Zero Knowledge Proofs

SNARKs based on Linear PCP

Instructors: Dan Boneh, Shafi Goldwasser, Dawn Song, Justin Thaler, **Yupeng Zhang**





Stanford University

Berkeley UNIVERSITY OF CALIFORNIA

GEORGETOWN UNIVERSITY

TEXAS A&M UNIVERSITY

SNARKs Learned So Far

Crypto primitive	Trusted Setup	Schemes
Pairing (KZG)	✓	Plonk, Interactive Proofs, …
Discrete-log	×	Bulletproofs, Dory, Dark, Hyrax, Halo2, …
Hashing	×	Brakedown, Orion, … (linear-time encodable code)
		Stark, Aurora, Fractal, Virgo, … (RS code and FRI)

This Lecture: SNARKs based on Linear PCP

Earliest Implemented SNARKs

- ✓ Shortest proof size (3 elements [Groth16])
- ✓ Fast verifier (bilinear pairing)
- ✗ FFT and group exponentiations on the prover
- ✗ Circuit-specific trusted setup

History of SNARKs

Kilian'92 → **Micali'00** PCP + Merkle Tree

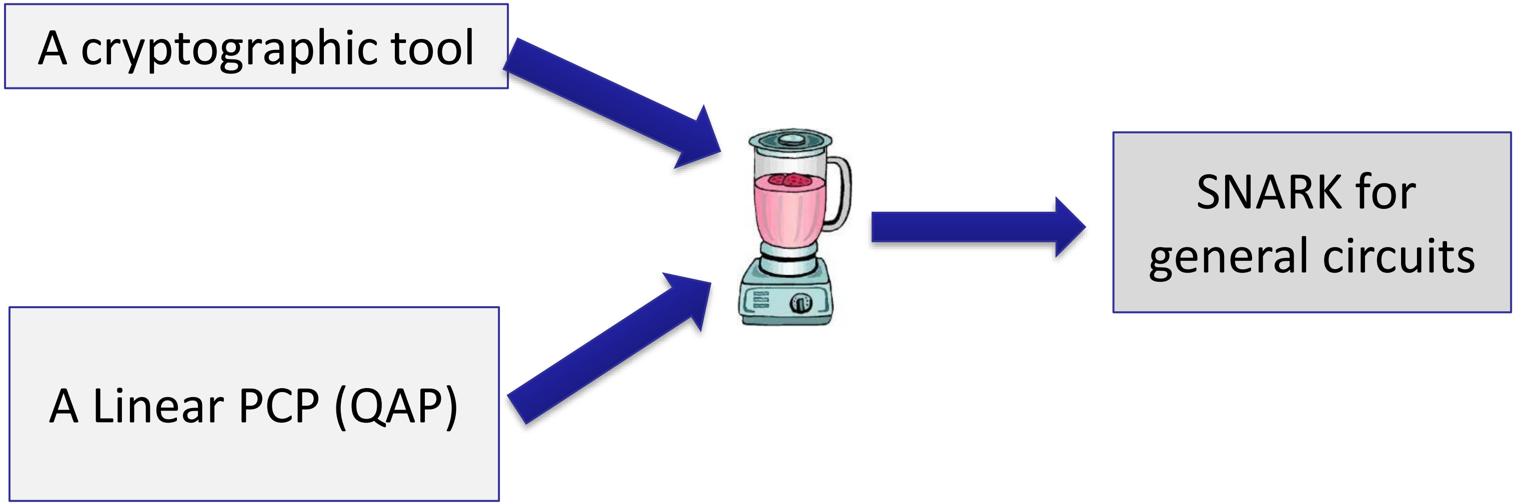
IKO'07 → **Groth'10** → **Lipmaa'12** → **GGPR'13**

Linear PCP

QSP and QAP

→ SBVBPW13, PGHR13, BCGTV13, BFRSBW13, BCTV14a, BCTV14b, BCGGMTV14, Groth16…
SMBW12, SVPBBW12, BCIOP13, …

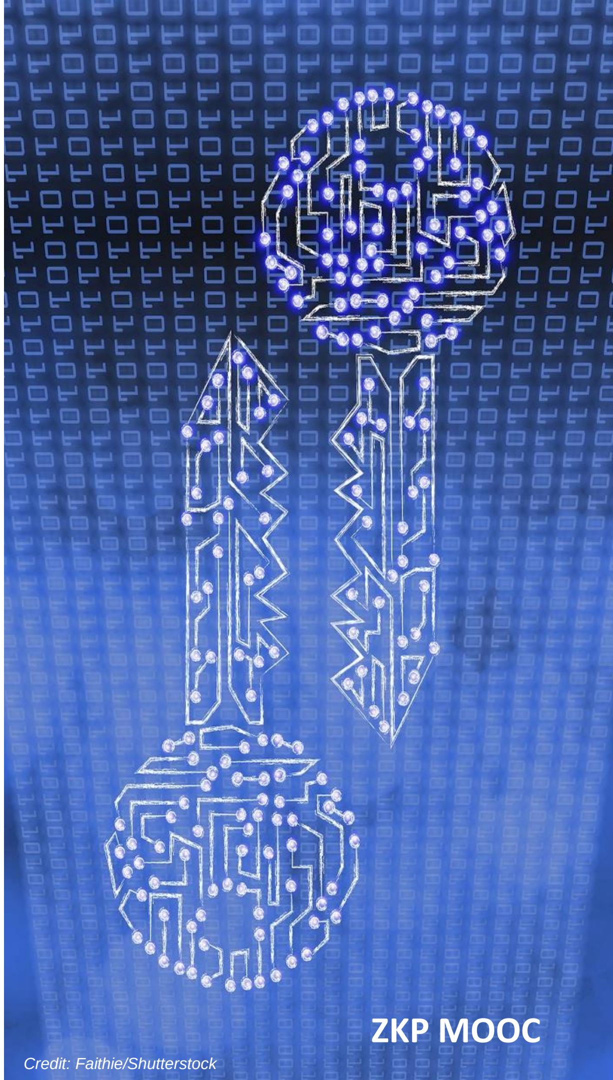
Paradigm for SNARK



Plan of this lecture

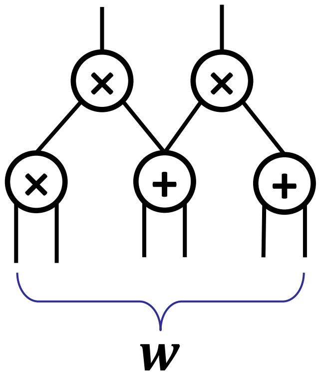
- Quadratic Arithmetic Program (QAP)
- From QAP to SNARK
- Other variants

Quadratic Arithmetic Program (QAP)

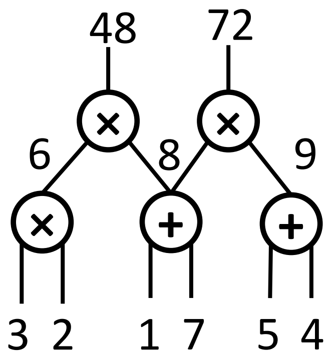


Recall: SNARKs for circuit-satisfiability

- Given an arithmetic circuit C over $\mathbb{F}$ and output y.
- P claims to know a w such that $C(x, w) = y$.
- For simplicity, let's take x to be the empty input.



