COMPUTATIONAL COMPLEXITY HW-3

SIDDHANT CHAUDHARY, AMIK RAJ BEHERA
BMC201953 , BMC201908

Problem 1. **ZPP** is the complexity class which contains all the languages L for which there is a machine M that runs in expected polynomial time but never makes a mistake on any input. Prove that **ZPP** $=$ **RP** $\cap$ **coRP**.

Proof. We will first show that **ZPP** $\subseteq$ **RP**$\cap$**coRP**. Since **ZPP** is closed under complementation (we can invert the output in constant time), it suffices to show that **ZPP** $\subseteq$ **RP**. Before we proceed, we revisit the Markov Inequality and look at a special case of it:

Let X be a random variable such that $X \geq 0$ with expectation $\mathbb{E}[X]$. Then Markov Inequality is

$$Pr(X \geq a) \leq \frac{\mathbb{E}[X]}{a} \tag{0.1}$$

If we substitute a = $3\mathbb{E}[X]$ in the above inequality, then we get

$$Pr(X \geq \mathbb{E}(x)) \leq \frac{1}{3}. \tag{0.2}$$

Let $L \in$ **ZPP** and M be a probabilistic Turing machine which decides L. By definition of **ZPP**, we know that M runs in expected polynomial time, call it $T(n)$. We define a new probabilistic Turing machine as follows:

Algorithm 1 $M'(x)$

Run $M(x)$ for time $3T(n)$, where $n = |x|$
if $M(x)$ halts in time $3T(n)$ **then**
  return $M(x)$
else
  return NO
end if

Note that M' runs in expected polynomial time (from definition of **ZPP**). By (0.2), the probability that $M(x)$ runs for time more than $3T(n)$ is less than $1/3$. Now we have three possibilities:

(1) $x \in L$ and $M(x)$ halts in time $3T(n)$. Then $M(x)$ = YES.
(2) $x \in L$ and $M(x)$ does not halt in time $3T(n)$. Then $M(x)$ = NO. Probability of this event occurring is less than $1/3$.
(3) $x \notin L$. Then no matter when $M(x)$ halts, $M(x)$ = NO.

Date: April 20 2021.

Thus we have showed that $L \in$ **RP**.

Now we will show that **RP** $\cap$ **coRP** $\subseteq$ **ZPP**. Let $L \in$ **RP** $\cap$ **coRP**. There is a probabilistic Turing machine A running in polynomial time such that if $x \in L$, then A outputs YES with probability atleast $2/3$, otherwise A outputs NO with probability 1. Similarly, there is a probabilistic Turing machine B running in polynomial time such that if $x \notin L$, then B outputs NO with probability atleast $2/3$, otherwise B outputs YES with probability 1.

Now we will define a probabilistic Turing machine M which on input x does the following:

Algorithm 2 M(x)

```
Run A(x) {Beginning of the iteration}
if A(x) == YES then
  return YES
else
  Run B(x)
  if B(x) == NO then
    return NO {End of the iteration}
  else
    Repeat this iteration
  end if
end if
```

We will show that this Turing machine never makes mistake on any input:

(1) If M outputs YES, then $A(x)$ outputs YES, which implies that $x \in L$.
(2) If M outputs NO, then $B(x)$ outputs NO, which implies that $x \notin L$.

Now we will show that M runs in expected polynomial time. On some input x:

- If $x \in L$, then

$$Pr[M(x) \text{ halts after one iteration}] = Pr[A(x) \text{ outputs YES}] \geq \frac{2}{3}$$

- If $x \notin L$, then

$$Pr[M(x) \text{ halts after one iteration}] = Pr[B(x) \text{ outputs NO}] \geq \frac{2}{3}$$

Let k denote the number of iterations required on some input x. Then by the above observation, we can find $\mathbb{E}[k]$ recursively as follows:

$$\mathbb{E}[k] = 1 + Pr[M(x) \text{ didn't halt after one iteration}] \cdot \mathbb{E}[k]$$
$$\Rightarrow \mathbb{E}[k] \leq \frac{3}{2}$$

Clearly, each iteration runs in polynomial time. Thus M runs in expected polynomial time and never makes mistakes, and recognizes L, which in turn implies that $L \in$ **ZPP**.

