Delegation with Updatable Unambiguous Proofs and PPAD-Hardness

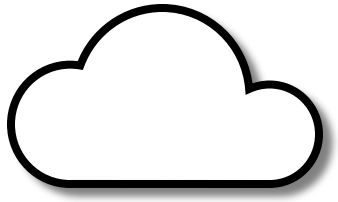
Lisa Yang
MIT

Based on joint work with
Yael Tauman Kalai and Omer Paneth

Delegation

time T computation $M(x) = ?$

P V



$M(x) = \mathrm{y}$

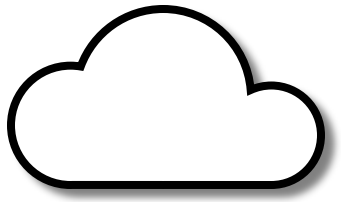
Proof Π

V checks Π
in time $\ll T$

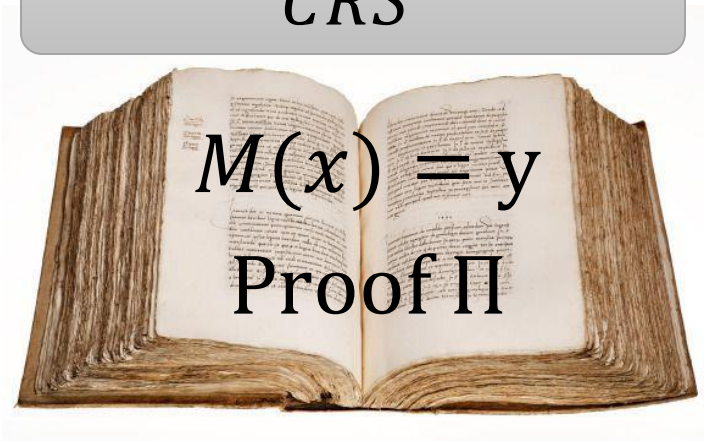
Can verifying be faster than computing?

Publicly Verifiable Delegation

P



CRS



V

Prior Work: Publicly Verifiable Delegation

Strong assumptions

- Random Oracle Model [Micali94]
- Knowledge assumptions [Groth10, Lipma12, Gennaro-Gentry-Parno-Raykova12, Bitansky-Canetti-Chiesa-Tromer12, Bitansky-Chiesa-Ishai-Ostrovsky-Paneth13…]
- Indistinguishability Obfuscation [Bitansky-Sanjam-Lin-Pass-Telang14, Canetti-Holmgren-Jain-Vaikuntanathan14, Koppula-Lewko-Waters14, Canetti-Holmgren16, Chen-Chow-Chung-Lai16]
- Multilinear maps [Paneth-Rothblum17]

Delegation for bounded-depth circuits via Fiat-Shamir

- Optimal security of LWE [Canetti-Chen-Holmgren-Lombardi-Rothblum-Rothblum-Wichs19]
- Sub-exponential LWE [Kalai-Zhang20]

Delegation for polynomial-time computations

- Bilinear groups [Kalai-Paneth-Y19]

Updatable Proofs [Valiant08]

