CS 6810: Theory of Computing	**Fall 2021**

Lecture 17: Oct 26, 2021

Lecturer: Eshan Chattopadhyay — *Scribe: Adarsh Srinivasan*

1 Introduction

In this lecture, we further discuss the complexity classes AM and MA. Unlike the class IP, these protocols use public coins. In the class MA, Merlin sends Arthur a proof, that Arthur then verifies using a randomised verifier using public coins. In the class AM on the other hand, Arthur first sends Merlin a set of random coin tosses. Merlin can then decide on a proof that depends on the coin tosses, and Arthur then verifies this proof using the coin tosses previously generated. In this lecture, we prove that both these classes have the property of *perfect completeness*, and along the way prove that MA is contained in AM. We also discuss the relation of AM and PH.

2 Obtaining Perfect completeness

Theorem 2.1 (Perfect completenss of MA). *For any language $L \in MA$, then there exists a probabilistic polynomial time verifier V such that*

$$x \in L \implies \textit{there exists } m \textit{ such that } Pr_r[V(x,r,m) = 1] = 1$$

$$x \notin L \implies \textit{for all } m, Pr_r[V(x,r,m) = 1] \leq 1/3$$

Proof. By using an error reduction technique (similar to the one used for BPP), we can say that there exists a verifier V such that

$$x \in L \implies \text{there exists } m \text{ such that } Pr_r[V(x,r,m) = 1] \geq 1 - 2^{-n}$$

$$x \notin L \implies \text{for all } m, Pr_r[V(x,r,m) = 1] \leq 2^{-n}$$

Now a similar argument to the one we used to prove $BPP \subseteq PH$ completes the proof. For a given x, we define the set 1_x as follows:

$$1_x = \{r : \exists m, V(x,r,m) = 1\}$$

If the probability of success is large ($x \in L$), then 1_x is large. Hence, if $x \in L$, for $k = poly(n)$, there exist vectors $v_1, \ldots, v_k$ such that for all r, there exists i such that $v_i \oplus r \in 1_x$. On the other hand, if $x \notin L$, the probability that there exists i such that $r \oplus v_i \in 1_x$ is tiny. The proof of this is the same as the proof given in lecture 14. Having proved this fact, the protocol can be easily defined:

Merlin sends Arthur a string m and strings $v_1, \ldots, v_k$. Arthur accepts if $V(x, r \oplus v_i, m) = 1$ for at least one v_i. If $x \in L$, Merlin, being all powerful can compute the strings v_i as they exist. If $x \notin L$ however, for every message m, the chance that at lest one of the strings $r \oplus v_i \in 1_x$ is very small, at most $k2^n$. Notice that this is the probability of the new verifier V' accepting. Hence, we have proved the perfect completeness of MA □

We now show that the complexity class MA is a subset of the class AM:

Theorem 2.2.
$$MA \subseteq AM$$

Proof. Given a language $L \in MA$, we define an AM protocol to compute it. We first perform error reduction to get an MA protocol with the verifier succeeding with probabilities at least $1 - 1/2^{b+2}$ if $x \in L$ and with probability at most $1/2^{b+2}$ if $x \notin L$ where $b = |m|$, the length of the message. This MA protocol uses a verifier $\tilde{V}(x, m, \tilde{r})$, where $\tilde{r}$ is a concatenation of b random strings. Now we define the following protocol:

Arthur first sends Merlin the random string . Merlin then responds with the message m. Note that in the MA protocol, this string does not depend on $\tilde{r}$ at all. Hence, if $x \in L$, an honest prover can just respond with the same string m that he would have sent in the MA protocol. If $x \notin L$ however, the probability of the verifier accepting is:

$$Pr_r[\exists m \text{ such that } \tilde{V}(x, m, \tilde{r}) = 1] \leq 2^b \max_m Pr_r[\tilde{V}(x, m, \tilde{r}) = 1]$$

This follows from the union bound on m. Notice that we have chosen $\tilde{V}$ such that $Pr_r[\tilde{V}(x, m, \tilde{r}) = 1] \leq 1/2^{b+2}$. hence, the probability of success it at most $1/4$, showing that this protocol is indeed an AM protocol. □

An interesting consequence of this theorem is that $AM[k] = AM[2]$ for any constant k. For example, if $k = 4$: $AMAM = AAMM = AM$. However k has to be a constant. If k is not a constant, the amount of communication becomes exponentially large.

