbb{F}^{3n}$

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

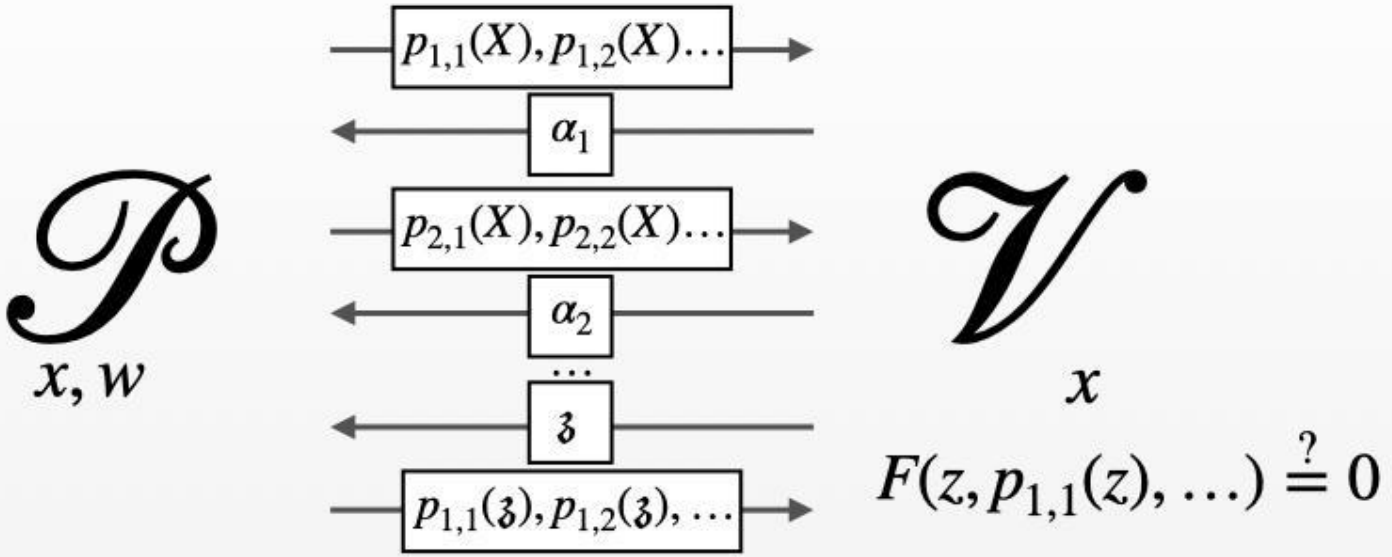
$$Wz = 0$$



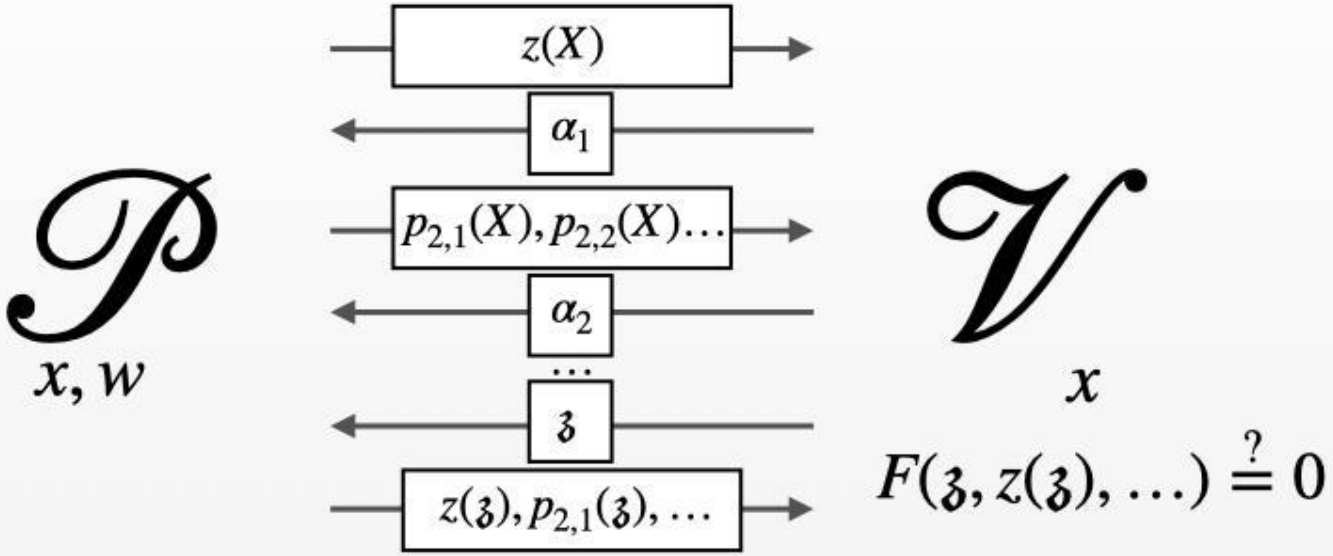
$$z(X)$$

complexity depends on the number of **non-zero** entries

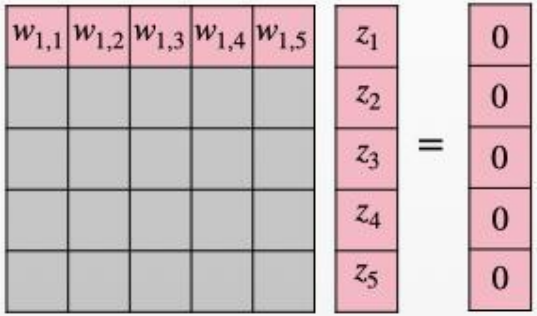
How zkSNARKs are built?



How zkSNARKs are built?

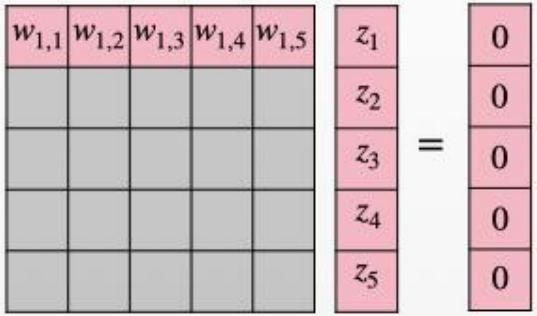


From matrices to polynomials



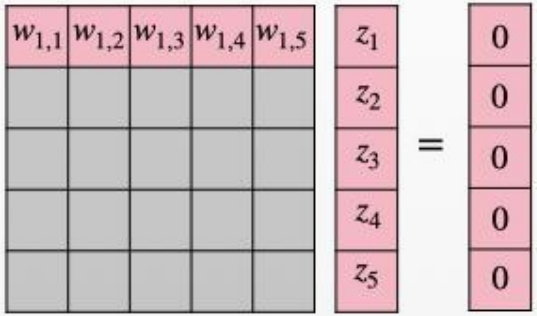
From matrices to polynomials

$$\forall x \in H : \sum_{y \in H} w_{x,y} \cdot z_y = 0$$



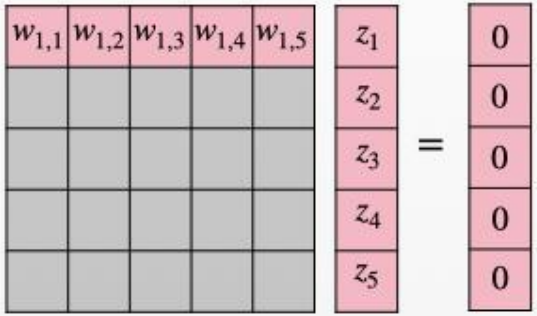
From matrices to polynomials

$$\forall x \in H : \sum_{y \in H} w_{x,y} \cdot z_y = 0$$



$$\forall x \in H : P(x) = \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

From matrices to polynomials

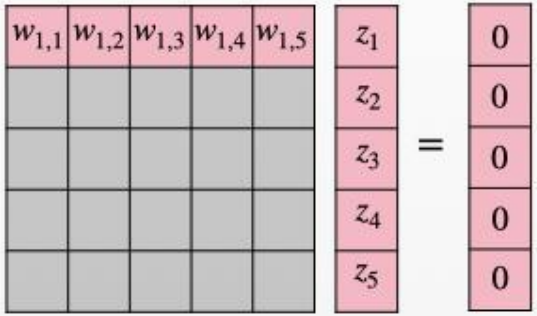


$$\forall x \in H : \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w(x,y) \cdot z(y) = 0$$

From matrices to polynomials



$$\forall x \in H : \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w(x,y) \cdot z(y) = 0$$

$$P(X) = \sum_{y \in H} w(X,y) \cdot z(y) = 0$$

$$\deg_X P(X) = n - 1$$

$$|H| = n$$

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

More **efficient** batching

Sumcheck argument

Sumcheck argument

Input: polynomial P, set H, the evaluation value v

Sumcheck argument

Input: polynomial P, set H, the evaluation value v

$$H = \{\omega, \omega^2, \ldots, \omega^n\}$$

$$P(\omega) + P(\omega^2) + \ldots + P(\omega^n) = v$$

$\mathcal{P}$

$P(X), v, H$

$\mathcal{V}$

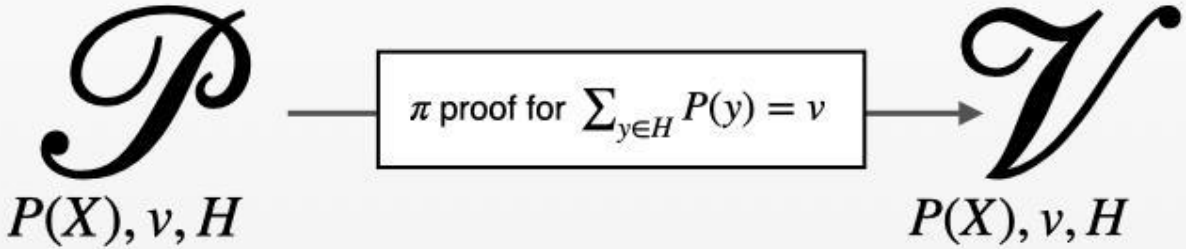
$P(X), v, H$

Sumcheck argument

Input: polynomial P, set H, the evaluation value v

$$H = \{\omega, \omega^2, \ldots, \omega^n\}$$

$$P(\omega) + P(\omega^2) + \ldots + P(\omega^n) = v$$



From bivariate polynomials to sumcheck

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

$$\Psi(X, Y) := w(X, Y) \cdot z(Y)$$

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

$$\Psi(X, Y) := w(X, Y) \cdot z(Y)$$
$$\Psi_\alpha(Y) = w(\alpha, Y) \cdot z(Y)$$