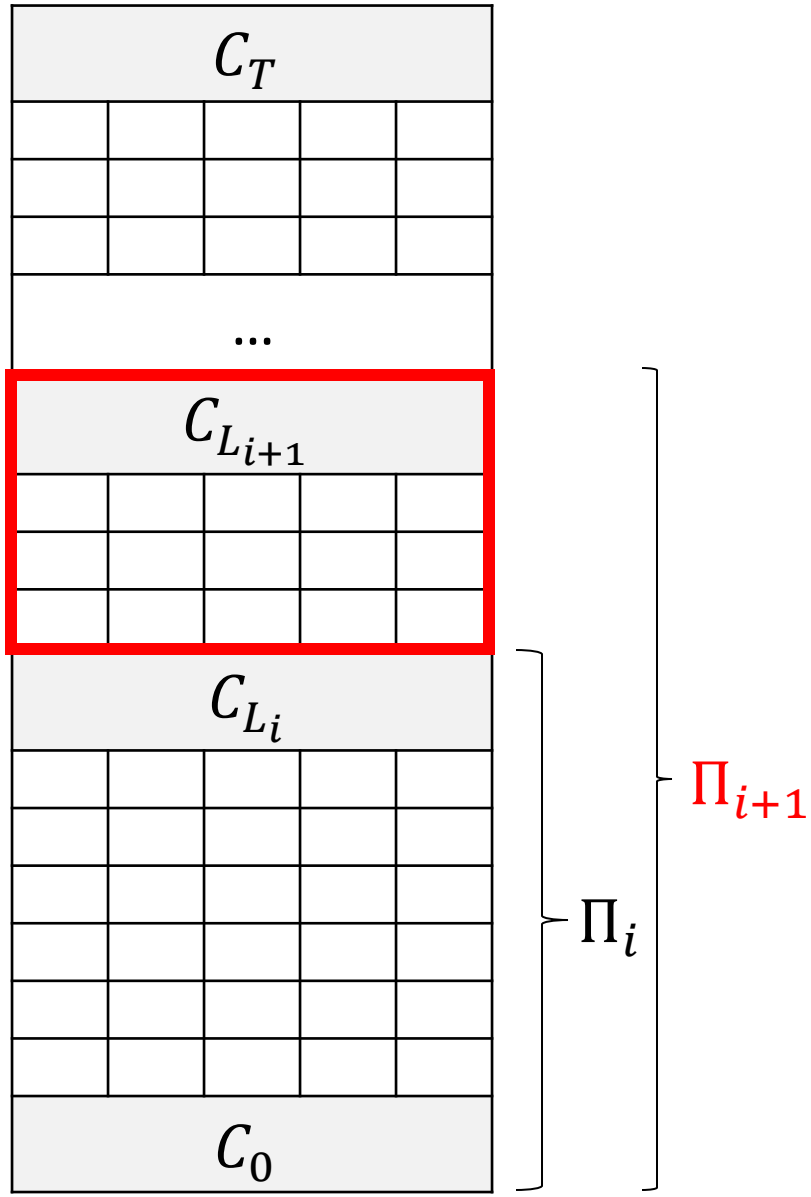
Consider a long computation $C_0 \rightarrow C_T$ carried out over B iterations

Updatable Proofs: update Π_i into Π_{i+1}

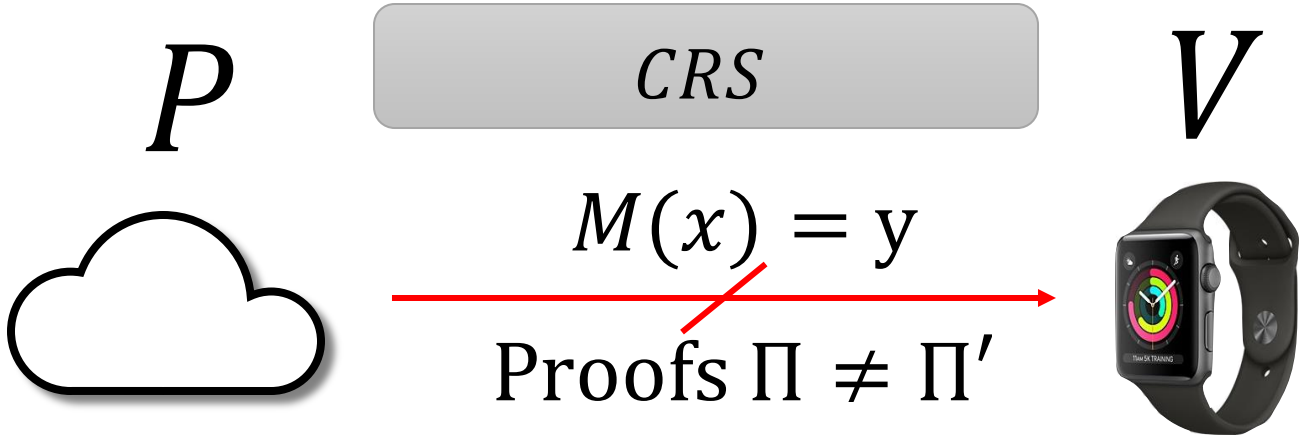
Want the proof update to take time $\sim$ computation performed

Want proofs to remain succinct

[Bitansky-Canetti-Chiesa-Tromer13] using SNARKs (based on strong assumptions)



Unambiguous Proofs



Unambiguous Proofs: $P^*(CRS)$ cannot output $\Pi \neq \Pi'$ for the same statement $M(x) = y$ (except with negligible probability over CRS)

[Reingold-Rothblum-Rothblum]

