Randomness in Computation

Friday, May 16, 2025 1:06 PM

¡ Probability !

Main Question: '80 –

How does allowing dice throws affect computations?

- More flexible when errors are allowed.
- Finding hard to construct avg. instances. (probabilistic methods)
- Symmetric breaking

Fingerprinting.

A, B n-bit binary string.

Question: How to check $A = B$ efficiently?

- Pick ℓ locations L, & check $A[i] = B[i] \ \forall i \in L$
- checksum $\sum_i A[i] = \sum_i B[i]$?
- Treat A & B as two numbers.

  Pick a prime p at random

  Is $A \equiv B \pmod{p}$?

- $(A-B) \bmod p = 0$, $\leq n$ prime divisors.

$$A - B = p_1 \cdot p_2 \cdot p_3 \cdot \ldots \cdot p_k \geq 2^k$$

$$k \le \log_2(A-B) = \log_2(2^n) = n.$$

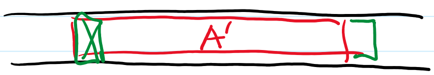
- Choose prime randomly from $\underline{[1:n^3]}$ (#bits $O(\log n)$) $\left(\sim \frac{n^3}{\log n} \text{ primes}\right)$

  Prime number thm: #primes in range $[1:N]$ is $\Theta\left(\frac{N}{\log N}\right)$

error prob. : $\frac{n}{n^3/\log n} \sim \frac{\log n}{n^2}$ with high prob. $\left(\text{error} : \frac{1}{\text{poly } n}\right)$

Rabin-Karp (A, B):

> input: $A[1..n]$ (long), $B[1..l]$ (short).
> output: Is B substring in A?
>
> choose random prime $p \in [1:n^4]$
>
> for i from 0 to $n-l$:
>
>   if $\boxed{A[i+1..i+l]}^{A'} = B \bmod p$:
>
>   ~~return YES~~. $A=B$?
>
> $A' \bmod p \leftarrow (A' \bmod p \ll 1)$
>
>   remove high bit
>
>   add $A[i+l+1]$



error probability :

$$\left(\frac{n}{n^4/\log n} = \frac{\log n}{n^3}\right) \cdot n = \frac{\log n}{n^2}$$

time:

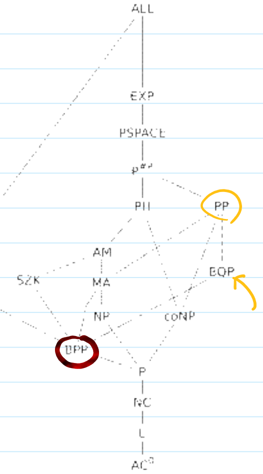
$$O(n) \cdot O(1) + O(m)$$
$$= O(n+m)$$

$$\frac{\log n}{n^2} \cdot O(nm) + \left(1 - \frac{\log n}{n^2}\right) \cdot O(n+m)$$

BPP : problems decided by TM + dice in poly-time

yes inst. $\Rightarrow$ $\Pr \ge 3/4$ $2/3$ $1/2 + \frac{1}{\text{poly } n}$

no inst. $\Rightarrow$ $\Pr \le 1/4$ $1/3$ $1/2 - \frac{1}{\text{poly } n}$



error reduction.

repeat alg., output majority.

Chernoff bound : $\Pr\left[\left|\sum X_i - 3k/4\right| \ge \alpha k\right] \le e^{-\alpha^2 k}$.

Chernoff bound : $\Pr\left[\left|\sum X_i - 3k/4\right| > \alpha \cdot k\right] \leq C^{-\alpha^2 k}$.

<u>Question</u>. Is BPP bigger than P ?

Think about algebraic version of SAT :

$(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2) =: \phi$

$[1-(1-x_1)(1-x_2)(1-x_3)]\ [1-(1-x_1)x_2] =: P(x) = P(x_1, x_2, x_3)$

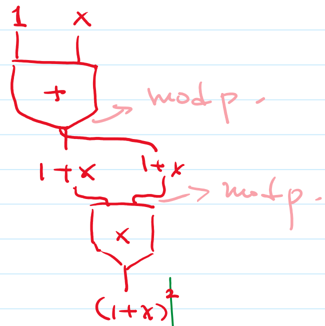
ϕ not sat. $\Leftrightarrow$ $\forall x \in \cancel{\{0,1\}^n}\ \mathbb{R}^n$ s.t. $P(x) = 0$.