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

$$\Psi(X, Y) := w(X, Y) \cdot z(Y)$$
$$\Psi_\alpha(Y) = w(\alpha, Y) \cdot z(Y)$$

$$\sum_{y \in H} \Psi_\alpha(y) = 0$$

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

$$\Psi(X, Y) := w(X, Y) \cdot z(Y)$$
$$\Psi_\alpha(Y) = w(\alpha, Y) \cdot z(Y)$$

$$\sum_{y \in H} \Psi_\alpha(y) = 0$$

Compute $R(X), Q(X)$ such that $\Psi_\alpha(X) = X \cdot R(X) + Q(X)Z(X)$

Count

KZG commitment scheme

$e : \mathbb{G} \times \mathbb{H} \rightarrow \mathbb{T}$

$SRS = \{g, g^s, \ldots, g^{s^k}, h\}$

$f(X) = f_0 + f_1 X + \ldots f_k X^k$

$\texttt{com}(f(X)) = g^{f_0} \cdot (g^s)^{f_1} \cdot \ldots \cdot (g^{s^k})^{f_k}$

Count

KZG commitment scheme

$e : \mathbb{G} \times \mathbb{H} \to \mathbb{T}$

$SRS = \{g, g^s, \ldots, g^{s^k}, h\}$

$$f(X) = f_0 + f_1 X + \ldots f_k X^k$$

$$\texttt{com}(f(X)) = g^{f_0} \cdot (g^s)^{f_1} \cdot \ldots \cdot (g^{s^k})^{f_k}$$

Assumption

s secret $\Longrightarrow$ It is infeasible to compute g^{s^n} for $n \notin \{1, \ldots, k\}$

Count

KZG commitment scheme

$e : \mathbb{G} \times \mathbb{H} \to \mathbb{T}$

$SRS = \{g, g^s, \ldots, g^{s^k}, h\}$

$$f(X) = f_0 + f_1 X + \ldots f_k X^k$$

$$\texttt{com}(f(X)) = g^{f_0} \cdot (g^s)^{f_1} \cdot \ldots \cdot (g^{s^k})^{f_k}$$

Assumption

s secret $\Longrightarrow$ It is infeasible to compute g^{s^n} for $n \notin \{1, \ldots, k\}$

Observation

If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Count

$$SRS = \left\{ \begin{matrix} g, g^{s}, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation

If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

Count

$$SRS = \left\{ \begin{matrix} g, g^{s}, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

$$\deg(f) \leq gap - 1 \leq |H|$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

$$\deg(f) \leq gap - 1 \leq |H|$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

$$\deg(f) \leq gap - 1 \leq |H|$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = f_0 X^{gap} - \frac{v}{|H|} X^{gap} + f_1 X^{gap+1} + \ldots$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

$$\deg(f) \leq gap - 1 \leq |H|$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = \underbrace{f_0 X^{gap} - \frac{v}{|H|} X^{gap}}_{0} + f_1 X^{gap+1} + \ldots$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact

Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$

$\mathcal{P}$

$f(X)$

$\mathcal{V}$

$g^{f(s)}$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$

$\mathcal{P}$

$f(X)$

$\mathcal{V}$

$g^{f(s)}$

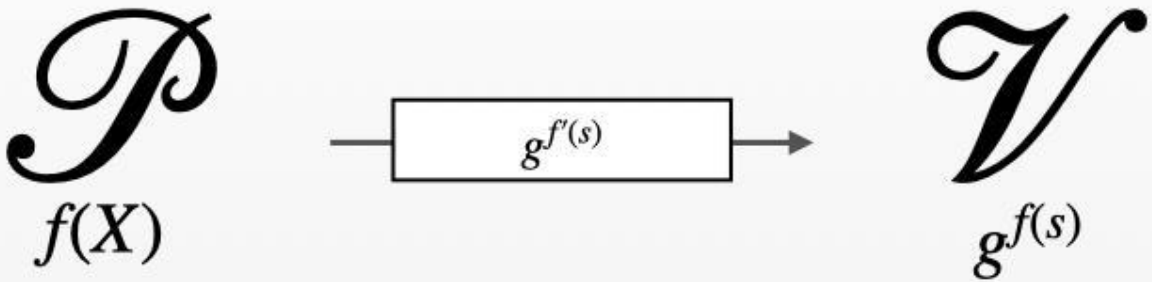
$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$



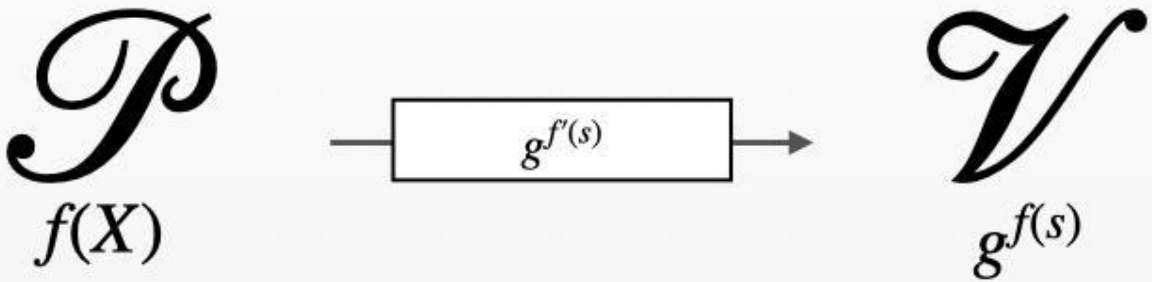
$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$



$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

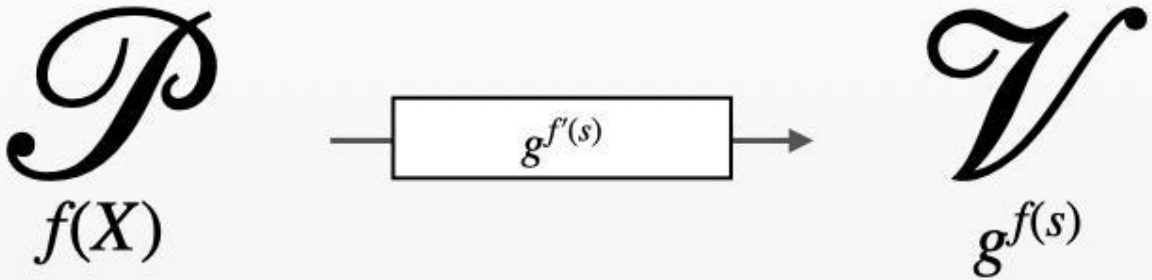
$$f(X) \cdot X^{gap} - f'(X) = \frac{v}{|H|} X^{gap}$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$, $|H| < gap$ then

$$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$$



$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f(X) \cdot X^{gap} - f'(X) = \frac{v}{|H|} X^{gap}$$

$$g^{f(s)} \bullet h^{s^{gap}} - g^{f'(s)} \bullet h = \frac{v}{|H|} g \bullet h^{s^{gap}}$$

$$\deg(f) \leq gap - 1 \leq |H|$$

Fact 1

Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot f_0 = v$

$$\deg(f) \leq gap - 1 \leq |H|$$

Fact 1

Let $f(X) \in \mathbb{F}_{\leq |H|}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot f_0 = v$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

$$\sum_{y \in H} f(y) = \sum_{y \in H} R(y) + Q(y)Z_H(y) = \sum_{y \in H} R(y) = |H| \cdot R(0)$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

$$\sum_{y \in H} f(y) = \sum_{y \in H} R(y) + Q(y)Z_H(y) = \sum_{y \in H} R(y) = |H| \cdot R(0)$$

$$f(X) = f_0 + f_1 X + \ldots + f_d X^d \pmod{Z_H(X)}$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

$$\sum_{y \in H} f(y) = \sum_{y \in H} R(y) + Q(y)Z_H(y) = \sum_{y \in H} R(y) = |H| \cdot R(0)$$

$$f(X) = f_0 + f_1 X + \ldots + f_d X^d \pmod{Z_H(X)}$$

$$f(X) = (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) + (f_1 + f_{|H|+1} + \ldots + f_{\lfloor d/|H| \rfloor} + 1)X + \ldots \pmod{Z_H(X)}$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

$$\sum_{y \in H} f(y) = \sum_{y \in H} R(y) + Q(y)Z_H(y) = \sum_{y \in H} R(y) = |H| \cdot R(0)$$

$$f(X) = f_0 + f_1 X + \ldots + f_d X^d \pmod{Z_H(X)}$$

$$f(X) = (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) + (f_1 + f_{|H|+1} + \ldots + f_{\lfloor d/|H| \rfloor} + 1)X + \ldots \pmod{Z_H(X)}$$

$$f(X) = R(X) \pmod{Z_H(X)}$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) = v$

$$f(X) = R(X) + Q(X)Z_H(X) \qquad \deg R \leq |H| \qquad Z_H(X) = X^{|H|} - 1$$

$$\sum_{y \in H} f(y) = \sum_{y \in H} R(y) + Q(y)Z_H(y) = \sum_{y \in H} R(y) = |H| \cdot R(0)$$

$$f(X) = f_0 + f_1 X + \ldots + f_d X^d \pmod{Z_H(X)}$$

$$f(X) = (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) + (f_1 + f_{|H|+1} + \ldots + f_{\lfloor d/|H| \rfloor} + 1)X + \ldots \pmod{Z_H(X)}$$

$$f(X) = R(X) \pmod{Z_H(X)}$$

$$R(0) = (f_0 + f_{|H|} + \ldots + f_{\lfloor d/|H| \rfloor}) \pmod{Z_H(X)}$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = \underbrace{\left(f_0 - \frac{v}{|H|} \right)}_{0} X^{gap} + f_1 X^{gap+1} + \dots$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap} \qquad f'(X) = \underbrace{\left(f_0 - \frac{v}{|H|} \right)}_{0} X^{gap} + f_1 X^{gap+1} + \dots$$

