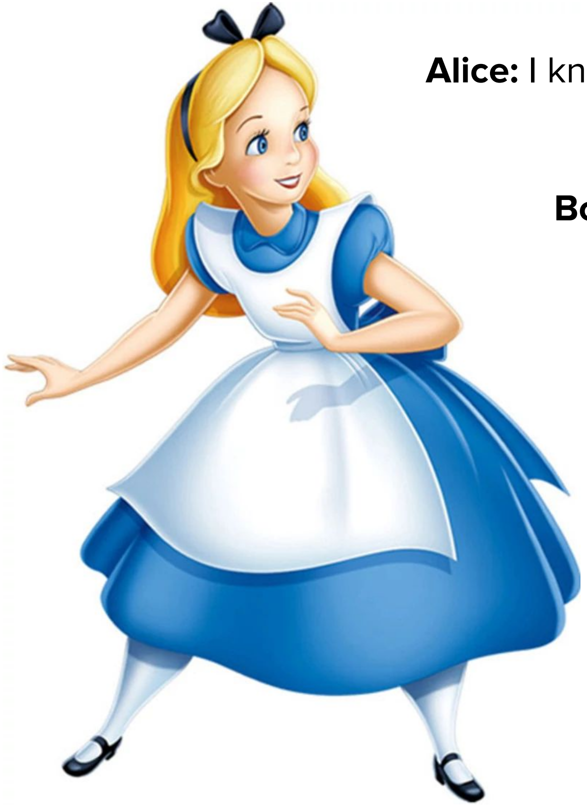
ZKP from MPC

April 16, 2019

TOY SCENARIO



Alice: I know how to make hats.

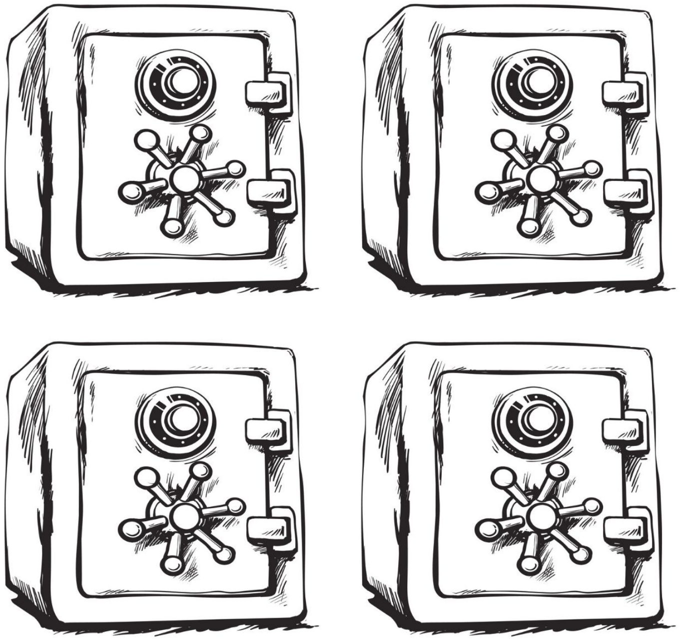
Bob: Prove it!

Alice: I'm not showing you; you might steal my design.

TOY SCENARIO: ZKP PROTOCOL

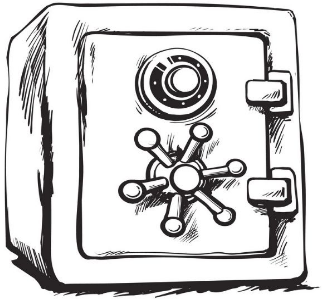
Step 1: Alice cuts up her hat (read: proof) into pieces.

TOY SCENARIO: ZKP PROTOCOL

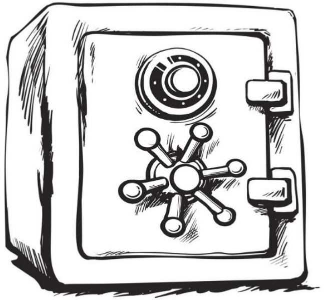


Step 2: Alice stores each piece into a different safe and then gives the safes to Bob.

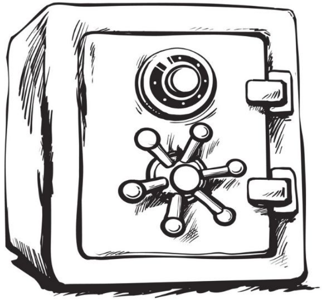
TOY SCENARIO: ZKP PROTOCOL

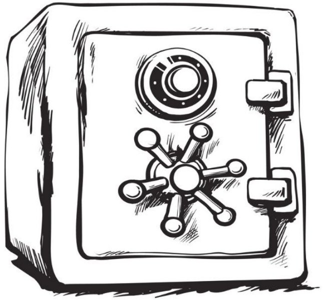


Step 3: Bob asks Alice to open two of the safes.



TOY SCENARIO: ZKP PROTOCOL





Step 3: Bob asks Alice to open two of the safes.

He checks to see if the pieces fit together (read: the views of the hat are consistent).

TOY SCENARIO: HAT LEMMA

Denote the hat pieces by $H_1, \cdots, H_n$. We say that a collection of hat pieces are **consistent** if they fit together and are clearly from the same hat.

TOY SCENARIO: HAT LEMMA

Denote the hat pieces by $H_1, \cdots, H_n$. We say that a collection of hat pieces are **consistent** if they fit together and are clearly from the same hat.

Lemma. $H_1, \cdots, H_n$ are consistent if and only if H_i and H_j are pairwise consistent for each $1 \leq i < j \leq n$.

TOY SCENARIO: EASY COROLLARY

Corollary. If $H_1, \cdots, H_n$ are not consistent, then there exists at least one pair of H_i and H_j that are not consistent with each other.