Transcript/trace of C



- Interactive proof (lecture 4, slide 76):

  value of every gate

- Plonk (lecture 5, slide 42):

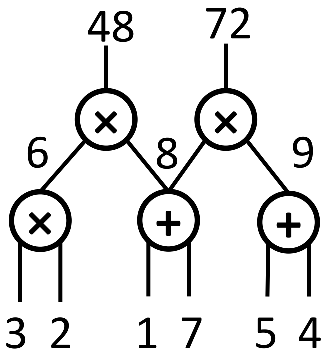
  left input, right input, output of every gate

- QAP:

  input + output of every multiplication gate

Transcript/trace of C

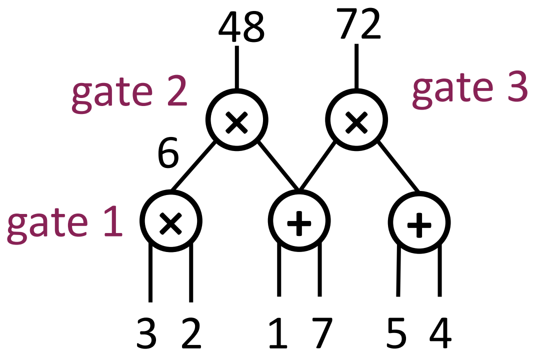
input + output of every multiplication gate



3	2	1	7	5	4	6	48	72

Transcript/trace of C

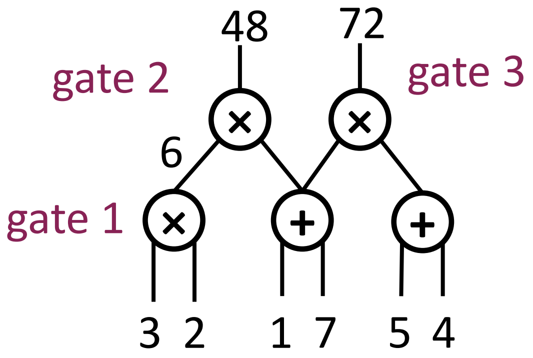
Labeling the multiplication gates



3	2	1	7	5	4	6	48	72

Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$l_i(x)$: is c_i the left input of gate j , for $j = 1,2,3$?

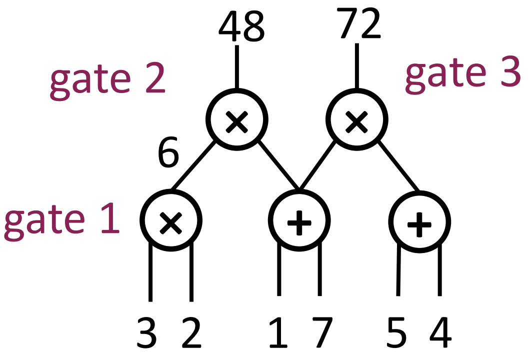
e.g., $l_1(x) : (1,0,0)$

Polynomial interpolation at a known set Ω

$$l_1(\omega) = 1, l_1(\omega^2) = 0, l_1(\omega^3) = 0$$

Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$l_i(x)$: is c_i the left input of gate j , for $j = 1,2,3$?

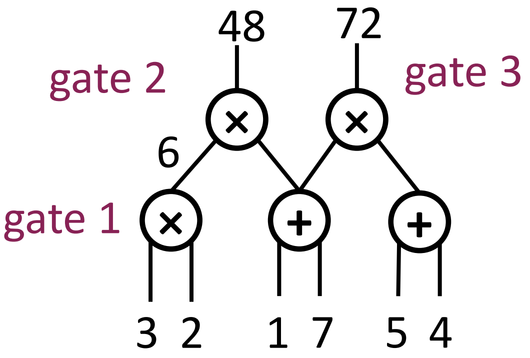
e.g., $l_2(x) : (0,0,0)$

Polynomial interpolation at a known set Ω

$$l_2(\omega) = 0, l_2(\omega^2) = 0, l_2(\omega^3) = 0$$

Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$l_i(x)$: is c_i the left input of gate j, for $j = 1,2,3$?

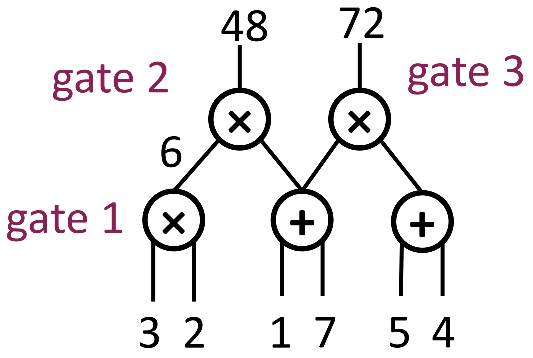
e.g., $l_3(x) : (0,0,\mathbf{1})$

Polynomial interpolation at a known set Ω

$$l_3(\omega) = 0, l_3(\omega^2) = 0, l_3(\omega^3) = 1$$

Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



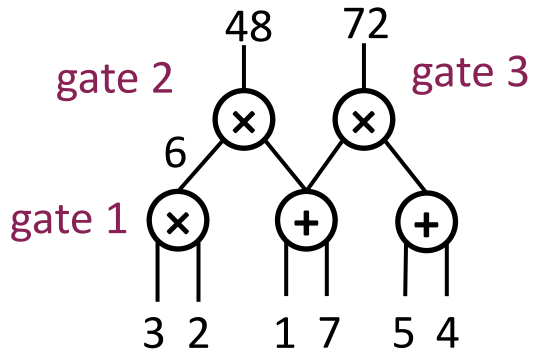
$l_i(x)$: is c_i the left input of gate j, for $j = 1,2,3$?