$$f'(X) = \underbrace{\left(f_0 + f_{|H|} + \dots + f_{\lfloor d/|H| \rfloor} - \frac{v}{|H|} \right)}_{=0} X^{gap} + \dots$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = \underbrace{\left(f_0 - \frac{v}{|H|} \right)}_{0} X^{gap} + f_1 X^{gap+1} + \dots$$

$$f'(X) = \underbrace{\left(f_0 + f_{|H|} + \dots + f_{\lfloor d/|H| \rfloor} - \frac{v}{|H|} \right)}_{=0} X^{gap} + \dots$$

$$S(X) = X^{gap} + X^{gap-|H|} + \dots + X^{gap-\lfloor d/|H| \rfloor}$$

$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = \underbrace{\left(f_0 - \frac{v}{|H|} \right)}_{0} X^{gap} + f_1 X^{gap+1} + \dots$$

$$f'(X) = \underbrace{\left(f_0 + f_{|H|} + \dots + f_{\lfloor d/|H| \rfloor} - \frac{v}{|H|} \right)}_{=0} X^{gap} + \dots$$

$$S(X) = X^{gap} + X^{gap-|H|} + \dots + X^{gap-\lfloor d/|H| \rfloor}$$

$$f'(X) = f(X) \cdot S(X) - \frac{v}{|H|} X^{gap}$$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot \sum_{i=0}^{\lfloor d/|H| \rfloor} f_{|H| \cdot i} = \nu$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot \sum_{i=0}^{\lfloor d/|H| \rfloor} f_{|H| \cdot i} = \nu$

$\mathcal{P}$

$f(X)$

$\mathcal{V}$

$g^{f(s)}$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot \sum_{i=0}^{\lfloor d/|H| \rfloor} f_{|H| \cdot i} = v$

$\mathcal{P}$ $\qquad$ $\mathcal{V}$

$f(X)$ $\qquad$ $g^{f(s)}$

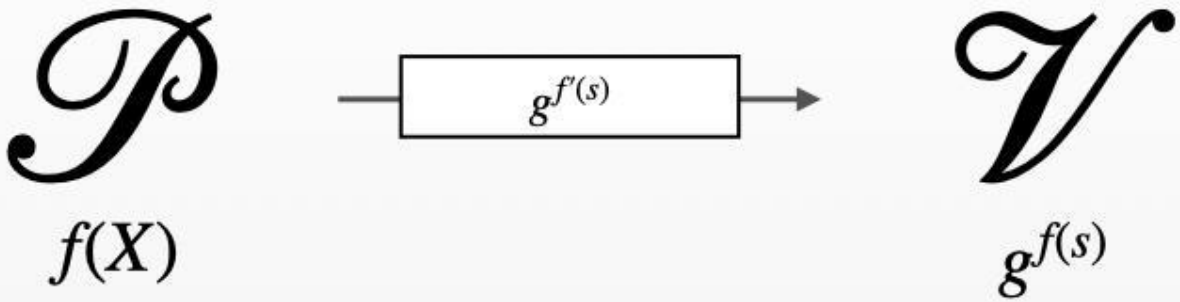
$$S(X) = \sum_{i=0}^{\lfloor d/|H| \rfloor} X^{gap - |H| \cdot i}$$

$$f'(X) = f(X) \cdot S(X) - \frac{v}{|H|} X^{gap}$$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot \sum_{i=0}^{\lfloor d/|H| \rfloor} f_{|H| \cdot i} = v$



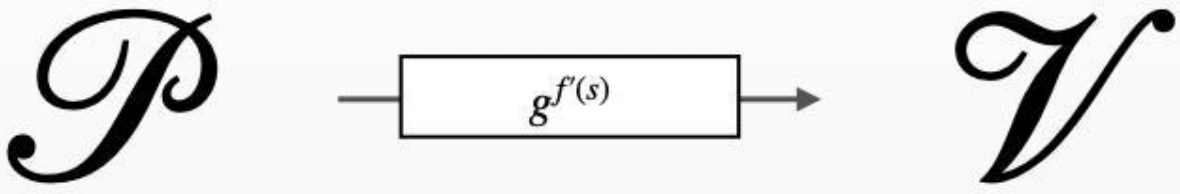
$$S(X) = \sum_{i=0}^{\lfloor d/|H| \rfloor} X^{gap - |H| \cdot i}$$

$$f'(X) = f(X) \cdot S(X) - \frac{v}{|H|} X^{gap}$$

$$\deg(f) \leq gap - 1$$

Fact 2

Let $f(X) \in \mathbb{F}_{\leq d}[X]$ then $\sum_{y \in H} f(y) = |H| \cdot \sum_{i=0}^{\lfloor d/|H| \rfloor} f_{|H| \cdot i} = v$



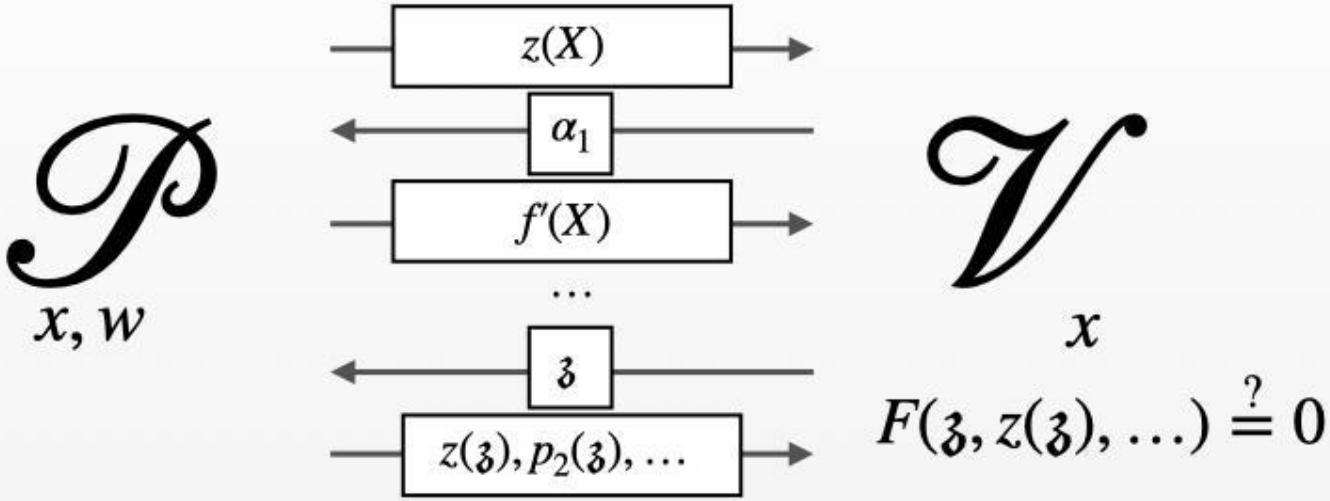
$$S(X) = \sum_{i=0}^{\lfloor d/|H| \rfloor} X^{gap - |H| \cdot i}$$

$$f'(X) = f(X) \cdot S(X) - \frac{v}{|H|} X^{gap}$$

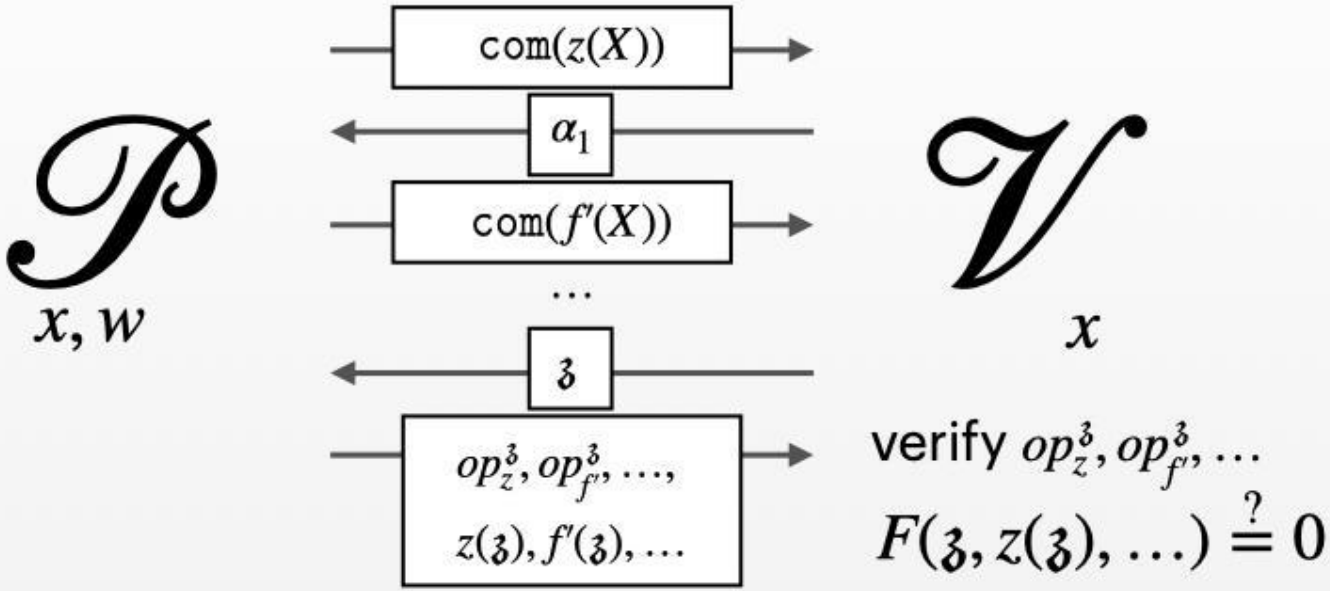
$$f(X) \cdot S(X) - f'(X) = \frac{v}{|H|} X^{gap}$$

$$g^{f(s)} \bullet h^{S(s)} - g^{f'(s)} \bullet h = \frac{v}{|H|} g \bullet h^{s^{gap}}$$

How zkSNARKs are built?



How zkSNARKs are built?



STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

More **efficient** batching

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A) =$ number of distinct elements

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$f_1(x_1)$
$f_2(x_1)$ } batch
$f_3(x_1)$
$f_1(x_2)$
$f_1(x_3)$

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$$\left.\begin{matrix} f_1(x_1) \\ f_2(x_1) \\ f_3(x_1) \end{matrix}\right\} \text{batch}$$
$$f_1(x_2)$$
$$f_1(x_3)$$

Need to send $\texttt{Dist}(X)$ openings

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$f_1(x_1)$
$f_2(x_1)$ } batch
$f_3(x_1)$
$f_1(x_2)$
$f_1(x_3)$

Need to send $\texttt{Dist}(X)$ openings

For each j, batch openings of all evaluations of f_j
(possible if each f_j evaluated at different set of points)

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$f_1(x_1)$
$f_2(x_1)$ } batch
$f_3(x_1)$
$f_1(x_2)$
$f_1(x_3)$

For each j, batch openings of all evaluations of f_j
(possible if each f_j evaluated at different set of points)

$f_1(x_1)$
$f_1(x_2)$ } batch
$f_1(x_3)$
$f_2(x_1)$
$f_3(x_1)$

Need to send $\texttt{Dist}(X)$ openings

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$f_1(x_1)$
$f_2(x_1)$ } batch
$f_3(x_1)$
$f_1(x_2)$
$f_1(x_3)$

Need to send $\texttt{Dist}(X)$ openings

For each j, batch openings of all evaluations of f_j
(possible if each f_j evaluated at different set of points)

$f_1(x_1)$
$f_1(x_2)$ } batch
$f_1(x_3)$
$f_2(x_1)$
$f_3(x_1)$

Need to send $\texttt{Dist}(F)$ openings

Transparent Vampire

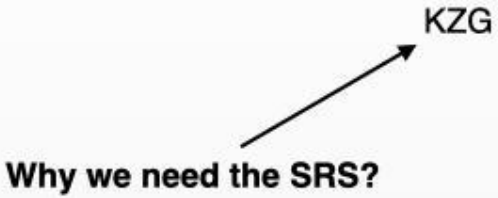
Why we need the SRS?

Transparent Vampire

KZG

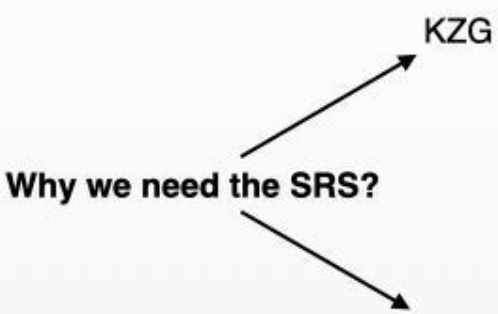
Why we need the SRS?

Transparent Vampire



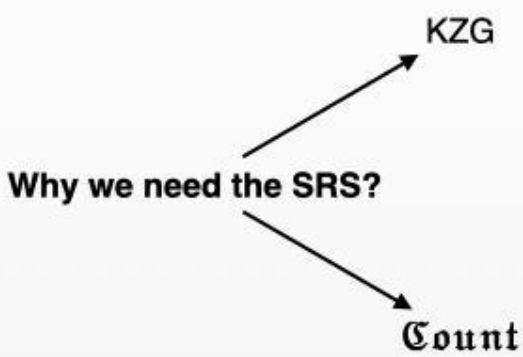
Transparent Vampire

Transparent Vampire



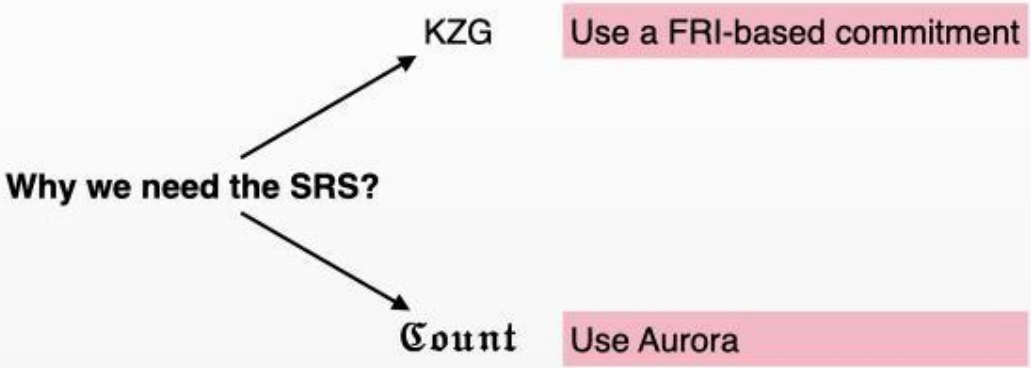
Use a FRI-based commitment

Transparent Vampire

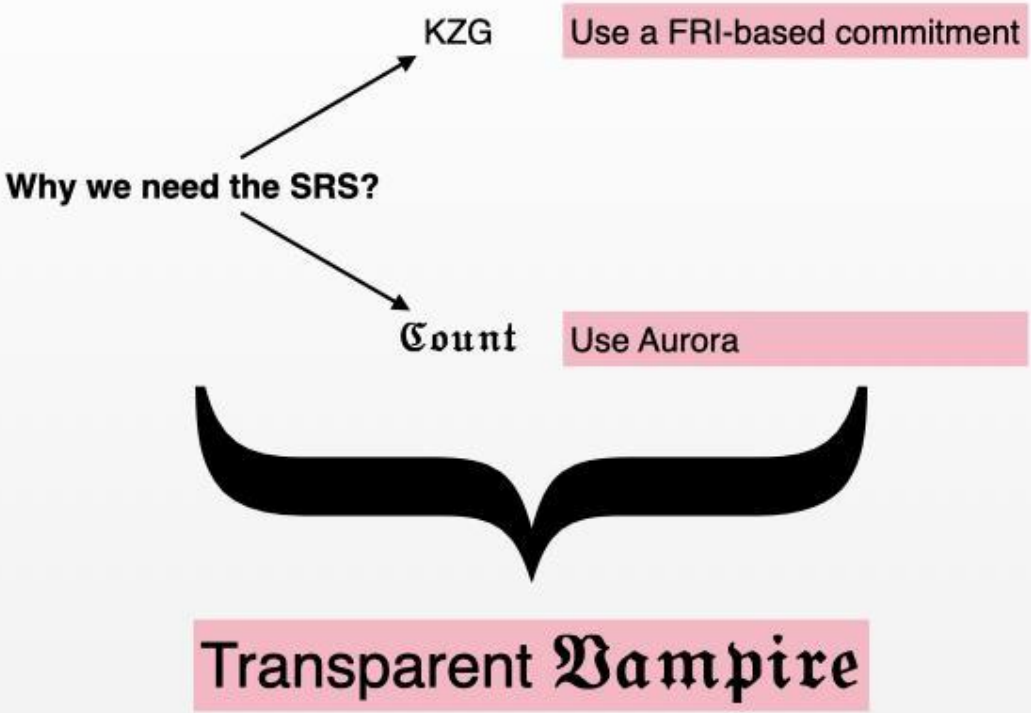


Use a FRI-based commitment

Transparent Vampire

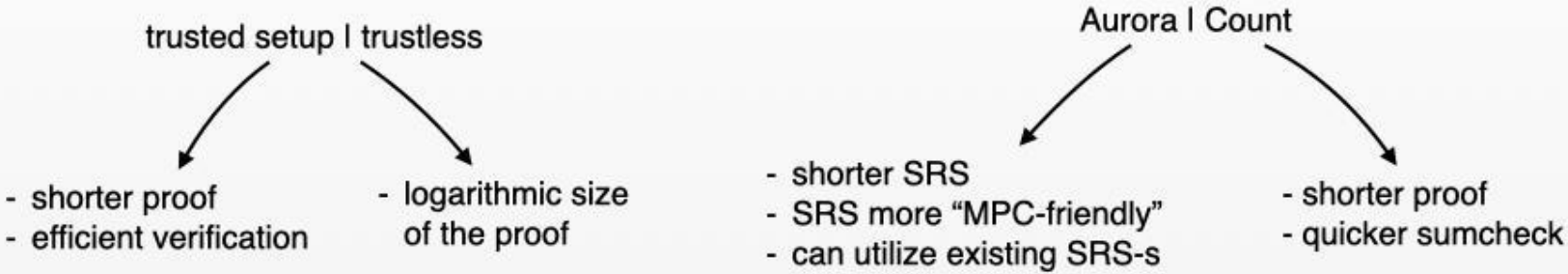


Transparent Vampire



Variants and trade-offs

Vampire



https://ia.cr/2022/406

Additional slides

Aurora SCA

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$P(\omega) + P(\omega^2) + \ldots + P(\omega^{3n}) = v$

$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^{3n})$

$\mathcal{P}$

$P(X), v, H$

$\mathcal{V}$

$P(X), v, H$

Aurora SCA

$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$P(\omega) + P(\omega^2) + \ldots + P(\omega^{3n}) = v$

$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^{3n})$

$\sum_{y \in H} P(y) = v$ iff there exist $R(X), Q(X), (\deg R(X) \leq |H| - 2)$ such that

$$P(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$

$\mathcal{P}$

$P(X), v, H$

$\mathcal{V}$

$P(X), v, H$

Aurora SCA

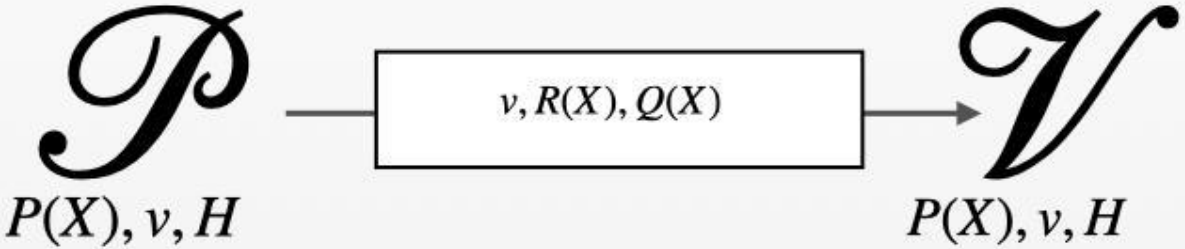
$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$P(\omega) + P(\omega^2) + \ldots + P(\omega^{3n}) = v$