TOY SCENARIO: EASY COROLLARY

Corollary. If $H_1, \cdots, H_n$ are not consistent, then there exists at least one pair of H_i and H_j that are not consistent with each other.

Question. If Alice is lying and her hat pieces are not consistent, what is the probability that Bob discovers her deceit?

TOY SCENARIO: EASY COROLLARY

Corollary. If $H_1, \cdots, H_n$ are not consistent, then there exists at least one pair of H_i and H_j that are not consistent with each other.

Question. If Alice is lying and her hat pieces are not consistent, what is the probability that Bob discovers her deceit?

Answer.

$$\Pr\left(\text{Bob discovers Alice's deceit}\right) \geq \binom{n}{2}^{-1}$$

TOY SCENARIO: EASY COROLLARY

Corollary. If $H_1, \cdots, H_n$ are not consistent, then there exists at least one pair of H_i and H_j that are not consistent with each other.

Question. If Alice is lying and her hat pieces are not consistent, what is the probability that Bob discovers her deceit?

Answer.

$$\Pr\left(\text{Bob discovers Alice's deceit}\right) > \frac{1}{n^2}$$

TOY SCENARIO: LIMITATION

Bob still learns something about Alice’s hat designs from her cut-up pieces.

TOY SCENARIO: LIMITATION

Bob still learns something about Alice's hat designs from her cut-up pieces.

Question. Is there a way to split up her proof so that no two shares can be combined to reveal any information?

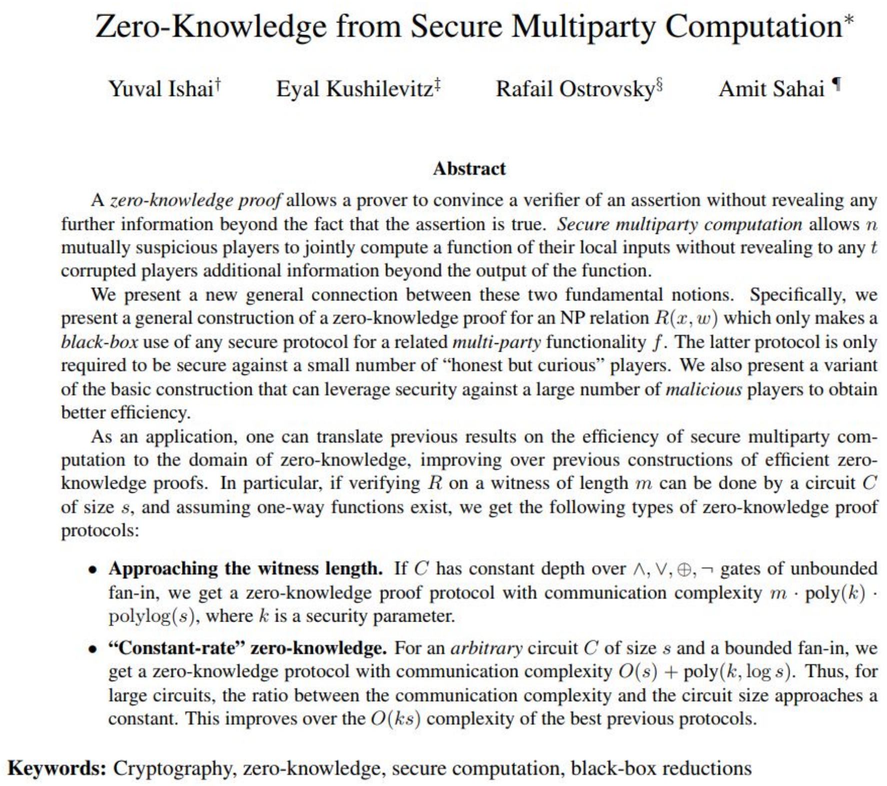
TOY SCENARIO: LIMITATION

Bob still learns something about Alice's hat designs from her cut-up pieces.

Question. Is there a way to split up her proof so that no two shares can be combined to reveal any information?

Answer. Use secure multi-party computation!

ZKP from MPC



Zero-Knowledge from Secure Multiparty Computation*

Yuval Ishai† Eyal Kushilevitz‡ Rafail Ostrovsky§ Amit Sahai ¶

Abstract

A *zero-knowledge proof* allows a prover to convince a verifier of an assertion without revealing any further information beyond the fact that the assertion is true. *Secure multiparty computation* allows n mutually suspicious players to jointly compute a function of their local inputs without revealing to any t corrupted players additional information beyond the output of the function.

We present a new general connection between these two fundamental notions. Specifically, we present a general construction of a zero-knowledge proof for an NP relation $R(x, w)$ which only makes a *black-box* use of any secure protocol for a related *multi-party* functionality f. The latter protocol is only required to be secure against a small number of "honest but curious" players. We also present a variant of the basic construction that can leverage security against a large number of *malicious* players to obtain better efficiency.

As an application, one can translate previous results on the efficiency of secure multiparty computation to the domain of zero-knowledge, improving over previous constructions of efficient zero-knowledge proofs. In particular, if verifying R on a witness of length m can be done by a circuit C of size s, and assuming one-way functions exist, we get the following types of zero-knowledge proof protocols:

- **Approaching the witness length.** If C has constant depth over $\wedge, \vee, \oplus, \neg$ gates of unbounded fan-in, we get a zero-knowledge proof protocol with communication complexity $m \cdot \text{poly}(k) \cdot \text{polylog}(s)$, where k is a security parameter.
- **"Constant-rate" zero-knowledge.** For an *arbitrary* circuit C of size s and a bounded fan-in, we get a zero-knowledge protocol with communication complexity $O(s) + \text{poly}(k, \log s)$. Thus, for large circuits, the ratio between the communication complexity and the circuit size approaches a constant. This improves over the $O(ks)$ complexity of the best previous protocols.

Keywords: Cryptography, zero-knowledge, secure computation, black-box reductions

PRELIMINARIES

Definition. An **NP-relation** $R(x,w)$ is an efficiently decidable binary relation that is polynomially bounded (i.e. $|w| \leq p(|x|)$ where p polynomial).

