COMS 4236 Homework 5

Mengqi Zong < *mz2326@columbia.edu* >

April 18, 2012

Problem 1

1. Here is a family of eamples that this algorithm takes exponential time:

$$E = \prod_{i=1}^{m-1} (a_{i+1}x_{i+1} + a_i x_i)$$

As we can see, each step we eliminate one pair of parentheses, the size of E doubles. This is because all the new terms are different. In total, this algorithm takes exponential time.

2. Guess an assignment $X = x_1, x_2, ..., x_m$ that E_1 and E_2 are different. If $E_1(X) = E_2(X)$, then accept. If not, reject.

This algorithm does not place this problem in RP, because in order to make Prob(M accepts when two equation are equal) greater equal 1/2, we have to try at least 2^{m-1} different assignments. In this case, the total running time is exponential, not a constant.

This algorithm does not place this problem in coRP, because Prob(M accepts when two equations are equal) is not 1.

This algorithm does not place this problem in BPP, same reason as that of RP.

Question 2

a) $1 \Rightarrow 2$

Since $L \in RP$, we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(acc) \geq 1/2, Pr(rej) \leq 1/2 \\ \forall x \notin L &\Rightarrow Pr(acc) = 0 \end{aligned}$$

Since $L \in coRP$, we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(acc) = 1, Pr(rej) = 0 \\ \forall x \notin L &\Rightarrow Pr(acc) \leq 1/2, Pr(rej) \geq 1/2 \end{aligned}$$

Since $L \in ZPP = RP \cap coRP$, combine the results together, we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) \geq 1/2, Pr(No) = 0 \\ \forall x \notin L &\Rightarrow Pr(Yes) = 0, Pr(No) \geq 1/2 \end{aligned}$$

Add $Pr(?)$, we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) \geq 1/2, Pr(No) = 0, Pr(?) \leq 1/2 \\ \forall x \notin L &\Rightarrow Pr(Yes) = 0, Pr(No) \geq 1/2, Pr(?) \leq 1/2 \end{aligned}$$

b) $2 \Rightarrow 3$

We can run the Turing Machine M in part two n times. If in the n times, there is at least one "Yes", we accept. If not, we reject. As a result, we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) \geq 1 - 2^{-n}, Pr(No) = 0, Pr(?) \leq 2^{-n} \\ \forall x \notin L &\Rightarrow Pr(Yes) = 0, Pr(No) \geq 1 - 2^{-n}, Pr(?) \leq 2^{-n} \end{aligned}$$

c) $3 \Rightarrow 1$

From part 3, since

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) \geq 1 - 2^{-n} \geq 1/2 \\ \forall x \notin L &\Rightarrow Pr(Yes) = 0 \end{aligned}$$

So $L \in RP$. Since

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(No) = 0 \\ \forall x \notin L &\Rightarrow Pr(No) \geq 1 - 2^{-n}, Pr(?) \leq 2^{-n} \end{aligned}$$

we get

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) = 1 \\ \forall x \notin L &\Rightarrow Pr(No) \leq 1 - 2^{-n} \leq 1/2, Pr(?) \leq 2^{-n} \end{aligned}$$

So $L \in coRP$.

To sum up, $L \in ZPP = RP \cap coRP$.

Question 3

a) If L is in ZPP, then it is decided by a probabilistic Turing Machine N that runs in expected polynomial time.

From condition 3 of Problem 2 , we know that if L is in ZPP, there is a probabilistic Turing Machine M which runs in polynomial time and return either "Yes" or "No" or "?" such that

$$\begin{aligned} \forall x \in L &\Rightarrow Pr(Yes) \geq 1 - 2^{-n}, Pr(No) = 0, Pr(?) \leq 2^{-n} \\ \forall x \notin L &\Rightarrow Pr(Yes) = 0, Pr(No) \geq 1 - 2^{-n}, Pr(?) \leq 2^{-n} \end{aligned}$$

So, the probability of M giving an correct answer is $Pr_s \geq 1 - 2^{-n}$. Let $T_M(x)$ denote the time that M runs once on input x, then we get

$$\begin{aligned}\overline{T}_N(x) &= Pr_s \cdot T_M(x) + (1-Pr_s)Pr_s \cdot 2T_M(x) + (1-Pr_s)^2 Pr_s \cdot 3T_M(x) + ... \\ &= Pr_s T_M(x)(1 + 2(1-Pr_s) + 3(1-Pr_s)^2 + 4(1-Pr_s)^3 + ...)\end{aligned}$$

Let $q = (1 - Pr_s) \leq 2^{-n}$, $S = 1 + 2q + 3q^2 + 4q^3 + ...$ Since $q < 1$, we can get

$$\begin{aligned}qS &= q + 2q^2 + 3q^3 + 4q^4 + ... \\ (1-q)S &= 1 + q + q^2 + q^3 + ... \\ &\leq \frac{1}{1-q} \\ S &\leq \frac{1}{(1-q)^2}\end{aligned}$$

