, 2017 12:32 PM

Verifiable Computing and the CRS model
Common Reference String
"Trusted Setup"

Start with relation as R1CS (U, V, W)

Prover(Stmt, witness)

Setup$(1^\lambda, U, V, W)$
↓
CRS
"public parameters"

CRS ← → CRS

Verifier(Stmt)

π := Prove(CRS, Stmt, witness)

π — very small

Verify(CRS, Stmt, π)
↓
very fast

- Correctness
- Extractability
- (optionally) Zero Knowledge

Trusted Setup:
- Like choosing generators for discrete log crypto
  $g, h \xleftarrow{\$} G$
- If uniform parameters, should use "NUMS"
- otherwise, need to flip coins, then forget the coins

Ex.
Setup(1^λ):
$s \xleftarrow{\$} \mathbb{Z}_p, \quad g \xleftarrow{\$} G$
$CRS = (g, g^s, g^{s^2}, g^{s^3}, \ldots g^{s^n})$ for some n.

Succinct Noninteractive Argument of Knowledge

- Succinct → π is small
  - independent of
    - #R1 constraints
    - witness size
  - Verify is fast
- Noninteractive → one-round
- Argument → "Computational assumptions" not "information theoretic"
- Knowledge → sound, extractable

Recall R1CS over a field $\mathbb{F}$
m variables $a_1, \ldots a_m$ $\quad \{a_i\}$ i ranges over $1 \ldots m$

also $a_0 = 1$

$\underline{n}$ R1 constraints U, V, W are $m \times n$ matrices of coefficients $\in \mathbb{F}$

(1) $\underline{\underline{\forall q}} \quad \left(\sum_i^m U_{i,q} \cdot \underline{a}_i\right) \cdot \left(\sum_i^m V_{i,q} \cdot a_i\right) = \left(\sum_i^m W_{i,q} \cdot a_i\right)$

Encode computations:

$$\underbrace{a_1, \ldots}_{\text{inputs}} ; \underbrace{a_{.}, \ldots}_{\text{outputs}} \underbrace{\ldots \quad a_m}_{\text{internal wires}}$$

For V.C. $\quad \underbrace{\text{inputs} \mid \text{outputs}}_{\text{statement}} \quad \underbrace{\text{internal wires}}_{\text{witness}}$

$F(in) = out$

For ZK $\quad \underbrace{\quad}_{\text{public inputs}} \underbrace{\quad}_{\text{private inputs}}$

private inputs also be part of the witness.

$ZK \{ (\underline{a}_{witness}) : R1CS_{U,V,W}(a_{witness} \mathbin{++} a_{statement}) \}$

Step 1. Encode $\underline{U}, \underline{V}, W$ as polynomials

Take arbitrary roots $\underline{r_q}$ $\quad r_1, \ldots, r_n \in \mathbb{F}$

Could be $r_1 = 1, r_2 = 2, \ldots$ $\quad$ one for each constraint.

$\forall i,$ Define $U_i(X)$ by interpolation so that $U_i(r_q) = U_{i,q}$

Same for $V_i(X)$ $\quad V_i(r_q) = V_{i,q}$

$W_i(X)$ $\quad W_i(r_q) = W_{i,q}$

$U_i(X)$

each polynomial is degree $(n-1)$

$U_i(r_q) = U_{i,q}$

(2) $\forall q \quad \left(\sum_i U_i(r_q) \cdot a_i\right) \cdot \left(\sum_i V_i(r_q) \cdot a_i\right) = \left(\sum_i W_i(r_q) \cdot a_i\right)$

Define $\underline{P}(X) = \left(\sum_i U_i(X)\, \underline{a}_i\right)\left(\sum V_i(X) \cdot \underline{a}_i\right) - \left(\sum_i W_i(X) \cdot a_i\right)$

(3) $\forall q \quad P(r_q) = 0 \quad \leftarrow$ a degree $2(n-1)$ over X

with roots at least n roots

Define a smaller degree polynomial $t(X)$ that has the same

$$t(X) = (X - r_1)(X - r_2) \cdots (X - r_n)$$

- has roots at each r_a
- degree n
- "monic" leading coefficient is 1 $\quad t(X) = X^n + c_1 X^{n-1} \ldots\ldots$

If $p(X)$ shares n roots with $t(X)$, then $p(X)$ is a multiple of $t(X)$.

(4) $p(\underline{X}) \equiv 0 \mod t(\underline{X})$

(5) $\exists$ polynomial $h(X)$ s.t. $t(X) \cdot \underline{h(X)} = p(X)$

ex $\overset{p(X)}{5X^4} + \ldots$

divides by $\underset{t(X)=}{} X^3 + \ldots \ldots$?

