So far, we have focused on proving properties in a privacy-preserving manner. Next, we will look at achieving short proofs that are efficient to verify.

**Application:** verifiable computation

\[
\begin{array}{ccc}
\text{client} & \xleftarrow{} & \text{program} \\
\downarrow & & \downarrow \\
(P, X) & \rightarrow & (y = P(x)) \\
\end{array}
\]

How do we know that the server computed the correct value? We can provide a proof \( y = P(x) \). To be useful, checking the proof should be much faster than computing \( P \).

**Main primitive:** aggregation scheme for proofs - batch argument for NP

**Setting:** given \( T \) statements \( X_1, ..., X_T \) and circuit \( C \), show that all of the statements are true (i.e., \( \exists w_i : C(X_i, w_i) = 1 \) for all \( i \in [T] \)).

Naively: Give out \( T \) proofs, one for each statement \( X_i \).

**Goal:** Can we do better. Namely, can we batch prove \( T \) statements with a proof of size \( o(T) \)?

For now, we will not worry about zero-knowledge. Turns out that succinctness can be used to achieve zero-knowledge.
Let $C: \{0,1\}^n \times \{0,1\}^h \rightarrow \{0,1\}$ be circuit that computes an NP relation. Take $x_1, ..., x_T \in \{0,1\}^n$. We want to prove that there exist $w_1, ..., w_n \in \{0,1\}^n$ where $C(x_i, w_i) = 1$ for all $i \in [T]$.

Waters-Wu construction: follows commit-and-prove paradigm
Relies on a GOS-like structure: commit to all of the wires and give a proof that commitments associated with each gate are valid (i.e., consistent with the gate operation)

In GOS, commitments were BGN encryptions (as long as the input)
⇒ Will not give succinct proofs

Starting point: succinct commitment scheme
- CRS: $g, h_1, ..., h_T$ where $h_i = g^{\alpha_i}$ (and $\alpha_i \in \mathbb{Z}_N$)
- Commit to a vector $V = (v_1, ..., v_T)$ as follows:

$$C = \prod_{i \in [T]} h_i^v_i$$

commitment is now a single group element (independent of $T$)

Construction overview:
- As in GOS, we index the wires in $C$ in topological order
- Prover starts by computing $C(x_i, w_i)$ for each $i \in [T]$. Let $t_i^{(1)}, ..., t_i^{(s)}$ be the wire assignments associated with $C(x_i, w_i)$.
- For each wire $j \in [s]$, prover commits to vector $(t_j^{(1)}, ..., t_j^{(s)})$ - namely the values associated with wire $j$ across all $T$ instances. Let $c_j$ be the commitment.

Similar to GOS, need to establish two properties
1. For all $i \in [T]$ and $j \in [s]$, $t_j^{(i)} \in \{0,1\}$
2. For each NAND gate $(j_1, j_2, j_3)$, $t_{j_1}^{(i)} = \text{NAND}(t_{j_2}^{(i)}, t_{j_3}^{(i)})$ for all $i \in [T]$

We start with the first property.
Suppose $c = \prod_{i \in [T]} h_i$. **Goal:** prove that $v_i \in [0,1]^3$ for all $i \in [T]$.

As before: $v_i \in [0,1]^3$ if and only if $v_i(v_i - 1) = 0$ or equivalently, $v_i^2 = v_i$.

$$e(c, c) = e\left(\prod_{i \in [T]} h_i, \prod_{i \in [T]} h_i\right)$$

$$= e\left(g^{\sum \alpha_i v_i}, g^{\sum \alpha_i v_i}\right)$$

$$= e\left(g, g\right)^{\left[\sum \alpha_i v_i\right]^2}$$

$$\left[\sum_{i \in [T]} \alpha_i v_i\right]^2 = \sum_{i \in [T]} \alpha_i^2 v_i^2 + \sum_{i \in [T]} \sum_{j \neq i} \alpha_i \alpha_j v_i v_j$$

if $v_i^2 = v_i$, then $\sum_{i \in [T]} \alpha_i^2 v_i^2 = \sum_{i \in [T]} \alpha_i^2 v_i$

\[\Rightarrow\text{Can we compute this in the exponent?}\]

$$e\left(c, \prod_{i \in [T]} h_i\right) = e\left(g^{\sum \alpha_i v_i}, g^{\sum \alpha_i}\right) = e\left(g, g\right)^{\left(\sum \alpha_i v_i\right)\left(\sum \alpha_i\right)}$$

$$\left[\sum_{i \in [T]} \alpha_i v_i\right]\left[\sum_{i \in [T]} \alpha_i\right] = \sum_{i \in [T]} \alpha_i^2 v_i + \sum_{i \in [T]} \sum_{j \neq i} \alpha_i \alpha_j v_i$$

**Main observation:** if $v_i^2 = v_i$ for all $i \in [T]$, then

$$e\left(c, c\right) = e\left(c, \prod_{i \in [T]} h_i\right) e\left(g, g\right)^{\sum_{i \in [T]} \sum_{j \neq i} \alpha_i \alpha_j (v_i v_j - v_i)}$$