Our Results: Delegation

Delegation with updatable and unambiguous proofs based on the decisional bilinear group assumption:

[Kalai-Paneth-Y19]

For a bilinear group G of order $p = 2^{\Theta(\kappa)}$ and $\alpha = O(\log \kappa)$ given

$$\left(g^{t^i \cdot s^j}\right)_{\substack{i \in [2] \\ j \in [\alpha]}} = \begin{pmatrix} g & g^s & \dots & g^{s^\alpha} \\ g^t & g^{t \cdot s} & \dots & g^{t \cdot s^\alpha} \\ g^{t^2} & g^{t^2 \cdot s} & \dots & g^{t^2 \cdot s^\alpha} \end{pmatrix}$$

for random $g \in G$ and $s \in \mathbb{Z}_p$ it is hard to distinguish whether $t = s^{2\alpha+1}$ or t is an independent random element in $\mathbb{Z}_p$.

Our Results: PPAD-Hardness

[Choudhuri-Hubacek-Kamath-Pietrzak-Rosen-Rothblum19]

PPAD-Hardness based on:

1. The quasi-polynomial hardness of KPY's bilinear group assumption
2. Any hard language L decidable in super-polynomial time (and polynomial space)
   - For example, the hardness of SAT for sub-exponential size circuits (non-uniform ETH) suffices

Related Work: PPAD-Hardness

Strong assumptions

- Indistinguishability Obfuscation [Abbot-Kane-Valiant04, Bitanski-Paneth-Rosen15, Hubacek-Yogev17]
- Functional Encryption assumptions [Garg-Pandey-Srinivasan16, Komargodski-Segev17]

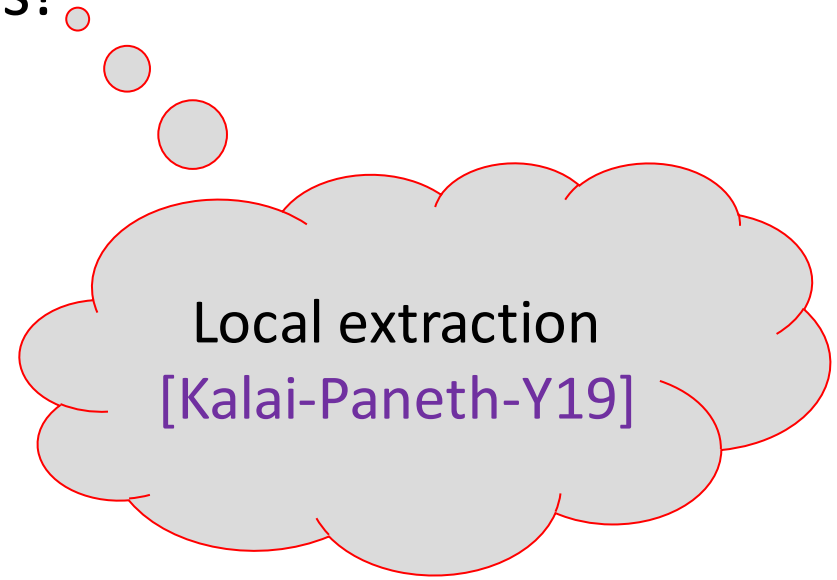
Fiat-Shamir interactive protocol for a particular language

- Security of Fiat-Shamir/Optimal security of LWE [Choudhuri-Hubacek-Kamath-Pietrzak-Rosen-Rothblum19, Ephraim-Freitag-Komargodski-Pass19]
- Sub-exponential LWE [Lombardi-Vaikuntanathan20, Kalai-Zhang20, Jawale-Khurana20]

Polynomial Local Search (PLS) Hardness [Bitansky-Gerichter20]

1. Delegation with Updatable Proofs

- Use recursive proof composition
- Without strong assumptions!



