Today: Delegation schemes.

Def: A delegation scheme for a Turing machine $M$ is an interactive protocol $(P,V)$ s.t.

- **Completeness**: $\forall n \in \mathbb{N} \quad \forall x \in \{0,1\}^n \quad \forall T$ s.t. $M(x) = 1$ within $T$ steps, $\forall k \in \mathbb{N}

\[ P_r[(P,V)(1^k,x,T) = 1] = 1 \]

- **Soundness**: $\forall$ poly size $P^* \nexists x,T$ s.t. $M(x) = 1$ within $T$ steps, $\forall k \in \mathbb{N}

\[ P_r[(P^*,V)(1^k,x,T) = 1] \leq \mu(k) \]

Remark: More generally, one can consider $T'$-soundness, which requires the soundness condition to hold w.r.t. any poly($T'$)-size $P^*$.

Efficiency: $P$ runs in time $\leq \text{poly}(T)$

(if $M$ is non-det. then $P$ runs in time $\leq \text{poly}(T)$ given witness)
The verifier runs in time \( n \cdot \text{poly}(k) \), assuming \( k \gg \log T \).

**Note:** The focus of delegation schemes is on efficient computations (in \( P \) or \( \text{NP} \)).

This is in contrast to the traditional work on \( \text{IP}/\text{MIP} \).

The main difference between IP & delegation is the efficiency requirement & the computational soundness.

**From PCP to Delegation** [Kilian 92, Micali 94]

Fix \( (P_{\text{PCP}}, V_{\text{PCP}}) \) for \( M \) w. resp. soundness.

**Tool:** Merkle hash

Given \( h : \{0,1\}^{2^k} \rightarrow \{0,1\}^k \) & given \( x \in \{0,1\}^N \) \((N \gg k)\), define \( T_h(x) = (d, \text{root}) \), defined by
Let $\mathcal{H} = \mathcal{H}_k$, where $\mathcal{H}_k = \{ h : \{0,1\}^k \to \{0,1\}^k \}$, be a family of collision resistant hash functions.

Delegation Scheme: $P$ ($1^k, x, T$) $\xrightarrow{\cdot} V$

Compute

$T \leftarrow P \circ PCP (M, x, T)$

$V_{PCP} (1^k)$

Sample $\pi \cdot \pi_e \leftarrow V_{PCP} (1^k)$

Accept iff $V_{PCP}$ accepts

$(M, x, T), \pi \cdot \pi_e, \pi_{e_1}, \ldots, \pi_{e_2}$

& openings are consistent with root.
Completeness: Follows from completeness of \((\text{Prep}, \text{Ver})\) together with completeness of Merkle Hash (i.e., honest local opening is accepted).

Soundness:

Claim: \(T_H = \left\{ t_H : t \in H \right\}\) is a collision resistant hash family.

Proof Idea: Show that if \(\exists\) poly-size \(A\) that finds \(\tilde{c}_{\text{poly-size}} A\) with non-negl prob collisions in \(T_H\) then \(\exists\) poly-size \(A'\) that finds collisions in \(H\) with non-negl prob.

It is here that we use the fact that the depth \(d\) is included in the output.
Proof Idea for Soundness: Suppose \( \exists \) poly-size \( P^* \) that breaks soundness:

\[
P^* \xleftarrow{h} V
\]

\[
\xrightarrow{\Pi_1, \Pi_2, \ldots, \Pi_n}
\]

\[
\xrightarrow{\Pi'_1, \Pi'_2, \ldots, \Pi'_n}
\]

Idea: Rewind \( P^* \) to obtain the entire PCP.

Due to the fact that \( T \) is collision resistant, \( P^* \) can open to a unique PCP.

\( \Rightarrow \) \( V_{pcp} \) accepts PCP pf of false statement w. non-negf prob — Contradicting soundness of PCP.

Holds if \( (P_{pcp}, V_{pcp}) \) satisfy the eff. conditions

Efficiency: \( P \) runs in time \( \text{poly}(t) \), assuming \( P_{pcp} \) is eff. \( V \) runs in time \( \text{poly}(k) + \text{runtime of } V_{pcp} \leq |x| \cdot \text{poly}(k) \), as desired.
Note: This scheme is interactive, i.e., consists of 4 msgs.

Goal: Construct non-interactive delegation scheme.

Has applications such as blockchain applications.

Approach 1: Apply Fiat-Shamir to reduce interaction

\[ P \xleftarrow{\$} (1^k, M, x_1), h \xleftarrow{\$} V \]

\[ \alpha = T_h(x_1), \quad h(\alpha) = \beta, \quad \Pi_{g_1} \ldots \Pi_{g_k} \]

\[ (d, \text{root}) \]

\[ (g_1, \ldots, g_k) = V_{PCP}(1^k, \beta) \]

Sound in the Random Oracle Model (ROM), i.e., if \( h \) is modelled as a random oracle.

Not clear if this is sound in the real world...
Approach 2: Using FHE (Gen, Enc, Dec, Eval).

\[ \Pi \leftarrow P_{\text{pop}}(M, x, T) \]

\[ \begin{array}{c}
\text{P} \\
\end{array} \]

Computing using eval

Equivailently:

\[ \begin{array}{c}
\text{Not publicly verifiable!} \\
\text{Completeness follows from completeness of MIP} \\
& \text{and FHE.}
\end{array} \]
Efficiency: Follows from efficiency of MIP.

Soundness: Intuitively seems to be sound since when computing $a_i$ "under the hood of the FHE" the cheating prover cannot see $y_i$, since it is encrypted w. a fresh independent key.

However, not always sound!

What went wrong??

- MIP is sound only if provers are local
  - Given $(\text{Enc}_{p_{k_1}}(y_1), \text{Enc}_{p_{k_2}}(y_2))$

  a cheating prover may not be local!

Soundness does hold if MIP is secure against more general than local, called no-signaling defined by quantum physicists.