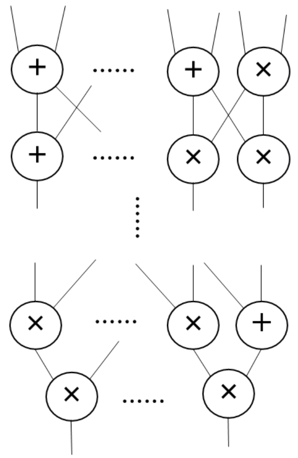
Generic verifiable computation and zero knowledge proof

SNARK



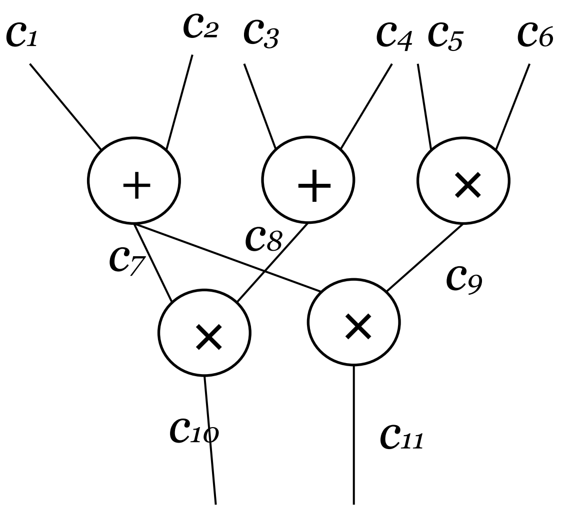
server

client

result = C(data) + 

data

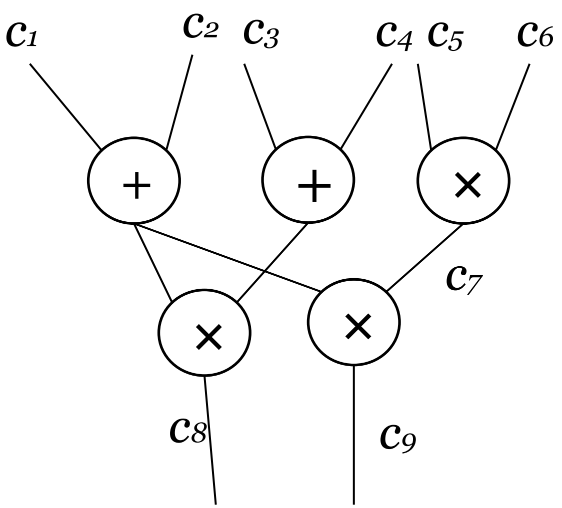
Satisfying assignment of circuits



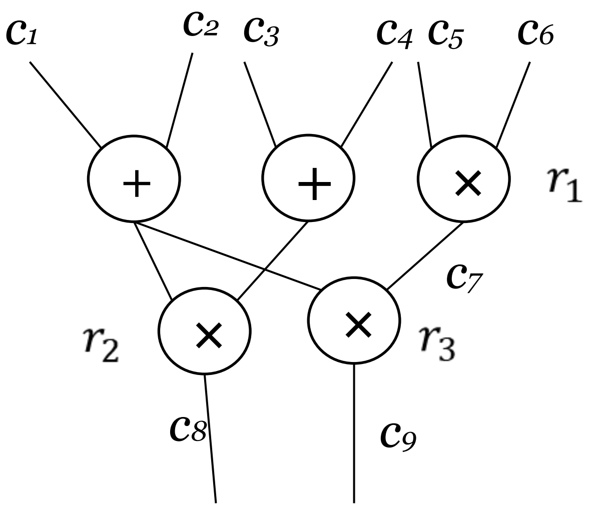
Proving C(data) = output → $(c_1, c_2, \ldots c_{11})$ with conditions defined by the circuit