$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^{3n})$

$\sum_{y \in H} P(y) = v$ iff there exist $R(X), Q(X)$, $(\deg R(X) \leq |H| - 2)$ such that

$$P(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$



Aurora SCA

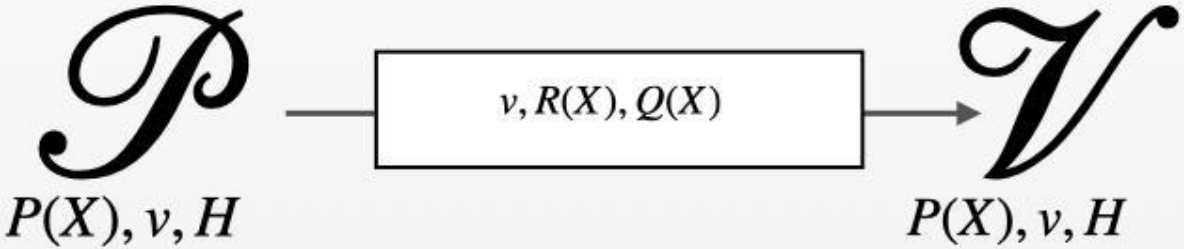
$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$P(\omega) + P(\omega^2) + \ldots + P(\omega^{3n}) = v$

$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^{3n})$

$\sum_{y \in H} P(y) = v$ iff there exist $R(X), Q(X), (\deg R(X) \leq |H| - 2)$ such that

$$P(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$



$$f(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$

Polynomial commitment scheme

$\texttt{commit}(f(X))$: **return** c

$\texttt{evaluate}(c, x)$: **return** $f(x) = y$

$\texttt{open}(c, x, y)$: **return** op

KZG polynomial commitment scheme

Efficiency table

Scheme	Argument length		Arithmetization
	Elements	Bits	
Updatable and universal zk-SNARKs			
Sonic [29]	$20\|\mathbb{G}_1\| + 16\|\mathbb{F}\|$	11776 [8]	constraints
Marlin [12]	$13\|\mathbb{G}_1\| + 8\|\mathbb{F}\|$	7040	R1CS, sparse matrices
Basilisk [31]	$10\|\mathbb{G}_1\| + 3\|\mathbb{F}\|$	4608	R1CSLite, sparse matrices
Plonk [16]	$7\|\mathbb{G}_1\| + 7\|\mathbb{F}\|$	4480	Plonk constraints
LunarLite [11]	$10\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	4352	R1CSLite, sparse matrices
Basilisk [31]	$8\|\mathbb{G}_1\| + 4\|\mathbb{F}\|$	4096	Plonk constraints
VOR1CS* [34]	$9\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	3968	R1CS, sparse matrices
VOPlonk* [34]	$7\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	3200	Plonk constraints
Basilisk (full version, [32])	$6\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	2816	Weighted R1CS with bounded fan-out
Vampire (this work)	$4\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	2048	R1CSLite, sparse matrices
Non-universal zk-SNARKs (relation-specific SRS)			
Groth16 [19]	$2\|\mathbb{G}_1\| + 1\|\mathbb{G}_2\|$	1536	R1CS

Vampire

$4\mathbb{G}_1 + 2\mathbb{F}$

2048 bytes

Plonk	$7\mathbb{G}_1 + 7\mathbb{F}$	4480 bytes	updatable	universal
Marlin	$13\mathbb{G}_1 + 8\mathbb{F}$	7040 bytes	updatable	universal
Sonic	$20\mathbb{G}_1 + 16\mathbb{F}$	11776 bytes	updatable	universal
Groth	$2\mathbb{G}_1 + 1\mathbb{G}_2$	1536 bytes	non-updatable	non-universal

How?

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

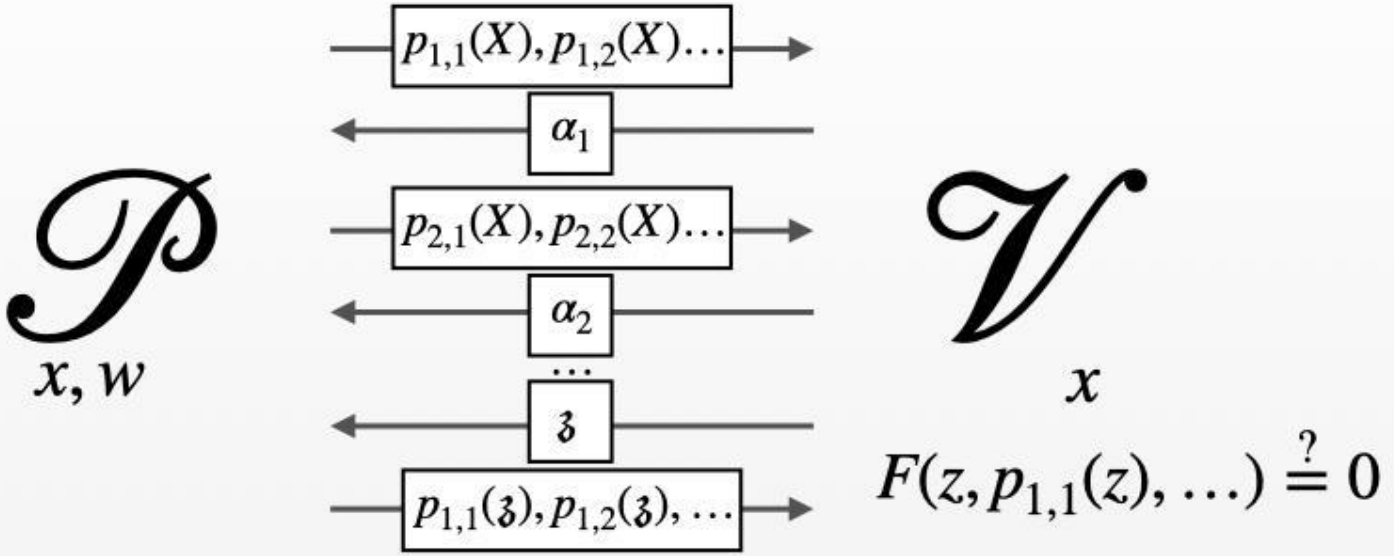
Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

More **efficient** batching

How zkSNARKs are built?

$R(x, w)$



(Idealized model)

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

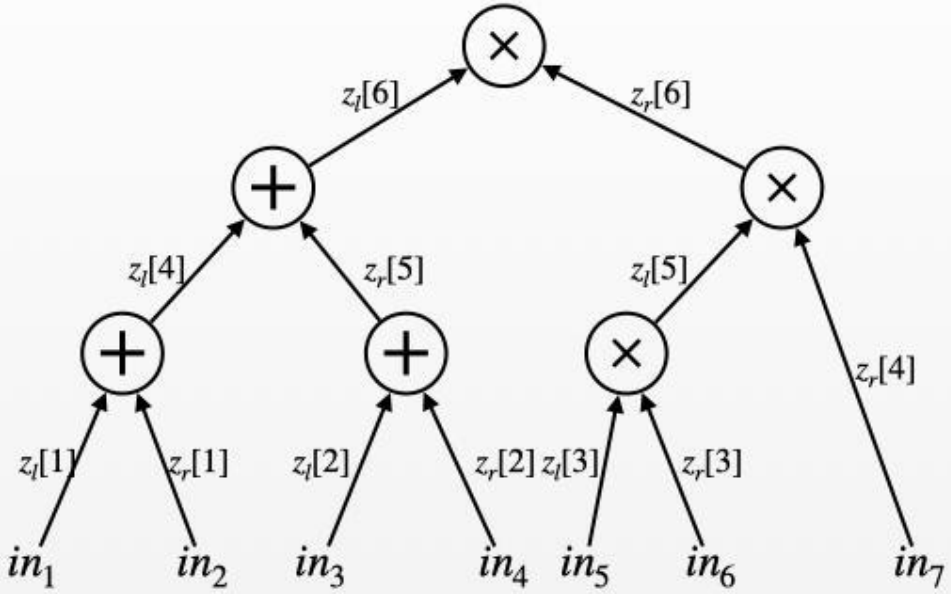
STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

More **efficient** batching

RiCS Lite



$$W = \begin{bmatrix} I & 0 & L \\ 0 & I & R \\ 0 & 0 & 0 \end{bmatrix}$$

$$z = \begin{bmatrix} z_l \\ z_R \\ z_l \circ z_R \end{bmatrix}$$

$$Wz = 0$$

RiCS vs RiCSLite

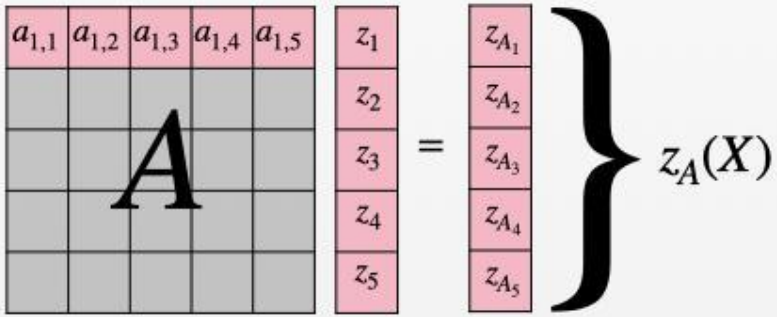
$A, B, C \in \mathbb{F}^{n \times n}$

$z \in \mathbb{F}^{n}$

$H = \{\omega, \omega^2, \ldots, \omega^n\}$

A mult. gates' left inputs
B mult. gates' right input
C mult. gates' outputs
$z = (x, w)$

