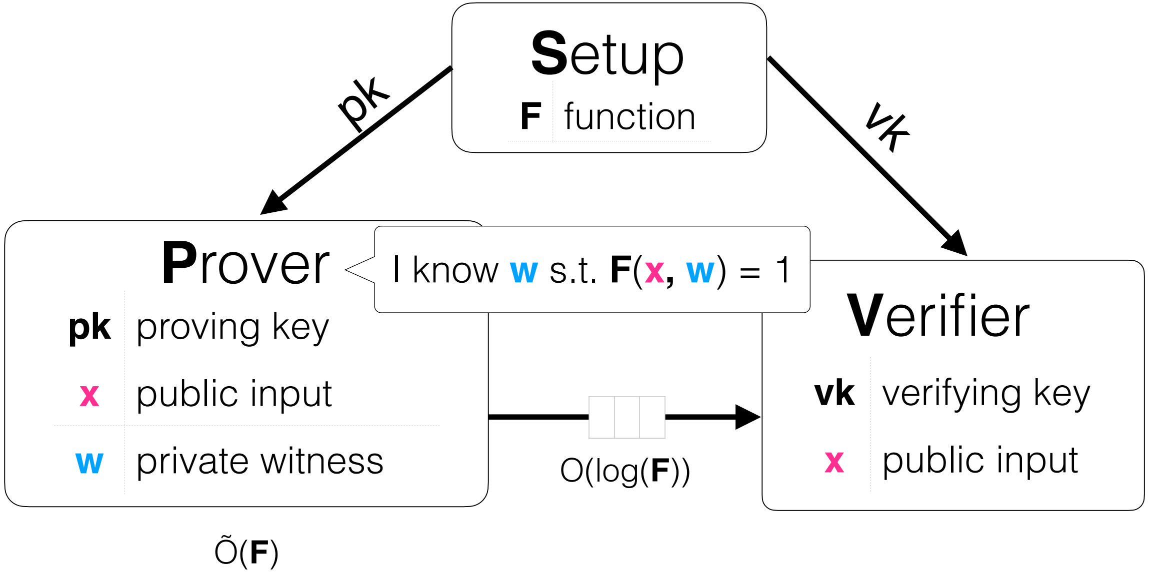
Theory and Practice of Succinct Zero Knowledge Proofs

Lecture 02: Modern zkSNARK Constructions

Pratyush Mishra
UPenn
Fall 2023

Succinct Non-Interactive Arguments (SNARGs)

[Mic94, Groth10, GGPR13, Groth16… …, GWC19, CHM**M**VW20, …]



Succinct Non-Interactive Arguments (SNARGs)

- **Completeness**: If $(F, x, w) \in \mathscr{R}$,
$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) = 1 \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w) \end{array}\right] = 1.$$

- **Soundness**: If $(F, x, w) \notin \mathscr{R}$, for all efficient provers $\tilde{\mathbf{P}}$
$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) = 1 \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \tilde{\mathbf{P}}(\mathsf{pk}, x) \end{array}\right] \approx 0$$

- **Succinctness**: $|\pi| = O(\log |F|)$

What if there's always a witness?

Soundness: If $(F, x, w) \notin \mathscr{R}$, then for all efficient provers $\tilde{\mathbf{P}}$

$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) = 1 \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \tilde{\mathbf{P}}(\mathsf{pk}, x) \end{array}\right] \approx 0$$

- $F(x, w) := \mathsf{SHA2}(w) \stackrel{?}{=} x$: there is always a preimage!
- $F((m, \mathsf{pk}), \sigma) := \mathsf{VerifySignature}(\mathsf{pk}, m, \sigma) \stackrel{?}{=} 1$: if pk is a valid public key, there is always a valid signature!
- Generally many examples where **witness always exists!**

SNARGs of Knowledge (SNARKs)

- **Completeness**: For all $(F, x, w) \in \mathscr{R}$,

$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) = 1 \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w) \end{array}\right] = 1.$$

- **Knowledge Soundness**: If $\mathbf{V}(\mathsf{vk}, x, \pi) = 1$, then $\tilde{\mathbf{P}}$ "knows" w such that $(F, x, w) \in \mathscr{R}$

