COM S 6810 Theory of Computing March 5, 2009

Lecture 14: Interactive Proofs

Instructor: Rafael Pass *Scribe: J. Aaron Lenfestey*

1 Interactive Proofs

Definition 1 *(P, V) is an interactive proof if V is a probabilisitic polynomial time interactive Turing machine and P is a (deterministic) interactive Turing machine, and:*

- *Completeness:* $x \in L \iff Pr[\langle P, V\rangle(x) = 1] \geq 2/3$
- *Soundness:* $x \notin L \iff \forall P^*.Pr[\langle P^*, V\rangle(x) = 1] \leq 1/3$

By running the proof in parallel, we can replace the 2/3 and 1/3 in the definition above with $1-2^{-|x|}$ and $2^{-|x|}$. Notice that P is not a PPT, for if it were this class would reduce to BPP. We will also see that the 2/3 (or $1-2^{-|x|}$) can be replaced by 1 and the definition will remain equivalent, i.e., if a string is in the language, the prover can establish this fact with probability 1. When designing protocols it suffices to have the gap in probability between the two cases go as $1/p(x)$ for some polynomial, so that the machines can be replicated polynomially many times and still get an exponentially diminishing probability of failure.

2 Graph non-isomorphism

As an example, we consider the problem of *graph non-isomorphism.*

Definition 2

- $L_{GISO} = \{(G_1, G_2) \mid G_1 \sim G_2\}$
- $L_{GNISO} = \{(G_1, G_2) \mid |G_1| = |G_2|,\ G_1 \nsim G_2\}$

where G_1 and G_2 are graphs.

Then clearly L_{GISO} is in NP since one can provide the isomorphism as a certificate, and so L_{GNISO} is in coNP. It is believed that L_{GNISO} is one of the interesting problems between P and NP, for if it were NP-complete, then one can show that the entire polynomial hierarchy would collapse.

There is, however, an interactive proof that two graphs are not isomorphic. Consider the following protocol:

- $P \leftarrow V$: Randomly choose $i \in \{1, 2\}$ and a permutation $\pi \in S_n$. Send $H = \pi(G_j)$.
- $V \rightarrow P$: Find j s.t. $G_j \sim H$. Send j.
- V accepts iff $i = j$.

If the graphs G_1 and G_2 are not isomorphic, then $i = j$ always, so V will always accept (this problem is perfectly complete). If they are isomorphic, then $i = j$ half of the time, so V will accept with probability .5. Since the probability gap is constant, this procedure can be amplified to satisfy any of the definitions above.

3 Set cardinality

Let $S \subseteq \{0, 1\}^m$ be for which membership can be efficiently certified. Fix K and consider the promise problem:

- $|S| \geq K \rightarrow True$
- $|S| \leq K/2 \rightarrow False$

This problem can be decided by the following interactive proof:

- $P \leftarrow V$: Choose h to be a random hash function from $H_{m,k}$ and y to be a random string from $\{0, 1\}^k$, where k is chosen such that $2^{k-2} \leq K \leq 2^{k-1}$.
- $V \leftarrow P$: Find x such that $h(x) = y$ and $x \in S$. Send x along with a certificate that $x \in S$ (if no x exists, then send any x and a faulty certificate).
- V accepts if $h(x) = y$ and the certificate is correct.

Notice that by the choice of k: $1/4 \leq k/2^K \leq 1/2$. First we consider soundness. Suppose $|S| \leq K/2$, so that $|h(S)| \leq K/2$. Then

$$\begin{aligned} Pr[\text{V accepts}] &= Pr[\text{randomly chosen } y \in h(S)] && (1)\\ &= |h(S)|/2^k && (2)\\ &\leq K/2 \cdot 1/2^k && (3)\\ &= \left(\frac{1}{2}\right) \cdot \left(\frac{K}{2^k}\right) && (4) \end{aligned}$$

Next we consider completeness. Suppose $K \leq |S|$. For simplicity, assume that $|S| \leq 2^{k-1}$ (the problem will be even easier if $|S|$ is larger). Then,

$$\begin{aligned}
Pr[\exists x \in S \;.\; h(x) = y] &= Pr[\bigvee_{x \in S} h(x) = y] && (5)\\
&\geq \sum_{x \in S} Pr[h(x) = y] - \sum_{x < x'} Pr[h(x) = y \text{ and } h(x') = y] && (6)\\
&= \sum_{x} 2^{-k} - \sum_{x < x'} 2^{-2k} && (7)\\
&= |S|2^{-k} - \frac{|S|(|S|-1)}{2} 2^{-2k} && (8)\\
&\geq |S|/2^k \cdot (1 - |S|/2^{k+1}) && (9)\\
&\geq K/2^k \cdot (1 - 2^{k-1}/2^{k+1}) && (10)\\
&= \left(\frac{3}{4}\right) \cdot \left(\frac{K}{2^k}\right) && (11)
\end{aligned}$$

So the gap between the two probabilities is $1/4K/2^K > 1/16$, so this procedure can be amplified to satisfy the definitions in the first section.

4 Arthur-Merlin games

Definition 3 *IP is the class of languages with interactive proofs. IP[k] is the class of languages with k-round interactive proofs.*

Definition 4 *AM[k] is the class of languages with k-round interactive proofs where the messages sent by V are transcripts of its coin tosses. MA = AM[1].*

Theorem 1 (Goldwasser-Sipser) *IP[k] = AM[k+2]*

While we won't prove this theorem, we will show an example for the case of graph non-isomorphism. Consider $S = \{H, \pi \mid H \sim G_1 \text{ or } H \sim G_2 \text{ and } \pi(H) = H\}$. Then $|S| = n!$ if $G_1 \sim G_2$ and $|S| = 2n!$ otherwise. Applying the above procedure for $K = 2n!$ and having as our message the random bits that give rise to our choice of h and y, the graph non-isomorphism problem can be solved as a 1-round Arthur-Merlin game.