Using the last inequality, we then get

$$\begin{aligned}\overline{T}_N(x) &= Pr_s T_M(x)(1 + 2(1-Pr_s) + 3(1-Pr_s)^2 + 4(1-Pr_s)^3 ...) \\ &\leq \frac{Pr_s T_M(x)}{(1-(1-Pr_s))^2} \\ &= \frac{Pr_s T_M(x)}{Pr_s^2} \\ &= \frac{T_M(x)}{Pr_s} \\ &\leq P(|x|)\end{aligned}$$

So, M is the Turing Machine we are looking at.

b) If L can be decided by a probabilistic Turing Machine N that runs in expected polynomial time, then L is in ZPP.

If L can be decided by a probabilistic Turing Machine N that runs in expected polynomial time, then it means every computation of N terminates with the correct answer Yes or No.

Let p denote the probability of Turing Machine N giving an correct answer in the polynomial time $P(|x|)$. Let Pr_s denote the probability of M giving an correct answer, then we get

$$
\begin{aligned}
Pr_s = 1 - (1-p)^m &\geq \frac{1}{2} \\
(1-p)^m &\leq \frac{1}{2} \\
m &\geq \log_{1-p} \frac{1}{2} \\
m &\geq -\log_{1-p} 2 \\
m &\geq -\frac{\log_2 2}{\log_2 (1-p)} \\
m &\geq -\frac{1}{\log_2 (1-p)}
\end{aligned}
$$

As we can see, in order to make Pr_s greater than $1/2$, we must run the Turing Machine at least $m \cdot P(|x|)$ time. That is, $(-\frac{1}{\log_2 (1-p)}) \cdot P(|x|)$. If the M runs longer than this time, we can terminate the computation and note the result as "?". Still, we can get $Pr_s \geq 1/2$. In this case, we get

$$
\begin{aligned}
\forall x \in L &\Rightarrow Pr(Yes) \geq 1/2, Pr(No) = 0, Pr(?) \leq 1/2 \\
\forall x \notin L &\Rightarrow Pr(Yes) = 0, Pr(No) \geq 1/2, Pr(?) \leq 1/2
\end{aligned}
$$

That is, L is in ZPP.

To sum up, L is in ZPP, if and only if it is decided by a probabilistic Turing Machine N that runs in expected polynomial time.

Question 4

1. From $A \leq_p^T B$, we know that $A = L(M^B)$. From $B \leq_p^T C$, we know that $B = L(M^C)$. Then $A = L(M^{L(M^C)})$. Since polynomial times polynomial is still polynomial, $A = L(M^C)$. That is, $A \leq_P^T C$.

2.

a) P is closed under Cook reductions. This has been shown in the class , $P^P = P$.

b) NP is closed under Cook reductions. Since B is in NP, we know that B can be verified in polynomial time. Since polynomial times polynomial is still polynomial, then we can verify A in polynomial time. So A is in NP.

c) coNP is closed under Cook reductions. Since B is in coNP, we know that $\overline{B}$ can be verified in polynomial time. Since polynomial times polynomial is still polynomial, then we can verify $\overline{A}$ in polynomial time. So A is in coNP.

d) $Delta_2$ is closed under Cook reductions. $\Delta_2 = P^{\Sigma_1} = P^{NP}$. And $P^{\Delta_2} = P^{P^{NP}} = P^{NP}$. Since B is in Δ_2, then A is also in Δ_2.

e) PH is closed under Cook reductions. $PH = \bigcup \Delta_i = \bigcup P^{\Sigma_{i-1}}$. So $P^{PH} = \bigcup P^{P^{\Sigma_{i-1}}} = P^{\Sigma_{i-1}}$. Since B is in PH, then A is also in PH.

f) PSPACE is closed under Cook reductions. We know that B is in PSPACE. Polynomial time Polynomial is still polynomial, so A is also in PSPACE.

Question 5

a) $NP \subseteq NP^{NP \cap coNP}$

We know that $NP = NP^P$. Since $P \subseteq NP \cap coNP$, we get $NP \subseteq NP^{NP \cap coNP}$.

b) $NP^{NP \cap coNP} \subseteq NP$

$L \in NP \cap coNP$ means that

$\forall x \in L$, x can be verified in poly-time using a NTM
$\forall x \notin L$, x can be verified in poly-time using a NTM

In this case, any input $x \in L$ can be verified in poly-time using a NTM. Then for $L' \in NP^{NP \cap coNP}$, any input $x' \in L'$ can be verified in poly-time using a NTM. Because polynomial times polynomial is still a polynomial. So we get $NP^{NP \cap coNP} \subseteq NP$

To sum up, $NP^{NP \cap coNP} = NP$.