Today: From MPC to Zero Knowledge

Logistics
- HW 2 due Friday 5pm
- Course Staff traveling to Eurocrypt next week:
  - Check Pizzeria for OH charges
  - OH today & tomorrow

Plan
- Recap: MPC with correlated randomness
- From MPC to ZK
- Application to PW sigs
Recap: Secret Sharing and MPC

- Simple secret sharing: additive
  \[ x_1 + x_2 + x_3 = x \in \mathbb{F}_p \]
  \[ + \rightarrow x \in \mathbb{F} \]

- Also covered fancier secret sharing schemes (Shamir)

Want to spend some time reviewing MPC

Most MPC protocols view computation as a circuit (blog)
- Boolean: AND, OR, NOT (mod 2)
- Arithmetic: * , + (mod \(p\))

Each party \(i\) has input \(x_i\), parties want to jointly compute \(f(x_1, \ldots, x_n)\)

such that no coalition of evil players learns anything apart from
- \(f(x_1, \ldots, x_r)\)
- their private input

[Think about how you would formalize this]

Well see in a second

We aim for "Semi-honest" Security
- "honest but curious" evil player
- Begins in reality... need more work to get "Sull"/"Malicious" Security

Implication: Anything that \(n\) people can compute (i.e. ppt), can compute securely (w/ private inputs)

\[ x \rightarrow \text{In theory} \]

\[ A \text{ has database of images} \]
\[ B \text{ has ML algorithm} \]
\[ C \text{ has private image} \]

\[ \text{Train ML alg on images and classify } C's \text{ image} \]
\[ \text{Classifies } C's \text{ image} \]

Why are we not doing this in practice today?

"efficient" ≠ efficient
"important" theme in life
Idea of MPC Protocol David discussed (Beaver ckt randomization)

Players 1, 2, 3 get "correlated randomness"

Example: Player 1 has private input \( x_1 \in \mathbb{F} \)

Want to compute

\[ f(x_1, x_2, x_3) = x_1 \left( x_1 + x_2 \right) x_3 + 7 \]

Idea: Players jointly walk through each wire in ckt, compute on additive secret sharing of wire

\[ \rightarrow \text{ Once have share of output, can publish and reconstruct} \]

I. Get share of \( x_1 \) input

\[ [x_1], [x_2]_1, [x_3]_1 \]

\[ \text{send each share to player 1} \]

II. Get share of internal wires

\[ [w_1], [x_2]_1, [x_3]_1 \]

\[ \text{Use correlated randomness to get share of} \ w_2 \]

\[ [w_2]_1 = \text{ Protocol David described plus randomness} \]

\[ [w_3]_1 = [w_2]_1 + \frac{7}{3} \in \mathbb{F} \]

III. Get output

Players send output shares to 1

Have \( [w_1], [x_2], [x_3] \) s.t. \( ab = c \in \mathbb{F} \)

1) Broadcast \( [w_1], [x_1], [x_3] \)

2) Compute \( d = w_1 - a \)

3) Compute \( e = x_3 - b \)

3) Set

\[ [w_2]_1 = [w_1]_1 \left( w_2 \right) + [x_3]_1 \]

\[ = [c]_1 + [w_1]_1 w_2 + [x_3]_1 \]
A couple of notes:

1. Where does correlated randomness come from?
   - Can generate it using MPC (see HW)
   - Semi-trusted dealer (see later in course)

2. Protocol cost (in terms of broadcast) scales w/ depth of circuit

3. Must represent S, f as a circuit
   - Not efficient... lots of optimization

   e.g. Recent work on QPC, semi-honest
   - 256 neurons, 2 hidden layers \( \Rightarrow \) 7 hours on tan\( ^\rightarrow \) (Zhang & Mohassel '17)
   - 1000 dimensions, 1000 points \( \Rightarrow \) 158 sec
   - \( \leq 0.1 \) s on my laptop

Bottom line:

- For simple functions, can actually implement MPC
- Allows "best possible" security for computation on secret data
Big Picture

1990s 1990s 2000

- Foundations
- Protocols & MPC
- Lattices, PQCrypto & more

MPC in the Head
- Connects MPC and zk in surprising way
- Has application to zk crypto
- Nice bridge to next section; of course
- Also just a really nice idea
MPC in the Head (IKOS '07)

We will show that it is possible to "efficiently" compile any general MPC protocol into a zero-knowledge proof system for an NP relation.

Surprising/cool connection b/w primitives

Recall: MPC

Each player $i$ has input $w_i \in \{0, 1\}^d$

At the end of the protocol, each player learns $f(w_1, \ldots, w_n)$ and nothing else.

Security notions for semi-honest MPC, honest majority:

1) Correctness: $\forall$ inputs, $\forall$ randomness, all players output $f(w_1, \ldots, w_n)$ on input $(w_1, \ldots, w_n)$

2) Privacy/Zk: $\exists$ ppt sim s.t. $\forall$ inputs, $\forall$ sets $T \subseteq [n]$ of corrupted players s.t. $|T| \leq \frac{n}{2}$

\[
\{ \text{view of players in } T \} \cong \{ \text{Sim}(T, (w_i); o_t, f(w_1, \ldots, w_n)) \}
\]

Intuition: the only thing that leaks is $f(w_1, \ldots, w_n)$.

\[
\Rightarrow \text{Can achieve this information theoretically using BGW protocol}
\]

5 players/parties
2 corruptions tolerated (again - semi-honest model)

\[
\text{Similar to (but slightly different from) the protocol we've seen}
\]
MPC in the Head Construction

We build Zk pf system for NP relation R(x,w).
Recall: define as Zk proof.
Define \( s_x(w_1, \ldots, w_n) = R(x, w_1 \oplus w_2 \oplus \ldots \oplus w_n). \)

\( s_x(.) \) outputs 1 if set shares of a witness w s.t. \( s_x(x, w) = 1 \)
o otherwise.

