6.841/18.405J: Advanced Complexity Theory April 3, 2002

Lecture 13

Lecturer: Madhu Sudan *Scribe: Jason Hickey*

In this lecture we will show the following:

1. IP $\subseteq$ PSPACE.
2. IP[poly] $\subseteq$ AM[poly]
3. IP[k] $\subseteq$ AM[k]

1 IP and PSPACE

Here we will show the "easier" direction of the proof that IP = PSPACE. That is, the proof that IP $\subseteq$ PSPACE. For concreteness, we will use $c = 2/3$ and $s = 1/3$ for the completeness and soundness of IP respectively. The idea is that for a fixed verifier V for some language $A \in$ IP and some string w, we can compute in polynomial space whether or not there exists a prover P such that:

$$\Pr_R[verdict(P \leftrightarrow V) = 1] \geq 2/3$$

This probability is the definition of $x \in A$. This will be done by comupting the greatest probability of acceptance inductively over the rounds of message history. If this probability is greater than 2/3, then such a prover exists.

Theorem 1 IP $\subseteq$ PSPACE

Proof Let $Acc(w, q_1, q_2, ..., q_i, a_1, a_2, ..., a_i)$ be the acceptance probability of a message stream that has been specified up to the ith stage.

It is clear that we can compute $Acc(w, q_1, q_2, ..., q_n, a_1, a_2, ..., a_n)$ where the interaction has a total of n stages, as the function is completely specified.

Assuming that we have computed the probability of accepting from the $i + 1$ stage of the message stream, we can compute the probability of the ith stage as follows. Given all questions so far, we can compute the set S of all random strings R that generate the message stream up to that point. For each $R \in S$, we can figure out what q_{i+1} will be. So, we can compute:

$$\max_{a_{i+1}}[Acc(w, q_1, q_2, ..., q_{i+1}, a_1, a_2, ..., a_{i+1})]$$

Thus, we can compute $Acc(w)$ in polynomial space, and this is exactly $\Pr_R[verdict(P \leftrightarrow V) = 1]$. Therefore, $A \in$ PSPACE. ■

2 IP[poly] $\subseteq$ AM[poly]

IP corresponds to the case where the prover does not have access to the results of the random choices the verifier makes. AM is the case where the prover can access the random choices. The task is to show that making the choices private does not add any power.

Theorem 2 IP[poly] $\subseteq$ AM[poly]

Proof First, we will fix an IP verifier V and an input string w. We will consider an "interaction tree" for an interactive proof. This tree consists of nodes corresponding to the history of the interaction up to a given point and edges connect these nodes to nodes that represent immediate successors to this history. The leaves of this tree will be labelled accept or reject depending on whether the interaction corresponding to the path from the root of the tree to that leaf accepted or rejected w. We can assume without loss of generality that questions are bits. (A question can be converted into binary and then sent one bit at a time). Also, it can be assumed that each path through the tree corresponds to a unique random string. (It is possible to assure this by adding questions that depend specifically on the random string).

We will define N_σ to be the number of accepting leaves in a the subtree of the interaction tree rooted at σ. Let the root of the tree be denoted *start*. The goal is to find N_{start}.

The goal of the AM Verifier (Arthur) is to verify that N_{start} is at least $2k/3$, where k is the number of random strings. At a given node r with children r_0 and r_1 in the tree, the Prover (Merlin) will send Arthur M_r, M_{r_0}, and M_{r_1}. Arthur wants to verify that $M_r = N_r$, $M_{r_0} = N_{r_0}$, and $M_{r_1} = N_{r_1}$. Arthur does this by checking $M_r = M_{r_0} + M_{r_1}$ and recursively verifying M_{r_0} or M_{r_1}. It is clear that Arthur cannot verify both children of every node in a polynomial number of steps, so Arthur must only choose one path to explore. Arthur picks the node to explore as follows: pick node r_0 with probability $M_{r_0}/(M_{r_0} + M_{r_1})$, and pick node r_1 otherwise. The completeness and soundness claims that follow establish the validity of this method. ■

The completeness of this method is 1 because there is zero chance of picking a node with value 0.
For the soundness,

$$Pr[\text{accepting at a node } \sigma] \leq \frac{N_\sigma}{M_\sigma}$$

This can be proved inductively. If σ is a leaf, it clearly holds because $N_\sigma = 0$ or 1.
Assume that the claim holds for the children σ_0 and σ_1 of σ. Then,

$$Pr[\text{accepting at } \sigma] = \frac{M_{\sigma_0}}{M_{\sigma_0} + M_{\sigma_1}} Pr[\text{accepting at } \sigma_0] + \frac{M_{\sigma_1}}{M_{\sigma_0} + M_{\sigma_1}} Pr[\text{accepting at } \sigma_1]$$