Can be computed by the verifier "cross terms" [depend on $\alpha_i \alpha_j$]

**Solution:** give out $h_{i,j} = g^{\alpha_i \alpha_j}$ for all $i \neq j$
Construction: \( \text{crs} = (g, \{ h_i^j : i \in G \}, \{ u_j^j : i \neq j \}, A = \prod_{i \in G} h_i) \)

Commitment to \( v = (v_1, \ldots, v_T) \): \( c = \prod_{i \in G} h_i^v_i \)

To prove \( v_i \in \{0, 1\} \), compute \( \pi = \prod_{i \in G} \prod_{j \neq i} v_i^j v_j - v_i \)

To check the proof, check that
\[
 e(c, c) = e(c, A) e(g, u)
\]

This corresponds to the following relation in the exponent:
\[
 \sum_{i \in [T]} \sum_{j \in [T]} \alpha_i \alpha_j v_i^j v_j = \sum_{i \in [T]} \sum_{j \in [T]} \alpha_i \alpha_j v_i + \sum_{i \in [T]} \sum_{j \neq i} \alpha_i \alpha_j (v_i^j v_j - v_i)
\]

Equality holds if \( v_i^2 = v_i \) for all \( i \in [T] \). (Completeness)

Soundness is more delicate - will defer to later.

Gate consistency can be implemented like in GOS
(by checking \( v_i^1 + v_i^2 - 2v_i^3 + 2 \in \{0, 1\} \) for all \( i \))

\[
 c_1 = \prod_{i \in G} h_i^v_i^1, \quad c_2 = \prod_{i \in G} h_i^v_i^2, \quad c_3 = \prod_{i \in G} h_i^v_i^3
\]

Compute \( c_1 c_2 c_3^2 \prod_{i \in G} h_i^2 \):
\[
 c^* = c_1 c_2 c_3^2 \prod_{i \in G} h_i^2 = \left( \prod_{i \in G} h_i^v_i^1 \right) \left( \prod_{i \in G} h_i^v_i^2 \right) \left( \prod_{i \in G} h_i^{-2v_i^3 + 2} \right) \left( \prod_{i \in G} h_i^2 \right) = \prod_{i \in G} v_i^1 + v_i^2 - 2v_i^3 + 2
\]
c* is a commitment to \( V_{i,i} + V_{2,i} - 2V_{3,i} + 2 \) for all \( i \in [T] \).

Can use previous approach to check that component of committed vector is a \( \{0,1\} \) value.

How do we argue soundness?

We will consider non-adaptive soundness where the adversary has to choose the false statement before seeing the public parameters (crs).

Approach is to program a secret index \( i^* \) into the CRS. Given a valid proof on a statement \((X_1, ..., X_T)\), it will be possible to extract a witness \( W_{i^*} \) such that \( C(X_{i^*}, W_{i^*}) = 1 \).

[Cannot extract witness for all indices \( i \in [T] \) from the same proof - why?]

As long as crs hides the index \( i^* \), this suffices to show non-adaptive soundness:

- Fix any statement \((X_1, ..., X_T)\). If this is false, there exists \( i^* \in [T] \) where \( X_{i^*} \) is false.
- Suppose we set the CRS to be extracting at \( i^* \).
- Adversary should still produce an accepting proof (otherwise, it breaks index hiding).
- If adversary produces valid proof in this case, then we extract a witness for \( X_{i^*} \). But this is not possible (since no such witness exists!)

With complexity leveraging for index hiding, this suffices to show adaptive soundness.
Programming the CRS to extract on index $i^*$:

- **Normal CRS**: $h_1 = g_1^x, \ldots, h_t = g_t^x, u_1 = g_1^{x_1}, \ldots, u_t = g_t^{x_t}$

  as described, all elements are in the mod-$p$ subgroup

- **Binding CRS at index $i^*$**: "lift" element $h_i$ to the full group.

  
  
  set $h_i^{i^*} = g_1^{x_i}$ where $g$ generates the full group

  cross terms involving $i^*$ also lifted to the full group as a result

- **Binding CRS indistinguishable from real CRS by subgroup decision**

- In binding mode, commitment to $V = (v_1, \ldots, v_t)$ is now

  $C = \prod_{i \in \mathcal{E}} h_i^{v_i} = h_i^{v_i \prod_{i \neq i^*} h_i}$

  in full group

  in order-$p$ subgroup

- **Extraction trapdoor** is the factor $p$, which can be used to project away the mode-$p$ component:

  $C^p = h_i^{v_i \prod_{i \neq i^*} h_i}$

  $= \prod_{i \neq i^*} (g_i^{x_i} p)^{v_i}$

  $= (g_i^{x_i} p)^{v_i}$

  if have isolated component $i^*$ and can see

  if $V_{i^*} = 0$ or $V_{i^*} = 1$

  if verification relations pass, these are the only two

  possibilities

Essentially, when the CRS binds at index $i^*$, the proof system is statistically sound at index $i^*$ (since we can extract the witness at $i^*$).

$\rightarrow$ Also called "somewhere statistical soundness"