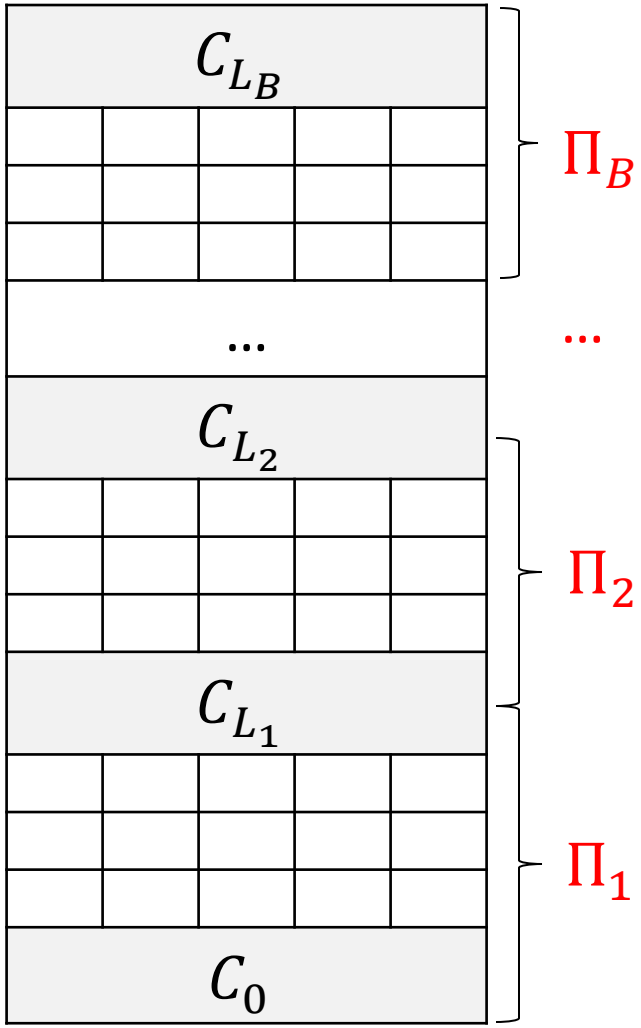
1. Delegation with Updatable Proofs

$\Pi: C_0 \to C_L$

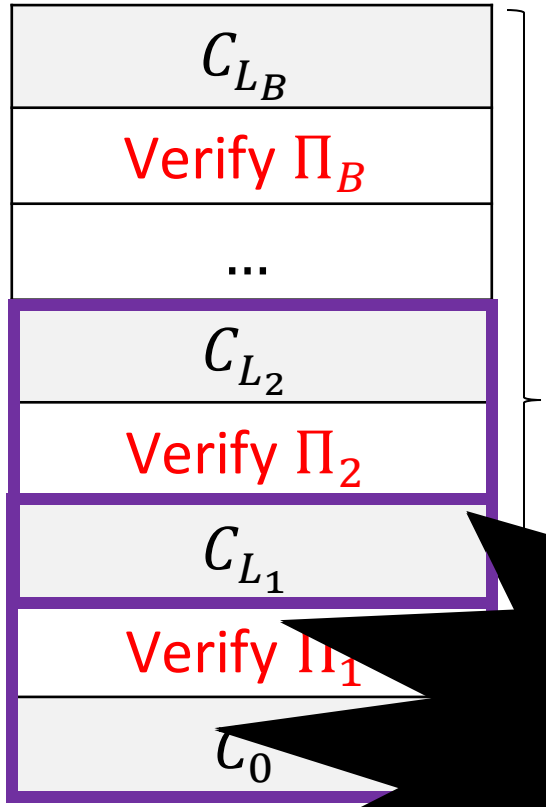
Update Π:
Append proof for computation performed

Proof grows!!

