Computational Complexity: Lecture 26

Agenda: - Zero knowledge proofs.

Recap: IP: $L \in IP \iff$ there is an efficient verifier with the following guarantees.

$x \in L \Rightarrow \exists$ Prover P : $P_r[V \leftrightarrow P(x) = \text{Accept}] \geq 2/3$

$x \notin L \Rightarrow \forall$ Provers P : $P_r[V \leftrightarrow P(x) = \text{accept}] \leq 1/3$.

Prop: ▷ We may assume perfect completeness.

▷ We may assume verifier only sends random strings each round.

▷ We may assume public coins protocol.

} IP=AM[poly].

Typical set up:

Guard against dishonest provers

What if we turn the tables?

"Banking"

"I have a pf of $P \neq NP$".

Can a prover convince the verifier that φ is SAT without revealing anything about the assignment?

Zero knowledge [Goldwasser-Micali]

Zero knowledge by example: Coke vs Pepsi.

GNI: $\{(G_1, G_2) : G_1 \not\cong G_2\}$.

V: $b \in_R \{1,2\}$, $\sigma \in_R S_n$, $H = \sigma(G_b)$ —H→

Acc. if $b = c$. ←$c \in \{1,2\}$—

Intuitively, this looks like zero knowledge, but why?

The prover basically returns b ... but V knew b already.

How do we formalise this in general?

"The verifier must learn "nothing else" from the conv. with the honest prover".

View of V.

What does the verifier see? Transcript, + internal rand. bits.

Key idea: Suppose V can generate the view himself, then V can't have learned anything from P!

Revisiting GNI:

Input = (G_1, G_2) with $G_1 \not\cong G_2$

Verifier: $H = \sigma(G_b)$ →, ← c

$b \in \{1,2\}$, $\sigma \in_R S_n$

(b, σ, H, b)

Simulator:
- Pick $b \in_R \{1,2\}$
- Pick $\sigma \in_R S_n$ $H = \sigma(G_b)$
- Return (b, σ, H, b)

Defn: An interactive protocol $(V \leftrightarrow P)$ is a zero knowledge protocol for a language L if:

- [Completeness] $x \in L \Rightarrow \Pr[V \leftrightarrow P(x) = acc] \geq 2/3$
- [Soundness] $x \notin L \Rightarrow \forall P^*: \Pr[V \leftrightarrow P^*(x) = acc] \leq 1/3$
- [Zero knowledge] If $x \in L$, then for any randomised verifier V^*, there is an efficient (ZPP) simulator S^* s.t $\{S^*(x)\} \equiv \{\text{View}_{V^*}(V^* \leftrightarrow P)(x)\}$.

The right simulator for GNI (works for even dishonest verifiers)

$S^*(G_1, G_2)$:

1) ~~Pick $b \in_R \{1,2\}$.~~
2) Run V^* and see what V^* chose in step 1.
3) Say V^* chose (b', σ).
4) ~~If $b = b'$:~~ return (b', σ, H, b').
   Else: ~~go to step 1~~.

Ex: Build the simulator for GI (Acadly protocol)

Pause... this takes a while to digest!

Variants of ZK:

- Statistical indistinguishability.

D_1 & D_2 are two distributions on Ω.

$$\Delta(D_1, D_2) := \frac{1}{2} \sum_{\omega \in \Omega} |D_1(\omega) - D_2(\omega)|$$

Fact: If $\Delta(D_1, D_2) \le \varepsilon$, then for any event $E \subseteq \Omega$, we have

$$\left| \Pr_{x \sim D_1}[x \in E] - \Pr_{x \sim D_2}[x \in E] \right| \le \varepsilon.$$

$$\left| \sum_{x \in E} D_1(x) - \sum_{x \in E} D_2(x) \right| \le \sum_{x \in E} |D_1(x) - D_2(x)|$$

- Computational indistinguishability:

D_1 & D_2 are dists. on $\{0,1\}^n$. They are (ε, s)-comp. indistinguishable if for any circuit C of size s, we have

$$\left| \Pr_{x \sim D_1}[C(x) = 1] - \Pr_{x \sim D_2}[C(x) = 1] \right| \le \varepsilon.$$

Typically $s = 1/\varepsilon$ $\qquad$ $\varepsilon << 1/n^c$ for any const c.