$$A \cdot z \circ B \cdot z - C \cdot z = 0$$



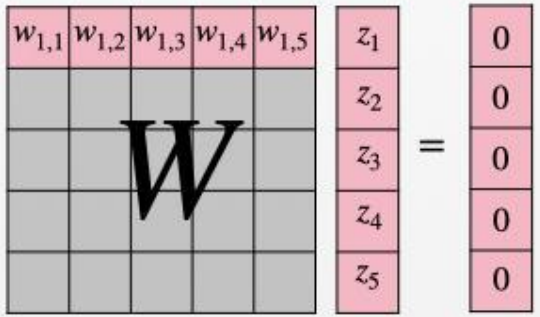
$z(X), z_A(X), z_B(X), z_C(X)$

$W \in \mathbb{F}^{3n \times 3n}$

$z \in \mathbb{F}^{3n}$

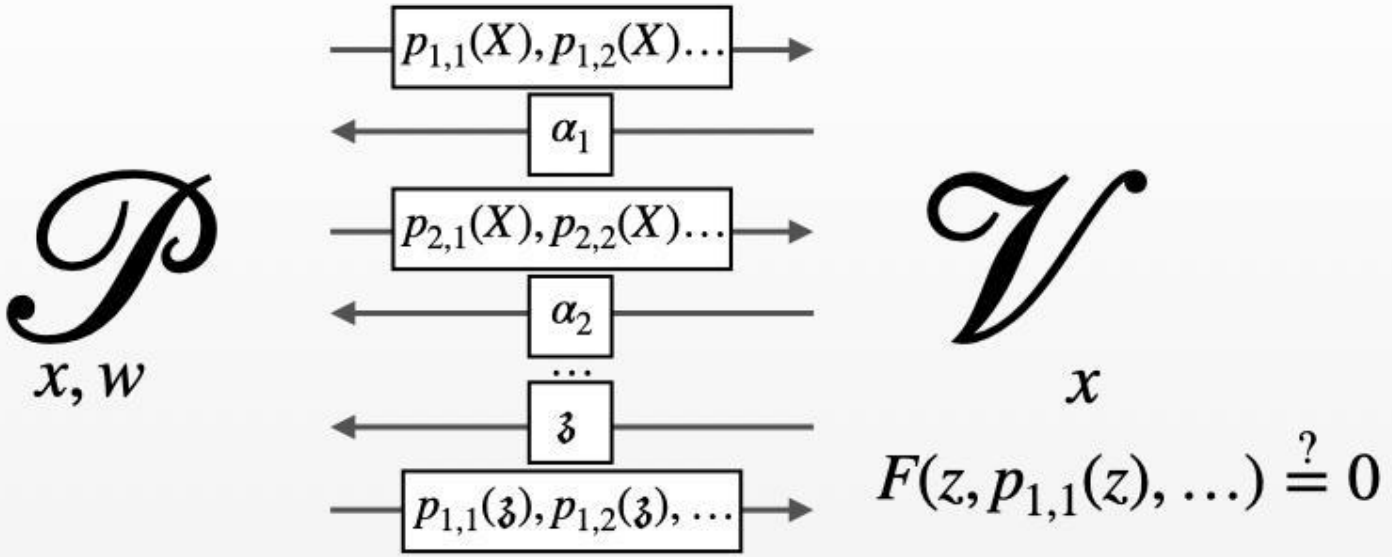
$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$$Wz = 0$$

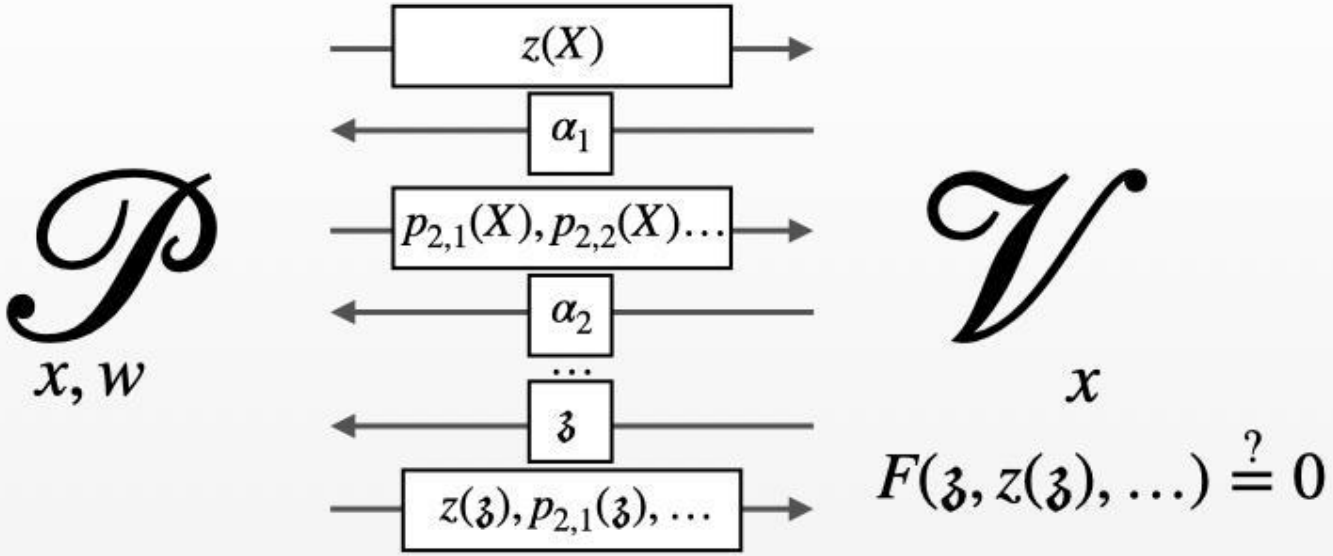


$z(X)$

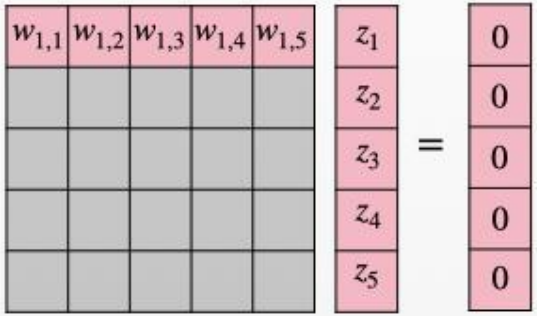
How zkSNARKs are built?



How zkSNARKs are built?



From matrices to polynomials



$$\forall x \in H : \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w_{x,y} \cdot z_y = 0$$

$$\forall x \in H : P(x) = \sum_{y \in H} w(x,y) \cdot z(y) = 0$$

$$P(X) = \sum_{y \in H} w(X,y) \cdot z(y) = 0$$

$$\deg_X P(X) = n - 1$$

$$|H| = n$$

STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

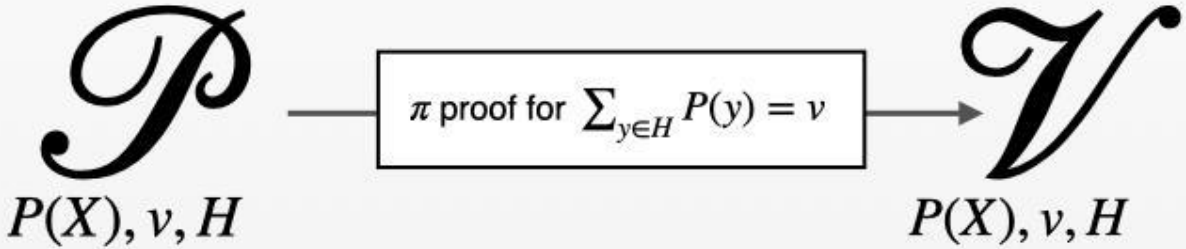
More **efficient** batching

Sumcheck argument

Input: polynomial P, set H, the evaluation value v

$$H = \{\omega, \omega^2, \ldots, \omega^n\}$$

$$P(\omega) + P(\omega^2) + \ldots + P(\omega^n) = v$$



Aurora SCA

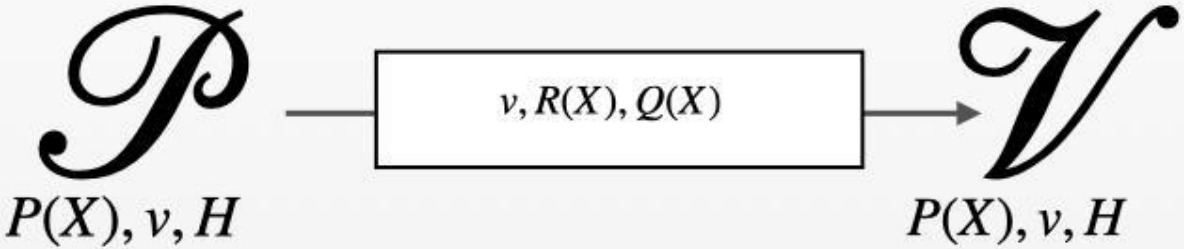
$H = \{\omega, \omega^2, \ldots, \omega^{3n}\}$

$P(\omega) + P(\omega^2) + \ldots + P(\omega^{3n}) = v$

$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^{3n})$

$\sum_{y \in H} P(y) = v$ iff there exist $R(X), Q(X), (\deg R(X) \leq |H| - 2)$ such that

$$P(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$



$$f(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$

From bivariate polynomials to sumcheck

$$P(X) = \sum_{y \in H} w(X, y) \cdot z(y) = 0$$

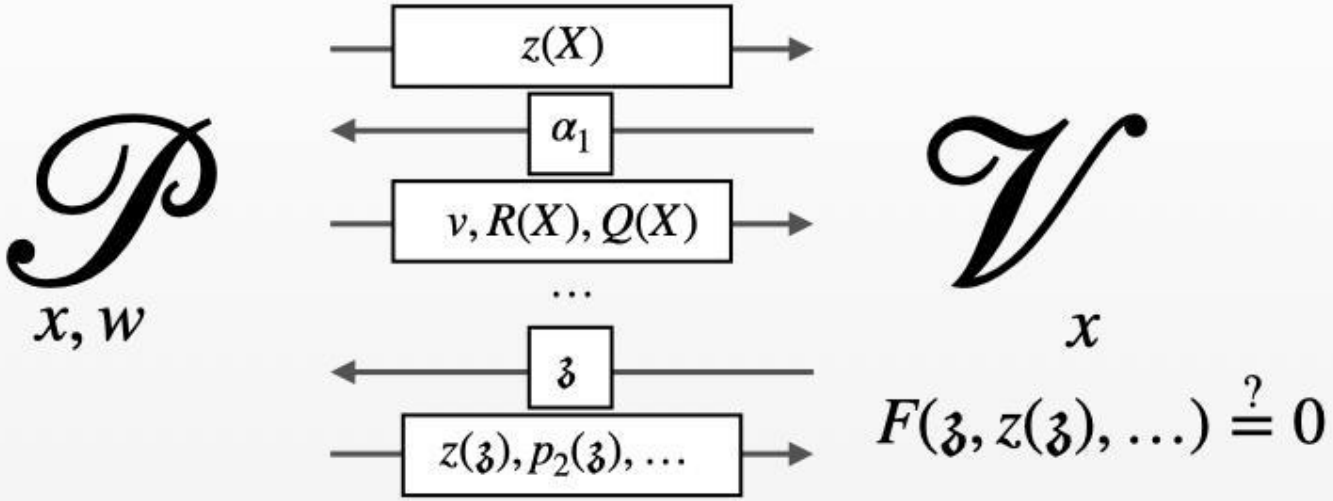
$$\Psi(X, Y) = w(X, Y) \cdot z(Y) = 0$$

$$\Psi_\alpha(Y) = w(\alpha, Y) \cdot z(Y) = 0$$

$$\sum_{y \in H} \Psi_\alpha(y) = 0$$

Compute $R(X), Q(X)$ such that $\Psi_\alpha(X) = X \cdot R(X) + Q(X)Z(X)$

How zkSNARKs are built?