Π contains B proofs
$\Pi_i: C_{L_{i-1}} \to C_{L_i}$

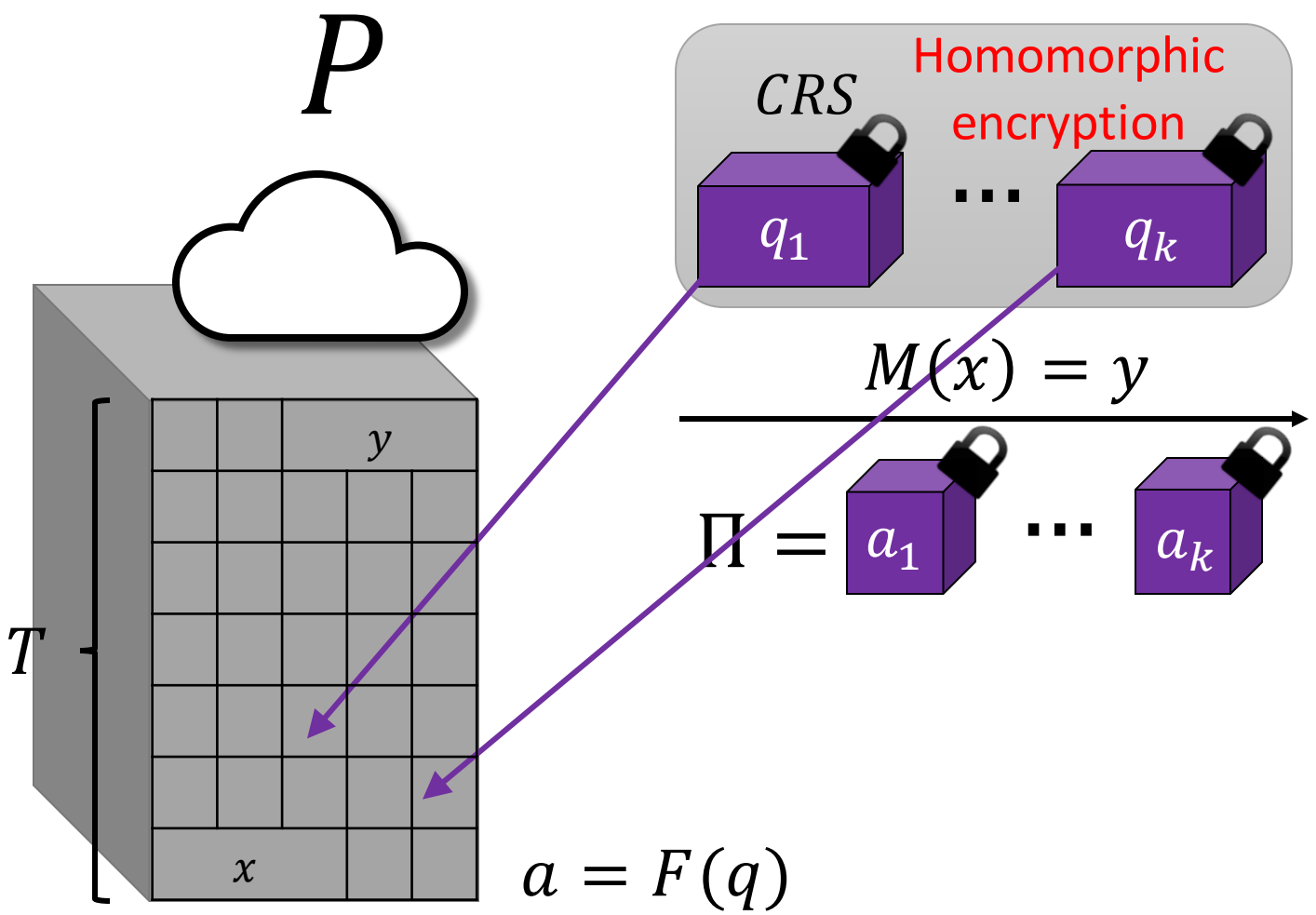


$Merge\left(C_{L_{i-1}}, \Pi_i, C_{L_i}\right)_{i\in[B]}$
replaces this with $\left(C_0, \Pi', C_{L_B}\right)$



Local extraction suffices!

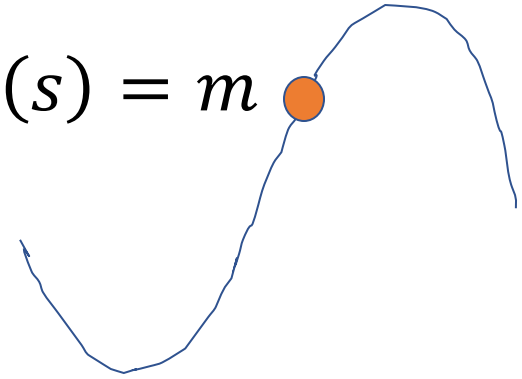
KPY Delegation



V checks Π using Zero-Test [Paneth-Rothblum17]

