

LECTURE 17

Note Title

Today : • Complete $IP \geq PSPACE$

• Few words on Knowledge

———— x ————

———— x ————

Review from last lecture

Polynomial Construction Sequence

$P_0, P_1, P_2, \dots, P_\ell$ field = \mathbb{F}

- Each polynomial of degree $\leq d$

- " # variables $\leq m$

- P_0 computable in time $\leq t$

- P_i computable ^{in time t} with oracle for

P_{i-1} with # calls $\leq w$

\Rightarrow Given $\bar{a} = (a_1, \dots, a_m) \in \mathbb{F}^m$ & $b \in \mathbb{F}$

Can prove interactively that " $P_\ell(\bar{a}) = b$ "

in time $\text{poly}(l, d, |\mathbb{F}|, m, t, w)$.
provided $|\mathbb{F}|$ large.

Typical Phase of interaction

$$P_i(a^{(i)}) = b^{(i)} ?$$

Verifier

Prover

Compute V_1, \dots, V_w

s.t.

$P_i(a^{(i)})$ can be

computed from

$P_{i-1}(V_1, \dots, V_w)$.

Compute Curve

C s.t. $C(j) = V_j$;

$h \leftarrow P_{i-1}(C(t))$

C, h



• Verify

$$C(i) = V_i$$

• Verify $b^{(i)} = f_i(h(1), \dots, h(w))$

Pick $t_0 \in \mathbb{F}$ at random

$$a^{(i-1)} = C(t_0); \quad b^{(i-1)} = h(t_0)$$

t_0



Poly Construction Sequence for PSPACE

Given: Machine M , n s.t.
Configurations of machine are
 S bits long.

Goal: To decide if initial config $a_1 \dots a_s$
leads to (unique) accepting config $b_1 \dots b_s$
in 2^s steps.

Define: Function

$$F_0, F_1, \dots, F_s : \mathbb{F}^{2^s} \rightarrow \mathbb{F}$$

$$F_i(\bar{\sigma} = (\sigma_1 \dots \sigma_s), \bar{\tau} = (\tau_1 \dots \tau_s)) = \begin{cases} 1 & \text{if } \sigma \Rightarrow \tau \\ & \text{in } 2^i \text{ steps} \\ 0 & \text{if } \sigma, \tau \in \{0,1\}^s \text{ or.} \\ \text{arbitrary} & \text{for } \sigma, \tau \notin \{0,1\}^s \end{cases}$$

① Can define F_0 s.t. it is a polynomial of degree $O(1)$ in each variable; and is computable in polytime.

$$② \quad F_i(\bar{x}, \bar{y}) = \sum_{z \in \{0,1\}^s} F_{i-1}(\bar{x}, \bar{z}) \cdot F_{i-1}(z, \bar{y})$$

↑

- F_i computable from F_{i-1} 😊

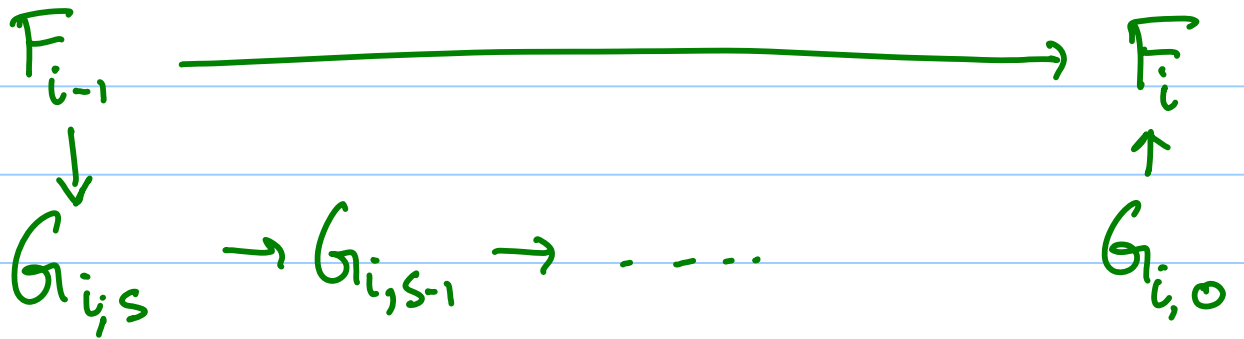
- degree in each variable $\leq C$ 😊

- F_i needs 2^s values of F_{i-1} 😞

↑

Need to fix this

Breaking Exponential Sum into Smaller Pieces.