PRELIMINARIES

Definition. An **NP-relation** *R*(x,w) is an efficiently decidable binary relation that is polynomially bounded (i.e. $|w| \leq p(|x|)$ where p polynomial).

Examples. Let x = (V,E) be a graph

I. w is a Hamiltonian path in x
II. w is a 3-coloring of x

Let $x \in (\mathbf{Z}/n\mathbf{Z})^\times$ be relatively prime to n

III. w is a square root of x (i.e. $w^2 \cong x \bmod n$)

PRELIMINARIES

Definition. An **NP-relation** $R(x,w)$ is an efficiently decidable binary relation that is polynomially bounded (i.e. $|w| \leq p(|x|)$ where p polynomial).

Examples. Let $x = (V,E)$ be a graph

I. w is a Hamiltonian path in x
II. w is a 3-coloring of x

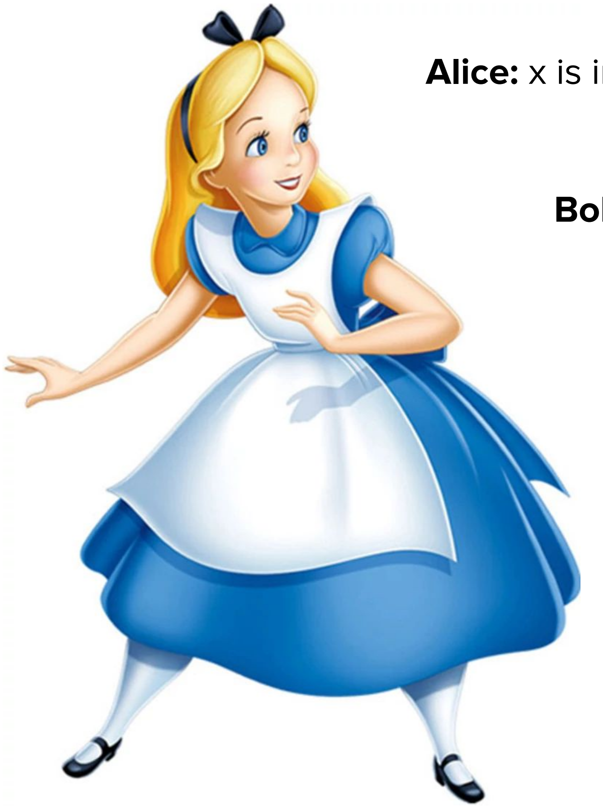
Let $x \in (\mathbf{Z}/n\mathbf{Z})^\times$ be relatively prime to n

III. w is a square root of x (i.e. $w^2 \cong x \bmod n$)

Remark. Any NP-relation R defines an NP-language:

$$L = \{ x : \exists w, R(x,w) = 1 \}.$$

PRELIMINARIES: GOAL



Alice: x is in *L*.

Bob: Prove it!

Alice: I don't want to share *w* with you.

HIGH-LEVEL OVERVIEW

Assume that there is an SMPC algorithm *f* that computes:

$$f(x, w_1, \cdots, w_n) \equiv R(x, w_1 \oplus \cdots \oplus w_n)$$

HIGH-LEVEL OVERVIEW

Assume that there is an SMPC algorithm *f* that computes:

$$f(x, w_1, \cdots, w_n) \equiv R(x, w_1 \oplus \cdots \oplus w_n)$$

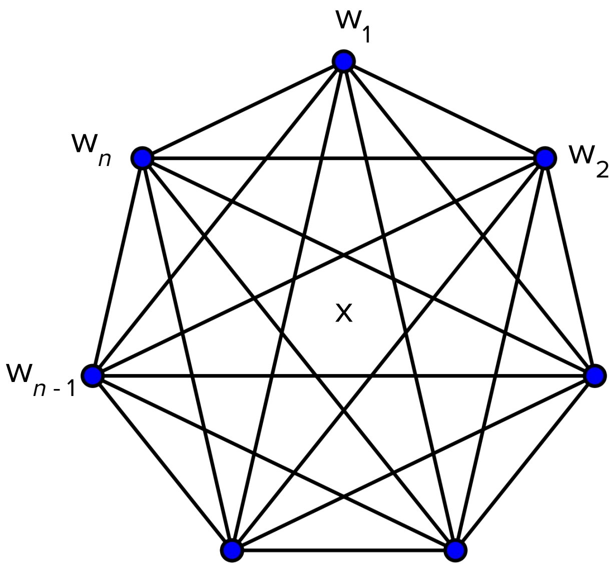


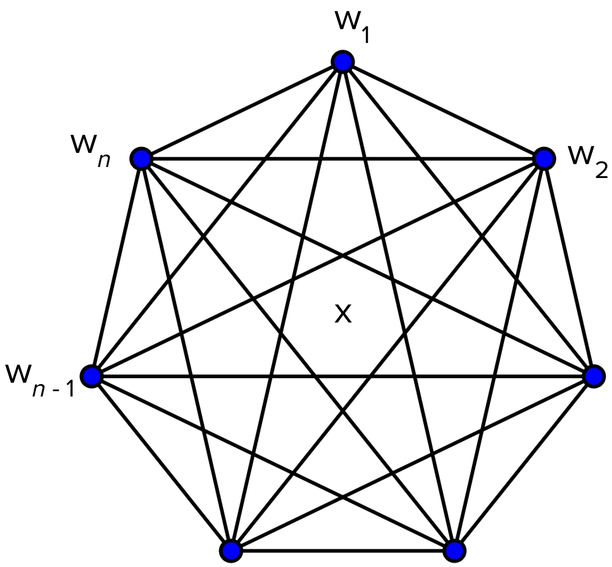
Figure. Each party has a secret share of *w*, where

$$w \equiv w_1 \oplus \cdots \oplus w_n$$

Together, they jointly and privately verify $R(x, w)$.