$\omega, \omega^2, \omega^3$

$l_1(x) : (1,0,0)$
$l_2(x) : (0,0,0)$
$l_3(x) : (0,0,1)$
$l_4(x) : (0,0,1)$
$l_5(x) : (0,0,0)$
$l_6(x) : (0,0,0)$
$l_7(x) : (0,1,0)$
$l_8(x) : (0,0,0)$
$l_9(x) : (0,0,0)$

Properties of the selector polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



- $L(x) = \sum_{i=1}^{9} c_i \times l_i(x)$

What is $L(\omega)$?

$$L(\omega) = c_1 = 3$$

What is $L(\omega^2)$?

$$L(\omega^2) = c_7 = 6$$

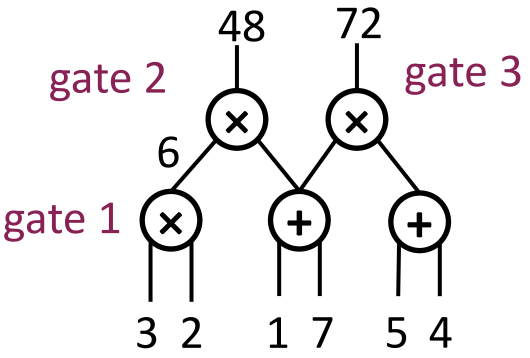
What is $L(\omega^3)$?

$$L(\omega^3) = c_3 + c_4 = 8$$

$\omega, \omega^2, \omega^3$
$l_1(x): (1,0,0)$
$l_2(x): (0,0,0)$
$l_3(x): (0,0,1)$
$l_4(x): (0,0,1)$
$l_5(x): (0,0,0)$
$l_6(x): (0,0,0)$
$l_7(x): (0,1,0)$
...

More Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$r_i(x)$: is c_i the right input of gate j , for $j = 1,2,3$?

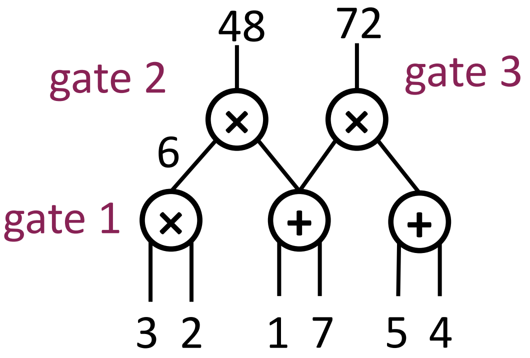
e.g., $r_1(x)$: $(0,0,0)$

Polynomial interpolation at a known set Ω

$$r_1(\omega) = 0, r_1(\omega^2) = 0, r_1(\omega^3) = 0$$

More Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



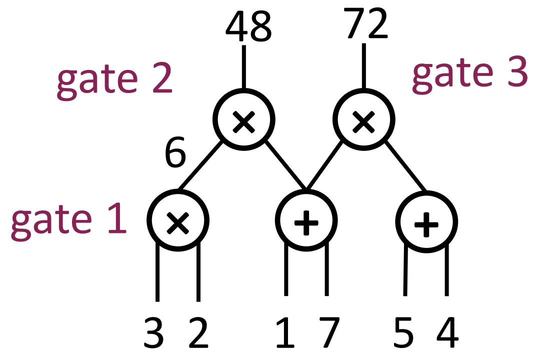
$r_i(x)$: is c_i the right input of gate j ?

ω , ω^2, ω^3

$r_1(x) : (0,0,0)$
$r_2(x) : (1,0,0)$
$r_3(x) : (0,1,0)$
$r_4(x) : (0,1,0)$
$r_5(x) : (0,0,1)$
$r_6(x) : (0,0,1)$
$r_7(x) : (0,0,0)$
$r_8(x) : (0,0,0)$
$r_9(x) : (0,0,0)$

Properties of the selector polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



- $R(x) = \sum_{i=1}^{9} c_i \times r_i(x)$

What is $R(\omega)$?

$$R(\omega) = c_2 = 2$$

What is $R(\omega^2)$?

$$R(\omega^2) = c_3 + c_4 = 8$$

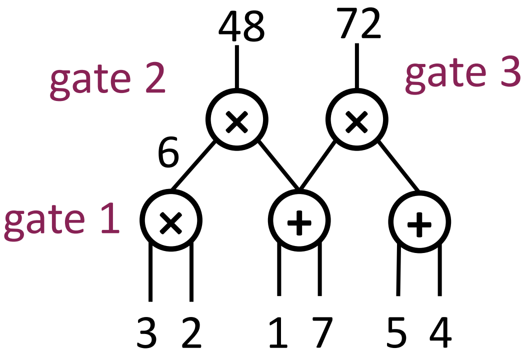
What is $L(\omega^3)$?

$$R(\omega^3) = c_5 + c_6 = 9$$

$r_1(x) : (0,0,0)$
$r_2(x) : (1,0,0)$
$r_3(x) : (0,1,0)$
$r_4(x) : (0,1,0)$
$r_5(x) : (0,0,1)$
$r_6(x) : (0,0,1)$
$r_7(x) : (0,0,0)$
$r_8(x) : (0,0,0)$
$r_9(x) : (0,0,0)$

More Selector Polynomials

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
3	2	1	7	5	4	6	48	72



$o_i(x)$: is c_i the output of gate j, for $j = 1,2,3$?

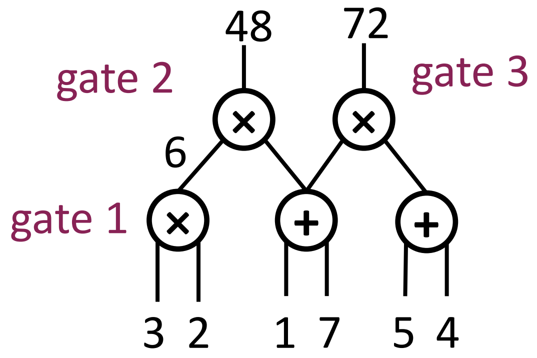
e.g., $o_1(x)$: $(0,0,0)$

Polynomial interpolation at a known set Ω

$$o_1(\omega) = 0, o_1(\omega^2) = 0, o_1(\omega^3) = 0$$

More Selector Polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$o_i(x)$: is c_i the output of gate j ?

$\omega\,, \omega^2, \omega^3$

$o_1(x) : (0,0,0)$

$o_2(x) : (0,0,0)$

$o_3(x) : (0,0,0)$

$o_4(x) : (0,0,0)$

$o_5(x) : (0,0,0)$

$o_6(x) : (0,0,0)$

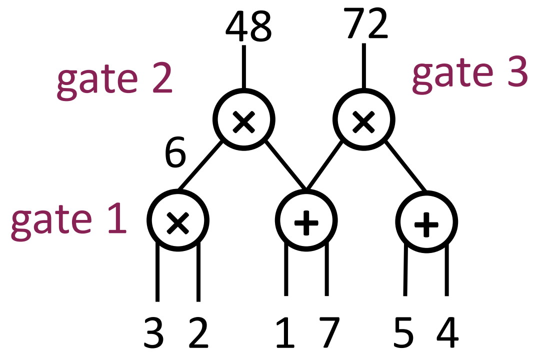
$o_7(x) : (1,0,0)$

$o_8(x) : (0,1,0)$

$o_9(x) : (0,0,1)$

Properties of the selector polynomials

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



- $O(x) = \sum_{i=1}^{9} c_i \times o_i(x)$

$O(\omega) = c_7 = 6$

$O(\omega^2) = c_8 = 48$

$O(\omega^3) = c_9 = 72$

$o_1(x) : (0,0,0)$
$o_2(x) : (0,0,0)$
$o_3(x) : (0,0,0)$
$o_4(x) : (0,0,0)$
$o_5(x) : (0,0,0)$
$o_6(x) : (0,0,0)$
$o_7(x) : (1,0,0)$
$o_8(x) : (0,1,0)$
$o_9(x) : (0,0,1)$