- **Succinctness**: $|\pi| = O(\log |F|)$

SNARGs of Knowledge (SNARKs)

- **Completeness**: For all $(F, x, w) \in \mathscr{R}$,

$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) = 1 \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w) \end{array}\right] = 1.$$

- **Knowledge Soundness**: For each efficient $\tilde{\mathbf{P}}$ there exists an **extractor** $\mathbf{E}$ such that

$$\Pr\left[\begin{array}{c} V(\mathsf{vk}, x, \pi) = 1 \\ \wedge \\ (F, x, w) \notin \mathscr{R} \end{array} \;:\; \begin{array}{c} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \tilde{\mathbf{P}}(\mathsf{pk}, x) \\ w \leftarrow \mathbf{E}_{\tilde{\mathbf{P}}}(\mathsf{pk}, x) \end{array}\right] \approx 0$$

- **Succinctness**: $|\pi| = O(\log |F|)$

What about privacy?

- $F(x, w) := \mathsf{SHA2}(w) \stackrel{?}{=} x$:
  Does proof reveal info about preimage?

- $F((m, \mathsf{pk}), \sigma) := \mathsf{VerifySignature}(\mathsf{pk}, m, \sigma) \stackrel{?}{=} 1$:
  Does proof reveal info about which signature was used?

- $F(x = \mathsf{score}, w = \mathsf{credit\_hist}) := \mathsf{CreditModel}(w) \stackrel{?}{=} x$
  Does proof reveal info about credit history?

Verifier is the adversary now!

Zero Knowledge SNARKs (zkSNARKs)

- **Completeness**: For all $(F, x, w) \in \mathscr{R}$, ...
- **Knowledge Soundness**: For each efficient $\tilde{\mathbf{P}}$ there exists an **extractor** $\mathbf{E}$ such that …
- **Zero Knowledge**: Proof reveals no information to $\mathbf{V}$ other than validity of w
- **Succinctness**: $|\pi| = O(\log |F|)$

Zero Knowledge SNARKs (zkSNARKs)

- **Completeness**: For all $(F, x, w) \in \mathscr{R}$, ...

- **Knowledge Soundness**: For each efficient $\tilde{\mathbf{P}}$ there exists an **extractor** $\mathbf{E}$ such that …

- **Zero Knowledge**: For all $(F, x, w) \in R$, and all efficient $\tilde{\mathbf{V}}$ there exists an **simulator** Sim such that

$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{array}{c}(\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathsf{Sim}(\mathsf{pk}, x)\end{array}\right] = \Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{array}{c}(\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w)\end{array}\right]$$

- **Succinctness**: $|\pi| = O(\log |F|)$

Doesn't this break soundness?

$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{matrix} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathsf{Sim}(\mathsf{pk}, x) \end{matrix}\right] = \Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{matrix} (\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F) \\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w) \end{matrix}\right]$$

Sim has same success probability as honest prover!

This is actually okay: we provide Sim with additional powers!

- Interactive case: Sim can rewind verifier
- Non-interactive case: Sim gets "trapdoor"/secret information

zk Marker Demo

What about succinct verification?

Succinctness: $|\pi| = O(\log |F|)$

- $F(x, w) = \mathsf{SHA2}^{10^6}(w) \stackrel{?}{=} x$:
  Do I need to compute 10^6 hashes to verify proof?
- $F(x = \mathsf{score}, w = \mathsf{credit\_hist}) = \mathsf{CreditModel}(w) \stackrel{?}{=} x$
  Do I need to evaluate complex model to verify proof?

Strongly Succinct zkSNARKs

- **Completeness**: For all $(F, x, w) \in \mathscr{R}$, …

- **Knowledge Soundness**: For each efficient $\tilde{\mathbf{P}}$ there exists an **extractor** $\mathbb{E}$ such that …

- **Zero Knowledge**: For all $(F, x, w) \in R$, and all efficient $\tilde{\mathbf{V}}$ there exists an **simulator** Sim such that