We now show the perfect completeness of the class AM.

Theorem 2.3 (Perfect completeness of AM)**.** *In the AM protocol,*

$$x \in L \implies \text{ there exists } m(\cdot) \text{ such that } Pr_r[V(x, r, m(r)) = 1] = 1$$

$$x \notin L \implies \text{ for all } m(\cdot), Pr_r[V(x, r, m(r)) = 1] \leq 1/3$$

Proof. Recall the original Arthur Merlin protocol. To check if $x \in L$, Arthur first sends Merlin a a random string r, who responds with a proof m. Arthur then runs a deterministic verifier $V(x, m, r)$ such that:

$$x \in L \implies Pr_r[V(x, m, r) = 1] \geq 1 - 1/2^n$$
$$x \in L \implies Pr_r[V(x, m, r) = 1] \leq 1/2^n$$

Notice that we have applied the standard error reduction technique. Now, we use the same covering property to prove that there exists $k = poly(n)$ such that:

$x \in L$ implies that there exists $v_1, \ldots, v_k$ such that for all r there exists m such that $V(x, r \oplus v_i, m) = 1$ for at least one i. On the other hand, if $x \notin L$, the probability that $V(x, r \oplus v_i, m) = 1$ for at least one i is at most $k2^{-i}$. Hence, we define the following protocol:

Merlin sends Arthur a sequence $v_1, \ldots, v_k$ satisfying this property. As such a sequence exists, the all powerful Arthur can compute them. Arthur then sends Merlin a random string r. Merlin then sends Arthur m and i such that $V(x, r \oplus v_i, m) = 1$, which Arthur verifies. Note that if $x \in L$, such a pair exists, and if $x \notin L$, the probability that such a pair exists can be bounded.

This protocol is not an AM protocol however. It is an MAM protocol. The reason we defined the protocol this way is that, if Arthur sent Merlin the random string before Merlin sent Arthur the sequence, it would be trivial for Merlin to find such a vector v such that $v \oplus r \in 1_x$. But now that we have defined an MAM protocol, we can use the previous theorem that $MA \subseteq AM$ to define a corresponding AM protocol from this. □

3 AM and the polynomial hierarchy

In this section, we explore the relationship between Arthur Merlin games and the polynomial hierarchy.

Theorem 3.1.

$$AM \subseteq \Pi_2$$

Proof. Recall the proof of the Sipser-Gacs-Lautemann theorem that $BPP \subseteq \Sigma_2 \cap \Pi_2$. This proof is similar to that. Using the AM protocol, we define an expression in Π_2 which checks if $x \in L$ For the verifier V, we define the set 1_x to be $\{r : \exists m V(x, m, r) = 1\}$. We can now use the covering argument to show that $x \notin L$ is equivalent to the statement:

For all $v_1, \ldots, v_k$, there exists r such that for all $j \in \{1, \ldots, k\}$, there exists m such that $V(x, m, r \oplus v_j) = 0$. We can rewrite this in prenex form to obtain a Π_2 expression for L. □

Theorem 3.2. *if $coNP \subseteq AM$, the polynomial hierarchy must collapse to the second level*

Proof. We want to show that, assuming $coNP \subseteq AM$, $\Sigma_2 SAT \in \Pi_2$. Consider an instance of $\Sigma_2 SAT$, $\exists x \forall y \phi(x, y)$. By know that $\forall y \phi(x, y) \in coNP \in AM$. Hence we can design an MAM protocol for $\Sigma_2 SAT$, which implies an AM protocol. But, by the previous theorem, $AM \subseteq \Pi_2$ which completes the proof. □

As Graph isomorphism has an AM protocol, we can derive the following corollary:

Corollary 3.3. *If the graph isomorphism is NP complete, the polynomial hierarchy must collapse to the second level.*