(3, 2, 5)

<u>Polynomial Identity testing</u>

input : Polynomial P, n-variable, degree-d → as algebraic circuit: poly-size.

output : Is $P = 0$?



<u>Thm</u>. PIT is in BPP.

DeMillo-Lipton-'78

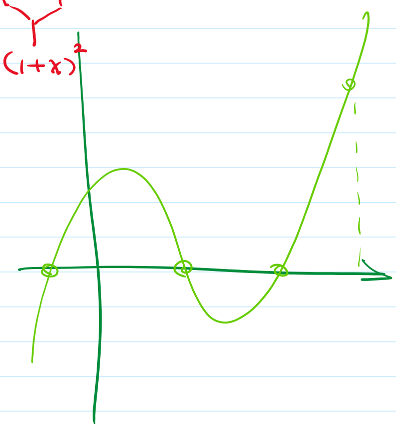
<u>Schwartz-Zippel Lemma</u>. ('80, '79)

Polynomial $P \neq 0$. deg d (5). n var (3).

S any set of integers in $[1 : d \cdot n]$

Choose $a_1, \ldots, a_n$ from S at random.

then $\Pr\left[P[a_1, \ldots, a_n] \neq 0\right] \geq 1 - d/|S|$



<u>PIT(P)</u> :

Let $S := [1 : d \cdot n]$ ← 2^{2n}

Pick random $a \in S$ ← 2^{2^n}

return $[P(a) = 0 ?]$

by SZ lemma. error prob. $\sim d/(d \cdot n) = \frac{1}{n}$

time : Eval $P(a)$ might take exp time!

$P(x) = (1+x)^{2^n}$

2   4   n steps 2^n

$$P(x) = (1+x)^{2^{\cdot\cdot}}$$

$$(1+x) \xrightarrow[\text{mod } p]{} (1+x)^2 \xrightarrow[\text{mod } p]{} (1+x)^4 \xrightarrow[\text{mod } p]{} \overset{n \text{ steps}}{\cdots} \longrightarrow (1+x)^{2^n}$$

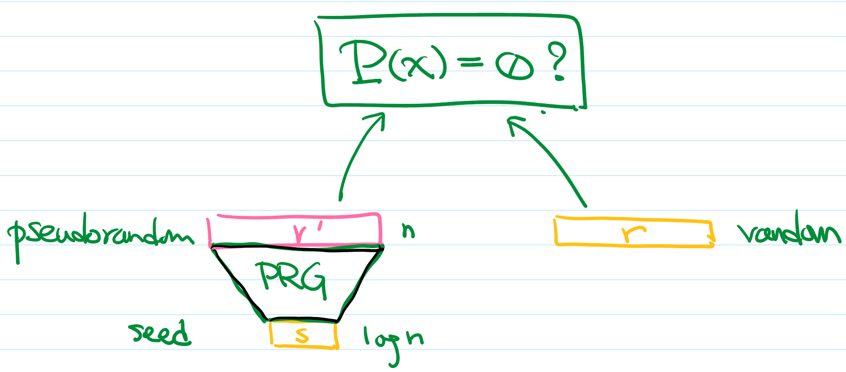
$$P(a) = (1+2^n)^{2^n} \text{ mod } p.$$

Fingerprinting! Pick random prime $p \in [1 : n^4]$

<u>Conclusion.</u> $BPP \neq P$?

No. Strong evidence that $BPP = P$.

<u>Observation.</u> If we can fake randomness, then $BPP = P$!



[IW'97]

- $\exists$ "hard" problem $\Rightarrow BPP = P$.

[IW'97]

- $\exists$ "hard" problem $\Rightarrow$ $BPP = P$.

[KI'04]

- $PIT \in P$ $\Rightarrow$ some problems are hard.

$$\boxed{\text{Hardness} \iff \text{Derandomization}}$$ mid '90 —

<u>Either</u> you believe some problems are hard,

<u>or</u> random. poly-time algorithms really need dices. BUT NOT BOTH!