$$G_{i,s}(x, y, z) = F_{i-1}(x, z) \cdot F_{i-1}(z, y)$$

$$G_{i,j}(x, y, z_1, \dots, z_j) = G_{i,j+1}(x, y, z_1, \dots, z_j, 0) \\ + G_{i,j+1}(x, y, z_1, \dots, z_j, 1).$$

The sequence

$$G_{0,0}, G_{1,s}, G_{1,s-1}, \dots, G_{1,0}, G_{2,s}, \dots, G_{2,0}, \dots, G_{s,0}$$

Over every large field is good.

$$\text{degree} \leq 4 \cdot c \cdot s$$

$$\text{length} \leq O(s^2)$$

$$\text{width} = 2$$

$$\text{time} = O(s)$$

$$\# \text{ variables} \leq 3s$$



(ZERO) KNOWLEDGE

- Classical theory of Information [Shannon]:
 - if I send you the outcome of n unbiased coin tosses, that gives you n bits of information.
- If goal of a website is to spread information, then would have websites filled with coin tosses....
- What do intelligent entities "trade" when they exchange bits?

Claim: Want "knowledge", not "information".

- Claim: Sequence of n random coin tosses has 0 bits of knowledge; (vs. n bits of information).

- (More interesting) Claim: If I take primes P, Q (n bits each) & send you $N (= P \cdot Q)$ then you don't know P (or Q).

- Anecdotal Story^{*}: Micali posed variant of above as question in problem set in U. Toronto in ~ 82 ; Cook responded "I don't know how to prove I don't know."

^{*} Anonymous source; unreliable

- Formal theory of knowledge emerged.] Goldwasser
Micali
Rackoff

Definition by Example:

- ZK proof of Graph Isomorphism

[Goldreich
Micali
Wigderson]

Given: G_1, G_2

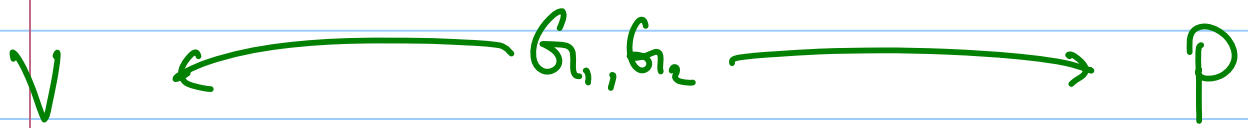
Goal: Prover \leftrightarrow Verifier

• Completeness: if $G_1 \approx G_2$ then V must accept w.h.p.

• Soundness: if $G_1 \not\approx G_2$ then V must reject w.h.p.

• Zero Knowledge: if $G_1 \approx G_2$ then V must not know isomorphism; or learn anything other than this fact from conv.

Protocol: Now P Randomized!

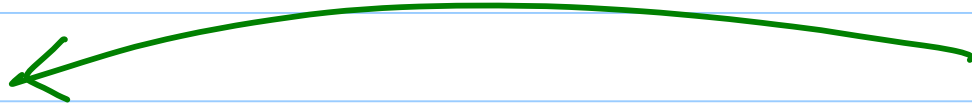


[say $G_1 = \pi_0(G_2)$]

Pick $i \in \{1, 2\}$

$\pi \in S_n$

$$H = \pi(G_i)$$



$b \in \{1, 2\}$



• π if $b=i$



• $\pi \circ \pi_0$ if $i=1$
 $b=2$

• $\pi \circ \pi_0^{-1}$ if $i=2$
 $b=1$

Claim: Sound : Know what this means

Claim: Zero-Knowledge : Don't know yet!

Definition of Zero-Knowledge

- ① Fix verifier's coins R
- ② Transcript is still random variable with distribution D_R

if Verifier can sample from D_R on its own, then Verifier gains no knowledge from prover.