HIGH-LEVEL OVERVIEW

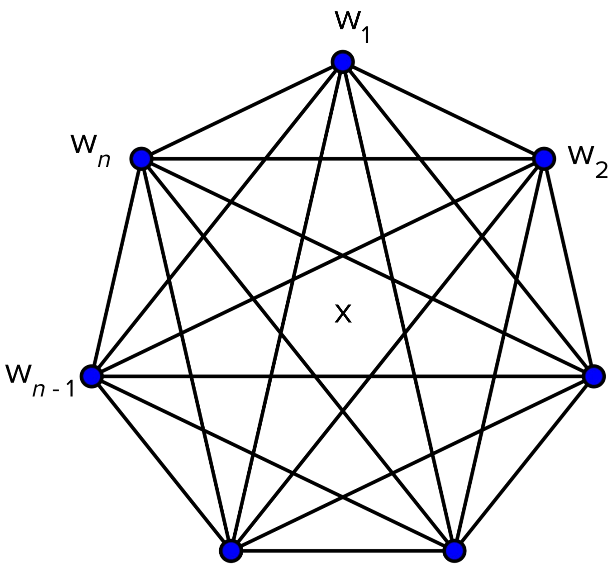
Step 1: Alice simulates the SMPC protocol that verifies that *w* is a witness to x ∈ *L*.



HIGH-LEVEL OVERVIEW

Step 1: Alice simulates the SMPC protocol that verifies that w is a witness to $x \in L$.

Remark. After running the protocol, each party will have a **view** of the messages it received and sent along with the randomness it used.

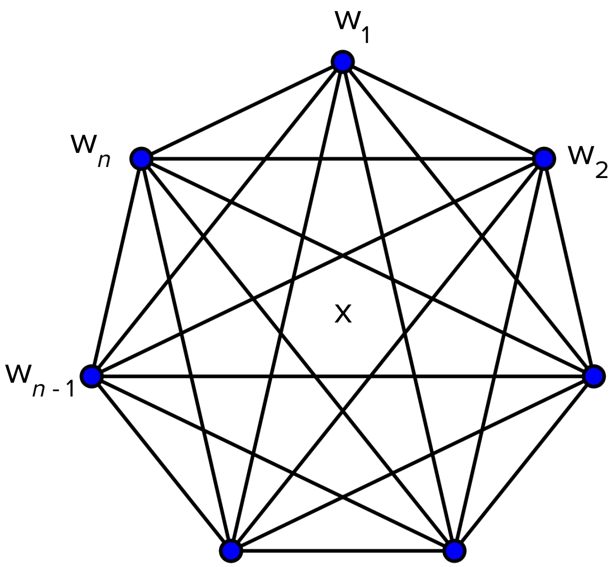


HIGH-LEVEL OVERVIEW

Step 1: Alice simulates the SMPC protocol that verifies that *w* is a witness to x ∈ *L*.

Remark. After running the protocol, each party will have a **view** of the messages it received and sent along with the randomness it used.

These views will act as cut-up pieces of the proof that x ∈ *L*.



HIGH-LEVEL OVERVIEW

Recall that a **commitment scheme**, COM, is a protocol that allows one to commit a message while hiding the message from others. Later, one is able to reveal the original message.

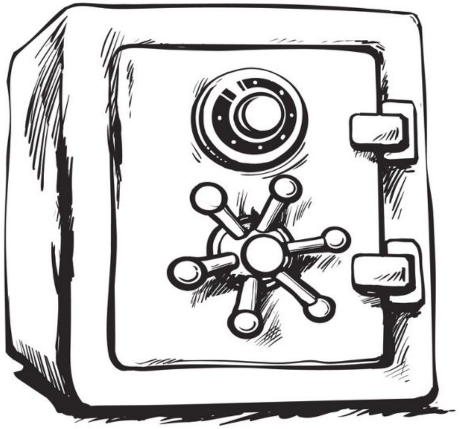
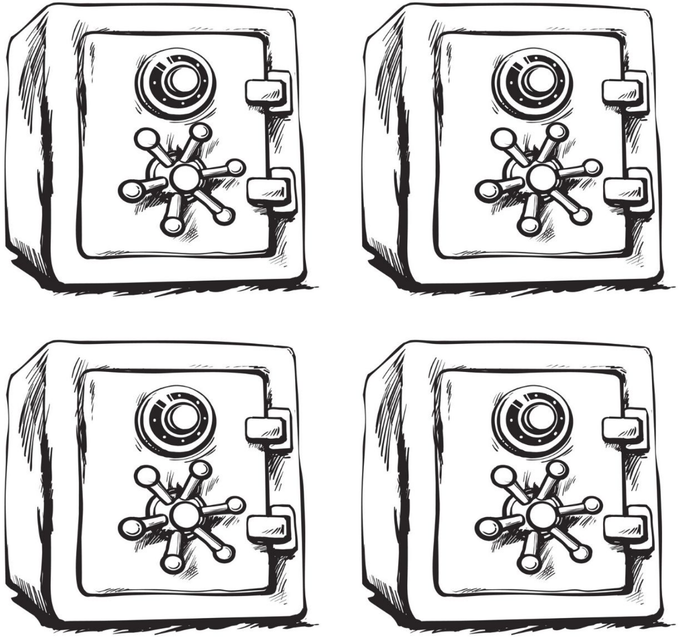


Figure. A commitment protocol is analogous to sending a message locked in a safe. Later in time, the sender can open the safe to reveal the committed message.