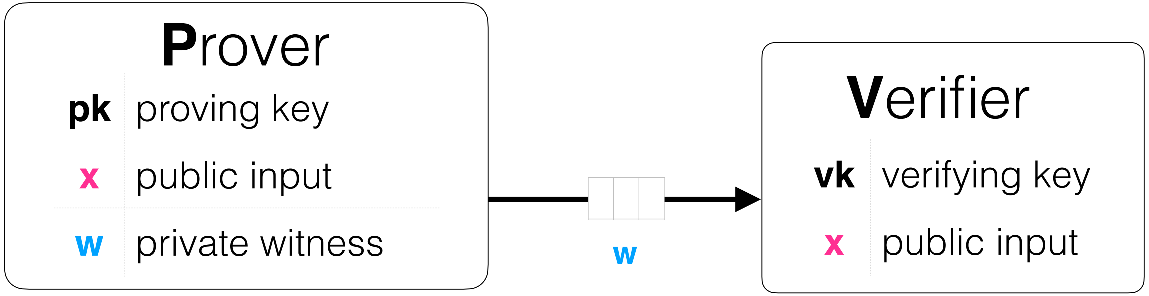
$$\Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{array}{c}(\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F)\\ \pi \leftarrow \mathsf{Sim}(\mathsf{pk}, x)\end{array}\right] = \Pr\left[\mathbf{V}(\mathsf{vk}, x, \pi) \;:\; \begin{array}{c}(\mathsf{pk}, \mathsf{vk}) \leftarrow \mathsf{Setup}(F)\\ \pi \leftarrow \mathbf{P}(\mathsf{pk}, x, w)\end{array}\right]$$

- **Succinctness**: $|\pi| = O(\log |F|)$

  and $\mathsf{Time}(\mathbf{V}) = O(\log |F|, |x|)$

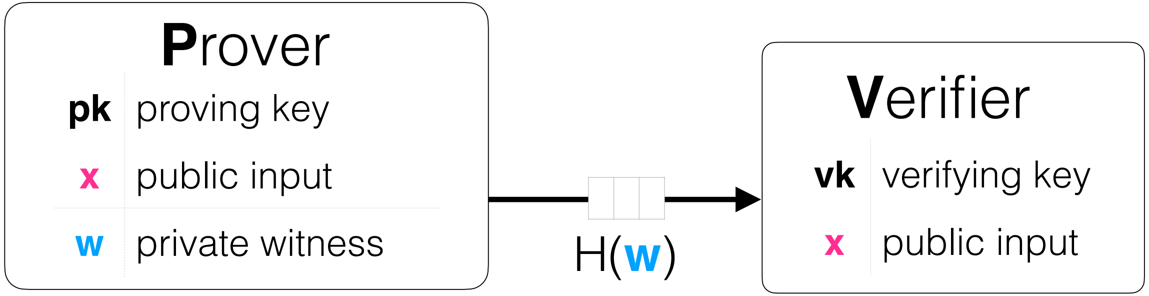
Constructing zkSNARKs

Starting point: Trivial NP Protocol



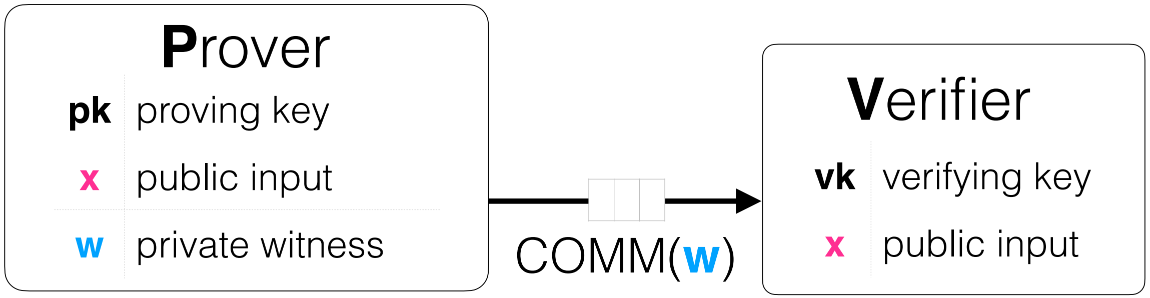
Problem 1: Non-succinct proof!
Problem 2: Non-succinct verification!
Problem 3: Not hiding at all!

Strawman 1: Hash the witness



Problem 1 solved: Succinct proof!
Problem 2: How to verify?
Problem 3: Still might not be hiding!

Strawman 2: Commit to the witness



Problem 1 solved: Succinct proof!
Problem 2: How to verify?
Problem 3: Still might not be hiding!