The rest of the class will deal with CZK only.

Thm: [Goldreich-Micali-Wigderson] Assuming "cryptography exists", every language in NP has a zero knowledge protocol.

WHAT?!

(This one of the most surprising results in complexity).

And this result followed shortly after.

Thm: [Ben-Or, Goldreich, Goldwasser, Håstad, Kilian, Micali, Rogaway].
"Everything provable is provable in zero knowledge".

Assuming cryptography exists, any $L \in IP$ has a zero knowledge protocol.

What does "cryptography exists" mean?

Formally: there are "one-way fns".

There must be secure encryption algorithms.

... what does that mean?

$Enc: \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^m$.

$Dec: \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^n$

→ Efficiency: Enc, Dec to be efficient

→ Correct.

$Dec(k, Enc(k,x)) = x$

$Enc_k(x)$

↓ Secure

$\{Enc_k(x)\} \overset{c.i}{\equiv} \{Enc_k(y)\}$

Uniqueness: $Enc_{k_1}(x_1) = Enc_{k_2}(x_2) \Rightarrow x_1 = x_2$

[Goldreich-Micali-Wigderson] CZK for NP.

Graph 3Col: $\{G : G \text{ is 3-colourable}\}$.

Prover: (Knows $\sigma : [n] \to \{1,2,3\}$.)

Picks $\pi \in S_3$. $\tilde{\sigma}(i) = \pi\,\sigma(i)$

Builds locked boxes $B_1, \ldots, B_n$ with B_i containing $\tilde{\sigma}(i)$.

Sends $B_1, \ldots, B_n$ to verifier.

Verifier: choose $(i,j) \in G$ and sends to prover.

Prover: sends keys to B_i & B_j

Verifier: checks if B_i & B_j have diff. colours.

[Completeness]: G - 3col $\Rightarrow$ Verifier accepts w.p 1.

[Soundness] $G \notin$ 3-coL $\Rightarrow$ There is some edge where σ fails.

$\therefore \Pr[V \text{ acc}] \leq \left(1 - \frac{1}{|E|}\right)^m \ll 1/3$

Repeat m times

Is this protocol ZK? Of course not!

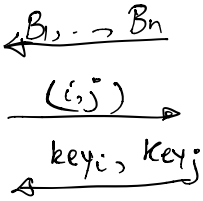
Fix: Prover shuffles the colours in each round!

[Zero knowledge]: Randomly choose $(i,j) \in E$

Randomly choose $c_1 \neq c_2 \in \{1,2,3\}$.

Create locked boxes $B_1, \dots, B_n$ with B_i & B_j containing c_1 & c_2

Return:



Building a digital locked box:

Commitment scheme: Commit: (key, value) $\longrightarrow$ Commitment

Reveal (k, v, c) Does $c = Com(k,v)$

Computationally hiding:

$$x \neq y \Rightarrow \{Com(k,x)\}_k \underset{c.i}{\equiv} \{Commit(k,y)\}$$

Perfectly binding:

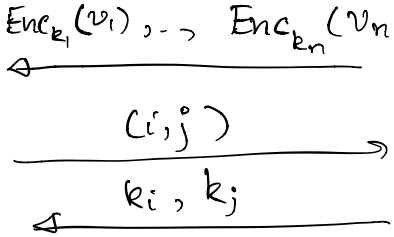
$c = Commit(k, v)$.

$Commit(k,v) = Commit(k',v') \Rightarrow v = v'$

Then $Reveal(k', v', c) = True \Rightarrow v = v'$.

Eg: $Commit(k, v) = Enc(k, v)$

ZK for 3Col:



Verify that they satisfy all the requirements.

□

Simulator:

1) Pick $(i,j) \in_R E$.
2) Picks $c_i \neq c_j \in \{1,2,3\}$.
3) Puts c_i & c_j in B_i, B_j.
   Sends $B_1 \to B_n$ to verifier (others empty).
4) If V^* chooses (i,j)
   return the view.
   Else go to step 1.

Can we do this for all of IP?

Can every interactive protocol be made Zero knowledge?

[BGGHKMR]: Yes!

– Next class!