Lecture 25:

Interactive Proofs
Complexity Theory

Efficiently Solvable

$P$  $\text{BPP (Randomized P)}$

Efficiently Verifiable

$\text{NP}$  $\text{?}$

Can randomness change what is (and how to) efficiently verify?
Today: The effect of Randomness on how we can efficiently verify proofs

- Interactive proofs
- Zero Knowledge Interactive Proofs
- Probabilistically Checkable Proofs
Classical Proofs

Theorem:
Axiom 1
Axiom 2
Axiom 1 ⇒ A
A ⇒ B
QED

Prime-Number Thm
Efficiently Verifiable Proofs

Theorem

Prover

proof

Verifier

accept/Reject proposition

Hard

Polynomial Time
NP: Efficiently Verifiable Proofs for decision problems

Prover

Verifier Algorithm

input x

proof y

accept/ Reject x

Hard Working

Polynomial Time

NP = problems for which there is a short and easy to verify proofs of the input being a yes-instances [Cook-Levin71]
Example: Graph $G$ is 3-colorable

Prover

Verifier

input: Graph $G = (V, E)$

Prover

Verifier

3 coloring of the graph

Polynomial Time in $|E| + |V|$

After interaction, Verifier knows:

1) $G$ is 3-colorable

2) Also a 3-coloring of $G$
Example: n is a product of 2 primes

Input: n

After interaction, Verifier knows:
1) n is product of 2 primes
2) Also the factors of n
Example: $G_1$ is isomorphic to $G_2$

Hard Problem: best known $\approx$ exponential time in description
Example: $G_1$ is isomorphic to $G_2$

After interaction, Verifier knows:

1) The Graphs are isomorphic
2) The correspondence $\phi$

If $\phi$ is legal, accept
Else reject
Is there any other way?
I will not give you the proof but I will prove to you that I could if I felt like it.

Randomness

Interaction
Interactive Proofs

- Two new ingredients:
  - randomness: verifier tosses coins, can err with some small probability
  - interaction: rather than “reading” proof, verifier interacts with the prover
I will produce a random graph $H$ s.t.

(1) I could give you a mapping $\gamma$ from $G_1$ to $H$

And

(2): I could give you a mapping $\eta$ from $H$ to $G_2$

Proving that $\exists$ mapping $\phi = \gamma \circ \eta$ from $G_1$ to $G_2$.

But, I will only do (1) OR (2)

You choose which!

Claim: If $G_1$ and $G_2$ are not isomorphic, I can at most
Answer (1) or (2) but not both
REPEAT K INDEPENDENT TIMES.

Permute the Vertices of G1 at random, call result H

Graph H

Toss coin b

b

If b=0: mapping from G1 to H
If b=1: mapping from H to G2

Claims:
(1) Statement true can answer correctly for b= 0 and 1
(2) Statement false prob\(_b\)'s(catch a mistake) = 1-1/2\(^k\).
Interactive Proofs

(P, V) is an interactive proof for L: if

Completeness: if \( L(x) = \text{yes} \), \( \text{Prob}[(P, V)(x) = \text{accept}] = 1 \)

Soundness: if \( L(x) = \text{no} \), then for all \( P' \),

If \( k \) times,

\( \text{Prob}[(P', V)(x) = \text{accept}] < 1/2 \)

probability \( \leq 2^{-k} \)
**Interactive Proofs**

(P, V) is an interactive proof for \( L \): if

**Completeness:** if \( L(x) = \text{yes} \), \( \text{Prob}[(P, V)(x) = \text{accept}] = 1 \)

This is what a proof ultimately is!