From Aurora to Count

KZG commitment scheme

$e : \mathbb{G} \times \mathbb{H} \to \mathbb{T}$

$SRS = \{g, g^s, \ldots, g^{s^k}, h\}$

$f(X) = f_0 + f_1 X + \ldots f_k X^k$

$\mathtt{com}(f(X)) = g^{f_0} \cdot (g^s)^{f_1} \cdot \ldots \cdot (g^{s^k})^{f_k}$

Assumption
It is infeasible to compute g^{s^n} for $n \notin \{1, \ldots, k\}$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Observation
If $f(X)$ has non-zero coefficient f_m then the SRS needs to have g^{s^m}

Fact
Let $f(X)$ of degree $< |H|$ then $\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

$$\deg(f) \leq gap - 1$$

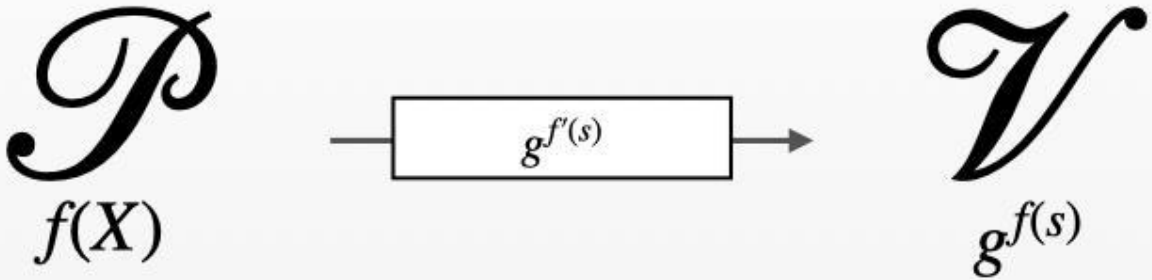
$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$f'(X) = \underbrace{f_0 X^{gap} - \frac{v}{|H|} X^{gap}}_{0} + f_1 X^{gap+1} + \ldots$$

Count

$$SRS = \left\{ \begin{matrix} g, g^s, \ldots, g^{gap-1}, g^{gap+1}, \ldots, g^{2gap} \\ h, h^{s^{gap}} \end{matrix} \right\}$$

Fact
Let $f(X)$ of degree $< |H|$ then
$\sum_{y \in H} f(y) = f(0) \cdot |H| = v$

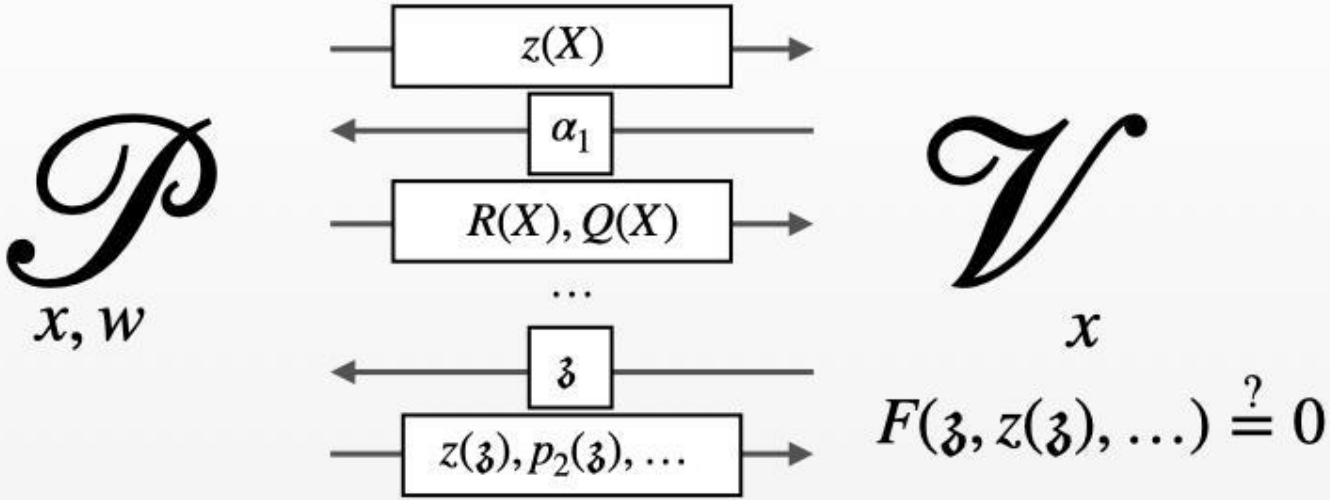


$$f(X) \cdot X^{gap} - f'(X) = \frac{v}{|H|} X^{gap}$$

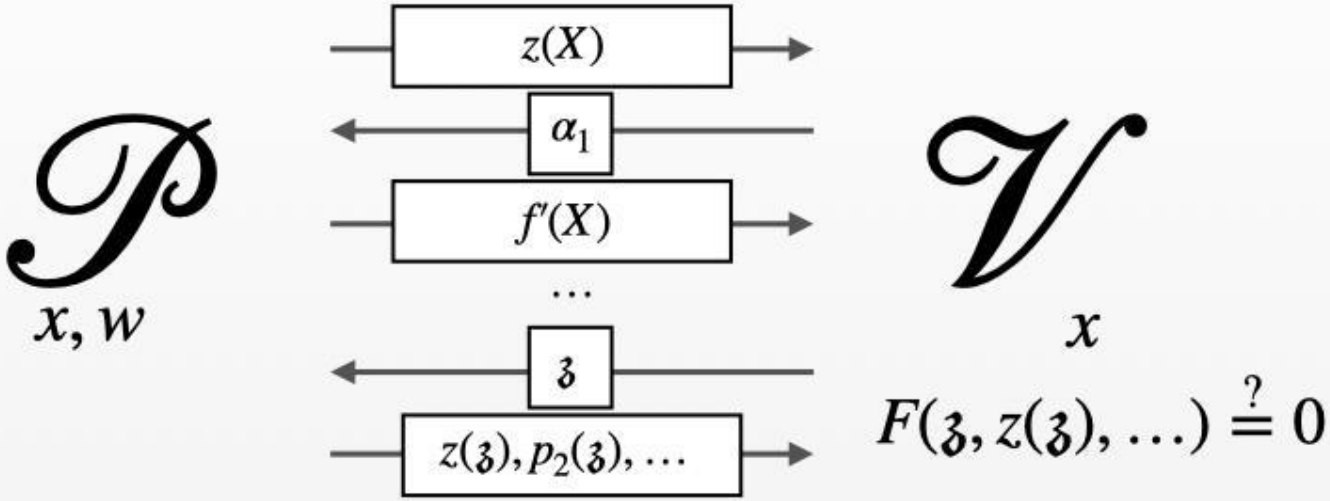
$$f'(X) = f(X) \cdot X^{gap} - \frac{v}{|H|} X^{gap}$$

$$g^{f(s)} \bullet h^{s^{gap}} - g^{f'(s)} \bullet h = \frac{v}{|H|} g \bullet h^{s^{gap}}$$

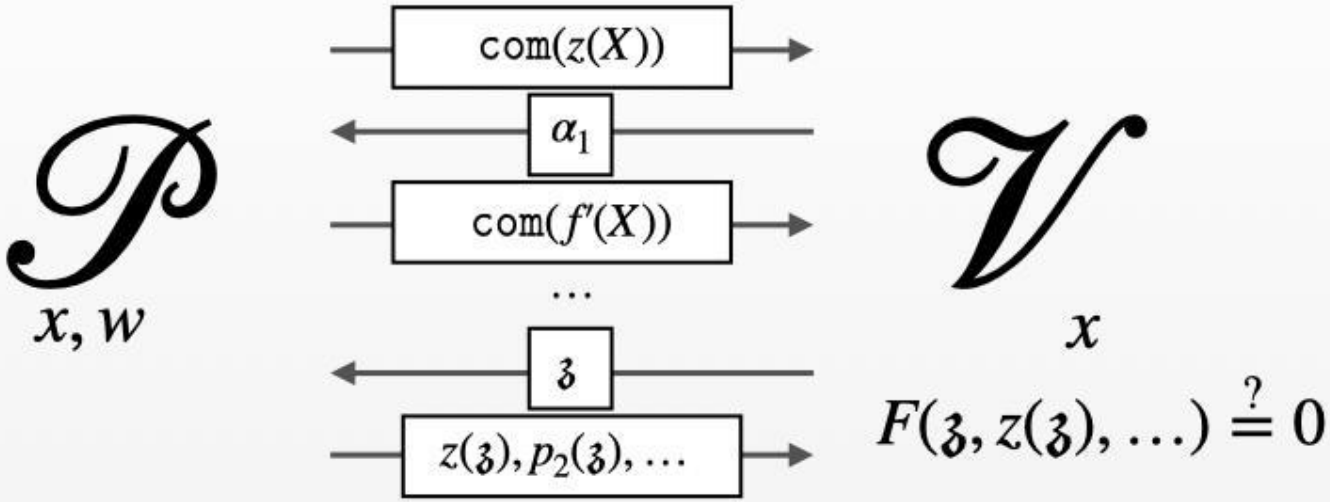
How zkSNARKs are built?



How zkSNARKs are built?



How zkSNARKs are built?