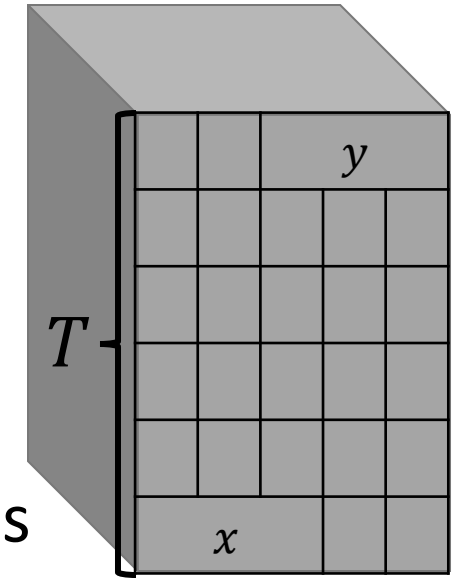
2. Delegation with Unambiguous Proofs

- Observation: need to use encryption with unambiguity property
- **Unambiguity of Ciphertexts:** any $P^*(CRS)$ cannot generate two different ciphertexts that encrypt the same message



- KPY Encryption:
  - $sk = s \leftarrow \mathbb{F}$
  - $c = g^{R \in \mathbb{F}[x]}$
  - $P^* \nrightarrow c = g^R, c' = g^{R'}$ such that $R(s) = R'(s) = m$
- Unambiguous Proofs: suffices to ensure unambiguity of answers

2. Unambiguity of Answers



$a = F(q)$

- [Kalai-Raz-Rothblum14] for $q \in \{0,1\}^{\ell}$ answers are unambiguous
- Need unambiguous answers for $q \in \mathbb{F}^{\ell}$
- Observation: If P evaluates a multilinear polynomial then can show unambiguity of answers for every $q \in \mathbb{F}^{\ell}$
- Idea: Ask P to send a "proof of multilinearity" for his evaluated ciphertexts

Notion of local multilinearity!

Proof of Local Multilinearity

- P homomorphically evaluates $F(q_1 \dots q_\ell)$
- First attempt: ask P for the restriction of F in each coordinate
  Evaluate encryptions of (A_i, B_i) such that $F(\vec{q}) = A_i \cdot q_i + B_i$
  V checks consistency using the Zero-Test

- Problem: P^* can compute (A_i, B_i) using $Enc(q_i)$
- Idea: encrypt $\vec{q}$ again without i'th coordinate
  Ask P for (A_i', B_i')
- Test that $(A_i, B_i) = (A_i', B_i')$

"Proof of Equality"
P evaluated same function on both encryptions

Delegation with Updatable Unambiguous Proofs

Not done yet…

To show unambiguity of entire proof:

- Unambiguity of other ciphertexts in KPY proof
- Unambiguity of ciphertexts we added ☺
  Equality and Multilinearity proofs

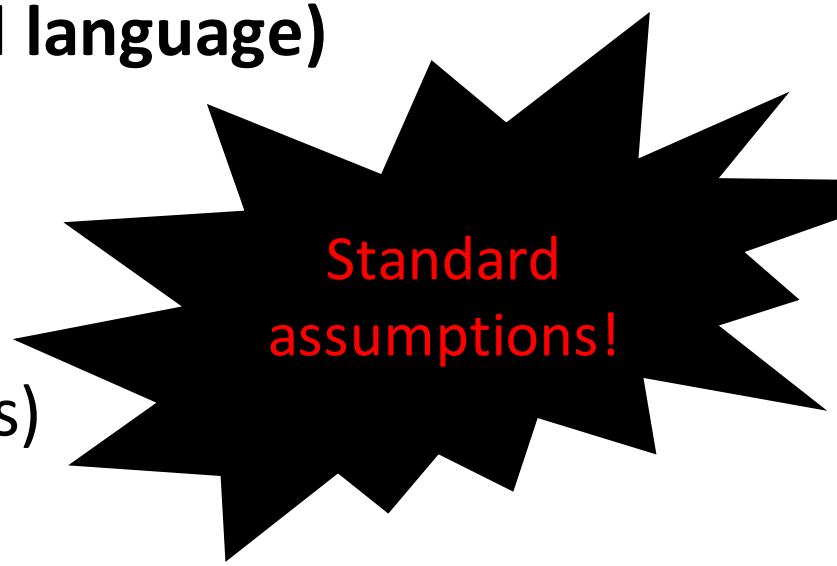
- Show unambiguity preserved in recursive proof composition
  Updatable proofs

Summary

- Our Results:
  - **Delegation with updatable and unambiguous proofs based on the KPY bilinear group assumption**
  - **PPAD-Hardness based on the quasi-polynomial hardness of the KPY bilinear group assumption (and any hard language)**

Power of local proofs:

- recursive proof composition (updatable proofs)
- proof of multilinearity (unambiguous proofs)



lisayang@mit.edu