Chapter 7: Randomized computation

- A probabilistic TM (PTM) is a TM with an extra read-only tape which contains a string of uniformly random bits
  ($\forall i : Pr[b_i = 0] = Pr[b_i = 1] = 1/2$).
  (Equivalently, at every step picks transition δ_0 or δ_1 with prob. $1/2$).
- A PTM M runs in $T(n)$-time if $\forall x : M(x)$ halts within $T(|x|)$ steps for every random string (still worst-case...).
- If M uses l random bits, then 2^l possible uniform random strings R
  $\Rightarrow Pr_R[M(x) = 1] = \frac{\#\text{ of } R\text{s that make } M(x) = 1}{2^l}$
- PTM M decides L in time $T(n)$ if
  1. $M(x)$ always halts in $T(|x|)$ steps
  2. $Pr[M(x) \text{ correct}] \geq 2/3(?)$.
- $L \in BPTIME(T(n))$ if $\exists$ PTM M that decides L in $O(T(n))$ time.

Definition 1

$BPP = \bigcup_{c \geq 0} BPTIME(n^c)$

Observations about PTMs (randomized algorithms)

- Use of random coins by an algorithm can have two consequences:
  1. Running time $T(|x|)$ is a random variable. Then
     $$\text{worst case expected running time } = \max_{|x|=n}\{E_R[T(|x|)]\}$$
     E.g., QUICKSORT is $O(n \log n)$, MEDIAN (p. 126) is $O(n)$.
  2. $M(x)$ is correct with a certain probability (over random bits R)
- In $BPTIME(T(n))$ definition:
  1. $T(|x|)$ is not expected running time, but time upper-bound of $M(x)$ for all random R. **But** can be made expected (stay tuned).
  2. We require $Pr[M(x) \text{ correct}] \geq 2/3$. Why 2/3? Why not 3/4? Or $1 - 1/n$? **Doesn't matter!** (stay tuned)
- Randomized algorithms $M(x)$ that are always correct (independently of random bits R) if, say $x \notin L$? **Yes!**
  E.g., PRIMALITY (p. 128)

Chapter 7: Randomized computation

Definition 2

$L \in BPTIME(T(n))$ if $\exists$ PTM M running in $O(T(n))$ time, and

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

$BPP = \bigcup_{c \geq 0} BTIME(n^c)$

Definition 3

$L \in RPTIME(T(n))$ if $\exists$ PTM M running in $O(T(n))$ time, and

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

$RP = \bigcup_{c \geq 0} RPTIME(n^c)$

Note: Book typo for the $x \notin L$ case!!!

Chapter 7: Randomized computation

Definition 4

$L \in coRPTIME(T(n))$ if $\exists$ PTM M running in $O(T(n))$ time, and

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

$coRP = \bigcup_{c \geq 0} coRPTIME(n^c)$

Definition 5

$L \in ZTIME(T(n))$ if $\exists$ PTM M running in expected $O(T(n))$ time, and for input x, whenever M halts, then $M(x)$ is correct.

Chapter 7: Randomized computation

Relations between classes

- $P \subseteq BPP \subseteq EXP$ (run PTM for $2^{|R|=p(n)}$ possible random strings)
- $RP \subseteq NP$, $coRP \subseteq coNP$ (certificate=random string R that makes $M(x) = 1$)
- $RP, coRP \subseteq BPP$ (obvious)

Theorem 6

$ZPP = RP \cap coRP$

Proof:

$L \in RP \cap coRP \Rightarrow \exists M_1 \in RP, M_2 \in coRP$ running in $p_1(n)$, $p_2(n)$
$\Rightarrow$ run $M_1(x)$, then $M_2(x)$ in $p(n) = p_1(n) + p_2(n)$ time
$\Rightarrow$ if $M_1(x) = 1 \wedge M_2(x) = 1$ return 1, if $M_1(x) = 0 \wedge M_2(x) = 0$ return 0, else repeat
$\Rightarrow$ at each repetition $Pr[\text{output } L(x)] \geq 2/3$, $Pr[\text{output } \overline{L(x)}] = 0$, $Pr[repeat] \leq 1/3$
$\Rightarrow E[T(n)] \leq \sum_{i=1}^{\infty} \frac{ip(n)}{3^{i-1}} = O(p(n)) \Rightarrow L \in ZPP$

Chapter 7: Randomized computation

Proof: (cont'd)