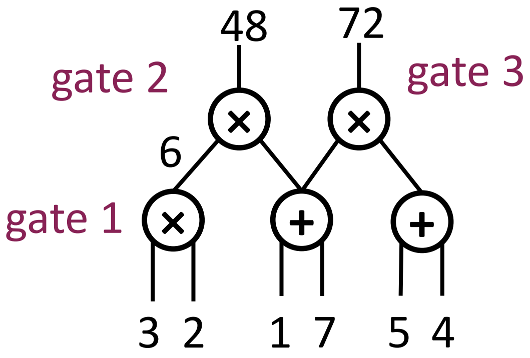
Master polynomial

$$p(x) = L(x)R(x) - O(x)$$
$$= \left(\sum_{i=1}^{9} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{9} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{9} c_i \times o_i(x)\right)$$

- Claim: $p(\omega^j) = 0$ for $j = 1,2,3$

Master polynomial

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
3	2	1	7	5	4	6	48	72



$$p(x) = L(x)R(x) - O(x)$$

- $L(x) = \sum_{i=1}^{9} c_i \times l_i(x)$

$L(\omega) = c_1 = 3$

- $R(x) = \sum_{i=1}^{9} c_i \times r_i(x)$

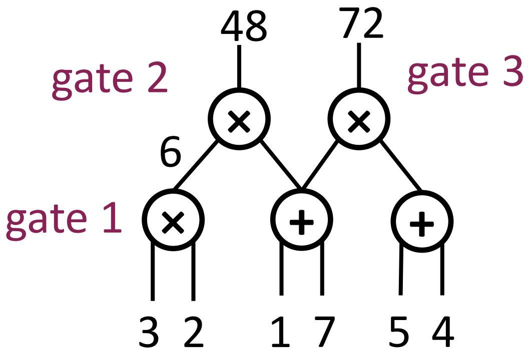
$R(\omega) = c_2 = 2$

- $O(x) = \sum_{i=1}^{9} c_i \times o_i(x)$

$O(\omega) = c_7 = 6$

Master polynomial

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
3	2	1	7	5	4	6	48	72



$$p(x) = L(x)R(x) - O(x)$$

- $L(x) = \sum_{i=1}^{9} c_i \times l_i(x)$

$L(\omega^2) = c_7 = 6$

- $R(x) = \sum_{i=1}^{9} c_i \times r_i(x)$

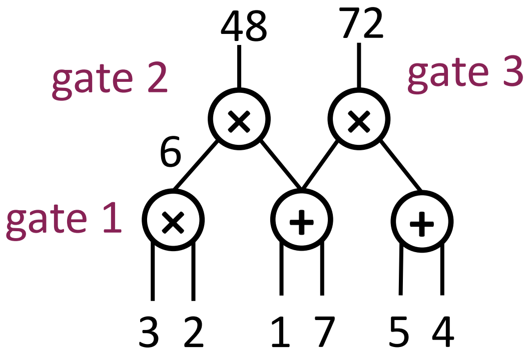
$R(\omega^2) = c_3 + c_4 = 1 + 7 = 8$

- $O(x) = \sum_{i=1}^{9} c_i \times o_i(x)$

$O(\omega^2) = c_8 = 48$

Master polynomial

3	2	1	7	5	4	6	48	72
c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9



$$p(x) = L(x)R(x) - O(x)$$

- $L(x) = \sum_{i=1}^{9} c_i \times l_i(x)$

$L(\omega^3) = c_3 + c_4 = 1 + 7 = 8$

- $R(x) = \sum_{i=1}^{9} c_i \times r_i(x)$

$R(\omega^3) = c_5 + c_6 = 5 + 4 = 9$

- $O(x) = \sum_{i=1}^{9} c_i \times o_i(x)$

$O(\omega^3) = c_9 = 72$

Vanishing polynomial

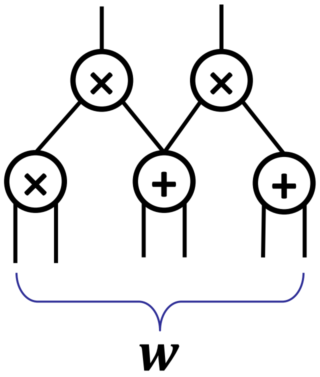
- $p(x) = L(x)R(x) - O(x)$
$= \left(\sum_{i=1}^{9} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{9} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{9} c_i \times o_i(x)\right)$

- $p(\omega^j) = 0$ for $j = 1,2,3$

- $p(x) = V(x)q(x)$, where $V(x) = (x - \omega)(x - \omega^2)(x - \omega^3)$ is the vanishing polynomial of the set $\Omega = \{\omega, \omega^2, \omega^3\}$

Circuit-SAT to QAP [GGPR13, PGHR13]

P claims to know a w such that $C(x, w) = y$



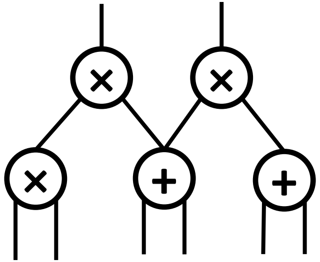
P claims to know a vector c such that $p(x) = V(x)q(x)$

m: size of c, extended witness

n: number of multiplication gates

Circuit-SAT to QAP [GGPR13, PGHR13]

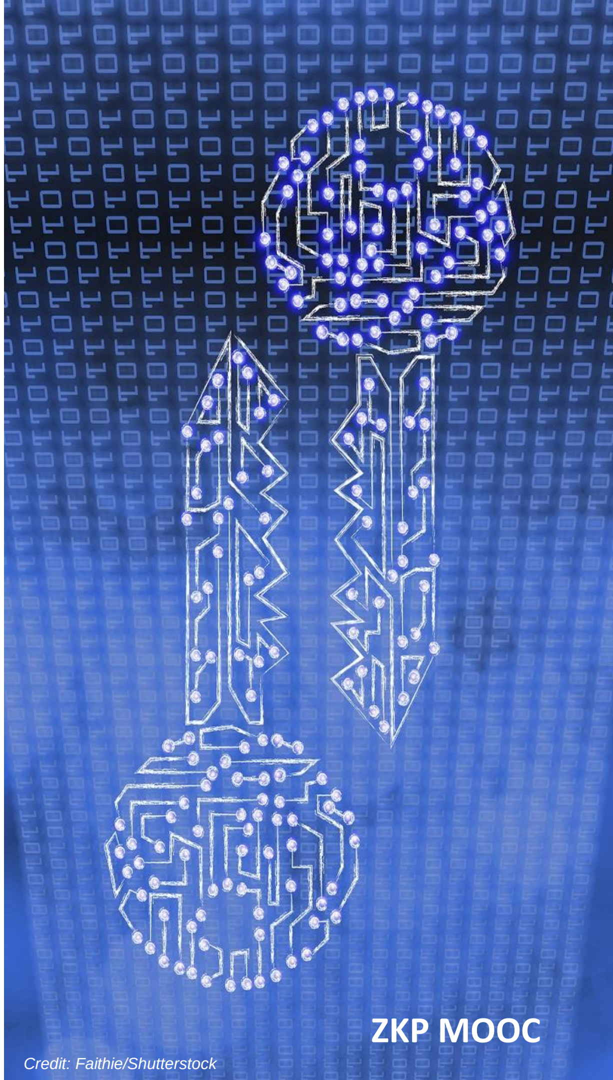
$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$