**Soundness:** if \( L(x) = \text{no} \), then for all \( P' \), \( \text{Prob}[(P', V)(x) = \text{accept}] < \frac{1}{2} \)

If \( k \) times, probability \( \leq 2^{-k} \)
The class IP

**Prover P**

<table>
<thead>
<tr>
<th>Input: x</th>
</tr>
</thead>
<tbody>
<tr>
<td>q1</td>
</tr>
<tr>
<td>a1</td>
</tr>
<tr>
<td>q2</td>
</tr>
</tbody>
</table>

**Verifier V**

- **Probabilistic Polynomial Time Algorithm in |x|**
- **V accepts/rejects x**

**IP:** decision problems L for which there exists interactive proofs
Probabilistically Verifiable
Proofs

Efficiently (Randomized P)

Efficiently Verifiable
• REMARK 1:
  - If the verifier tosses no coins, \( \text{IP}=\text{NP} \)
  - If the prover runs in polynomial time, then \( \text{IP}=\text{Probabilistic Polynomial time} \)

• REMARK 2
  - Probabilities are with respect to the coins the verifier uses.
Why did we disturb the classical notion of proof?

• Preventing Identity Theft

• Secure Protocols

• Can prove statements not verifiable efficiently with classical NP proofs
Zero Knowledge Interactive Proofs

Alternative but equivalent definition

P gives Zero-Knowledge to V: when T is true, whatever V can compute after an interaction, can be computed by a probabilistic polynomial time machine on input T alone.
What Made/Makes ZK possible?

The statement to be proven has many possible proofs of which the prover chooses one \textit{at random}.

- Each such proof is made up of exactly 2 parts: seeing either part on its own gives the verifier \textit{no knowledge}; seeing both parts imply 100\% correctness.

- Verifier chooses \textit{at random} which of the two parts of the proof he wants the prover to give him. The ability of the prover to provide either part, convinces the verifier
Polyomial Time Prover + Certificate

Prover P
Probabilistic
Polynomial Time + secret

Verifier V
Accepts / Re Rejects x

In our example prover only needed to know a correspondence (isomorphism) from G1 to G2
## Classical Identification - Password

### Public password file

<table>
<thead>
<tr>
<th>Name</th>
<th>$f(\text{pswd})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>shafi</td>
<td>$P_{\text{Shafi}}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Alice</td>
<td>$P_{\text{Alice}} = f(\text{summer})$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Bob</td>
<td>$P_{\text{Bob}}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Computer 1 checks if $f(\text{pswd}) = P_{\text{Alice}}$

2 erases password from screen.
Problems with Classical Method

For Settings:
• Alice = Smart Card.
• Over the Net

Passwords are no good
Zero Knowledge: Preventing Identity Theft

To identify itself prover proves that he knows a proof of the theorem.
You can show Zero Knowledge for all of NP

Theorem [GMW86]: If one-way functions exist, then every $L$ in $NP$ has computational zero knowledge interactive proofs
How can you prove something so general?

Idea: Suffices to show a zero knowledge interactive proof for an NP complete language 3COLOR

Completeness: For any other L in NP, \( L \prec 3\text{COLOR} \)

Any instance \( x \) of \( L \) can be reduced to graph \( G \):

- \( L(x) = \text{yes} \) \( \implies \) \( G \) is 3 colorable
- \( L(x) = \text{no} \) \( \implies \) \( G \) is not 3 colorable

Show a Zero-knowledge Proof for 3-coloring
Can you show Zero Knowledge for all of NP

Theorem[GMW86]: If one-way permutations exist, then every L in NP has computational zero knowledge interactive proofs

Building Block: One Way Permutations imply Commitments schemes
Commitment Scheme

• An efficient two-stage protocol between a sender $S$ and receiver $R$ on input $(1^k)$ s.t.:

• In the outset of the **commit stage**, $S$ has private input $b \in \{0, 1\}$; At the end of the **commit stage** both parties hold a joint output $c$, called the **commitment**, and $S$ holds a private output $d$ called the **de-commitment**.

• In the **reveal stage**, $S$ sends the pair $(d, b)$ to $R$ which in turn either accepts or rejects.
Properties of a Commitment Scheme

Completeness: R always accepts in an honest execution.