Hence we have showed that **ZPP** $=$ **RP** $\cap$ **coRP**. ■

Problem 2. B reduces to C under a randomized polynomial time reduction, denoted by $B \leq_r C$, if there is a probabilistic TM M such that for all x,

$$\mathbb{P}[C(M(x)) = B(x)] \geq 2/3$$

Define

$$\textbf{BP.NP} := \{L \mid L \leq_r 3\text{SAT}\}$$

Prove that **BP.NP** $\subseteq$ **NP/Poly**.

Proof. Let $L \in$ **BP.NP**. This means there exists a polynomial time reduction M from L to $3SAT$ and a polynomial $p : \mathbb{N} \to \mathbb{N}$ such that for every $x \in \{0,1\}^*$,

$$Pr_{r \in \{0,1\}^{p(|x|)}}\left[L(x) = 3SAT(M(x,r))\right] > 1 - \frac{1}{2^{|x|}},$$

(such a Turing machine exists by Error reduction using Chernoff Bounds). We will show that $L \in$ **NP/poly** by giving a family $\mathcal{C}_n$ of non-deterministic polynomial sized circuits.

Let $x \in \{0,1\}^*$ with $|x| = n$. For a given x, we say a sequence of choices for M is "good" if $3SAT(M(x,r)) = L(x)$, "bad" otherwise. While running M with input x, there are in total $2^{p(n)}$ possible sequences of choices for M. From definition of **BP.NP**, there are strictly less than $2^{p(n)-n}$ sequence of "bad" choices. Summing over all $x \in \{0,1\}^n$, there are strictly less than $2^{p(n)}$ sequence of "bad" choices.

This means, there exists a sequence of "good" choice, of length $p(n)$, call it as α_n, such that for all $x \in \{0,1\}^n$, $3SAT(M(x,\alpha_n)) = L(x)$.

Then $\mathcal{C}_n$ is described as follows: It has input x of length n, and a witness y. $C_n(x,y)$ computes whether the witness y satisfies the $3SAT$ instance $M(x,\alpha_n)$. Since α_n fixed for every x of length n, we hard-wire α_n in our circuit C_n. Clearly, we get

$$x \in L \Leftrightarrow M(x,\alpha_n) \in 3SAT \Leftrightarrow \exists y \, C_n(x,y) = 1.$$

Computing $M(x,\alpha_n)$ is of polynomial size since M is a polynomial time reduction. Also verifying whether y is a satisfying assignment of $M(x,\alpha_n)$ or not can be done in polynomial size. Thus C_n is a family of non-deterministic polynomial sized circuits, which implies $L \in$ **NP/poly**. Hence **BP.NP** $\subseteq$ **NP/poly**. ■

Problem 3. Prove that there exists a perfectly complete **AM**$[O(1)]$ protocol for proving a lower bound on set size.

Proof. Suppose we have a family $\mathcal{H}_{m,k}$ of pairwise independent hash functions. We can construct such a family as shown in the next problem.

We will prove the claim in two steps. First, we will show that there is an **AM**$[O(1)]$ protocol for proving a lower bound on set size which has exponentially small error probability (which is essentially just using the Chernoff Bound, as we will show). After doing this, we exhibit a perfectly complete **AM**$[O(1)]$ protocol for set lower-bound.

So, let $S \subseteq \{0,1\}^m$ be a set such that any $x \in S$ has an efficient (polynomial sized) proof of membership in S. Let K be a fixed number. Recall that the

Goldwasser-Sipser protocol was the **AM**$(O(1))$ protocol for set lower-bound that we covered in class. Now, consider the following protocol.

(1) The verifier V randomly picks n pairs (h_i, y_i) where $h_i \in \mathcal{H}_{m,k}$ and $y_i \in \{0,1\}^k$ for each i and sends these n pairs to the prover P.
(2) P produces $x_i \in S$ such that $h_i(x_i) = y_i$ for each i and sends the same to V, along with a proof of membership of x_i in S.
(3) V accepts if there are more than $n/2$ indices i such that the proof of membership of x_i in S is valid and $h_i(x_i) = y_i$.