w

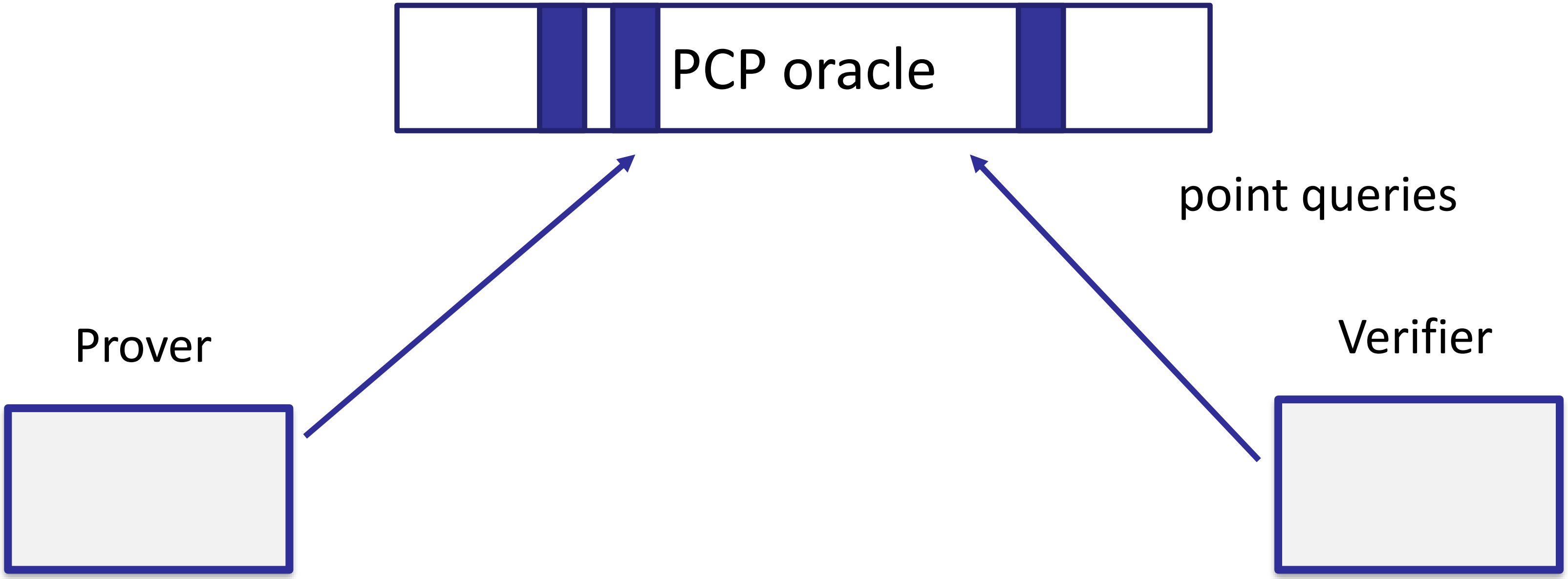
$\Omega = \omega, \omega^2, \dots \omega^n$		
$l_1(x) : (1,0,0, \dots)$	$r_1(x) : (0,0,0, \dots)$	$o_1(x) : (0,0,0, \dots)$
$l_2(x) : (0,0,0, \dots)$	$r_2(x) : (1,0,0, \dots)$	$o_2(x) : (0,0,0, \dots)$
$l_3(x) : (0,0,1, \dots)$	$r_3(x) : (0,1,0, \dots)$	$o_3(x) : (0,0,0, \dots)$
…	…	…
$l_m(x) : (0,0,1, \dots)$	$r_m(x) : (0,0,1, \dots)$	$o_m(x) : (0,0,0, \dots)$