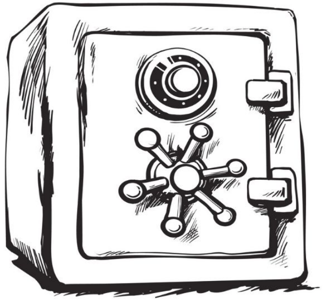
HIGH-LEVEL OVERVIEW

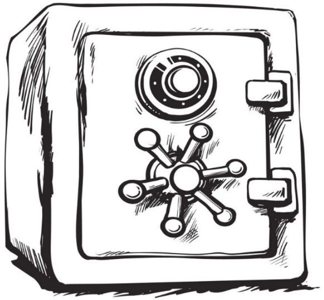
Step 2: Alice commits each of the views using a commitment scheme, sending the commits to Bob.



HIGH-LEVEL OVERVIEW

Step 3: Bob chooses two commitments for Alice to decommit, revealing two views of the MPC protocol.

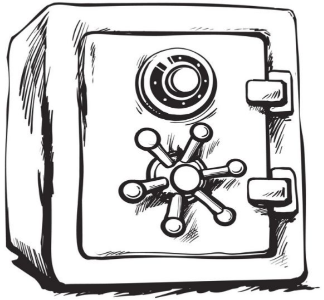


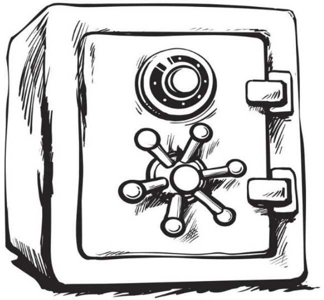


HIGH-LEVEL OVERVIEW

Step 3: Bob chooses two commitments for Alice to decommit, revealing two views of the MPC protocol.

Decision. If the views are inconsistent, Bob rejects. Otherwise, Bob accepts.





HIGH-LEVEL OVERVIEW

Remark. The **soundness error** (i.e. probability that Bob accepts a invalid proof) at this point is as before:

$$\Pr\left(\text{Bob accepts a false proof}\right) < 1 - \frac{1}{n^2}$$

HIGH-LEVEL OVERVIEW

Remark. The **soundness error** (i.e. probability that Bob accepts a invalid proof) at this point is as before:

$$\Pr\left(\text{Bob accepts a false proof}\right) < 1 - \frac{1}{n^2}$$

We could make this probability negligibly small (say, w.p. 2^{-k}) through repetitions (say, kn^2 times).

HIGH-LEVEL OVERVIEW: UPSHOT

Main Result. We can build ZKP protocols given black-box access to MPC protocols.

FORMALITIES

Let P be the prover and V be the verifier. Let L be an NP-language.

FORMALITIES

Let *P* be the prover and *V* be the verifier. Let *L* be an NP-language.

Definition (informal). A **zero-knowledge proof** (ZKP) is a protocol (*P*,*V*) where for each $x \in L$, the prover tells the verifier essentially nothing but $x \in L$.

FORMALITIES

Let *P* be the prover and *V* be the verifier. Let *L* be an NP-language.

Definition (informal). A **zero-knowledge proof** (ZKP) is a protocol (*P*,*V*) where for each x ∈ *L*, the prover tells the verifier essentially nothing but x ∈ *L*.

FORMALITIES

Let P be the prover and V be the verifier. Let L be an NP-language.

Definition (informal). A **zero-knowledge proof** (ZKP) is a protocol (P,V) where for each $x \in L$, the prover tells the verifier essentially nothing but $x \in L$.

Remark. For a given input x, the prover and verifier will exchange messages according to some underlying probability distribution, say A_x, that depends on x.

We call a collection of distributions $\{ A_x \}_{x \in X}$ a **probability ensemble** indexed by X.

FORMALITIES

Definition. Two probability ensembles $\{ A_x \}_{x \in X}$ and $\{ B_x \}_{x \in X}$ are **computationally indistinguishable** if for any non-uniform efficient distinguisher D,*

$$\left| \Pr\left[D(A_x) = 1\right] - \Pr\left[D(B_x) = 1\right] \right| \leq \varepsilon(|x|)$$

where $\varepsilon(\bullet)$ is a *negligible* function.**

*by efficient, we mean probabilistic polytime (PPT) algorithm, and by non-uniform, we mean that the algorithm can depend on the length of x.

**by negligible, we mean that for all c > 0, asymptotically, $\varepsilon(n) < o(n^{-c})$.

FORMALITIES

Definition. A protocol (P, V) is a **zero-knowledge proof protocol** for the NP relation R (with corresponding language L), if it satisfies:

I. **completeness:** if $x \in L$, and if both players follow the protocol, the verifier always accepts
II. **soundness:** for every malicious and computationally unbounded prover P^*, if $x \notin L$, the verifier accepts with negligible probability $\varepsilon(|x|)$
III. **zero-knowledge:** for any malicious PPT verifier V^*, there is a PPT simulator M^*, such that the view of V^* is computationally indistinguishable from the output distribution $M^*(x)$.

FORMALITIES: ZERO-KNOWLEDGE

Remark. What does it mean for the view of V^* to be computationally indistinguishable from the output distribution $M^*(x)$?

FORMALITIES: ZERO-KNOWLEDGE

Remark. What does it mean for the view of V^* to be computationally indistinguishable from the output distribution $M^*(x)$?

World I

The verifier V^* is told x, and then interacts with Alice. All new information is encapsulated in a string:

$$\textbf{View}_{V^*}(x,w)$$

the collection of messages and random bits that V^* saw through the interaction.

FORMALITIES: ZERO-KNOWLEDGE

Remark. What does it mean for the view of V^* to be computationally indistinguishable from the output distribution $M^*(x)$?

World I

The verifier V^* is told x, and then interacts with Alice. All new information is encapsulated in a string:

$\mathbf{View}_{V^*}(x,w)$

the collection of messages and random bits that V^* saw through the interaction.

World II

In the simulation world, a random string:

$M^*(x)$

is generated from the input x. There is no access to Alice.

FORMALITIES: ZERO-KNOWLEDGE

Remark. What does it mean for the view of *V** to be computationally indistinguishable from the output distribution *M**(x)?

