Computational Complexity

Mohammad Mahmoody

Session 20

April 2014

Complexity Class **BPP**

say that a PTM M decides L in time $T(n)$ if for every $x \in \{0, 1\}^*$, M halts in $T(|x|)$ steps regardless of its random choices, and $\Pr[M(x) = L(x)] \geq 2/3$.

We let **BPTIME**$(T(n))$ be the class of languages decided by PTMs in $O(T(n))$ time and define **BPP** $= \cup_c$ **BPTIME**(n^c).

Definition 7.3 *(**BPP**, alternative definition)* A language L is in **BPP** if there exists a polynomial-time TM M and a polynomial $p : \mathbb{N} \to \mathbb{N}$ such that for every $x \in \{0, 1\}^*$, $\Pr_{r \in_R \{0,1\}^{p(|x|)}}[M(x, r) = L(x)] \geq 2/3$.

Other ways randomness can help

- Find the answer always correctly (zero error) but faster.

input: $(x_1, \dots, x_n)$ all distinct

- Example: finding median.

Sorted: $(x'_1 < x'_2, \dots < x'_n)$ $n = 2k-1$

x'_k : median

Try 1: Sort and pick x'_k : time $\Theta(n \lg)$

Try 2: given $x_1, \dots, x_n$: what is the k^{th} smallest (general k)

pick $i \in \{1, \dots, n\}$ at random. look at x_i $\begin{cases} \text{go over } x_1 - x_n \\ \text{divide them into} \end{cases}$ $S_1 < x_i < S_2$

for every element in them

if $|S_1| = k-1 \longrightarrow$ done

otherwise $\begin{cases} \text{if } |S_1| < k-1 \longrightarrow \text{find}(S_2, k - |S_1| - 1) \\ \text{if } |S_1| > k-1 \longrightarrow \text{find}(S_1, k) \end{cases}$

time $\approx$
$n + \frac{n}{2} + \frac{n}{4} \cdots \approx$
$n(1 + \frac{1}{2} + \frac{1}{4} \cdots) < 2n$

One sided error

$\subseteq BPP$

$P \subseteq$

- **RP** : there is a poly-time $M(\cdot)$ such that

$$x \in L \Rightarrow \Pr[M(x) = 1] \geq \frac{2}{3}$$
$$x \notin L \Rightarrow \Pr[M(x) = 0] = 1$$

- The randomness r such that $M(x, r) = 1$ is a "witness" that $x \in L$

$RP \overset{?}{\neq} NP$

$RP \subseteq NP$

proof: M is verifier

for $x \in L$: r is witness

accepted: if $M(x, r) = 1$

(Impagliazzo-Wigderson: if SAT require 2^{cn}-size circuit for some constant c

$\Longrightarrow P = RP = BPP$

$X \subseteq Y$

$coX \subseteq coY$

$P \subseteq co\text{-}RP \subseteq \begin{matrix} BPP \\ coNP \end{matrix}$

How to decrease the error for **RP**?

- Suppose $\begin{array}{c} x \in L \Rightarrow \Pr[M(x) = 1] \geq 2/3 \\ x \notin L \Rightarrow \Pr[M(x) = 0] = 1 \end{array}$

$\frac{1}{100}$ → err $\left(\frac{99}{100}\right)^k$

Run $M(x)$ twice using independent randomness

M_2

$r_1 \longrightarrow a_1$

$r_2 \longrightarrow a_2$

$0\ 0 \longrightarrow 0$

$0\ 1 \longrightarrow 1$ ✓ correct

$1\ 0 \longrightarrow 1$ ✓

$1\ 1 \longrightarrow 1$ ✓

$x \in L: \Pr[M_2(x) = 1] \overset{?}{=} 1 - \frac{1}{9}$ $\quad \left(\frac{1}{3}\right)^k$

$x \notin L: \Pr[M_2'(x) = 0] = 1$

generalize: run M k times.

M_k: $0\,0\,0\,0\,0 \ldots \longrightarrow 0$

$\left.\begin{array}{l} \ldots 1 \ldots \\ 1 \ldots \end{array}\right\} \longrightarrow 1$

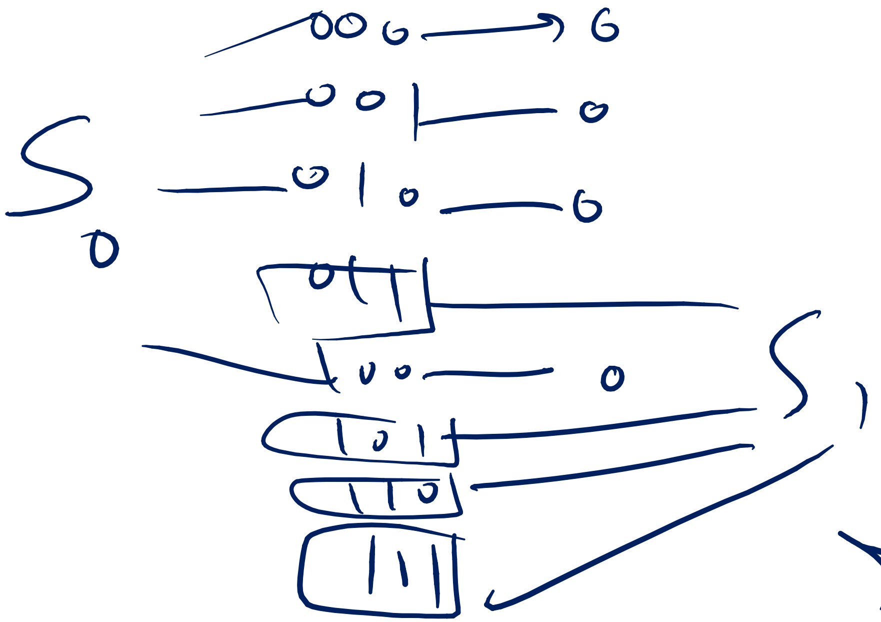
How to decrease the error for **BPP**

- Suppose $x \in L \Rightarrow \Pr[M(x) = 1] \geq 2/3$
  $x \notin L \Rightarrow \Pr[M(x) = 0] \geq 2/3$

$k = 2\alpha + 1$

M_k : runs M k times. output the "majority" of answers

$k = 3$:



$x \in L$: $\Pr[M_3$ gives right answer$] =$

$\cancel{3} \cdot \left(\frac{2}{3}\right)^2 + \sum \left(\frac{2}{3}\right)^3$

$\left(\frac{2}{3}\right)^2 + \left(\frac{2}{3}\right)^3 > \frac{2}{3}$

$\frac{20}{27} > \frac{18}{27}$