From QAP to SNARK

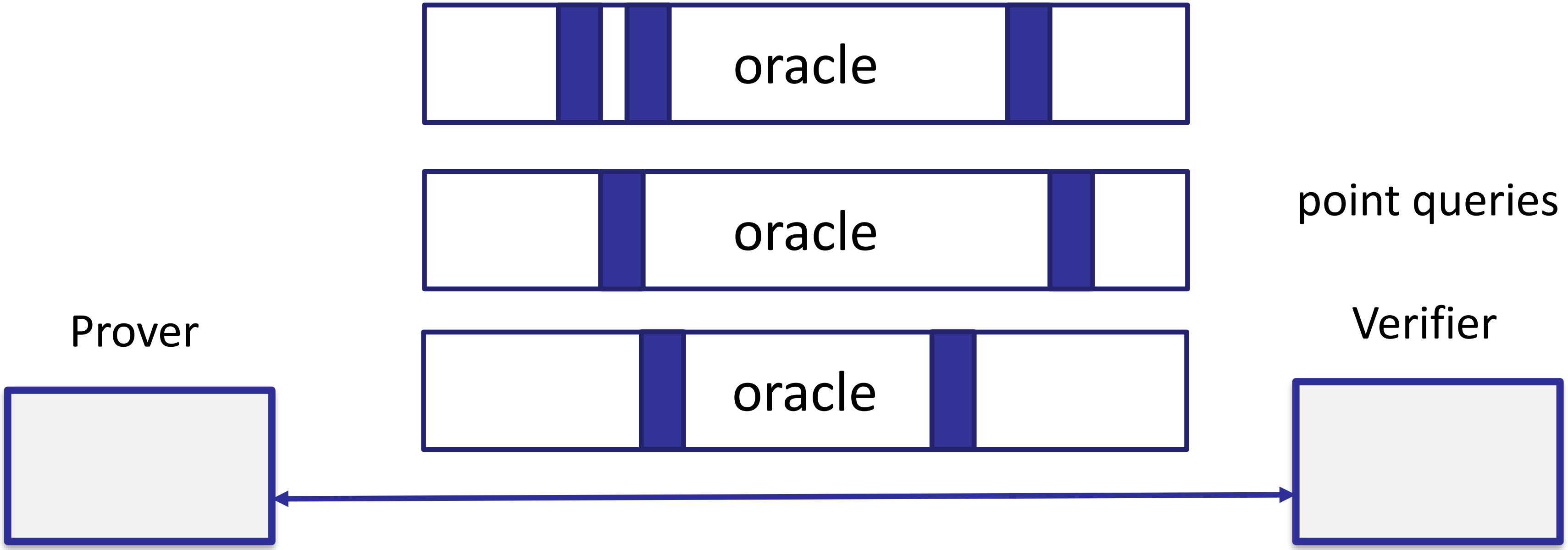


Credit: Faithie/Shutterstock

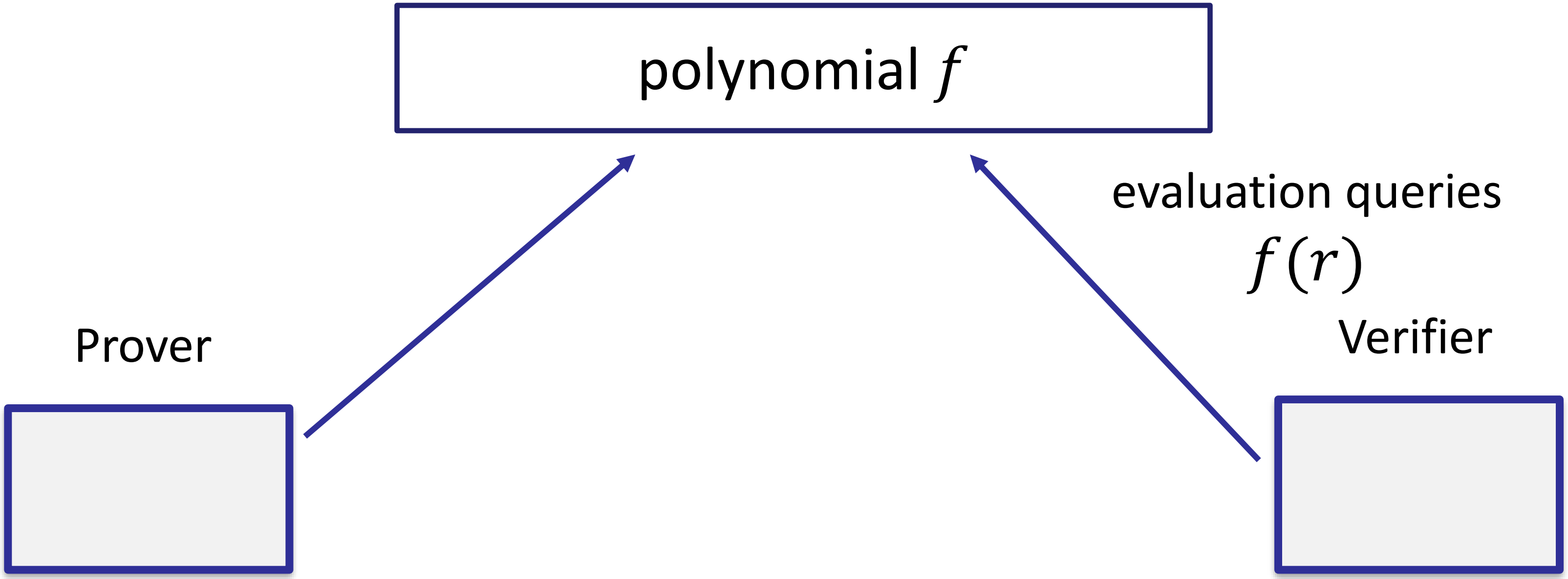
Probabilistically Checkable Proofs (PCP)



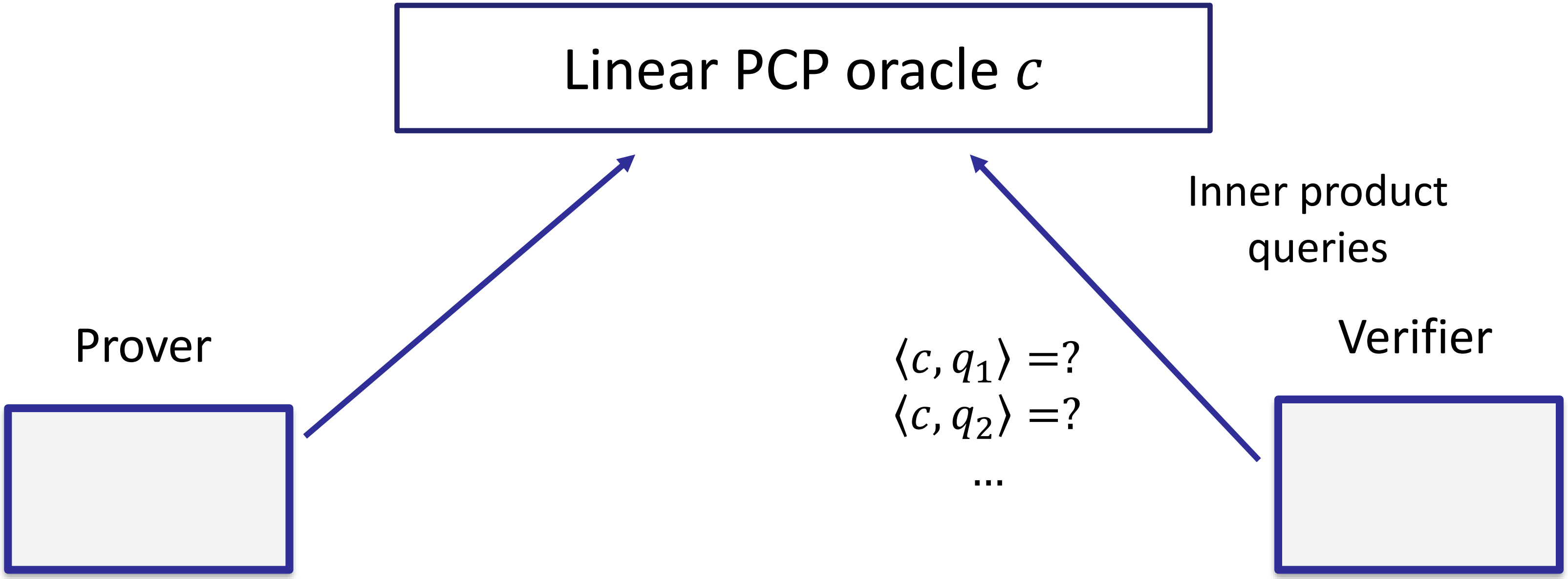
IPCP [Kalai-Raz'08] and IOP [Ben-Sasson-Chiesa-Spooner'16]



Polynomial IOP [Bünz-Fisch-Szepieniec'20]



Linear PCP [Ishai-Kushilevitz-Ostrovsky'07]



QAP and Linear PCP

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

extended witness c and coefficients of $q(x)$

Prover

Verifier

$$\langle c, \boldsymbol{l_i}(\gamma) \rangle \times \langle c, \boldsymbol{r_i}(\gamma) \rangle - \langle c, \boldsymbol{o_i}(\gamma) \rangle = V(\gamma)\langle q, \boldsymbol{\gamma} \rangle$$

Recall: Bilinear pairing

- $(p, \mathbb{G}, g, \mathbb{G}_T, e)$
  - $\mathbb{G}$ and $\mathbb{G}_T$ are both multiplicative cyclic group of order p, g is the generator of $\mathbb{G}$.