This means that it is not possible to computationally determine which world we're actually in:

$$\left| \Pr\left[D(\mathbf{View}_{V^*}(x,w)) = 1\right] - \Pr\left[D(M^*(x)) = 1\right] \right| \leq \delta(|x|)$$

for some negligible function δ(•).

FORMALITIES: ZERO-KNOWLEDGE

Remark. What does it mean for the view of V^* to be computationally indistinguishable from the output distribution $M^*(x)$?

This means that it is not possible to computationally determine which world we're actually in:

$$\left| \Pr\left[D(\mathbf{View}_{V^*}(x, w)) = 1\right] - \Pr\left[D(M^*(x)) = 1\right] \right| \leq \delta(|x|)$$

for some negligible function $\delta(\bullet)$.

It follows that Bob will not learn anything about *w* that he can efficiently recover even after interacting with Alice.

FORMALITIES: MPC PRIMITIVES

Definition (informal). An n-party MPC protocol Π_f computes f with ***t*-privacy** if no matter how a subset of t corrupted players collude, they can gain no additional information beyond their shared secrets.

FORMALITIES: MPC PRIMITIVES

Definition (informal). An *n*-party MPC protocol Π_f computes *f* with ***t*-privacy** if no matter how a subset of *t* corrupted players collude, they can gain no additional information beyond their shared secrets.

Remark. We can have different versions of *t*-privacy:

- perfect *t*-privacy: same distribution
- statistical *t*-privacy: statistical indistinguishability
- computational *t*-privacy: computational indistinguishability

FORMALITIES: MPC PRIMITIVES

Definition. Let $1 \leq t < n$. An MPC protocol Π_f realizes f with **perfect *t*-privacy** if there is a PPT simulator SIM such that for any input x, $w_1, \cdots, w_n$, and for any set of corrupted players $T \subset [n]$ of size t, the joint view of those T players is distributed identically to:

$$\mathrm{SIM}(T, x, (w_i)_{i \in T}, f_T(x, w_1, \cdots, w_n)).$$

Similar definitions for **statistical *t*-privacy** and **computational *t*-privacy**, with respect to a security parameter k.

ZK PROTOCOL

Zero-knowledge protocol Π_R in the commitment-hybrid model

1. The prover picks at random $w_1, \ldots, w_n \in \{0,1\}^m$ whose exclusive-or equals the witness w. She emulates "in her head" the execution of Π_f on input $(x, w_1, \ldots, w_n)$ (this involves choosing randomness for the n players and running the protocol). Based on this execution, the prover prepares the views $V_1, \ldots, V_n$ of the n players; she separately commits to each of these n views.
2. Verifier picks at random distinct player indices $i, j \in [n]$ and sends them to the prover.
3. Prover "opens" the commitments corresponding to the two views V_i, V_j.
4. Verifier accepts if and only if:
   (a) the prover indeed successfully opened the two requested views,
   (b) the outputs of both P_i and P_j (which are determined by their views) are 1, and
   (c) the two opened views are consistent with each other (with respect to x and Π_f, see Definition 2.2).

ZK PROTOCOL

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R from the previous slide is a zero-knowledge proof protocol for the NP-relation *R* with soundness error $\varepsilon \leq 1 - n^{-2}$.

ZK PROTOCOL

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R from the previous slide is a zero-knowledge proof protocol for the NP-relation *R* with soundness error $\varepsilon \leq 1 - n^{-2}$.

Proof. COMPLETENESS. Follows from correctness of Π_f.

ZK PROTOCOL

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R from the previous slide is a zero-knowledge proof protocol for the NP-relation *R* with soundness error $\varepsilon \leq 1 - n^{-2}$.

Proof. COMPLETENESS. Follows from correctness of Π_f.

SOUNDNESS. Follows from "Hat Lemma".

ZK PROTOCOL

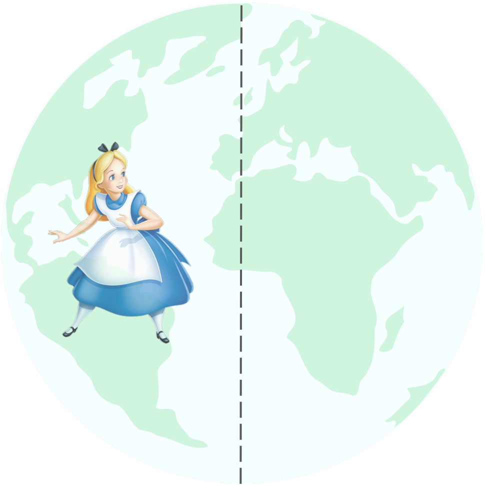
Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R from the previous slide is a zero-knowledge proof protocol for the NP-relation *R* with soundness error $\varepsilon \leq 1 - n^{-2}$.

Proof. COMPLETENESS. Follows from correctness of Π_f.
SOUNDNESS. Follows from "Hat Lemma".
ZERO-KNOWLEDGE. Construct the following simulation $M^*(x)$.