STEP 1

R1CSLite vs R1CS
one polynomial for the whole witness
$-2\mathbb{G}$ vs Marlin/Plonk

STEP 2

Novel sumcheck argument $\mathfrak{Count}$
$-1\mathbb{G}$ vs Aurora sumcheck

STEP 3

More **efficient** batching

More efficient batching

Evaluation points: $X = (x_1, \ldots, x_k)$
Polynomials: $F = (f_1(X), \ldots, f_m(X))$

$\texttt{Dist}(A)$ = number of distinct elements

For each i, batch openings of all polynomials f_j evaluated at x_i

$$\left.\begin{matrix} f_1(x_1) \\ f_2(x_1) \\ f_3(x_1) \end{matrix}\right\} \text{batch}$$
$$f_1(x_2)$$
$$f_1(x_3)$$

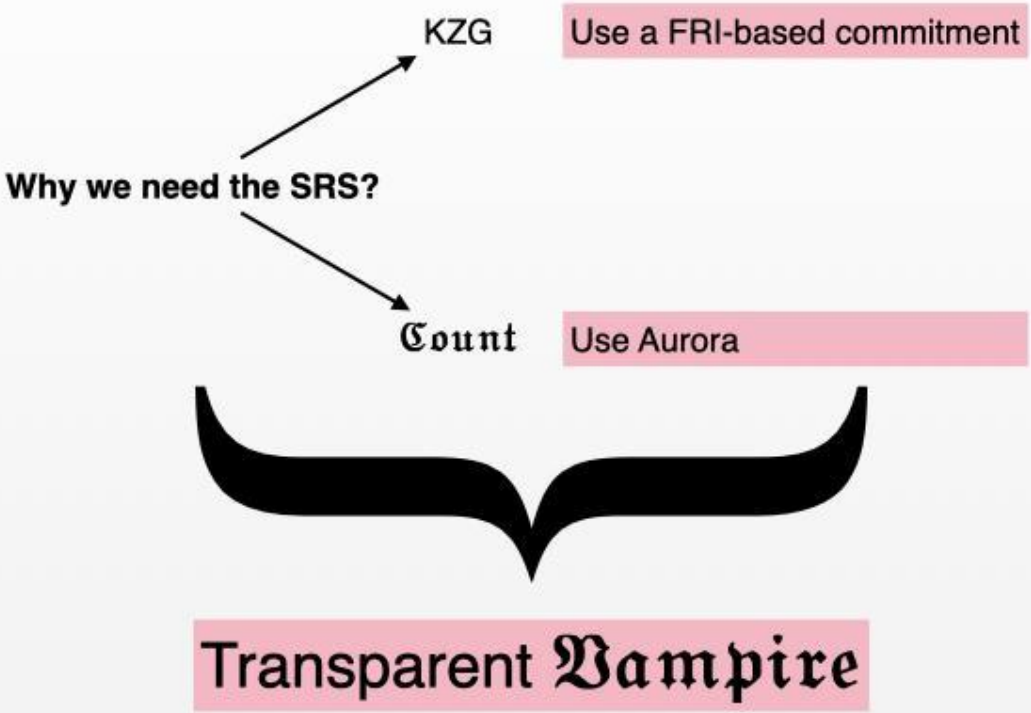
Need to send $\texttt{Dist}(X)$ openings

For each j, batch openings of all evaluations of f_j
(possible if each f_j evaluated at different set of points)

$$\left.\begin{matrix} f_1(x_1) \\ f_1(x_2) \\ f_1(x_3) \end{matrix}\right\} \text{batch}$$
$$f_2(x_1)$$
$$f_3(x_1)$$

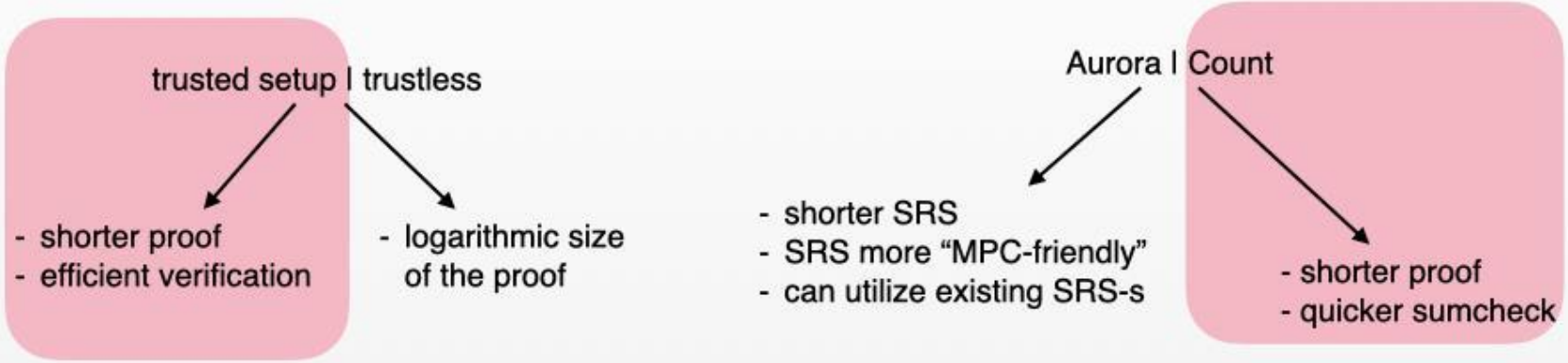
Need to send $\texttt{Dist}(F)$ openings

Transparent Vampire



Variants and trade-offs

Vampire



https://ia.cr/2022/406

Counting Vampires: From Univariate Sumcheck to Updatable ZK-SNARK*

Version 2.0, Thursday 23rd June, 2022, 12:58

Helger Lipmaa[1], Janno Siim[1], and Michał Zając[2]

Abstract. We
sentially optima
the previously n
based on polyno
mitments. We us
universal zk-SN
(four group and
uses the aggrega
ent Count of es-
element. While
ent of Aurora is
er-product com-
d updatable and
argument length
ldition, Vampire
Boneh *et al.*

Additional slides

Polynomial commitment scheme

`commit`$(f(X))$: **return** c

`evaluate`(c, x) : **return** $f(x) = y$

`open`(c, x, y) : **return** op

KZG polynomial commitment scheme

Efficiency table

Scheme	Argument length		Arithmetization
	Elements	Bits	
Updatable and universal zk-SNARKs			
Sonic [29]	$20\|\mathbb{G}_1\| + 16\|\mathbb{F}\|$	11776 [8]	constraints
Marlin [12]	$13\|\mathbb{G}_1\| + 8\|\mathbb{F}\|$	7040	R1CS, sparse matrices
Basilisk [31]	$10\|\mathbb{G}_1\| + 3\|\mathbb{F}\|$	4608	R1CSLite, sparse matrices
Plonk [16]	$7\|\mathbb{G}_1\| + 7\|\mathbb{F}\|$	4480	Plonk constraints
LunarLite [11]	$10\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	4352	R1CSLite, sparse matrices
Basilisk [31]	$8\|\mathbb{G}_1\| + 4\|\mathbb{F}\|$	4096	Plonk constraints
VOR1CS* [34]	$9\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	3968	R1CS, sparse matrices
VOPlonk* [34]	$7\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	3200	Plonk constraints
Basilisk (full version, [32])	$6\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	2816	Weighted R1CS with bounded fan-out
Vampire (this work)	$4\|\mathbb{G}_1\| + 2\|\mathbb{F}\|$	2048	R1CSLite, sparse matrices
Non-universal zk-SNARKs (relation-specific SRS)			
Groth16 [19]	$2\|\mathbb{G}_1\| + 1\|\mathbb{G}_2\|$	1536	R1CS

Sumcheck argument

Input: polynomial f, set H, the evaluation value v

$$H = \{\omega, \omega^2, \ldots, \omega^n\}$$

$$f(\omega) + f(\omega^2) + \ldots + f(\omega^n) = v$$

$$Z(X) = (X - \omega)(X - \omega^2)\ldots(X - \omega^n)$$

Aurora SCA

$\sum_{y \in H} f(y) = v$ iff there exist

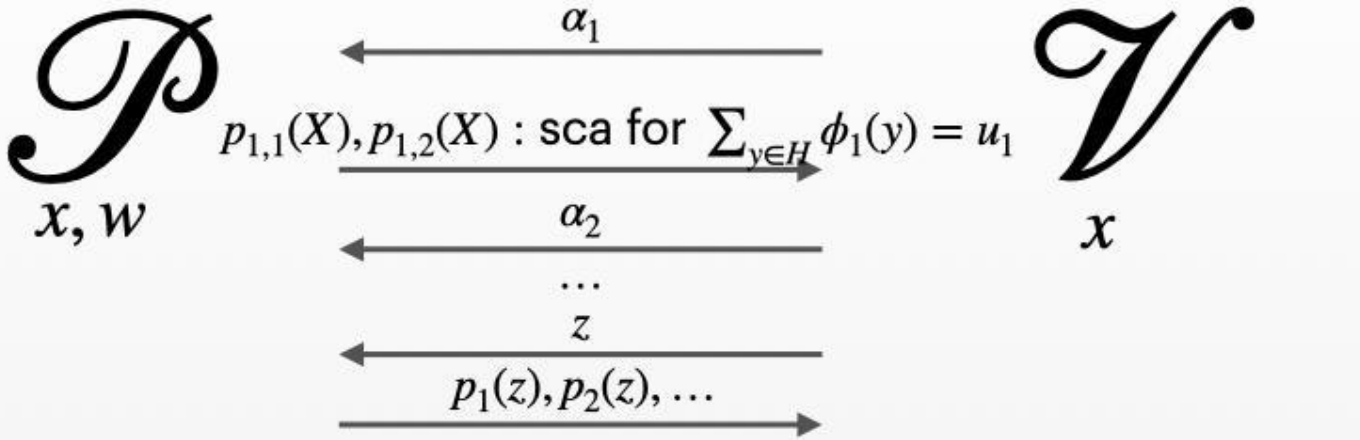
$R(X), Q(X)$ $(\deg R(X) \leq |H| - 2)$, such that

$$f(X) = v/|H| + X \cdot R(X) + Q(X)Z(X)$$

Argument:

- **Prover** sends $v, f(X), R(X), Q(X)$
- **Verifier** checks that the equality above holds.

How zkSNARKs are built?



$$F(z, p_1(z), \ldots) \stackrel{?}{=} 0$$

How zkSNARKs are built?