Topics

- 2-Point Sampling
- Interactive Proofs
  - Public coins vs Private coins

1 Two Point Sampling

1.1 Error Reduction

Let’s say we are given a language \( L \) and an algorithm \( A \) in RP which uses random bits \( r \in \{0,1\}^R \)

- \( x \in L \implies Pr[A(x, r) = 1] \geq \frac{1}{2} \)
- \( x \notin L \implies Pr[A(x, r) = 1] = 0 \)

How do we reduce error? Repeat \( A \) with \( k \) different values of \( r \) − \{\( r_1, \ldots, r_k \}\).

Let \( a_i = A(x, r_i) - i \in \{1, \ldots, k\} \) and \( r' = \{r_1, \ldots, r_k\} \).

Define \( A'(x, r') = \bigwedge_{i=1}^{k} a_i \).

Claim 1 Given \( r \in \{0,1\}^{kR} \), error probability is reduced to \( \frac{1}{2^k} \) – i.e.

- \( x \in L \implies Pr[A'(x, r) = 1] \geq 1 - \frac{1}{2^k} \)
- \( x \notin L \implies Pr[A'(x, r) = 1] = 0 \)

Proof

If \( x \notin L \), \( A'(x, r) = \bigwedge_{i=1}^{k} A(x, r_i) = \bigwedge_{i=1}^{k} 0 = 0 \)

If \( x \in L \), \( A'(x, r) = \bigwedge_{i=1}^{k} A(x, r_i) \implies Pr[A'(x, r) = 0] \leq \frac{1}{2^k} \implies Pr[A'(x, r) = 1] \geq 1 - \frac{1}{2^k} \)

\( A' \) uses \( k \cdot R \) random bits. Can we do better?

1.2 Using Pairwise Independence to Reduce Randomness

Definition 2 A family of hash functions \( H = \{h : A \rightarrow B\} \) is pairwise independent if –

\[ Pr[h(a_1) = b_1 \land h(a_2) = b_2] = \frac{1}{|B|^2} \] (1)

Consider the family of pairwise independent hash functions \( H : \{0,1\}^{k+2} \rightarrow \{0,1\}^{kR} \).

Let \( h \in R H \) – sampling \( h \) requires \( O(kR) \) random bits.
Algorithm

- Pick $h \in R \mathcal{H}$
- for $i = 1\ldots 2^{k+2}$
  - $r_i = h(i)$
  - if $A(x, r_i) = 1$ – Output 1 (Accept)
- Output 0 (Reject)

If $x \not\in L - A(x, r_i) = 0$ for all random strings $r_i$. So, the algorithm outputs ”Reject”.

If $x \not\in L$, Define –

$$c(r_i) = \begin{cases} 
0, & \text{if } A(x, r_i) = 0. \\
1, & \text{otherwise.}
\end{cases} \quad (2)$$

$$E[c(r_i)] = Pr[c(r_i) = 1] > \frac{1}{2}$$

Let $Y = \sum_{i=1}^{q=2^{k+2}} c(r_i) \implies E[Y] = \frac{E[Y]}{q} > \frac{1}{2}$

**Chebyshev’s Inequality** – If $X$ is a random variable and $E[X] = \mu$ then $Pr[|X - \mu| \geq \epsilon] \leq \frac{\text{Var}[X]}{\epsilon^2}$

**Lemma 3** If $X_1, X_2, \ldots, X_n$ are pairwise independent random variables, $\text{Var}[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} \text{Var}[X_i]$.

**Proof**

$$\text{Var}[\sum_{i=1}^{n} X_i] = E[(\sum_{i=1}^{n} X_i)^2] - E[(\sum_{i=1}^{n} X_i)]^2 \quad (3)$$

$$= E[\sum_{i,j} X_i X_j] - \sum_{i=1}^{n} E[X_i]^2 \quad (4)$$

$$= \sum_{i,j} E[X_i X_j] - \sum_{i,j} E[X_i]E[X_j] \quad (5)$$

$$= \sum_{i} (E[X_i^2] - E[X_i]^2) - \sum_{i \neq j} (E[X_i X_j] - E[X_i]E[X_j]) \quad (6)$$

$$= \sum_{i} \text{Var}[X_i] - 0 = \sum_{i} \text{Var}[X_i] \quad (7)$$

Since pairwise independence $\implies E[X_i X_j] = E[X_i]E[X_j] \ \forall i \neq j$. 