Commitment Schemes

$\text{Commit}(w; r) \rightarrow \text{cm}$

satisfying the following properties

- **Binding**: For all efficient adv. $\mathscr{A}$,
  $\Pr\left[\text{Commit}(w; r) = \text{Commit}(w'; r') \ : \ (w, r, w', r') \leftarrow \mathscr{A}\right] \approx 0$
  (no adv can open commitment to two diff values)

- **Hiding**: For all w, w', and all adv. $\mathscr{A}$,
  $\mathscr{A}(\text{Commit}(w; r)) = \mathscr{A}(\text{Commit}(w'; r'))$
  (no adv can learn committed value, i.e. comms are indistinguishable)

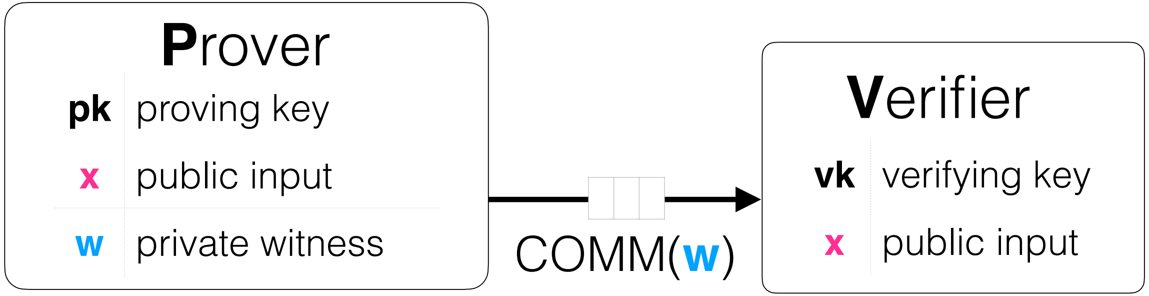
A standard construction

Let H be a cryptographic hash function. Then

$$\text{Commit}(w; r) := H(w, r)$$

is a commitment scheme

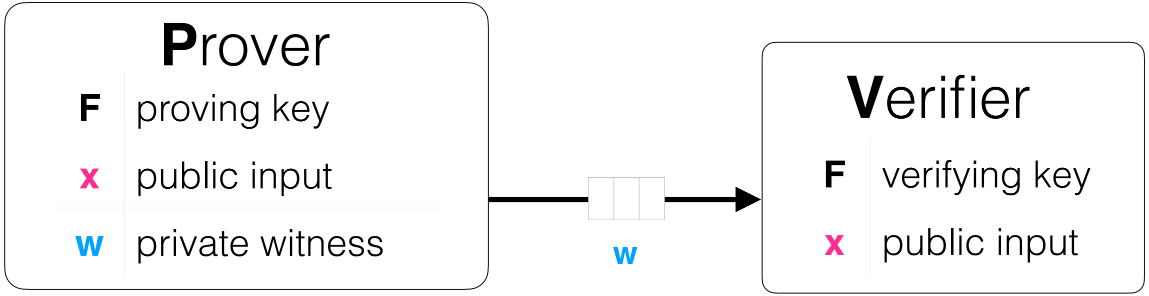
Strawman 2: Commit to the witness



Problem 1 solved: Succinct proof!
Problem 2: How to verify?
Problem 3 solved: COMM hides w!

Performing checks on committed data?

What does V do in the Trivial NP proof?



Evaluate F(x, w)!

To apply this to our commitment-based protocol, do we need a “fully-homomorphic” commitment?

Homomorphic Commitments?

Pair of algorithms with the following syntax:

- Commit$(w; r) \rightarrow$ cm
  - Commits to the message
- Eval$(F_x, \text{cm}) \rightarrow F(x, w)$
  - Evaluates a function over the committed message, and outputs the result in the clear.

Strawman 3: Homomorphic Commitments

Prover(pk, x, w)

1. cm := Commit(w; r)

$\xrightarrow{\text{cm}}$

Verifier(vk, x)

eval(F_x, cm) $\stackrel{?}{=} 1$

Completeness: Follows from that of commitment

Knowledge Soundness: Follows from Trivial NP Proof

Succinct pf size: Follows if eval. proof is succinct

ZK: ???