By the inductive hypothesis,

$$\leq \frac{M_{\sigma_0}}{M_{\sigma_0} + M_{\sigma_1}} \frac{N_{\sigma_0}}{M_{\sigma_0}} + \frac{M_{\sigma_1}}{M_{\sigma_0} + M_{\sigma_1}} \frac{N_{\sigma_1}}{M_{\sigma_1}} = \frac{N_\sigma}{M_\sigma}$$

(The above theorem is due to [Goldwasser-Sipser] and [Furer-Goldreich-Mansour-Sipser-Zachos]. The proof is due to [Kilian].)

3 IP$[k] \subseteq$ AM$[k]$

First, we will introduce a protocol for approximate set size that will be used in the proof of IP$[k] \subseteq$ AM$[k]$.

The problem is as follows:
Suppose $S \subseteq \{0,1\}^n$ and has size either $|S| \geq \text{BIG} = 2^m$ or at most $\text{SMALL} = \frac{2^m}{100}$, where m is on the order of $\sqrt{n}$. Also, Arthur can test membership of S. The question is, can Merlin convince Arthur that S is BIG? The protocol for doing this is called the Goldwasser-Sipser protocol (GS):

- Merlin picks a hash function $h : \{0,1\}^n \to \{0,1\}^{m-4}$ and it sends to Arthur.
- Arthur pics $y \in \{0,1\}^{m-4}$ and sends it to Merlin.
- Merlin responds with $x \in S$ such that $h(x) = y$.

Soundness:
If $|S| \leq \frac{2^m}{100}$, then for any h, at most

$$\frac{2^m/100}{2^{m-4}} = \frac{16}{100} \leq \frac{1}{3}$$

of the y's will have an $x \in S$ such that $h(x) = y$.

Completeness (sketch):
The idea is that we expect 16 elements of S to map to a given y. Pairwise independence implies that any fixed y is in the range of an h with probability $9/10$. Markov's inequality implies that the number of y's covered is $\geq 2/3$ with probability $2/3$.

Theorem 3 IP[k] $\subseteq$ AM[k]

Proof We will only prove IP[1] $\subseteq$ AM[$O(1)$], but extension to arbitrary k follows similarly.
We will provide an $AM[O(1)]$ protocol to decide an aribtrary language in IP[1]. The protocol is as follows:

- Fix a verifier V with completeness $2/3$ and soundness $1/\text{poly}$, and an input w.
- Let $Q = \{1, ..., q_i, ...\}$ be the set of all possible questions and $A = \{1, ..., a_i, ..\}$ is the set of all possible answers.
- For all $q \in Q$ and $a \in A$, let S_q^a be the set of all random strings R such that $V(R, w) = q$ and $V(R, w, a) =$ accept. Let a_q^* be the answer that maximizes S_q^a.
- Let r be the length of random strings.
- So,

$$\sum_{q \in Q} \left| S_q^{a_q^*} \right| = (\text{probability of acceptance}) * 2^r$$

- Assume for simplicity, that $\left| S_q^{a_q^*} \right| = 0$ or 2^l or for every q. Now Arthur needs to be convinced that $\exists \frac{2}{3} * 2^{r-l}$ q's such that $|S_q| \geq 2^l$.
- $Q = Q_0 \bigcup Q_1 ... \bigcup Q_r$, where $Q_i = \{q | 2^i \leq |S_q| \leq 2^{i+1}\}$.
- Since,

$$\sum_{q \in Q} \left| S_q^{a_q^*} \right| \geq \sum_{i=1}^{r} 2^i |Q_i| \geq \frac{2^r}{3}$$

  Only the last inequality needs to be verified.
- It needs to be verified that $\exists i$ such that $2^i |Q_i| \geq \frac{2^r}{3r}$.
- Now run 2 GS protocols one after the other.
- Merlin will prove $|Q_i| \geq \frac{2^r}{3r * 2^i}$.
- Merlin sends h, Arthur queries with y and Merlin sends $q \in Q_i$ such that $h(q) = y$ (This is the first GS protocol).
- Arthur must now verify that $|S_q| \geq 2^l$. Run another GS protocol to achieve this.

Thus, only a constant number of rounds is needed to decide a problem in IP[1]. ■