$L \in ZPP \Rightarrow \exists M$ running in expected $p(n)$ time

$\Rightarrow Pr_R[|T(x)| \geq 3p(|x|)] \leq \frac{1}{3}$ (Markov's inequality)

$$M_1(x) = \begin{cases} \text{1. Run } M(x) \text{ for } 3p(|x|) \text{ time} \\ \text{2. If halts, output } M(x) \text{ else output } 0 \end{cases}$$

$$M_2(x) = \begin{cases} \text{1. Run } M(x) \text{ for } 3p(|x|) \text{ time} \\ \text{2. If halts, output } M(x) \text{ else output } 1 \end{cases}$$

$\Rightarrow L \in RP$ because of M_1 and $L \in coRP$ because of M_2

□

Note: $M(x)$ for $L \in ZPP$ may not even halt for some random string(s)!

Relations between classes

- $P \subseteq BPP \subseteq EXP$ (run PTM for $2^{|R|=p(n)}$ possible random strings)
- $RP \subseteq NP$, $coRP \subseteq coNP$ (certificate=random string R that makes $M(x) = 1$)
- $RP, coRP \subseteq BPP$ (obvious)

Theorem 7

$ZPP = RP \cap coRP$

Open problem: $BPP \overset{?}{=} P$, $BPP \overset{?}{\subset} NEXP$

Some basic probabilities

Lemma 8 (Linearity of expectation)

$E[\sum_i X_i] = \sum_i E[X_i]$

Lemma 9

If X_i*'s mutual independent* $E[\Pi_i X_i] = \Pi_i E[X_i]$

Lemma 10 (The probabilistic method)

- *If* $E[X] = \mu$ *then* $Pr[X \geq \mu] > 0$
- *If* $Pr_r[A(r)\ true] > 0$ *then at least one* r_0 *makes* $A(r_0) = true$*.*

Some probability inequalities

Lemma 11 (Markov's)

If $X \geq 0$, *then* $Pr[X \geq kE[X]] \leq \frac{1}{k}$

Lemma 12 (Chebyshev's)

If $Var(X) = \sigma$, *then* $Pr[|X - E[X]| > k\sigma] \leq \frac{1}{k^2}$

Lemma 13 (Chernoff's)

If $X_1, X_2, \ldots, X_n \in \{0, 1\}$ *mutually independent with* $\mu = E[\sum_i X_i]$, *for every* $\delta > 0$

$$Pr[\sum_i X_i \geq (1+\delta)\mu] \leq \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu}$$

$$Pr[\sum_i X_i \leq (1-\delta)\mu] \leq \left[\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\mu}$$

Corollary 1 (Chernoff's)

If $X_1, X_2, \dots, X_n \in \{0,1\}$ *mutually independent with* $\mu = E[\sum_i X_i]$, *for every* $\delta > 0$

$$Pr[|\sum_i X_i - \mu| \geq \delta\mu] \leq 2e^{-\min\{\delta^2/4,\delta/2\}\mu}$$

Lemma 14 (Success boost)

$Pr[M(x)$ *correct*$] \geq \frac{1}{2} + |x|^c$ *can be boosted to*
$Pr[N(x)$ *correct*$] \geq 1 - 2^{-|x|^d}$.

Proof:

$N(x)$ runs $M(x)$ $k := 8|x|^{2c+d}$ times, and output majority result
$\Rightarrow$ Let $X_i = 1$ if $M(x)$ correct the i-th time ($X_i = 0$ o/w)
$\Rightarrow E[X_i] = Pr[X_i = 1] = \frac{1}{2} + |x|^c =: p \Rightarrow E[\sum_i X_i] = pk$
$\Rightarrow$ Chernoff with $\delta := |x|^{-c}/2$: $Pr[\sum_i X_i < \frac{k}{2}] \leq 1 - 2^{-|x|^d}$

□

Chapter 7: Randomized computation

Corollary 2 (Chernoff's)

If $X_1, X_2, \ldots, X_n \in \{0,1\}$ *mutually independent with* $\mu = E[\sum_i X_i]$, *for every* $\delta > 0$

$$Pr[|\sum_i X_i - \mu| \geq \delta\mu] \leq 2e^{-\min\{\delta^2/4, \delta/2\}\mu}$$

Lemma 15 (Success boost)

$Pr[M(x)$ *correct*$] \geq \frac{1}{2} + |x|^c$ *can be boosted to* $Pr[N(x)$ *correct*$] \geq 1 - 2^{-|x|^d}$.