ZK PROTOCOL: proof of zero-knowledge

World I

1. The verifier V^* is told x, uses randomness to choose $1 \leq i < j \leq n$.

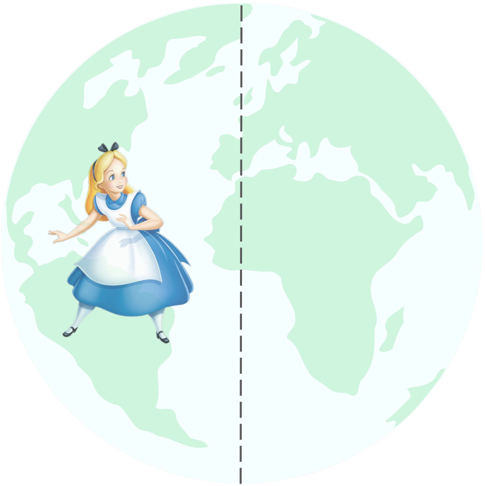


ZK PROTOCOL: proof of zero-knowledge

World I

1. The verifier *V** is told x, uses randomness to choose $1 \leq i < j \leq n$.

2. Independently, *P* uniformly at random splits up *w* into *n* shares, and computes the Views for i and j.



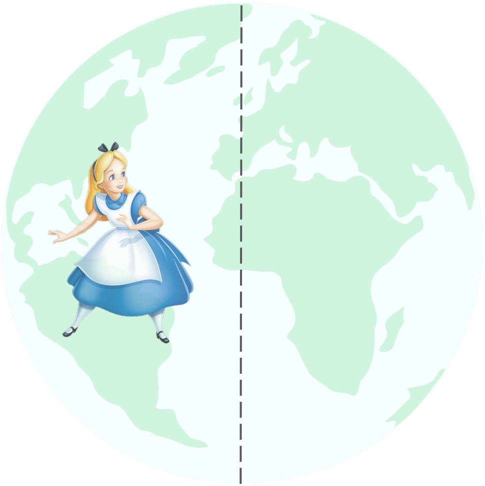
ZK PROTOCOL: proof of zero-knowledge

World I

1. The verifier *V** is told x, uses randomness to choose $1 \leq i < j \leq n$.

2. Independently, *P* uniformly at random splits up *w* into *n* shares, and computes the Views for i and j.

3. **View**$_i$ and **View**$_j$ are revealed to *V**.



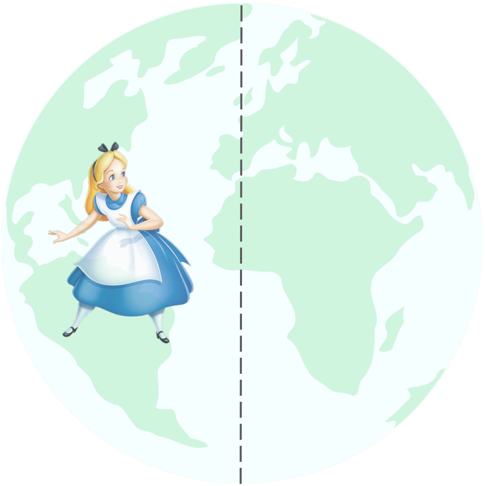
ZK PROTOCOL: proof of zero-knowledge

World I

1. The verifier V^* is told x, uses randomness to choose $1 \leq i < j \leq n$.

2. Independently, P uniformly at random splits up w into n shares, and computes the Views for i and j.

3. $\textbf{View}_i$ and $\textbf{View}_j$ are revealed to V^*.



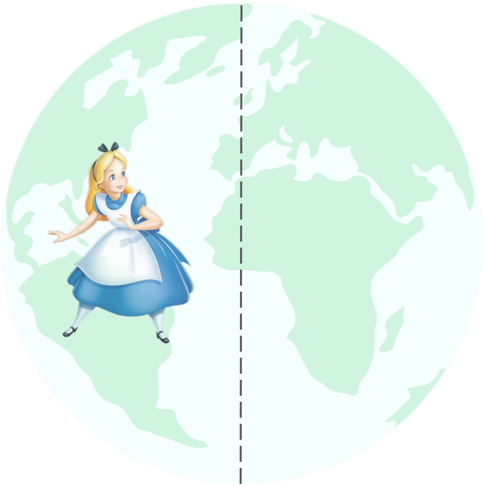
World II

1. M^* uses the same randomness that V^* uses to choose $1 \leq i < j \leq n$.

ZK PROTOCOL: proof of zero-knowledge

World I

1. The verifier V^* is told x, uses randomness to choose $1 \leq i < j \leq n$.
2. Independently, P uniformly at random splits up w into n shares, and computes the Views for i and j.
3. $\textbf{View}_i$ and $\textbf{View}_j$ are revealed to V^*.



World II

1. M^* uses the same randomness that V^* uses to choose $1 \leq i < j \leq n$.
2. M^* uniformly at random selects w_i and w_j and runs:

   SIM(T={i,j}, x, (w_i, w_j), 1)

   to generate simulated views for i, j.

ZK PROTOCOL

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R from the previous slide is a zero-knowledge proof protocol for the NP-relation *R* with soundness error $\varepsilon \leq 1 - n^{-2}$.

Proof. COMPLETENESS. Follows from correctness of Π_f.
SOUNDNESS. Follows from "Hat Lemma".
ZERO-KNOWLEDGE. Follows from 2-privacy of SIM. ∎

ZK PROTOCOL: EXTENSIONS

Question. What are issues with the current description of the ZK protocol?

ZK PROTOCOL: EXTENSIONS

Question. What are issues with the current description of the ZK protocol?

Issue 1. Soundness error $1 - n^{-2}$ is not great.

ZK PROTOCOL: EXTENSIONS

Question. What are issues with the current description of the ZK protocol?

Issue 1. Soundness error $1 - n^{-2}$ is not great.

Potential solution? Repetitions of protocol.

ZK PROTOCOL: EXTENSIONS

Question. What are issues with the current description of the ZK protocol?

Issue 1. Soundness error $1 - n^{-2}$ is not great.

Potential solution? Repetitions of protocol.

- Do multiple rounds reveal information?
- Can we overcome inefficiency of multiple rounds?

ZK PROTOCOL: EXTENSIONS

Question. What are issues with the current description of the ZK protocol?

Issue 1. Soundness error $1 - n^{-2}$ is not great.

Potential solution? Repetitions of protocol.

- Do multiple rounds reveal information?
- Can we overcome inefficiency of multiple rounds?

Issue 2. Assumption of ideal primitives for MPC and COM.

ZK PROTOCOL v.2

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R as before. Sequential repetitions kn^2 times results in soundness error $\varepsilon \leq 2^{-k}$.