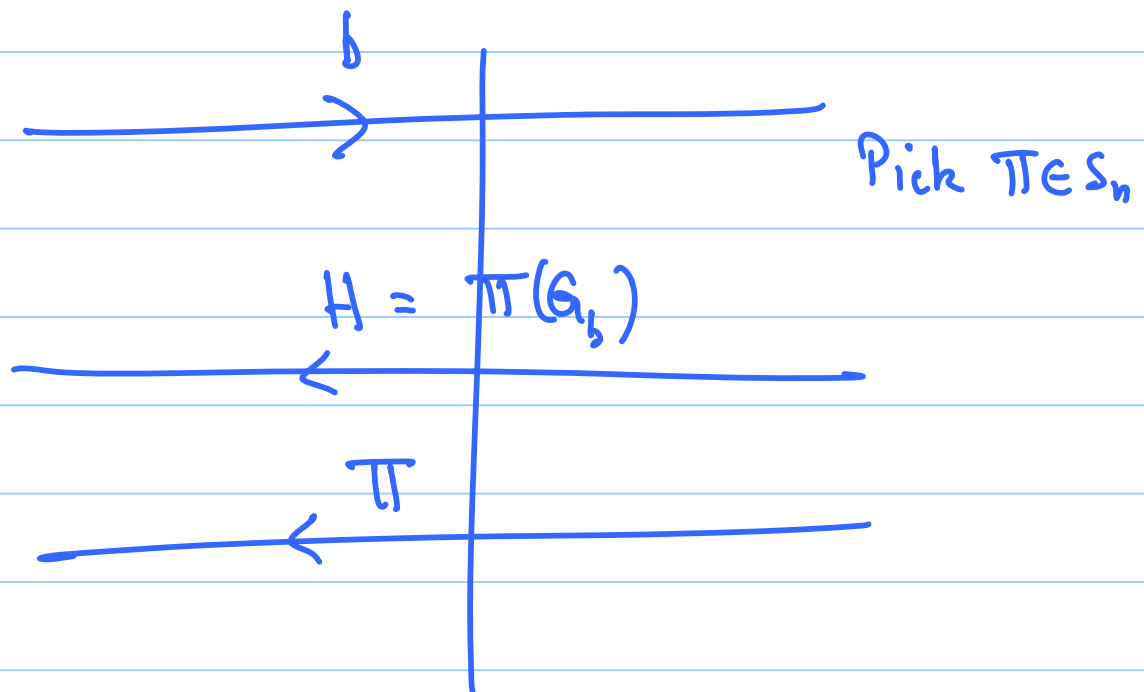
"Simulator" \equiv Sampler of D_R

[Perfect Zero Knowledge]

Simulator for GI

Verifier

Simulator



Output (H, b, π)

exactly same distribution as with prover!

(for every $b \in \{0, 1\}$.)

Sequence important for soundness!

Complexity - Theory of Knowledge

1. Can weaken definitions of Zero-Knowledge.

(i) Simulator produces

$$D'_R \approx_{\epsilon} D_R$$

$$\text{ie, } \sum_x |D'(x) - D(x)| \leq 2\epsilon$$

"Statistical ZK" (SZK)

Equivalently \forall tests $T: \{0,1\}^n \rightarrow \{0,1\}$

$$\left| \Pr_{x \in D_R} [T(x)=1] - \Pr_{x \in D'_R} [T(x)=1] \right| \leq \epsilon$$

(ii) Simulator produces D_R'' s.t.

\forall polytime alg. A

$$\left| \Pr_{x \in D_R} [A(x) = 1] - \Pr_{x \in D_R''} [A(x) = 1] \right| \leq \epsilon.$$

"Computational ZK" (CZK)

Results

• [GMR]: Definitions + protocols for NP-comp. problems.

• [GMW]: GI \in PZK \subseteq SZK \leftarrow statistical

IP \subseteq CZK if o.w.f. exist.

\uparrow

cryptographic

- [Fortnow, Sipser, Hastad]:

$$SZK \in co-AM$$

$\Rightarrow \exists \in SZK$ can't be NP-hard
unless PH collapses

- [Okamoto]:

$$SZK = co-SZK$$

- [Sahai-Vadhan]; [Goldreich-S-V] etc:

SZK complete problems.

E.g. $SD = \left\{ (C_1, C_2) \mid \begin{array}{l} C_1, C_2 : \{0,1\}^n \rightarrow \{0,1\}^m \\ \text{circuits poly size} \\ \{C_1(x)\}_{x \in \{0,1\}^n} \neq \{C_2(x)\}_{x \in \{0,1\}^n} \end{array} \right\}$

Easy: $SD \subseteq SZK$

Harder: $SD \equiv \overline{SD}$.

$(C_1, C_2) \longrightarrow (D_1, D_2)$

dit. $C_1 \approx_{\epsilon} C_2 \Leftrightarrow D_1 \not\approx_{\epsilon} D_2 !)$

Recently rich theory of CZK

$L_1 \in CZK \quad \& \quad L_2 \in CZK$

$\Rightarrow L_1 \cup L_2 \in CZK. \quad [Vadhan].$