Hiding: In commit stage: \( \forall R^*, b \neq b' \in \{0,1\}, \) 
\( \{\text{View} (S(b),R^*)(1^k)\} \approx_c \{\text{View}(S(b'),R^*)(1^k)\}. \)

Binding: Any \( A^* \) will fail to output two pairs \((d,b)\) and \((d',b')\) in reveal stage after interacting with \( R \) in the commit stage resulting in a commitment \( c \) s.t. \( R(c, d, b) = R(c, d', b') = \text{Accept} \)

Ex: \( c \in \text{Enc}(r,b) \) for semantically secure \( \text{Enc} \) is a commitment to \( b \) (with d-commitment \( r \))
ZK interactive proof for $G3\text{COL}$

On common input graph $G = (V, E)$ and private prover input coloring $\pi: V \rightarrow \{0, 1, 2\}$

- $P \rightarrow V$: Pick a random permutation $\sigma$ of the coloring $\{0, 1, 2\}$ & color the graph with coloring $\phi(\sigma) = \sigma(\pi(v))$. Send commitments $Enc(r_v, \phi(v)) \forall$ vertex $v$.
- $V \rightarrow P$: Select a random edge $(a, b)$ and send it.
- $P \rightarrow V$: de-commit colors of $a$ and $b$ committed in $Enc(r_a, \phi(a))$ and $Enc(r_b, \phi(b))$ by releasing $r_a$ and $r_b$.
- If $\phi(a) \neq \phi(b)$ $V$ rejects, otherwise repeat and $V$ accepts after $k$ iterations.
Completeness and Soundness

• **Completeness**: if prover used a proper 3-coloring, the verifier will accept.

• **Soundness**: Let \( k = |E|^2 \).
  If \( G \) is not 3-colorable, then for all \( P^* \)
  \[
  \Pr[(P^*, V)(G) \text{ accepts}] < (1 - 1/|E|)^k < \frac{1}{e^{|E|}}
  \]
Many Uses of Zero Knowledge

Lots of Applications to cryptography..
Due to generality

Zero Knowledge and Nuclear Disarmament
[BarakGlasserGoldstone11]
Catalyst

Decoupled “Correctness” from “Knowledge of the proof”

Ask new questions about nature of proof

Questions have been asked and answered in last 25+ years leading up to current research on cloud computing
Interactive Proofs are More Powerful than classical proofs
How to prove colors are different to a blind verifier

Theorem: There are 2 colors on this page

Toss coin to decide if to flip page over or not
Heads flip
Tails don't

Sends resulting page

Guesses coin

If coin ≠ coin, reject
Recall, some natural problems are not known to be NP

- Can you verify quickly that a 3CNF formula is NOT satisfiable? What would be a certificate?

- Can you verify quickly that a pair of graphs are NOT isomorphic? What would be a certificate?

- Can we show that these are in IP?

- E.g. Is there an interactive proof that two graphs are NOT isomorphic?
Example: $G_1$ is NOT isomorphic to $G_2$

Shortest classical proof:
$\approx$ exponential $n$!
But can convince with an efficient interactive proof
Graph Non Isomorphism in IP

input: \((G_0, G_1)\)

Verifier
- flip coin \(c \in \{0,1\}\); pick random \(\gamma\)
- Output YES iff \(b = c\)

Prover
- if \(H\) isomorphic to \(G_0\) then \(b = 0\), else \(b = 1\)

\[ H = \gamma(G_c) \]
GNI in IP

- **completeness:**
  - if $G_0$ not isomorphic to $G_1$ then $H$ is isomorphic to exactly one of $(G_0, G_1)$
  - prover will choose correct $b$

- **soundness:**
  - if $G_0$ is isomorphic to $G_1$ then prover sees same distribution on $H$ for $c = 0$, $c = 1$
  - no information on $c$ $\Rightarrow$ prover $P^*$ can succeed and find $b=x$ with probability at most $1/2$
Classically: Can Efficiently Verify

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP</td>
<td>Can efficiently verify</td>
<td>$\exists$ solution</td>
</tr>
<tr>
<td>Co-NP</td>
<td></td>
<td>0 solutions</td>
</tr>
<tr>
<td>#P</td>
<td></td>
<td>$2^{100} - 13$ solutions</td>
</tr>
<tr>
<td>PSPACE</td>
<td></td>
<td>$\forall \forall ... \exists$</td>
</tr>
</tbody>
</table>

Can you prove more via interactive proofs?
Interactively Provable = IP

- NP
- Co-NP
- \#P
- PSPACE = IP