18.405J/6.841J Advanced Complexity Theory	February 17, 2000
Lecture 6	
Lecturer: Dan Spielman	*Scribe: Alexey Radul*

In the first half of this lecture, we shall prove one of the strange theorems that show up in Complexity. This is a theorem by Baker, Gill and Soloway that states that there is an oracle with respect to which P is equal to NP, and another with respect to which it is not. From this, we can conclude that P and NP will not be separated by any techniques that are conserved under the addition of oracles.

In the second half of the lecture, we shall explore randomization.

1 Baker-Gill-Soloway

Theorem 1 *(Baker-Gill-Soloway) There exist oracles A and B such that*

1. $\mathrm{P}^A = \mathrm{NP}^A$, *and*
2. $\mathrm{P}^B \neq \mathrm{NP}^B$.

The first part is straightforward.

Proof of (1): Let A be TQBF. As TQBF is PSPACE-complete, we have, using Savitch,

$$\mathrm{NP}^A \subseteq \mathrm{NPSPACE} = \mathrm{PSPACE} \subseteq \mathrm{P}^A \subseteq \mathrm{NP}^A.$$

■

This would work just as well with any other oracle of power sufficient to render the differences between P and NP insignificant.

Proof of (2): For this part of the theorem, we would be satisfied if we found an oracle B and a language L such that $L \in \mathrm{NP}^B$, but $L \notin \mathrm{P}^B$. To this end we make the

Definition 2 *For any language B, let*

$$L_B = \{w \mid \exists x \in B \text{ such that } |x| = |w|\}.$$

Claim 3 $L_B \in \mathrm{NP}^B$ *for all B.*

Proof of claim: Trivial. Existentially choose an x the same length as w, and see if it's in B. ■

Now, we wish to find a B such that $L_B \notin \mathrm{P}^B$, i.e. an B such that for any polynomial time oracle Turing machine $M^?$ there exists a word w in L_B such that either $M^B(w)$ accepts and $w \notin L_B$, or $M^B(w)$ rejects and $w \in L_B$.

In order to do this for all Turing machines, let us first examine how this is done for one. Let $M^?$ be an OTM. Let n be such that $M^?$ runs in time less than $2^n - 1$ on input 0^n. We shall now define an oracle B such that M^B will yield the wrong answer for the membership of 0^n in L_B. To accomplish this, simulate $M^?$ on 0^n, answering "no" to all of its questions. (As we do this, we note down the words $M^?$ asks, and say that they are not in B.) If $M^?$ accepts despite our negative answers, set $B = \emptyset$. Then $0^n \notin L_B$, but $M^B(0^n)$ accepts. If $M^?$ should reject 0^n, then let w be a word of length n that $M^?$ didn't ask in its computation. Let $B = \{w\}$. Now $0^n \in L_B$, but $M^B(0^n)$ rejects. Either way, $L(M^B) \neq L_B$.

We have found a way to construct an oracle so as to confuse one Turing machine. Now we need a way to use this to confuse them all. Let us begin that process by enumerating them. Let $\{M_i^?\}$ be all polynomial time oracle Turing machines. We build up $\{B_i\}$ with the condition that for all $j \leq i, L(M_j^{B_i}) \neq L_{B_i}$, and $B_{i-1} \subseteq B_i$. Given such a collection, if we let B be the union of the $\{B_i\}$, B will be the desired oracle.

How do we do this? By induction on i, of course. For $i = 1$ (base case), we can do what we have done for one Turing machine, above. Now, suppose we have, for some particular i, the following:

$$\forall j < i, \text{ either } M_j^{B_{i-1}}(0^{n_j}) \text{ accepts, while } 0^{n_j} \notin L_{B_{i-1}}, \text{ or } M_j^{B_{i-1}}(0^{n_j}) \text{ rejects, while } 0^{n_j} \in L_{B_{i-1}}.$$

Then let us define n_i by

1. $M_j^?(0^{n_i})$ runs in time $< 2^{n_i} - 1$, and

2. $\forall j < i, M_j^{B_{i-1}}(0^{n_j})$ did not query any strings of length n_i or greater.