$p(X) = (5X) \cdot t(X) + (\text{rem}(X))$ degree 3 or less

$(5x^4 + \ldots)$

$p(X) = t(X) \cdot h(X) + \text{rem}(X)$

$\deg(\text{rem}) \leq \deg(p) - \text{degree}(t)$

Step 3) Suffices to check $p(s) = h(s) \cdot t(s)$ for a randomly chosen $s \xleftarrow{\$} \mathbb{F}$.

Setup: $\underline{U_i}(X)$, $\underline{V_i}(X)$, $\underline{W_i}(X)$, $t(X)$ only depends on $r_1, \ldots r_n$

prover computes $\bar{a}$, $p(X)$ depends on $U_i(X) \ldots V_i W_i$ and on $\bar{a}$

compute $h(\underline{X}) = p(\underline{X}) / t(\underline{X})$

Commit $\underline{h}, \underline{p}, \underline{t}$

$\leftarrow$ verifier chooses s

Prover evaluate $\pi = p(s), h(s), t(s)$

Verifier verify $p(s) = h(s) \cdot t(s)$

Remaining problem: need to check that $p(s)$... are done consistently.

Step 3). Setup: $s \xleftarrow{\$} \mathbb{F}$

map $u_i(X), v_i(X)$ $w_i(X)$ to elements in G

$$CRS := \{g^{u_i(s)}\}, \{g^{v_i(s)}\}, \{g^{w_i(s)}\}, g^{t(s)} \ldots$$

Prover computes $p(X)$, $h(X)$

$$g^{\sum_i u_i(s) \cdot a_i} = \prod_i \left(g^{u_i(s)}\right)^{a_i}$$

Same for $g^{\sum_i v_i(s) \cdot a}$ and $g^{\sum_i w_i(s) \cdot a}$

Also add $CRS := \ldots \; g^{s}, g^{s^2}, g^{s^3} \ldots g^{s^n}$

Prover has $h(x) = p(x)/t(X)$ $\quad h(X) = h_0 + h_1 x + \ldots h_n x^n$

$$g^{h(s)} = g^{h_0} \cdot (g^{s})^{h_1} \cdots = \prod_i (g^{s^i})^{h_i}$$

Sends $g^{u(s)}, g^{v(s)}, g^{w(s)}, g^{h(s)}$ to verifier

$u(s) = \sum_i u_i(s) \cdot a_i$

$$p(s) = u(s) \cdot v(s) - w(s) \stackrel{?}{=} h(s) \cdot t(s)$$

Verifier checks

$$e(g^{u(s)}, g^{v(s)}) / e(g^{w(s)}, g) \stackrel{?}{=} e(g^{t(s)}, g^{h(s)})$$

$$e(g,g)^{u(s) \cdot v(s) - w(s)} \stackrel{?}{=} e(g,g)^{t(s) \cdot h(s)}$$

$u(s)$

Remaining problem: Still need to check consistency ~~with~~ of $g^{\cdots}$, etc.

Step 4) Expanding CRS to include consistency checks.

Setup (U,V,W):

Sampling s, α, β_U, β_V, β_W,

CRS $:= g^{-}$ for each of

$\beta_U, \beta_V, \beta_W$ $\{s^i\}$, $\{\alpha s^i\}$, $t(s)$, $\{U_i(s)\}$, $\{V_i(s)\}$, $\{W_i(s)\}$

g^{s^i}, $(g^{\alpha})^{s^i}$ like h in pedersen

Prover(CRS, $\bar{a}$)

Computes $p(X)$, $h(X) = p(X)/t(X)$,

$$\pi := \left(g^{\underline{U(s)}}, g^{V(s)}, g^{W(s)}, g^{h(s)}, g^{\alpha U(s)}, g^{\alpha V(s)}, g^{\alpha W(s)}, g^{\alpha h(s)}, g^{\underline{\beta_U \cdot U(s) + \beta_V \cdot V(s) + \beta_W \cdot W(s)}} \right)$$

Verify: - Check α and β terms:

- $e(g^{U(s)}, g^{\alpha}) \stackrel{?}{=} e(g^{\alpha U(s)}, g)$

Same for V, W, h

- $e(g^{\beta_U U(s) + \dots}, g) \stackrel{?}{=} e(g^{\beta_U}, g^{U(s)}) \cdot e(\cdots) \cdot e(\cdots)$

- Finally $e(g^{U(s)}, g^{V(s)}) / e(g^{W(s)}, g) \stackrel{?}{=} e(g^{h(s)}, g^{t(s)})$

Potentially remaining:

- how to make zero knowledge
- include consistency for statements