Lemma 16 (Expected vs. absolute time)

In $BPTIME(T(n)), RTIME(T(n))$ *definitions can have expected (instead of absolute) time bound* $T(n)$.

Proof: Run $M(x)$ for $100T(|x|)$ steps. $Pr[M(x)$ no halt$] \leq 1/100$ (Markov) $\Rightarrow Pr[M(x)$ correct$] \geq 2/3 - 1/100$ □

Lemma 17 (Biased coin from unbiased coins)

$\exists$ PTM that can simulate a biased coin with $Pr[Heads] = \rho = [0.\rho_1\rho_2\rho_3\ldots]_2$ in $O(1)$ expected time.

Proof:

PTM uses its unbiased coins $b_1, b_2, \ldots, b_i, \ldots$ as follows: At step i

1. If $b_i < \rho_i$ then output "heads" & halt $(Pr[b_i < \rho_i] = \rho_i)$
2. If $b_i > \rho_i$ then output "tails" & halt
3. If $b_i = \rho_i$ then go to step $i+1$ $(Pr[(3) \text{ happens}] = 1/2)$

$\Rightarrow Pr[\text{reaches } i] = 1/2^i$

$$Pr[\text{heads}] = \sum_i Pr[\text{reaches } i \wedge \text{heads at } i]$$

$$= \sum_i Pr[\text{reaches } i] Pr[\text{heads at } i | \text{reaches } i] = \sum_i \frac{1}{2^i}\rho_i = \rho$$

$E[\text{running time}] = \sum_i i \cdot Pr[\text{reaches } i] = \sum_i i/2^i = O(1)$ □

Lemma 18 (Biased coin from unbiased coins)

$\exists$ PTM that can simulate a biased coin with $Pr[Heads] = \rho = [0.\rho_1\rho_2\rho_3\ldots]_2$ in $O(1)$ expected time.

Lemma 19 (Unbiased coin from biased coins)

PTM with biased coins ($Pr[heads] = \rho$) can simulate an unbiased coin ($Pr[heads] = 1/2$) in $O(\frac{1}{\rho(1-\rho)})$ expected time.

Proof:

PTM tosses two coins: HT=heads, TH=tails, HH, TT=repeat.

$\Rightarrow Pr[\text{heads}] = Pr[\text{tails}] = \rho(1-\rho), Pr[\text{repeat}] = 1 - 2\rho(1-\rho)$

$E[\text{running time}] = \sum_i \left[i(1-2\rho(1-\rho))^{i-1}(2\rho(1-\rho))\right] = O(\frac{1}{\rho(1-\rho)})$

□

Theorem 20

$BPP \subseteq \Sigma_2^p \cap \Pi_2^p$

Proof: Some preliminary observations

- Since $BPP = coBPP$, enough to show $BPP \subseteq \Sigma_2^p$
- Set $S \subset \{0,1\}^m$ can be "shifted" by $u \in \{0,1\}^m$ by bit-wise XOR: $S + u = \{x \oplus u : x \in S\}$. Also $r = s \oplus u \Leftrightarrow r \oplus u = s$.
- If S is big then $\exists$ few shifts $u_1, u_2, \ldots, u_k$ that can cover all $\{0,1\}^m$ strings with $S \oplus u_1, S \oplus u_2, \ldots, S \oplus u_k$.
- If S is small then $\not\exists$ few shifts $u_1, u_2, \ldots, u_k$ that can cover all $\{0,1\}^m$ strings with $S \oplus u_1, S \oplus u_2, \ldots, S \oplus u_k$.

Chapter 7: Randomized computation

Theorem 21

$BPP \subseteq \Sigma_2^p \cap \Pi_2^p$

Proof: Some preliminary observations

- Since $BPP = coBPP$, enough to show $BPP \subseteq \Sigma_2^p$
- Set $S \subset \{0,1\}^m$ can be ”shifted” by $u \in \{0,1\}^m$ by bit-wise XOR: $S + u = \{x \oplus u : x \in S\}$. Also $r = s \oplus u \Leftrightarrow r \oplus u = s$
- $|S| \geq (1-2^{-n})2^m, k = \lceil \frac{m}{n} \rceil + 1 \Rightarrow \exists u_1, \ldots, u_k : \cup_i (S \oplus u_i) = \{0,1\}^m$
  Proof: Pick random u's. Show $Pr_u[\cup_i(S \oplus u_i) = \{0,1\}^m] > 0$
  Bad for r: $B_r^i = 1$ if $r \notin S \oplus u_i \Rightarrow B_r = \wedge_i B_r^i$

$$Pr_u[B_r] \prod_{i=1}^{k} Pr_{u_i}[B_r^i] = \prod_{i=1}^{k} Pr_{u_i}[r \notin S \oplus u_i] = \prod_{i=1}^{k} Pr_{u_i}[r \oplus u_i \notin S] \quad (1)$$