Verifying is easier than computing

Labeling of wires and gates

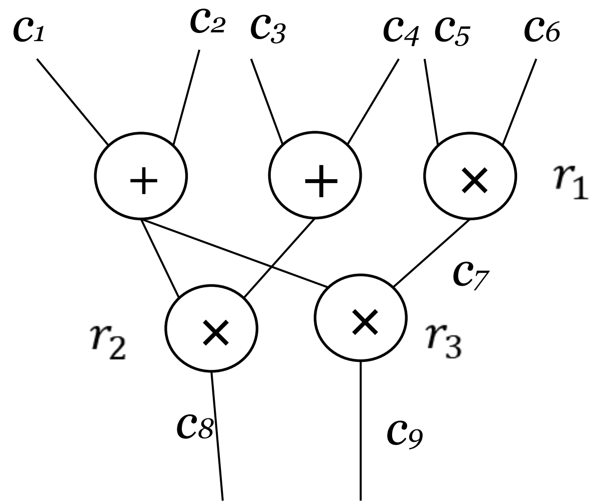


Encoding circuits to polynomials



Circuit SAT to polynomial division

- $p(x) = (\sum_{i=1}^{m} c_i \times v_i(x)) \times (\sum_{i=1}^{m} c_i \times w_i(x)) - (\sum_{i=1}^{m} c_i \times y_i(x))$



Circuit SAT to polynomial division

- $p(x) = (\sum_{i=1}^{m} c_i \times v_i(x)) \times (\sum_{i=1}^{m} c_i \times w_i(x)) - (\sum_{i=1}^{m} c_i \times y_i(x))$

- Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Generic verifiable computation from QAP

s

(p, g, G, e, G_T)

server

Public key:
$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$
for $i = 1, \ldots, m$

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

client

$$\pi = g^{\sum_{i=1}^m c_i \times v_i(s)}, g^{\sum_{i=1}^m c_i \times w_i(s)}, g^{\sum_{i=1}^m c_i \times y_i(s)}, g^{h(s)}$$

$g^{t(s)}$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

$$p(x) = \left(\sum_{i=1}^m c_i \times v_i(x)\right) \times \left(\sum_{i=1}^m c_i \times w_i(x)\right) - \left(\sum_{i=1}^m c_i \times y_i(x)\right)$$

Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Complexity: setup

s

(p, g, G, e, G_T)

server

Public key:
$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$
for $i = 1, \dots, m$

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

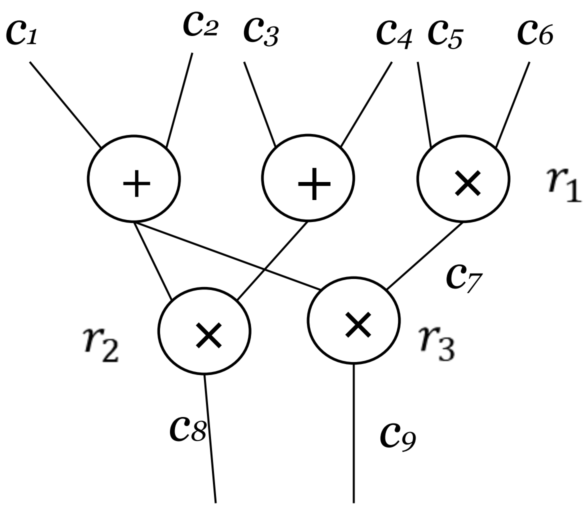
client

$g^{t(s)}$

- Naively computing takes $O(m^2)$ time
- Can be done in O(m) time because of sparsity

Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Encoding circuits to polynomials



Complexity: prover time

s

(p, g, G, e, G_T)

server

Public key:
$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$
for $i = 1, \ldots, m$

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

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)}, g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$

$g^{t(s)}$

$$p(x) = \left(\sum_{i=1}^{m} c_i \times v_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times w_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times y_i(x)\right)$$

Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Compute: $h(x) = \frac{p(x)}{t(x)}$
$O(m \log^2 m)$ using FFT

Complexity: proof size and verification

s

(p, g, G, e, G_T)

server

Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$

for $i = 1, \dots, m$

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

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)}, g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$

$g^{t(s)}$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

$$p(x) = \left(\sum_{i=1}^{m} c_i \times v_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times w_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times y_i(x)\right)$$

Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Problem 1: form of polynomials

server

Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$

for $i = 1, \dots, m$

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

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)}, g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

How to make sure π is the right form?