ZK PROTOCOL v.2

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R as before. Sequential repetitions kn^2 times results in soundness error $\varepsilon \leq 2^{-k}$.

Proof. COMPLETENESS + SOUNDNESS. Follows from single round.

ZK PROTOCOL v.2

Theorem. Let Π_f be a correct and computational 2-private MPC protocol. Then Π_R as before. Sequential repetitions kn^2 times results in soundness error $\varepsilon \leq 2^{-k}$.

Proof. COMPLETENESS + SOUNDNESS. Follows from single round.

ZERO-KNOWLEDGE. Need indistinguishability from repetitions.

ZK PROTOCOL v.2

Definition. Two probability ensembles $\{ A_x \}_{x \in X}$ and $\{ B_x \}_{x \in X}$ are **indistinguishable by polynomial-time sampling** if for any non-uniform efficient distinguisher *D* and *m* = p(|x|),

$$\left| \Pr\left[D(A_x^{(1)}, \ldots, A_x^{(m)}) = 1 \right] - \Pr\left[D(B_x^{(1)}, \ldots, B_x^{(m)}) = 1 \right] \right| < \varepsilon(|x|)$$

where ε(•) is a *negligible* function and p is a polynomial.

HYBRID TECHNIQUE

Theorem. Two probability ensembles $\{ A_x \}$ and $\{ B_x \}$ are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

HYBRID TECHNIQUE

Theorem. Two probability ensembles { A_x } and { B_x } are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D'* can distinguish between two sequences:

$A^1, \ldots, A^m$ and $B^1, \ldots, B^m$,

HYBRID TECHNIQUE

Theorem. Two probability ensembles $\{ A_x \}$ and $\{ B_x \}$ are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D'* can distinguish between two sequences:

$$A^1, \ldots, A^m \quad \text{and} \quad B^1, \ldots, B^m,$$

then consider the chain of hybrid sequences:

HYBRID TECHNIQUE

Theorem. Two probability ensembles { A_x } and { B_x } are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D’* can distinguish between two sequences:

$$A^1, \ldots , A^m \quad \text{and} \quad B^1, \ldots , B^m,$$

then consider the chain of hybrid sequences:

1. $A^1, A^2, \ldots , A^{m-1}, A^m$

HYBRID TECHNIQUE

Theorem. Two probability ensembles $\{ A_x \}$ and $\{ B_x \}$ are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D'* can distinguish between two sequences:

$$A^1, \ldots, A^m \quad \text{and} \quad B^1, \ldots, B^m,$$

then consider the chain of hybrid sequences:

1. $A^1, A^2, \ldots, A^{m-1}, A^m$
2. $A^1, A^2, \ldots, A^{m-1}, B^m$

HYBRID TECHNIQUE

Theorem. Two probability ensembles $\{ A_x \}$ and $\{ B_x \}$ are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D'* can distinguish between two sequences:

$$A^1, \ldots, A^m \quad \text{and} \quad B^1, \ldots, B^m,$$

then consider the chain of hybrid sequences:

1. $A^1, A^2, \ldots, A^{m-1}, A^m$
2. $A^1, A^2, \ldots, A^{m-1}, B^m$
3. $A^1, A^2, \ldots, B^{m-1}, B^m$

$\vdots$

HYBRID TECHNIQUE

Theorem. Two probability ensembles $\{ A_x \}$ and $\{ B_x \}$ are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Proof (sketch). If an algorithm *D'* can distinguish between two sequences:

$$A^1, \ldots, A^m \quad \text{and} \quad B^1, \ldots, B^m,$$

then consider the chain of hybrid sequences:

1. $A^1, A^2, \ldots, A^{m-1}, A^m$
2. $A^1, A^2, \ldots, A^{m-1}, B^m$
3. $A^1, A^2, \ldots, B^{m-1}, B^m$

⋮

4. $A^1, B^2, \ldots, B^{m-1}, B^m$
5. $B^1, B^2, \ldots, B^{m-1}, B^m$

HYBRID TECHNIQUE

Theorem. Two probability ensembles { A_x } and { B_x } are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Question. What is the probability that *D'* distinguishes between neighboring hybrid sequences?

HYBRID TECHNIQUE

Theorem. Two probability ensembles { A_x } and { B_x } are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Question. What is the probability that *D'* distinguishes between neighboring hybrid sequences?

Idea. The sum of the probability that *D'* distinguishes neighboring hybrid sequences telescopes to the probability that *D'* distinguishes original two sequences.

HYBRID TECHNIQUE

Theorem. Two probability ensembles { A_x } and { B_x } are computationally indistinguishable if and only if they are indistinguishable by polynomial-time sampling.

Question. What is the probability that *D’* distinguishes between neighboring hybrid sequences?

Idea. The sum of the probability that *D’* distinguishes neighboring hybrid sequences telescopes to the probability that *D’* distinguishes original two sequences.

Ergo. *D’* can distinguish one of these *m* neighboring probabilities with probability greater than ε(|x|)/m.
Use to construct single distinguisher! ∎

REMARKS ON EXTENSIONS

If you're interested, the rest of paper goes into:

1. More efficient technique using t-robustness assumptions (allows verifier to open more than two safes).

2. Incorporation of imperfect MPC and commitment protocols into security analysis.

3. More on efficiency and coin-flipping.

ACKNOWLEDGMENTS

This lecture by Geelon So follows the paper [IKOS07], and was given for the course COMS 6998 Privacy-Preserving Technologies, by Prof. Jeannette Wing. It was part of a longer lecture jointly presented along with Alex Nicita and Abhi Shah, with guest speaker Eran Tromer.

REFERENCES

[IKOS07] Ishai, Yuval, et al. "Zero-knowledge from secure multiparty computation." *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*. ACM, 2007.

[Gold07] Goldreich, Oded. Foundations of cryptography: volume 1, basic tools. Cambridge university press, 2007.