  $\mathbb{G}$:base group, $\mathbb{G}_T$ target group
- Pairing: $e(P^x, Q^y) = e(P, Q)^{xy} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$
- Example: $e(g^x, g^y) = e(g, g)^{xy} = e(g^{xy}, g)$

Given $g^x$and g^y, a pairing can check that some element $h = g^{xy}$ without knowing x and y

Key Generation

$$p(x) = (\sum_{i=1}^{m} c_i \times l_i(x)) \times (\sum_{i=1}^{m} c_i \times r_i(x)) - (\sum_{i=1}^{m} c_i \times o_i(x)) = V(x)q(x)$$

Preprocessor

Proving key: $p, \mathbb{G}, g, \mathbb{G}_T, e$

$g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}$ for $i = 1, \ldots, m$

$g^{\tau}, g^{\tau^2}, \ldots, g^{\tau^m}$

Verification key: $g^{V(\tau)}$

Prover

Verifier

delete τ !! (trusted setup)

Prove

$$p(x) = (\sum_{i=1}^{m} c_i \times l_i(x)) \times (\sum_{i=1}^{m} c_i \times r_i(x)) - (\sum_{i=1}^{m} c_i \times o_i(x)) = V(x)q(x)$$

PK: $g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}$
for $i = 1, \ldots, m$
$g^{\tau}, g^{\tau^2}, \ldots, g^{\tau^m}$

$$\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}$$
$$\pi_2 = g^{\sum_{i=1}^{m} c_i \times r_i(\tau)}$$
$$\pi_3 = g^{\sum_{i=1}^{m} c_i \times o_i(\tau)}$$
$$\pi_4 = g^{q(\tau)}$$

Prover $\longrightarrow$ Verifier

Verify

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

PK: $g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}$
for $i = 1, \dots, m$
$g^{\tau}, g^{\tau^2}, \dots, g^{\tau^m}$

Verification key: $g^{V(\tau)}$

$$\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}$$
$$\pi_2 = g^{\sum_{i=1}^{m} c_i \times r_i(\tau)}$$
$$\pi_3 = g^{\sum_{i=1}^{m} c_i \times o_i(\tau)}$$
$$\pi_4 = g^{q(\tau)}$$

$$e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{V(\tau)}, \pi_4)$$

Prover

Verifier

Towards the real protocol

- Q1: How to make sure π_1 is computed from $g^{l_i(\tau)}$?

- Solution: Knowledge of Exponent assumption (KoE) or Generic Group Model (GGM)

Recall: KoE

- Sample random α, compute $g^{\alpha l_i(\tau)}$ for $i = 1, \ldots, m$
- $\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}, \pi_1{}' = g^{\alpha \sum_{i=1}^{m} c_i \times l_i(\tau)}$
- $e(\pi_1, g^{\alpha}) = e(\pi_1{}', g)$
- Used in [PGHR13]

Recall: GGM

- (Informal) Adversary is only give an oracle to compute the group operation.

E.g., given $g^{\alpha l_i(\tau)}$ for $i = 1, \ldots, m$, Adv can only compute their linear combinations.

- Used in [Groth16]

Towards the real protocol

- Q2: how to make sure the same c is used in π_1, π_2, π_3?

Solution

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

PK: $g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}$

$g^{\tau}, g^{\tau^2}, \dots, g^{\tau^m}$

$g^{\beta(l_i(\tau)+r_i(\tau)+o_i(\tau))}$ for $i \in [m]$

g^{β}

Prover

$$\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}$$

$$\pi_2 = g^{\sum_{i=1}^{m} c_i \times r_i(\tau)}$$

$$\pi_3 = g^{\sum_{i=1}^{m} c_i \times o_i(\tau)}$$

$$\pi_4 = g^{q(\tau)}$$

$$e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{V(\tau)}, \pi_4)$$

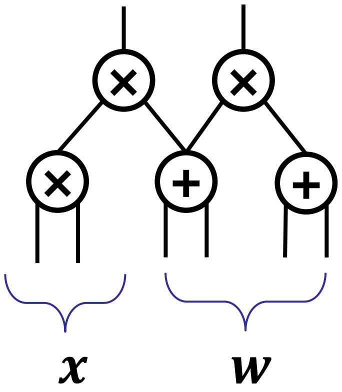
$$e(\pi_1 \pi_2 \pi_3, g^{\beta}) = e(\pi_5, g)$$

Verifier

$$\pi_5 = \Pi_{i=1}^{m} \left(g^{\beta(l_i(\tau)+r_i(\tau)+o_i(\tau))}\right)^{c_i}$$

Towards the real protocol

- Q3: what about public input and output?



$$C(x, w) = y$$

Solution

Original $\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}$

New $\pi_1 = g^{\sum_{i \in I_{mid}} c_i \times l_i(\tau)}$

$g^{\sum_{i \in I_{io}} c_i \times l_i(\tau)}$

$$\pi_1^* = \pi_1 \cdot g^{\sum_{i \in I_{io}} c_i \times l_i(\tau)}$$

$$e(\pi_1^*, \pi_2^*)/e(\pi_3^*, g) = e(g^{V(\tau)}, \pi_4)$$

Prover

$\pi_1 = g^{\sum_{i \in I_{mid}} c_i \times l_i(\tau)}$

Verifier

Putting everything together

- Keygen:

  PK: $g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}, g^{\beta(l_i(\tau)+r_i(\tau)+o_i(\tau))}$ for $i \in I_{mid}, g^{\beta}, g^{\tau}, g^{\tau^2}, \ldots, g^{\tau^m}$

  VK:$g^{l_i(\tau)}, g^{r_i(\tau)}, g^{o_i(\tau)}$ for $i \in I_{io}, g^{V(\tau)}$

- Prove: $\pi_1 = g^{L(\tau)}, \pi_2 = g^{R(\tau)}, \pi_3 = g^{O(\tau)}, \pi_4 = g^{q(\tau)}, \pi_5 = g^{\beta(L(\tau)+R(\tau)+O(\tau))}$

- Verify: $\pi_1^* = \pi_1 \cdot g^{\sum_{i\in I_{io}} c_i \times l_i(\tau)}, \pi_2^*, \pi_3^*,$
  - $e(\pi_1^*, \pi_2^*)/e(\pi_3^*, g) = e(g^{V(\tau)}, \pi_4)$
  - $e\left(\pi_1\pi_2\pi_3, g^{\beta}\right) = e(\pi_5, g)$