Knowledge of exponent assumption

(p, g, G, e, G_T) $g^s, g^{s^2}, \dots, g^{s^q}$

s

α $g^\alpha, g^{\alpha s}, g^{\alpha s^2}, \dots, g^{\alpha s^q}$

$\pi = g^{a_0 + a_1 s + a_2 s^2 + \cdots + a_q s^q}$

$\pi' = g^{\alpha(a_0 + a_1 s + a_2 s^2 + \cdots + a_q s^q)}$

Check: $e(\pi', g) = e(\pi, g^\alpha)$

Assumption: π must be of this form

Knowledge of exponent assumption

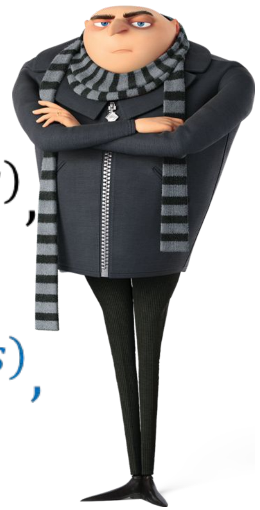
Assumption 2 (*q*-**PKE** [21]) *The q-power knowledge of exponent assumption holds for $\mathcal{G}$ if for all $\mathcal{A}$ there exists a non-uniform probabilistic polynomial time extractor $\chi_{\mathcal{A}}$ such that*

$$
\begin{aligned}
Pr[\quad & (p,\mathbb{G},\mathbb{G}_T,e) \leftarrow \mathcal{G}(1^\kappa)\ ;\ g \leftarrow \mathbb{G}\backslash\{1\}\ ;\ \alpha, s \leftarrow \mathbb{Z}_p^*\ ; \\
& \sigma \leftarrow (p,\mathbb{G},\mathbb{G}_T,e,g,g^s,\ldots,g^{s^q},g^\alpha,g^{\alpha s},\ldots,g^{\alpha s^q})\ ; \\
& (c,\hat{c}\ ;\ a_0,\ldots,a_q) \leftarrow (\mathcal{A} \parallel \chi_{\mathcal{A}})(\sigma,z)\ : \\
& \hat{c} = c^\alpha \wedge c \neq \textstyle\prod_{i=0}^{q} g^{a_i s^i}] = \mathtt{negl}(\kappa)
\end{aligned}
$$

for any auxiliary information $z \in \{0,1\}^{\mathrm{poly}(\kappa)}$ that is generated independently of α. Note that $(y;z) \leftarrow (\mathcal{A} \parallel \chi_{\mathcal{A}})(x)$ signifies that on input x, $\mathcal{A}$ outputs y, and that $\chi_{\mathcal{A}}$, given the same input x and $\mathcal{A}$'s random tape, produces z.

Problem 1: form of polynomials

server



Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)},$

$g^{\alpha_1 v_i(s)}, g^{\alpha_2 w_i(s)}, g^{\alpha_3 y_i(s)}$

for $i = 1, \ldots, m$

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

$g^{\alpha_1}, g^{\alpha_2}, g^{\alpha_3}$

client

$$\pi = g^{\sum_{i=1}^m c_i \times v_i(s)}, g^{\sum_{i=1}^m c_i \times w_i(s)}, g^{\sum_{i=1}^m c_i \times y_i(s)}, g^{h(s)}$$

$$g^{\alpha_1 \sum_{i=1}^m c_i \times v_i(s)}, g^{\alpha_2 \sum_{i=1}^m c_i \times w_i(s)}, g^{\alpha_3 \sum_{i=1}^m c_i \times y_i(s)}$$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

$$e(\pi_1', g) = e(\pi_1, g^{\alpha_1})$$
$$e(\pi_2', g) = e(\pi_2, g^{\alpha_2})$$
$$e(\pi_3', g) = e(\pi_3, g^{\alpha_3})$$

Problem 2: consistency of coefficients

server

Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)},$
$g^{\alpha_1 v_i(s)}, g^{\alpha_2 w_i(s)}, g^{\alpha_3 y_i(s)}$
for $i = 1, \ldots, m$

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)},$$
$$g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$
$$g^{\alpha_1 \sum_{i=1}^{m} c_i \times v_i(s)}, g^{\alpha_2 \sum_{i=1}^{m} c_i \times w_i(s)},$$
$$g^{\alpha_3 \sum_{i=1}^{m} c_i \times y_i(s)}$$

$g^{s}, g^{s^2}, \ldots, g^{s^m}$
$g^{\alpha_1}, g^{\alpha_2}, g^{\alpha_3}$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

$$e(\pi_1', g) = e(\pi_1, g^{\alpha_1})$$
$$e(\pi_2', g) = e(\pi_2, g^{\alpha_2})$$
$$e(\pi_3', g) = e(\pi_3, g^{\alpha_3})$$

How to make sure the same c_i s are used for all polynomials?