We now prove that this is the required protocol for set lower-bound with exponentially small error probability. Let X_i be the random variable

$$X_i = \begin{cases} 1 & \text{if } h_i(x_i) = y_i \text{ for some } i \\ 0 & \text{otherwise} \end{cases}$$

and put

$$X = \frac{1}{n}\sum_{i=1}^{n} X_i$$

We need to handle the following cases.

(1) Suppose $|S| \geq K$. Then since the Goldwasser-Sipser protocol belongs to **AM**$[2]$, we see that $X_i = 1$ with probability $\geq 2/3$. So,

$$\mathbb{E}[X] \geq \frac{1}{n} \cdot n\frac{2}{3} = \frac{2}{3}$$

So by the Chernoff Bound, we see that

$$\mathbb{P}\left[X \leq \frac{1}{2}\right] = \mathbb{P}\left[X - \frac{2}{3} \leq \frac{-1}{6}\right] \leq \mathbb{P}\left[|X - \mathbb{E}[X]| \geq \frac{1}{6}\right] \leq e^{-n/c}$$

where c is positive constant ($c = -(1/4)^2/4$). This means that

$$\mathbb{P}\left[X > \frac{1}{2}\right] \geq 1 - e^{-n/c}$$

(2) Next, suppose $|S| \leq \dfrac{K}{2}$. In this case, again since the Goldwasser-Sipser protocol belongs to **AM**$[2]$, we see that $X_i = 1$ with probability atmost $1/3$. So,

$$\mathbb{E}[x] \leq \frac{1}{n} \cdot n\frac{1}{3} = \frac{1}{3}$$

Hence

$$\mathbb{P}\left[X > \frac{1}{2}\right] \leq \mathbb{P}\left[X > \frac{1}{3} + \frac{1}{12}\right] \leq \mathbb{P}\left[|X - \mathbb{E}[X]| > \frac{1}{12}\right] \leq e^{-n/c'}$$

where c' is some positive constant.

So, in both cases we see that the error probability is exponentially small. Hence, there is an **AM**$[2]$ protocol for set lower-bound with exponentially small error probability.

Now, consider the following **AM**$[O(1)]$ protocol for set lower-bound. Again, suppose the input set is S and the number K is fixed. We will use the ideas in the Sipser-Gacs Theorem extensively.

(1) Let S' be the set of all sequences $(h_1, y_1), ..., (h_n, y_n)$ such that $h_i \in \mathcal{H}_{m,k}$ and $y_i \in \{0,1\}^k$ for each i, and such that there are atleast $n/2$ indices i for which there exists $x_i \in S$ such that $h_i(x_i) = y_i$. The verifier V sends the description of such a set S' to P (here the sequence $(h_1, y_1), ..., (h_n, y_n)$ acts as the random string which V generates and sends to P). Suppose the length of such a random string is bounded above by l. Note that if $|S| \geq K$, then as in our previous exponentially small error protocol, we see that

$$|S'| \geq \left(1 - \frac{1}{2^n}\right) 2^l$$

and if $|S| \leq K/2$, then

$$|S| \leq \frac{1}{2^n} 2^l$$

(2) Let $k = \dfrac{l}{n} + 1$ (just like in Sipser-Gacs). P then produces $u_1, ..., u_k \in \{0,1\}^l$ and sends it to V.
(3) V produces $r_0 \in \{0,1\}^l$ and sends it to P.
(4) P proves $r_0 \in \bigcup_{i=1}^k (S' + u_i)$ where the $+$ operator represents translating the set S'. If $r_0 + u_i \in S'$ for some i, then V accepts, otherwise it rejects.

Again, note that if $|S| \geq K$, then $|S'| \geq (1 - \dfrac{1}{2^n})2^l$ (because of the exponentially low error). Then using the probabilistic method just like in the proof of Sipser-Gacs, it can be shown that there exist $u_1, ..., u_k$ such that

$$\bigcup_{i=1}^k S' + u_i = \{0,1\}^l$$

and hence this means that there is a strategy for P to convince the verifier, implying that V accepts with probability 1.