If u_i uniformly random $\Rightarrow r \oplus u_i$ uniformly random
$(1) \Rightarrow Pr_u[B_r] \leq \prod_{i=1}^{k}(1 - \frac{|S|}{2^m}) \leq 2^{-nk} < 2^{-m}$
$\Rightarrow 1 - Pr_u[\cup_i(S \oplus u_i) = \{0,1\}^m] = Pr[\exists r : B_r] < 2^m 2^{-m} = 1$ □

Theorem 22

$BPP \subseteq \Sigma_2^p \cap \Pi_2^p$

Proof: Some preliminary observations

- Since $BPP = coBPP$, enough to show $BPP \subseteq \Sigma_2^p$
- Set $S \subset \{0,1\}^m$ can be "shifted" by $u \in \{0,1\}^m$ by bit-wise XOR: $S + u = \{x \oplus u : x \in S\}$. Also $r = s \oplus u \Leftrightarrow r \oplus u = s$
- $|S| \geq (1-2^{-n})2^m, k = \lceil \frac{m}{n} \rceil + 1 \Rightarrow \exists u_1, \ldots, u_k : \cup_i (S \oplus u_i) = \{0,1\}^m$
- $|S| \leq 2^{m-n}, k = \lceil \frac{m}{n} \rceil + 1 \Rightarrow \forall u_1, \ldots, u_k : \cup_i (S \oplus u_i) \neq \{0,1\}^m$

  Proof:

  $|S \oplus u_i| = |S| \Rightarrow | \cup_{i=1}^k (S \oplus u_i)| \leq \sum_{i=1}^k |S \oplus u_i| \leq k|S| < 2^m$

  $\Rightarrow \exists r \in \{0,1\}^m : r \notin \cup_{i=1}^k (S \oplus u_i)$ □

Proof: (cont'd)

$L \in BPP \Rightarrow$ PTM M uses $m = poly(n)$ random bits and (boosting)

$$x \in L \Rightarrow Pr_r[M(x,r) = 1] \geq 1 - 2^{-n}$$
$$x \notin L \Rightarrow Pr_r[M(x,r) = 1] \leq 2^{-n}$$

If S_x are the random strings r that make $M(x,r) = 1$, then

$$x \in L \Rightarrow |S_x| \geq (1 - 2^{-n})2^m$$
$$x \notin L \Rightarrow |S_x| \leq 2^{-n}2^m$$

$x \in L \Rightarrow \exists u_1, \ldots, u_k \ \forall r \in \{0,1\}^m : r \in \cup_{i=1}^{k}(S_x \oplus u_i)$

$x \in L \Rightarrow \exists u_1, \ldots, u_k \ \forall r \in \{0,1\}^m : \bigvee_{i=1}^{k}(r \oplus u_i \in S_x)$

$x \in L \Rightarrow \exists u_1, \ldots, u_k \ \forall r \in \{0,1\}^m : \bigvee_{i=1}^{k}[M(x, r \oplus u_i) = 1]$

□

Chapter 7: Randomized computation

Are there *BPP*-complete problems?
Syntactic classes (e.g., *P*, *NP*, *PSPACE*) vs. Semantic classes (e.g., *BPP*, *RP*)

Time hierarchy theorem for *BPTIME*?
Same problem as before...

Definition 23 (**Randomized reductions**)

$B \leq_r C$ if $\exists$ PTM M s.t. $\forall x : Pr_r[C(M(x,r)) = B(x)] \geq 2/3$.

CAREFUL: Book has a **typo** in Definition 7.16!!!

Definition 24

$BP \cdot NP = \{L : L \leq_r 3SAT\}$

Definition 25

BPL, RL defined similarly to BPP, RP but now use $O(\log n)$ space.

Theorem 26

$UPATH \in RL$