Now repeat the construction as for just a single OTM — simulate $M^{B_{i-1}}$ on 0_i^n, answering "no" to all questions whose answers are not already decided by B_{i-1}. If $M^{B_{i-1}}(0_i^n)$ accepts, let $B_i = B_{i-1}$. If $M^{B_{i-1}}(0_i^n)$ rejects, let w be a word of length n_i that $M^{B_{i-1}}(0_i^n)$ did not ask about, and let $B_i = B_{i-1} \cup w$. This will preserve the desired properties of the collection $\{B_i\}$, allowing us to define all of them by induction, and thereby to create B. ■

2 Randomizations

Given polynomials $p_1, p_2, \ldots, p_k$, $q_1, q_2, \ldots, q_k$, and $r_1, r_2, \ldots, r_k$, in variables $x_1, x_2, \ldots, x_n$, we want to verify (in polynomial time, of course) the statement

$$\prod_{i=1}^{k} p_i(x_1, \ldots, x_n) + \prod_{i=1}^{k} q_i(x_1, \ldots, x_n) - \prod_{i=1}^{k} r_i(x_1, \ldots, x_n) \equiv 0.$$

The idea we use to do this is that if the identity is false, choosing a random point is very unlikely to give us a root. Therefore, if we choose a random point, the truth value of the identity there will be a fairly good estimate of the truth of the identity everywhere.

To make formal use of this, we will first need

Lemma 4 *(Schwartz) If $P(x_1, \ldots, x_n)$ is a degree d polynomial over a field $\mathcal{F}$ and $S \subseteq \mathcal{F}$, then*

$$\Pr_{x_1, \ldots, x_n \in S^n} [P(x_1, \ldots, x_n) = 0] \leq \frac{nd}{|S|}.$$

Proof of Lemma By induction on n. The base case is trivial: a non-zero, degree d univariate polynomial has no more than d roots. Now we proceed by induction. Write

$$P(x_1, \ldots x_n) = \sum_{i=0}^{d_1} x_1^i P_i(x_2, \ldots x_n),$$

where d_1 is the degree of P in x_1, and $P_{d_1} \not\equiv 0$. By the induction hypothesis,

$$\Pr_{x_2, \ldots, x_n \in S^{n-1}} [P_{d_1}(x_2, \ldots, x_n) = 0] \leq \frac{(n-1)d}{|S|}.$$

Moreover, for any particular $(x_2, \ldots, x_n)$ for which $P_{d_1}(x_2, \ldots, x_n) \neq 0$,

$$\Pr_{x_1 \in S}[P(x_1, \ldots, x_n) = 0] \leq \frac{nd}{|S|}.$$

Summing, we win. ■

With this machinery, we can now consider polynomial identities. Formally, we define the language

$$\mathrm{POLY-IDENT} = \{< P(y_1, \ldots, y_k), q_1(x_1, \ldots, x_m), \ldots, q_k(x_1, \ldots, x_m) > : P(q_1, \ldots, q_k) \equiv 0\}.$$

The statement we would like to make about it is the

Theorem 5

$$\mathrm{POLY-IDENT} \in \mathrm{P/poly}.$$

Proof Let $S = \{1, \ldots, 5^n\}$. Then our theorem is equivalent to the

Claim 6

$$\Pr_{x_1, \ldots, x_m \in S^m}\left[\begin{array}{l} \exists w, |w| = n \text{ s.t. } w \notin \mathrm{POLY-IDENT} \\ \text{yet the polynomial corresponding to } w \text{ is zero at } (x_1, \ldots, x_m) \end{array}\right] < \frac{1}{2}.$$

Why is this enough for our theorem? Well, if this claim holds, then there must exist a point $x_1, \ldots, x_m \in S^m$ which can test every polynomial identity of length n. Now we only need
Proof of claim: There are only 2^n polynomial identities of length n. The maximum degree of each is no more than 2^n. So on any particular identity, if the polynomial of such an identity is not everywhere zero, the probability that a random point will guide us astray is, by Schwartz, less than $\frac{n2^n}{5^n}$. Therefore, summing, the probability of a random point of leading us astray at all is no more than $\frac{2^n n 2^n}{5^n} < \frac{1}{2}$ for large enough n. ■

That proves the claim, and with it, the theorem. ■