Problem 1: This would violate ZK: no hiding!

Problem 2: All constructions are inefficient!

Idea: Ask Prover to help

Triple of algorithms with the following syntax:

- Commit$(m; r) \rightarrow$ cm
    - Commits to the message
- ProveEval$(F, m; r) \rightarrow (F(m), \pi)$
    - Returns proof of correct evaluation of $F(m)$
- CheckEval$(F, \text{cm}, v, \pi) \rightarrow b \in \{0,1\}$
    - Checks that π is a valid proof that $F(m) = v$, where m is the msg inside cm

Does this work?

Strawman 4: Functional Commitments

Prover(pk, x, w)

1. $\mathsf{Commit}(w; r)$
2. $\mathsf{ProveEval}(F_x, w; r)$

cm, π_{F_x} $\longrightarrow$

Verifier(vk, x)

$\mathsf{CheckEval}(F_x, \mathsf{cm}, 1, \pi)$

Completeness: Follows from that of $(\mathsf{ProveEval}, \mathsf{CheckEval})$

Knowledge Soundness: Ditto

ZK: Follows from hiding

Succinct pf size: Follows if eval. proof is succinct

Are we done?

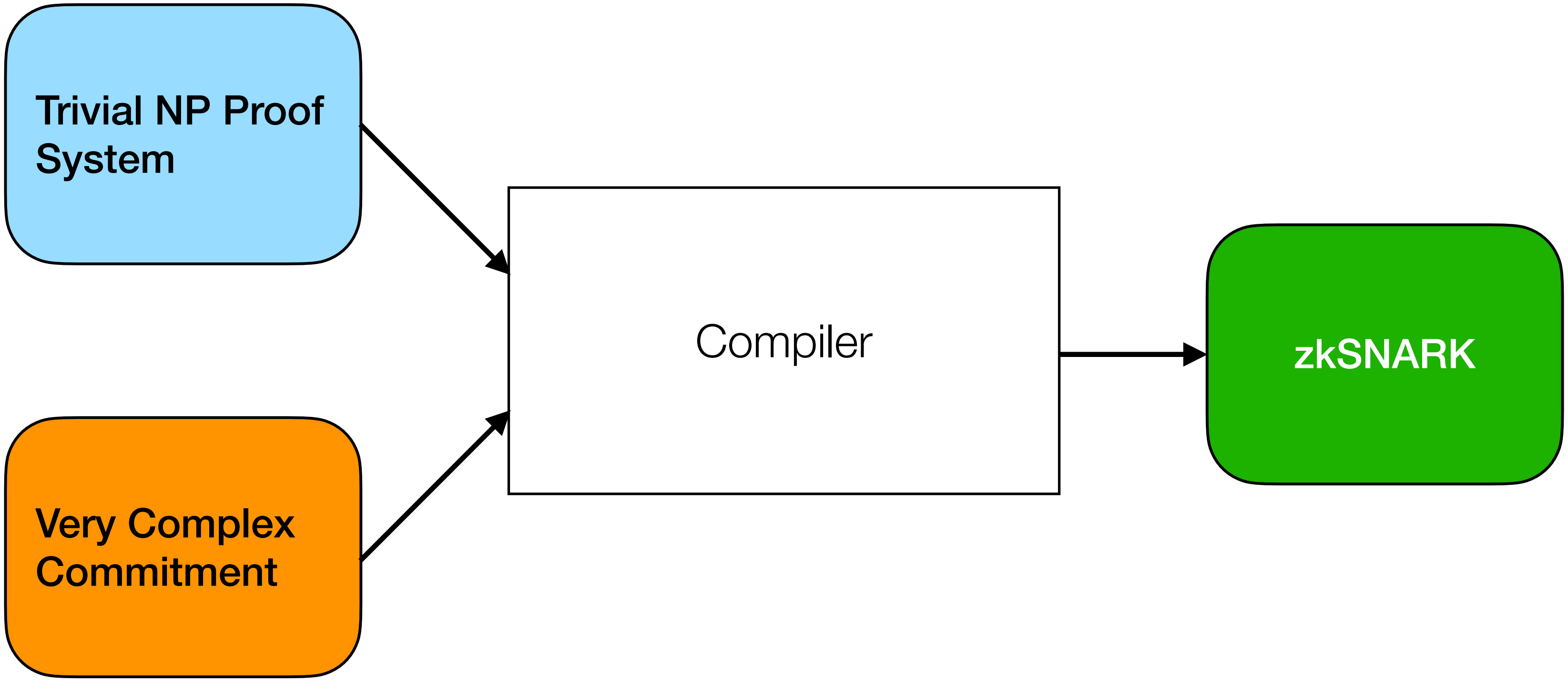
No! We just pushed the problem one layer down!