Problem 2: consistency of coefficients

server

Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)},$

$g^{\alpha_1 v_i(s)}, g^{\alpha_2 w_i(s)}, g^{\alpha_3 y_i(s)}$

$g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)}$

for $i = 1, \ldots, m$

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

$g^{\alpha_1}, g^{\alpha_2}, g^{\alpha_3}$

g^{β}

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)},$$
$$g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$
$$g^{\alpha_1 \sum_{i=1}^{m} c_i \times v_i(s)}, g^{\alpha_2 \sum_{i=1}^{m} c_i \times w_i(s)},$$
$$g^{\alpha_3 \sum_{i=1}^{m} c_i \times y_i(s)}$$
$$\Pi_{i=1}^{m} (g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)})^{c_i}$$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

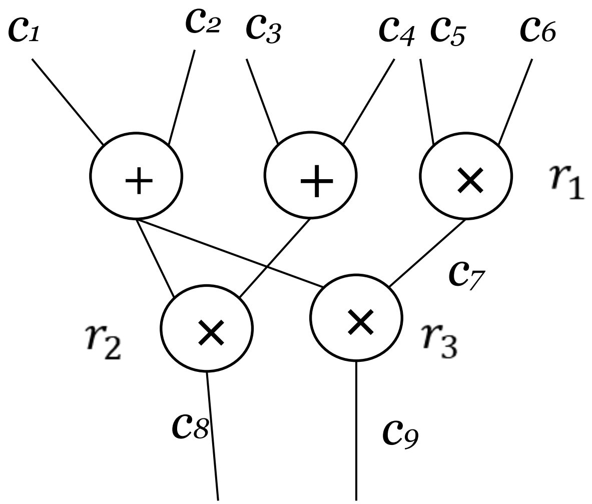
$$e(\pi_1', g) = e(\pi_1, g^{\alpha_1})$$
$$e(\pi_2', g) = e(\pi_2, g^{\alpha_2})$$
$$e(\pi_3', g) = e(\pi_3, g^{\alpha_3})$$
$$e(\pi_1 \cdot \pi_2 \cdot \pi_3, g^{\beta}) = e(Z, g)$$

Problem 3: input and output*

- Only proves there is a satisfying assignment

QAP

- $p(x) = (\sum_{i=1}^{m} c_i \times v_i(x)) \times (\sum_{i=1}^{m} c_i \times w_i(x)) - (\sum_{i=1}^{m} c_i \times y_i(x))$



SNARK

server

Public key:

$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)},$

$g^{\alpha_1 v_i(s)}, g^{\alpha_2 w_i(s)}, g^{\alpha_3 y_i(s)}$

$g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)}$

for $i = 1, \ldots, m$

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

$g^{\alpha_1}, g^{\alpha_2}, g^{\alpha_3}$

g^{β}

client

$$\pi = g^{\sum_{i=1}^{m} c_i \times v_i(s)}, g^{\sum_{i=1}^{m} c_i \times w_i(s)},$$

$$g^{\sum_{i=1}^{m} c_i \times y_i(s)}, g^{h(s)}$$

$$g^{\alpha_1 \sum_{i=1}^{m} c_i \times v_i(s)}, g^{\alpha_2 \sum_{i=1}^{m} c_i \times w_i(s)}$$

$$g^{\alpha_3 \sum_{i=1}^{m} c_i \times y_i(s)}$$

$$\Pi_{i=1}^{m} (g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)})^{c_i}$$

