Last time: Randomized Complexity Classes

RP: one-sided error, randomized polytime
BPP: two-sided error, randomized polytime

Also,
\[ \text{ZPP} = \text{RP} \cap \text{coRP} : \text{zero error, random polytime} \]

- either outputs the correct answer, or "don't know",
- \( \Pr \left[ \text{"don't know"} \right] < \frac{1}{3} \).

Example of a co-RP algorithm:
Polynomial Identity Testing

Input: An algorithm \( A \) computing a multi-variate polynomial \( p(x_1, \ldots, x_n) \) of total degree \( \leq d \).

Decide: Is \( p(x_1, \ldots, x_n) \equiv 0 \)?

\[ \exists \xi: \prod_{i > j} (x_i - x_j) \text{ is an algorithm} \]
\[ i,j \in \{1, \ldots, n\} \] for a polynomial \( p(x_1, \ldots, x_n) \) of total degree \( \leq n \).

The degree of a polynomial:
\[
p(x, y, z) = 10 \cdot xy^2z^3 - 7 \cdot y^5 + 13 \cdot x^5y^2
\]

\[
\text{deg } 5 \quad \text{deg } 5 \quad \text{deg } 19
\]

\[
\text{total deg} = \max \text{ deg over all monomials} = 19
\]

**Schwartz-Zippel Lemma:**

\[
\forall \ p(x_1, \ldots, x_n) \neq 0 \text{ of total degree } \leq d,
\forall \text{ finite set } S \subseteq \mathbb{Z},
\]

\[
\Pr_{r_1, r_2, \ldots, r_n \in S} \left[ p(r_1, r_2, \ldots, r_n) = 0 \right] \leq \frac{d}{|S|}.
\]

**Proof (by induction on } n):**

**Base case:** \( n = 1 \). \( p(x) \neq 0 \) of deg \( \leq d \) can have \( \leq d \) roots.
can have \(d\) root's.

So, \(\Pr_{r \in \mathcal{S}} [ p(r) = 0 ] \leq \frac{d}{151} \).

**Induction Step:** Assume the statement is true for \(\leq n\) variables. Prove for \(n+1\) variables.

Idea: \(p(x_1, \ldots, x_n, x_{n+1}) = q_0(x_1, \ldots, x_n) + q_1(x_1, \ldots, x_n) \cdot x_{n+1}^2 + \ldots + q_k(x_1, \ldots, x_n) \cdot x_{n+1}^k\)

a polynomial in \(x_{n+1}\), whose coefficients are polynomials \(q_0, q_1, \ldots, q_k\) in \(x_1, \ldots, x_n\).

**Ex:** \(p(x_1, x_2, x_3) = 3 \cdot x_1 x_2^2 + 7 \cdot x_2 x_3^2 - 2 \cdot x_3^7\)

\(= (3 \cdot x_1 x_2^2) \cdot x_3^0 + 0 \cdot x_3^1 + (7 \cdot x_2 x_3^2) \cdot x_3^2\)

\(= 0 \cdot x_3^3 + 0 \cdot x_3^4 + 0 \cdot x_3^5 + 0 \cdot x_3^6 + \frac{(2)}{2^7} x_3^7\)

\(p(x_1, \ldots, x_n, x_{n+1}) = \sum_{i=1}^{K} q_i(x_1, \ldots, x_n) \cdot x_{n+1}^i\)

\(\Pr_{r \in \mathcal{S}} [ p(r_1, \ldots, r_n, r_{n+1}) = 0 ] = \)
\[ Pr \left[ p(r_1, \ldots, r_{n+1}) = 0 \mid q_K(r_1, \ldots, r_n) = 0 \right] \cdot Pr \left[ q_K(r_1, r_n) = 0 \right] + \]
\[ Pr \left[ p(r_1, \ldots, r_{n+1}) = 0 \mid q_K(r_1, \ldots, r_n) \neq 0 \right] \cdot Pr \left[ q_K(r_1, r_n) \neq 0 \right] \leq Pr \left[ q_K(r_1, \ldots, r_n) = 0 \right] + \]
\[ \frac{\text{deg}(q_K)}{|S|} \cdot \frac{K}{|S|} = \frac{\text{deg}(q_K)+K}{|S|} \leq \frac{d}{|S|}. \]

QED

Random algo for PIT

Given a for \( p(x_1, \ldots, x_n) \) \( \deg \leq d \).

Define \( S = \{1, 2, 3, 4, \ldots, 10d\} \)
\[ \text{Pick } r_1, \ldots, r_n \in S \text{ at random.} \]

Output "Zero" if \( \# A(r_1, \ldots, r_n) = 0. \]

Open Problem: Is there a deterministic polytime (or even sub-exponential-time) algo. for Poly Identity Testing?

Then [K., Impagliazzo]: If Poly Id. Test. is in \( P \), then we get STRONG circuit lower bounds (\( \sim NP \neq P \)).

Message: Derandomizing BPP algorithms is impossible without \( \sim \) at the same time, proving new, strong complexity lower bounds.
Randomness + Interaction: Interactive Proofs (IP)

NP:

Prover \[ \rightarrow \] y \[ \rightarrow \] Verifier

\[ \check{y} \text{ is a valid witness for } x \]

Generalize in two ways:
(1) allow Verifier to be a randomized polytime algorithm.
(2) allow many (polynomial number) of rounds of communication.

IP:

\[ \rightarrow \] X \[ \rightarrow \]
Example: Graph Non-Isomorphism

Given: \( G_0 = (V_0, E_0) \)
\( G_1 = (V_1, E_1) \)

Decide: \( G_0 \not\cong G_1 \) ?

\[ \begin{align*}
\begin{array}{c}
1 \quad 2 \\
\text{X} \\
3
\end{array}
\quad \text{are isomorphic}
\quad \begin{array}{c}
1 \quad 2 \\
\text{square}
\end{array}
\]

\[ \begin{align*}
\begin{array}{c}
1 \quad 2 \\
\text{are non-isomorphic}
\end{array}
\quad \begin{array}{c}
1 \quad 2 \\
\text{square}
\end{array}
\]
are non-isomorphic.

If protocol for Graph Non Isomorphism

\((G_0, G_1)\) both graph on \(n\) nodes \((1, 2, \ldots, n)\)

\(P\) \quad \rightarrow \quad \mathcal{V} \quad \text{random} \quad i \in \{0, 1\} \quad \text{random permutation} \quad \pi : [n] \to [n]

\(\pi(G_i)\)

\(j \in \{0, 1\}\) \quad \rightarrow \quad \text{accept if} \quad j = i

Correctness Analysis:

(1) \(G_0 \neq G_1 \Rightarrow\)

the prover can determine which of \(G_0\) or \(G_1\) is isomorphic to \(\pi(G_i)\)

& so can send \(i = i\).
& so can send \( j = 1 \).

\[(2) \quad G_0 \preceq G_1 \implies \]

\[\mathcal{H}(G_0) \text{ is distributed the same as } \mathcal{H}(G_1)\]

So, Prover has no way to pick the correct \( j = i \), except by randomly guessing \( j \in \{0, 1\} \). Hence, \( \Pr \left[ j = i \right] \leq \frac{1}{2} \).