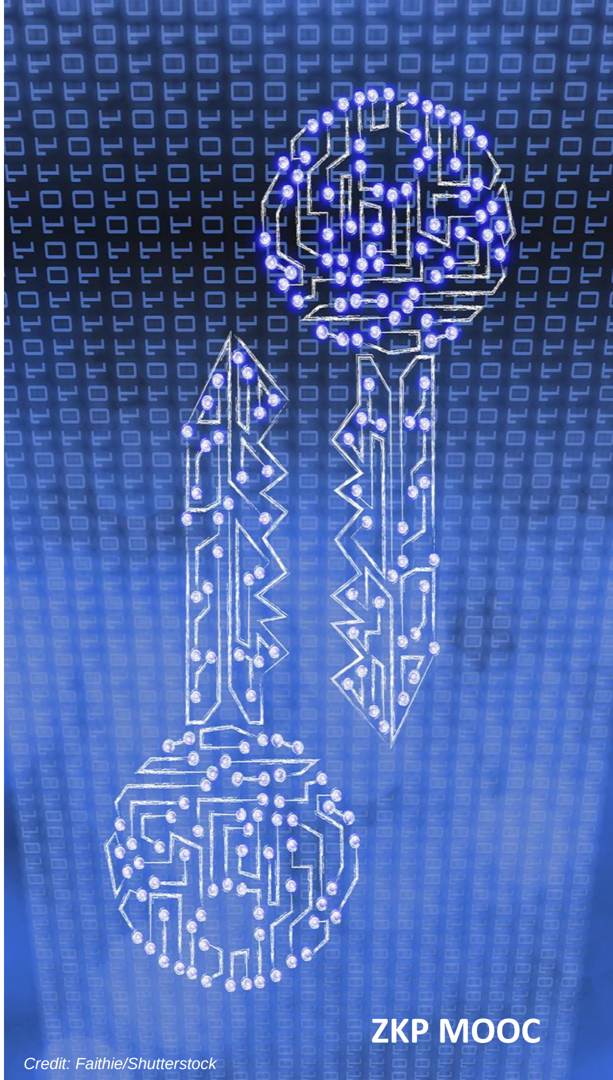
Properties of SNARK [PGHR13]

- Per-circuit trusted setup: $O(C)$ group exponentiations due to sparsity
- Prover time: $O(C \log C)$ FFT, $O(C)$ group exponentiations

✓ Proof size: $O(1)$, hundreds of bytes only

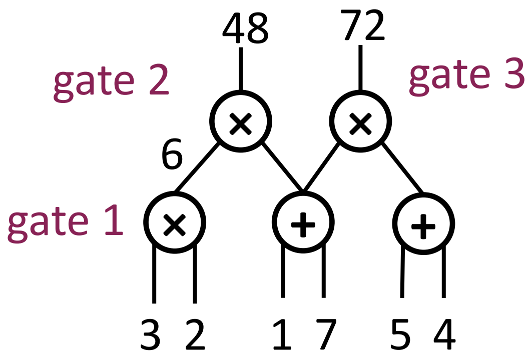
✓ Verifier time: $O(1)$ pairing + $O(|IO|)$ group exponentiations

Other Variants



Rank-1-Constraint-System (R1CS)

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
3	2	1	7	5	4	6	48	72



$\omega, \omega^2, \omega^3$

$l_1(x) : (1,0,0)$
$l_2(x) : (0,0,0)$
$l_3(x) : (0,0,1)$
$l_4(x) : (0,0,1)$
$l_5(x) : (0,0,0)$
$l_6(x) : (0,0,0)$
$l_7(x) : (0,1,0)$
$l_8(x) : (0,0,0)$
$l_9(x) : (0,0,0)$

$l_i(x)$: is c_i the left input of gate j, for $j = 1,2,3$?

Rank-1-Constraint-System (R1CS)

c_1	c_2	c_3	…	…	…	c_m

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right)$$

$$l_i(\omega^j) = \begin{cases} 1, & \text{if } c_i \text{ is the left input of multiplication gate } j \\ \text{Any public constant} & \\ 0, & \text{otherwise} \end{cases}$$

Example: Constraint: $(3c_1 + 5c_5 - 7c_7) \times (6c_2 + 10c_9) - (c_3 - 2c_8) = 0$

$l_1(\omega) = 3, \quad l_5(\omega) = 5, \quad l_7(\omega) = -7$

$r_2(\omega) = 6, \quad r_9(\omega) = 10, \quad \ldots$

Matrix View of R1CS

- m: size of the extended witness; n: number of constraints

$$n\left\{\underbrace{\boxed{\quad L \quad}}_{m}\right. \times \boxed{c} \otimes \boxed{\quad R \quad} \times \boxed{c} = \boxed{\quad O \quad} \times \boxed{c}$$

Building blocks for SNARKs: Linear check + Hadamard product check

Used in Bulletproofs, Marlin, Spartan, …

Groth16

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

- $\pi_1 = g^{\alpha + \sum_{i=1}^{m} c_i \times l_i(\tau)}$
- $\pi_2 = g^{\beta + \sum_{i=1}^{m} c_i \times r_i(\tau)}$
- $\pi_3 = g^{\sum c_i \times \left(\beta l_i(\tau) + \alpha r_i(\tau) + o_i(\tau)\right) + V(\tau)q(\tau)}$

- Verify: $e(\pi_1, \pi_2) = e(\pi_3, g)e(g^\alpha, g^\beta)$

Groth16

- Change the keygen accordingly
- Proof size: 3 group elements, 144 bytes
- Verifier time: 1 pairing equation

Achieving Zero-Knowledge

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

$$\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau)}$$
$$\pi_2 = g^{\sum_{i=1}^{m} c_i \times r_i(\tau)}$$
$$\pi_3 = g^{\sum_{i=1}^{m} c_i \times o_i(\tau)}$$
$$\pi_4 = g^{q(\tau)}$$

$$e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{V(\tau)}, \pi_4)$$

Prover

Verifier

Achieving Zero-Knowledge

$$p(x) = \left(\sum_{i=1}^{m} c_i \times l_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times r_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times o_i(x)\right) = V(x)q(x)$$

$$\pi_1 = g^{\sum_{i=1}^{m} c_i \times l_i(\tau) + \delta_1 V(\tau)}$$
$$\pi_2 = g^{\sum_{i=1}^{m} c_i \times r_i(\tau) + \delta_2 V(\tau)}$$
$$\pi_3 = g^{\sum_{i=1}^{m} c_i \times o_i(\tau) + \delta_3 V(\tau)}$$
$$\pi_4 = g^{q(\tau)}$$

$$e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{V(\tau)}, \pi_4)$$

Prover → Verifier

End of Lecture

Next: Recursive SNARKs