

Lecture 21

Lance Fortnow (notes for 02/29/08 scribed by Sanchit Misra)

March 5, 2008

1 Relation between AM and MA

Claim: $MA \subseteq AM$

Proof:

- $L \in MA$ if there is an $A \in P$ s.t. $\forall x$
 $x \in L \Rightarrow \exists y Pr_r((x, y, r) \in A) \geq \frac{2}{3}$
and $x \notin L \Rightarrow \forall y Pr_r((x, y, r) \in A) \leq \frac{1}{3}$
 $|y| \leq poly(|x|), |r| \leq poly(|x|)$
- $L \in AM$ if
 $x \in L \Rightarrow Pr_r(\exists y(x, y, r) \in A) \geq \frac{2}{3}$
and $x \notin L \Rightarrow Pr_r(\forall y(x, y, r) \in A) \leq \frac{1}{3}$
- Using chernoff bounds, make error in MA $\leq \frac{1}{2^{|y|+1}}$
Hence, $\Pr(V \text{ commits an error for a message } y) \leq \frac{1}{2^{|y|+1}}$
 $\Rightarrow Pr(\exists \text{ message } y \text{ s.t. } V \text{ commits error}) \leq \frac{2^{|y|}}{2^{|y|+1}} = \frac{1}{2}$
Hence even if we pick the coin before, the error $\leq \frac{1}{2}$

Hence $MA \subseteq AM$

- We don't think $AM \subseteq MA$
- $MAM \subseteq AM$
- $AMAMAM = AM$, infact any constant number of rounds of A and M is computable in AM public coin

2 Is GI NP-complete?

Suppose GI is NP-complete \Rightarrow GNI is Co-NP-complete.

We showed that $GNI \in AM$ this implies $Co - NP \subseteq AM$

2.1 Boppana - Hastad Lemma '86

Lemma : $Co - NP \subseteq AM \Rightarrow PH = \Sigma_2^p$

- Hence, GI NP-complete $\Rightarrow PH = \Sigma_2^p$
- If PH infinite ;i.e.; $\forall k \Sigma_k^p \neq \Sigma_{k+1}^p$, then $PH \neq \Sigma_2^p \Rightarrow$ GI is not NP-complete
- It is not known but PH is believed to be infinite.

3 IP = unbounded round interactive proofs

Lemma: $IP = PSPACE$

Step1 : $Co - NP \subseteq IP$

To prove this we prove that $\overline{3SAT} \in IP$

proof:

- Fix large prime q , work over $GF(q)$ (regular + & * modulo q)
- For any clause of the form $(x \vee \bar{y} \vee z)$, write it in the form of a polynomial:
$$h(x, y, z) = x + (1 - y) + z - x(1 - y) - xz - (1 - y)z + x(1 - y)z$$

if $x = z = 0$ and $y = 1$, $h(x, y, z) = 0$ otherwise if $x = 1$ or $z = 1$ or $y = 0$, and $(x, y, z) \in \{0, 1\}^3$, $h(x, y, z) = 1$

The above process of converting a boolean formula to a polynomial is called arithmetization.
- If $\phi = (x \vee \bar{y} \vee z) \wedge (x \vee w \vee \bar{y}) \wedge \dots \wedge (...)$, arithmetize each clause of ϕ to get the corresponding polynomial:
$$\tilde{\phi} = (\text{polynomial for clause\# 1}) * (\text{polynomial for clause\# 2}) * \dots * (\text{polynomial for clause\# m})$$

- $degree(\tilde{\phi}) \leq 3m$ since there are m clauses and each clause has 3 variables out of x_1, x_2, \dots, x_n
- Define $V_0 = \sum_{x_1=0}^1 \sum_{x_2=0}^1 \dots \sum_{x_n=0}^1 \tilde{\phi}(x_1, x_2, \dots, x_n)$
 $V_0 = \#$ of satisfying assignments of ϕ
Hence $\phi \notin SAT \Leftrightarrow V_0 = 0$
- Define $g(x_1) = \sum_{x_2=0}^1 \sum_{x_3=0}^1 \dots \sum_{x_n=0}^1 \tilde{\phi}(x_1, x_2, \dots, x_n)$
 $degree(g_1) \leq 3m$

So the following rounds of prover(P) and verifier(V) are executed:

- $P \rightarrow V$: polynomial $h_1(x_1)$ of $degree \leq 3m$ and claims $h_1 = g_1$
- Verifier V : Rejects if $h_1(0) + h_1(1) \neq 0$
 $(g_1(0) + g_1(1) = V_0$ hence either $V_0 \neq 0(\phi \in SAT)$ or $h_1 \neq g_1)$
- V : Pick random $y_1 \in GF(q)$
- $V \rightarrow P : y_1$
If $\phi \notin SAT$ & prover follows protocol (is honest), $h_1 = g_1 \Rightarrow h_1(y_1) = g_1(y_1)$
Suppose $\phi \in SAT$ & $h_1(0) + h_1(1) = 0, \Rightarrow h_1 \neq g_1$ (since $g_1(0) + g_1(1) = V_0 \neq 0$)
If $h_1 \neq g_1 \Rightarrow Pr_{y_1 \in GF(q)}(h_1(y_1) = g_1(y_1)) \leq \frac{d}{q}$
Since $h_1 - g_1$ is polynomial of degree $d \leq 3m$. Hence $h_1 - g_1$ has atmost d roots. This means h_1 and g_1 can be equal at atmost d values of y_1 . And there are q values of y to choose from. Hence if we choose q large enough, with high probability, $h_1(y_1) \neq g_1(y_1)$
- This proof continues in the next lecture.