Problem: This is a zkSNARK for F!

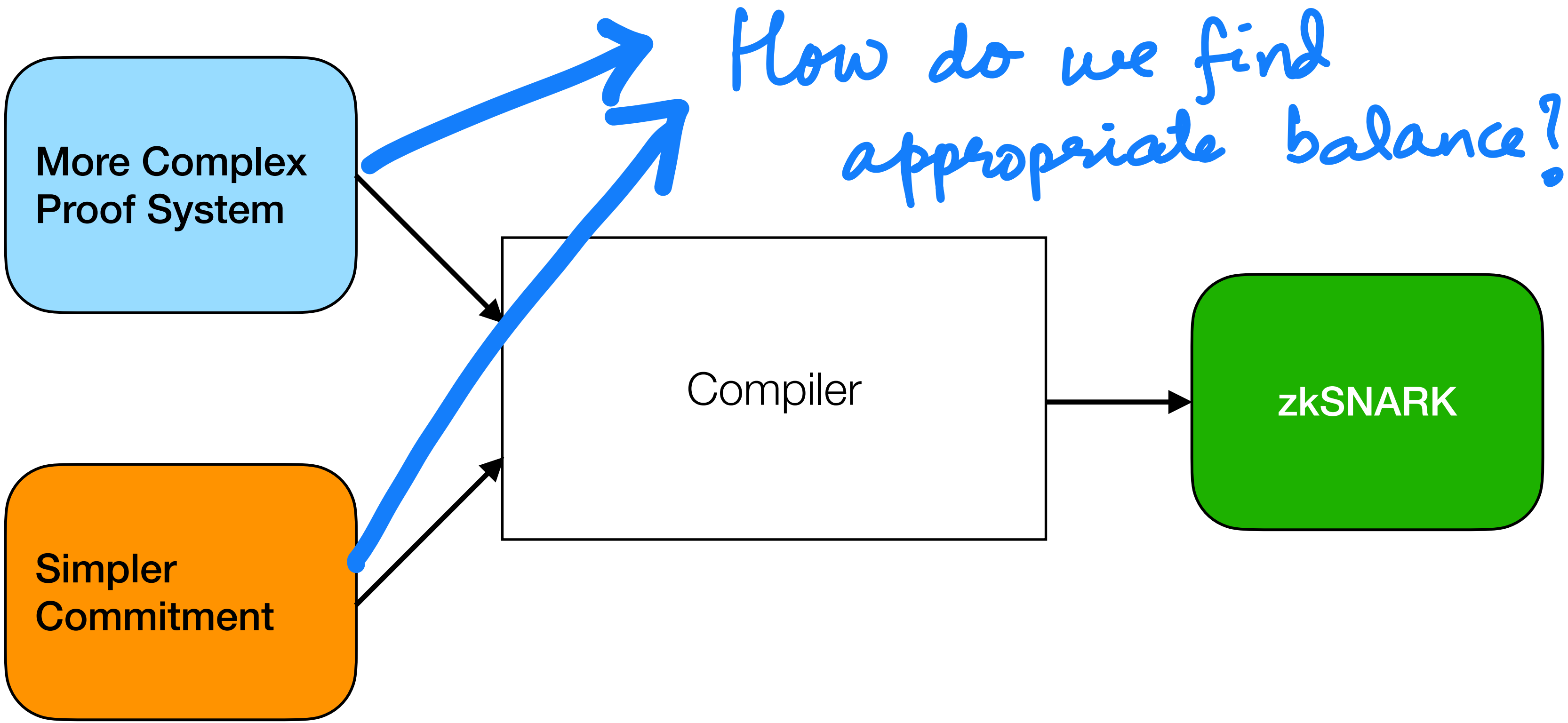
Triple of algorithms with the following syntax:

- $\mathsf{Commit}(m; r) \to \mathsf{cm}$
  - Commits to the message
- $\mathsf{ProveEval}(F, m; r) \to (F(m), \pi)$
  - Returns proof of correct evaluation of $F(m)$
- $\mathsf{CheckEval}(F, \mathsf{cm}, v, \pi) \to b \in \{0,1\}$
  - Checks that π is a valid proof that $F(m) = v$, where m is the msg inside cm

Let's Reassess Our Status



How about we rebalance?