Verification: $e(\pi_1, \pi_2)/e(\pi_3, g) = e(g^{t(s)}, \pi_4)$

$$e(\pi_1', g) = e(\pi_1, g^{\alpha_1})$$

$$e(\pi_2', g) = e(\pi_2, g^{\alpha_2})$$

$$e(\pi_3', g) = e(\pi_3, g^{\alpha_3})$$

$$e\left(\pi_1 \cdot \pi_2 \cdot \pi_3, g^{\beta}\right) = e(Z, g)$$

SNARK

server

client

$$\pi = g^{\sum_{i\in I_{mid}} c_i \times v_i(s)}, g^{\sum_{i\in I_{mid}} c_i \times w_i(s)},$$
$$g^{\sum_{i\in I_{mid}} c_i \times y_i(s)}, g^{h(s)}$$
$$g^{\alpha_1 \sum_{i\in I_{mid}} c_i \times v_i(s)}, g^{\alpha_2 \sum_{i\in I_{mid}} c_i \times w_i(s)},$$
$$g^{\alpha_3 \sum_{i\in I_{mid}} c_i \times y_i(s)}$$
$$\Pi_{i\in I_{mid}} (g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)})^{c_i}$$

Public key:
$g^{v_i(s)}, g^{w_i(s)}, g^{y_i(s)}$,
$g^{\alpha_1 v_i(s)}, g^{\alpha_2 w_i(s)}, g^{\alpha_3 y_i(s)}$
$g^{\beta v_i(s)} \cdot g^{\beta w_i(s)} \cdot g^{\beta y_i(s)}$
for $i \in I_{mid}$

$$g^{s}, g^{s^2}, \ldots, g^{s^m}$$
$$g^{\alpha_1}, g^{\alpha_2}, g^{\alpha_3}$$
$$g^{\beta}$$

Verification: $e\left(\pi_1 \cdot g^{v_{io}(s)}, \pi_2 \cdot g^{w_{io}(s)}\right)/e(\pi_3 \cdot g^{y_{io}(s)}, g) = e(g^{t(s)}, \pi_4)$

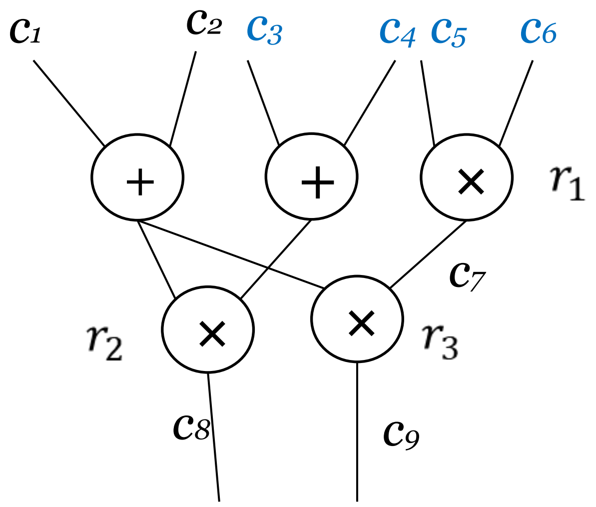
$$e(\pi_1', g) = e(\pi_1, g^{\alpha_1})$$
$$e(\pi_2', g) = e(\pi_2, g^{\alpha_2})$$
$$e(\pi_3', g) = e(\pi_3, g^{\alpha_3})$$
$$e\left(\pi_1 \cdot \pi_2 \cdot \pi_3, g^{\beta}\right) = e(Z, g)$$

Complexity

Verification time: O(1)+ size of input and output

Argument system

C(data, witness) = output



server

client

Zero knowledge

server

client

$$\pi = g^{\sum_{i \in I_{mid}} c_i \times v_i(s)}$$

$$\pi = g^{\sum_{i \in I_{mid}} c_i \times v_i(s) + \delta}$$

$$\pi = g^{\sum_{i \in I_{mid}} c_i \times v_i(s) + \delta t(s)}$$

$$p(x) = \left(\sum_{i=1}^{m} c_i \times v_i(x)\right) \times \left(\sum_{i=1}^{m} c_i \times w_i(x)\right) - \left(\sum_{i=1}^{m} c_i \times y_i(x)\right)$$

Target polynomial: $t(x) = (x - r_1)(x - r_2)(x - r_3)$

Summary of zkSNARK

- Circuit evaluation to satisfying assignment
- SAT to QAP
- QAP to argument (bilinear map, knowledge of exponent assumption)
- Argument to zero knowledge

Pros and Cons of zkSNARK

- ✔ Supports all functions (modeled as arithmetic circuit)
- ✔ Constant proof size
- ✔ Fast verification time

- × Function dependent **trusted setup**
- × Slow prover time (modular exponentiations for every gate)