Idea:

Proven for Zk pf system imagines 5 parties running MPC of \( s_x \)
Each party's input is a share \( w_i \) of NP witness w
Uses simulated MPC to convince V that x \( \in R(z) \).

Protocol
- Let \( C: \{0,1\}^* \rightarrow C \) be a commitment scheme
- We will use \( n=5 \) party protocol

\[ P(x, w) \]
Choose \( w_1, \ldots, w_n \in \{0,1\}^* \)
s.t. \( w_1 \oplus \ldots \oplus w_n = w \)
Simulate MPC of \( s_x(w_1, \ldots, w_n) \)

\[ C_i = C(V, w_i, c) \]
\[ C_c = C(V, w_n, c) \]

\[ V(x) \]
Choose \( i \leftarrow \{1, \ldots, n\} \)
s.t. \( i \neq j \)
Check that
1) \( C_i = C(V, w_i, c) \)
2) \( C_j = C(V, w_j, c) \)
3) \( w_i \) and \( V_{w_i} \) are consistent\( \Rightarrow \)
4) \( V_{i} = V_{w_i} \) output 1

\( acc/rej \)
MPC in the Head

* input $w_i$ to party $i$.

$\textbf{View}_i$ includes all msgs sent & rec'd by party $i$:
* all randomness of party $i$.

Say that 2 views $\textbf{View}_i, \textbf{View}_j$ are consistent if every msg sent by $i$ to $j$ is identical to msg received by $j$ from $i$ (and vice versa). (And $ij$ followed protocol.)

Claim: MPC in the head ZkP is correct, sound, Zk.

Correctness: By construction.

ZK: Need to construct a simulator $S$ s.t. $\forall V^*$

$$\textbf{View}_i \in S(x)$$

$$\textbf{View}_j \in \{S^*(x)\}$$

$S(x)$:

Guess $(ij)_x$ that $V^*$ will ask for $w_i, w_j \in \{0,1\}$

$(\textbf{View}_i, \textbf{View}_j) \leftarrow \text{Sim}_{\text{mpc}}((ij)_x, \{w_i, w_j\}, i)$

$C_i \leftarrow C(\textbf{View}_i)$

$C_j \leftarrow C(\textbf{View}_j)$

$C_k$ for $k \neq (ij)_x$ is commit to $0^*$

$(\cdot, \cdot) \leftarrow V^*(C_1, \ldots, C_n)$

if $(ij)_x \neq (\cdot, \cdot)$ abort
else output

$\langle (C_1, \ldots, C_n), (ij)_x, (\textbf{View}_i, C_i) \rangle$
MPC in the Head

Has knowledge, but let's just show soundness.

* If $x \in L(R)$ then $\forall w_1, \ldots, w_n \in \{0,1\}^*$ honest MPC plays output $0$ (by MPC correctness)

* Either
  - all views have output "0"
  - $\exists$ pair $(i,*)$ of views that are inconsistent.

* $\Pr[\text{opposite}] \geq \frac{1}{f^2} > \text{constant for } (S,d) \text{ protocol}$

$\# \text{ parties}$
MPC in the Head & PQ Crypto

Can use MPC in the head to prove knowledge of preimage of OWF

\[ R(x, w) = \begin{cases} 1 & \text{if } x = \text{SHA256}(w) \\ 0 & \text{otherwise} \end{cases} \]

MPC-in-head is a 3-round Sigma protocol

\[ \text{Can convert to signature scheme in ROM = k-Schnorr.} \]

\[ P(x, w) \quad \quad V(x) \]

\[ \begin{array}{c}
C_1, \ldots, C_n \\
(1, i) \\
\text{Verify, view, reply} \\
\text{Verify, view, reply}
\end{array} \]

\[ \text{Run A times in parallel to amplify soundness/knowledge} \]

A signature scheme:

\[ \text{Gen}(1^n) \rightarrow \text{Sample } w \leftarrow \{0, 1\}^n \\
\text{set } x \leftarrow H_n(w) \\
\text{Output } (pk = x, sk = w) \]

\[ \text{Sign}(sk, m) \rightarrow \text{Compute tx of MPCIH protocol} \\
\text{where challenge } (i, j) \leftarrow H(x, (c_1, \ldots, C_n), m) \\
\text{for random oracle } H \]

\[ \text{Verify}(pk, w, o) \rightarrow \text{Run MPC in head verification using } (i, j) \leftarrow H(x, (c_1, \ldots, C_n), m) \text{ as challenge} \]

\[ \Rightarrow \text{Security essentially follows same argument as we used for Schnorr} \]
Why do we care?! 

- Signatures from only Symmetric-key primitives (AES/SHA2) 
  - Comparing vs RSA, Schnorr.
  - Prove/verify time \( \approx 60 \text{ ms} \) using SHA2 w/ 80 bit security
- No known polytime Q attacks on these primitives
- LQ secure signatures from "standard assumptions" in ROM

Limitations

- \( \text{Sig size} \approx 1 \text{ MB} \) for 80 bit security
- Have to open \( \approx 80 \) views
- Each view is \( O(\ell \log \ell) \) bits long
- Lots of bits!

This is an active line of research (also happening at Stanford)

- In theory can get very short PQ signature in ROM from SHA2 \( O(\log \log \ell) \)
- In practice, most hash-based PQS have relatively long signatures

see Ligero (CCS '17) w/ \( O(\sqrt{\ell}) \)-sized \( \text{Sig} \)