What commitment schemes exist?

Polynomial commitments:

- $F_z(m)$**:** Interpret m as <u>univariate poly</u> $f(X)$ in $\mathbb{F}[X]$ and evaluate at z

Multilinear commitments:

e.g., $f\left(x_1, \ldots, x_k\right) = x_1 x_3 + x_1 x_4 x_5 + x_7$

- $F_{\vec{z}}(m)$**:** Interpret m as <u>multilinear poly</u> $f(X)$ in $\mathbb{F}[\overrightarrow{X}]$ and evaluate at $\vec{z}$

Vector commitments:

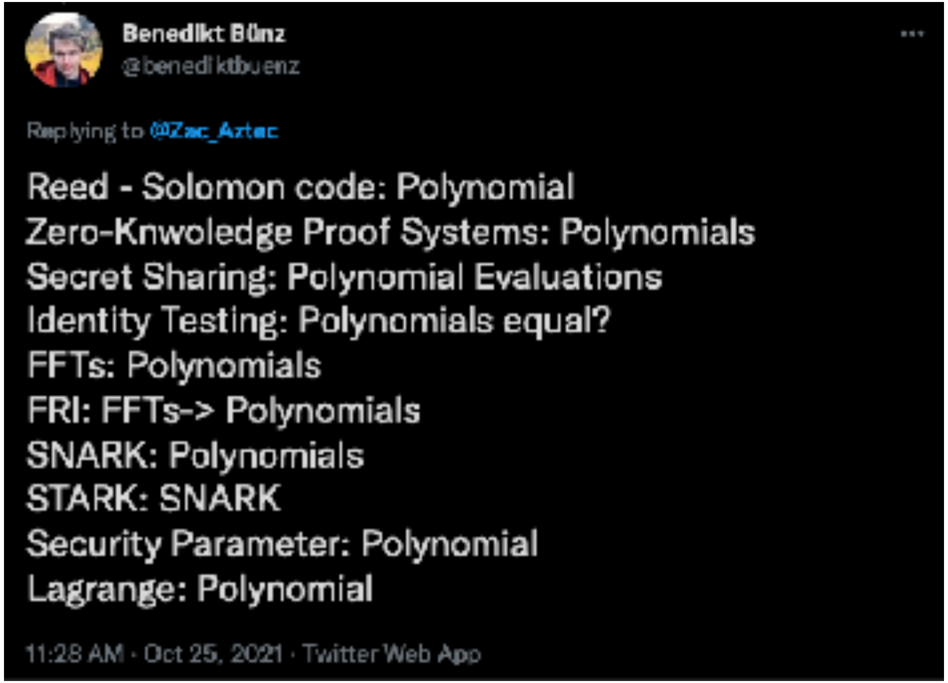
- $F_i(m)$**:** Interpret m as <u>vector</u> v in $\mathbb{F}^n$ and return v_i

Inner-product commitments:

- $F_{\vec{q}}(m)$**:** Interpret m as <u>vector</u> $\vec{v}$ in $\mathbb{F}^n$ and return $\langle \vec{v}, \vec{q} \rangle$

Which to pick?

A: Polynomials!



Benedikt Bünz
@benediktbuenz

Replying to @Zac_Aztec

Reed - Solomon code: Polynomial
Zero-Knwoledge Proof Systems: Polynomials
Secret Sharing: Polynomial Evaluations
Identity Testing: Polynomials equal?
FFTs: Polynomials
FRI: FFTs-> Polynomials
SNARK: Polynomials
STARK: SNARK
Security Parameter: Polynomial
Lagrange: Polynomial

11:28 AM · Oct 25, 2021 · Twitter Web App

Let's pick polynomials