So, if $X = \sum X_i$ and $\mu = E[X]$, then $Pr[|X - \mu| > \epsilon] = \frac{\text{Var}[\sum_{i=1}^{n} X_i]}{\epsilon^2} = \frac{\sum_{i=1}^{n} \text{Var}[X_i]}{\epsilon^2} = \frac{\text{Var}[X]}{\epsilon^2}$
Pairwise Independent Tail Inequality
If $X$ is a random variable and $E[X] = \mu$, $Pr[|X - \mu| \geq \epsilon] \leq \frac{\text{var}[X]}{\epsilon^2}$

So, $Pr[Y/q = 0] \leq Pr[|Y/q - E[Y/q]| \geq E[Y/q]] \leq \frac{1}{q} \cdot \frac{E[Y]^2}{q} < \frac{4}{q} = \frac{1}{2^k}$.

Remark Using this algorithm reduces the randomness complexity but greatly increases the running time of the algorithm.
The running time is now $O(2^{k+2} \cdot T_A(n))$ rather than $O(k \cdot T_A(n))$.

2 Interactive Proofs – Generalization of NP

2.1 NP vs IP

Definition 4 NP is the class of all languages $L$ for which an "yes" ($x \in L$) answer is verifiable in polynomial time by a deterministic Turing Machine.

Definition 5 Consider a model with a Prover – $P$ and a Verifier – $V$.
- $V$ is bounded in polynomial time and can toss coins (non-deterministic).
- $P$ has unbounded time and is deterministic. (No point being randomized since time is unbounded)
- $V$ and $P$ can send information to each other through conversation tapes.
- $V$’s random bits are private – $P$ doesn’t know what they are.

An Interactive Proof System for a language $L$ is a protocol such that given input $x$, $P$ tries to convince $V$ that $x \in L$ and at the end $V$ either "accepts" or "rejects" the proof. It must satisfy the following conditions –
1. If $x \in L$ and $V$ and $P$ follow the protocol,
   - $Pr_{\text{coins}_V}[V \text{ accepts}] \geq \frac{2}{3}$
2. If $x \notin L$ and $V$ follows the protocol, no matter what $P$ does,
   - $Pr_{\text{coins}_V}[V \text{ rejects}] \geq \frac{2}{3}$

Definition 6 IP is the class of languages $L$ such that there exists an Interactive Proof System for $L$.

Known – $NP \subset IP$ and $IP = PSPACE$
2.2 Graph Isomorphism and Graph Non-isomorphism

2.2.1 Graph Isomorphism

**Input** – Graphs $G$ and $H$.

$G \cong H \iff (\exists \psi \in S_{|V_G|} \text{ s.t. } (u, v) \in E_G \iff (\psi(u), \psi(v)) \in E_G))$

**Output** – 1 if $G \cong H$, 0 else.

Graph Isomorphism is in NP – since $G \cong H$ can be proven by providing $\psi$. Can be verified in polynomial time. So, Graph Isomorphism is in IP.

2.2.2 Graph Nonisomorphism

**Input** – Graphs $G$ and $H$.

**Output** – 1 if $G \not\cong H$, 0 else.

**Protocol**

- Repeat $k$ times –

  1. $V$ computes $G'$ and $H'$ which are random permutations of $G$ and $H$.
  2. $V$ flips a coin and with equal probability –

     - Heads : Sends $(G, G')$ to $P$
     - Tails : Sends $(G, H')$ to $P$

     - $P$ replies indicating whether the pair of graphs it received were isomorphic or not.

     - If ($V$ sends $(G, G')$ and $P$ sends $\cong$) or if ($V$ sends $(G, H')$ and $P$ sends $\not\cong$) – Continue.
     - If ($V$ sends $(G, G')$ and $P$ sends $\not\cong$) or if ($V$ sends $(G, H')$ and $P$ sends $\cong$) – Reject.

- Accept.

If $x \in L \Rightarrow G \not\cong H$, then $P$ will follow protocol and always answer correctly and $V$ will continue till the loop ends and then Accept.

If $x \not\in L \Rightarrow G \cong H$, then $(G, G')$ and $(G, H')$ are indistinguishable by $P$. So, $P$ will return a value that causes Reject with probability $\frac{1}{2}$ at every iteration.

Hence, $Pr[V \text{ accepts}] = \frac{1}{2} \Rightarrow Pr[V \text{ rejects}] = 1 - \frac{1}{2}$

So, $Pr[V \text{ accepts}] = \begin{cases} 
1, & \text{if } x \in L \\
2^{-k}, & \text{if } x \not\in L.
\end{cases}$

Hence, Graph Nonisomorphism in in IP.

**Remark** This protocol only works if $V$ has private coins. If $P$ can see $V$’s random bits, $V$ can be made to accept for all inputs.
2.3 Arthur-Merlin Protocol

The Arthutr-Merlin protocol is an interactive proof system where the Verifier’s coins are public.