Similarly, if $|S| \leq K/2$, then $|S'| \leq \dfrac{2^l}{2^n}$, and hence

$$\mathbb{P}[V \text{ accepts}] = \mathbb{P}_{r_0 \in \{0,1\}^l} \left[r_0 \in \bigcup_{i=1}^k S' + u_i \right] \leq k \frac{1}{2^l} \cdot \frac{2^l}{2^n} = \frac{l+n}{n 2^n}$$

which is exponentially small. So, this is the required perfectly complete protocol. ∎

Problem 4. Let $k \leq n$. Construct a family $\mathcal{H}_{n,k}$ of pairwise independent functions $\{0,1\}^n \to \{0,1\}^k$ as discussed in class.

Proof. Let $\mathrm{D} = \mathbb{F}_{2^n}$ and $\mathrm{R} = \mathbb{F}_{2^k}$. We describe a class of functions $h_{a,b}$ from D to R as follows:

$$\mathcal{H}_{n,k} = \{h_{a,b}(x) = (a \cdot x + b) \bmod 2^k \mid a, b \in \mathbb{F}_{2^n}\},$$

where the multiplication is defined in $\mathbb{F}_{2^n}$. Consider $x, x' \in \mathbb{F}_{2^n}$ such that $x \neq x'$ and $y, y' \in \mathbb{F}_{2^k}$. Then we have

$$\begin{aligned}
&\Pr_{h \in \mathcal{H}_{n,k}} [h(x) = y \wedge h(x') = y'] \\
&= \Pr_{h \in \mathcal{H}_{n,k}} [(a \cdot x + b = y) \bmod 2^k \wedge (a \cdot x' + b = y') \bmod 2^k] \\
&= \Pr_{h \in \mathcal{H}_{n,k}} [(a = (y - y') \cdot (x - x') \bmod 2^k) \wedge (b = y - a \cdot x \bmod 2^k)] \\
&= \frac{1}{2^k} \cdot \frac{1}{2^k} = \frac{1}{2^{2k}} \\
&= \frac{1}{|\mathrm{R}|^2}
\end{aligned}$$

Thus we have showed that $\mathcal{H}_{n,k}$ is a pairwise independent functions. It is also easy to see that each function of $\mathcal{H}_{n,k}$ is efficiently computable. ■

Problem 5. Prove that QUADEQ is **NP**-complete.

Proof. We will show that CIRCUIT − SAT is reducible to QUADEQ, which will imply that QUADEQ is **NP**-complete. Let's first revisit the definition of QUADEQ:

There is a system of m quadratic equations over $\mathbb{F}_2$, with variables $x_1, \ldots, x_n$. Each quadratic equation is of the form

$$\sum_{i,j \in [n]} c_{ij} x_i x_j = b, \quad \{c_{ij} \mid i, j \in [n]\}, b \in \mathbb{F}_2$$

We say that the system of m quadratic equations is in QUADEQ if there exists a satisfying assignment $\{x_1, \ldots, x_n\} \in \{0, 1\}^n$.

We now give a polynomial time reduction from a circuit to set of quadratic equations. Let C be a circuit with n input variables. Let $\{x_1, \ldots, x_n\}$ represent the n input gates. We will define a set of equivalent quadratic equations for each of AND, OR and NOT gates using **arithmetization** as follows

Notation: If gate i has fan-in of 2, then gate j and gate k are the inputs, otherwise only gate j is the input. All the operations are in $\mathbb{F}_2$.

$$x_i = \begin{cases} x_j x_k & \text{if } i \text{ is an AND gate} \\ a_j x_j + a_k x_k - x_j x_k & \text{if } i \text{ is an OR gate and } a_j, a_k \in \mathbb{F}_2 \\ (1 - x_j) & \text{if } i \text{ is an NOT gate} \end{cases}$$

Each of the above equation is a quadratic equation. For example, $x_i - x_j x_k = 0 \Rightarrow a_i x_i + x_j x_k = 0$. a_i's are variables in $\mathbb{F}_2$.
It is clearly evident that C is satisfiable if and only if the above set of equations is satisfiable. If C has m gates, then the above reduction takes poly(m) time. Thus the above reduction is in polynomial time. Hence QUADEQ is **NP**-complete. ■