18.405J/6.841J Advanced Complexity Theory March 2, 1999

Lecture 7

Lecturer: Dan Spielman *Scribe: Eric Kuo*

1 The Baker-Gill-Solovay Theorem

Theorem 1 (Baker-Gill-Solovay) *There exist oracles A and B for which $\mathrm{P}^A = \mathrm{NP}^A$ and $\mathrm{P}^B \neq \mathrm{NP}^B$.*

Proof: (A) Let $A = TQBF$. Clearly, $\mathrm{P}^A \subseteq \mathrm{NP}^A$ for any oracle A. For the other direction, $\mathrm{NP}^{TQBF} \subseteq \mathrm{NPSPACE}^{TQBF} \subseteq \mathrm{NPSPACE}$ since we can compute instances of $TQBF$ in polynomial space rather than asking the oracle. From Savitch's theorem, NPSPACE $\subseteq$ PSPACE. And finally, since $TQBF$ is PSPACE-complete, we have PSPACE $\subseteq \mathrm{P}^{TQBF}$. Hence $\mathrm{P}^{TQBF} = \mathrm{NP}^{TQBF}$.

(B) We need $L \in \mathrm{NP}^B$, but $L \notin \mathrm{P}^B$. We define L_B as follows:

$$L_B = \{w | \exists x \in B : |x| = |w|\}$$

In other words, L_B is the language of words which are the same length of another word in B. We see that $L_B \in \mathrm{NP}^B$ because we can nondeterministically guess a word x such that $|x| = |w|$ and then check if it is in B.

We want to show that there is a B for which $L_B \notin \mathrm{P}^B$. We first show how to construct an oracle to defeat one specific oracle TM $M^?$ (i.e. $L(M^?) \neq L_B$). First, find some n such that on input 0^n, $M^?$ runs in fewer than $2^n - 1$ steps. Simulate $M^?$ on input 0^n, and answer "no" to all queries. If $M^?$ ends up accepting 0^n, simply let B be the empty language. That means L_B is also empty, but the language of $M^?$ is not empty. On the other hand, if $M^?$ rejects 0^n, then let x be the lexically least word of length n for which $M^?$ did not query the oracle. Such a word must exist since $M^?$ ran in under $2^n - 1$ steps. So set $B = \{x\}$. That makes $0^n \in L_B$, so once again $L(M^?) \neq L_B$.

Now we need to show that there is a B such that for all polynomial time oracle TMs $M^?$, $L(M^B) \neq L_B$. First we let M_1, M_2, etc. be an enumeration of polynomial time oracle TMs. We use the following algorithm:

1. First let $B_0 = \emptyset$.
2. For $i = 1, 2, 3, \ldots$ do
   (a) Choose n_i such that $\forall j < i$, $M_j^{B_{i-1}}$ does not ask about any strings of length n_i on input 0^{n_j}. In addition, $M_i^{B_{i-1}}$ should run for at most $2^{n_i} - 1$ steps on input 0^{n_i}.
   (b) Simulate M_i with oracle B_{i-1} on input 0^{n_i}. (At this point, B_i has no strings of length n_i.)
   (c) If $M_i^{B_{i-1}}$ accepts, set $B_i = B_{i-1}$. (So $M_i^{B_i}$ has accepted 0^{n_i}, but $0^{n_i} \notin L_{B_i}$).
   (d) Else if $M_i^{B_{i-1}}$ rejects, then find w such that $|w| = n_i$ and $M_i^{B_{i-1}}$ on input 0^{n_i} did not ask w. Then set
   $$B_i = B_{i-1} \cup \{w\}.$$
   (So we have $M_i^{B_{i-1}}$ rejecting 0^{n_i}, but $0^{n_i} \in L_B$.)
3. Set $B = \bigcup_{i=0}^{\infty} B_i$.

Note that M_i^B is the same as $M_i^{B_i}$ for all i. $\square$

2 $\mathrm{NP}^B \neq \mathrm{coNP}^B$

Theorem 2 *There exists an oracle B such that* $\mathrm{NP}^B \neq \mathrm{coNP}^B$.

Proof: This proof is almost exactly like the previous one, except for a few highlighted modifications.

We need to show that there is a B such that for all *coNP* oracle TMs $M^?$, $L(M^B) \neq L_B$. First we let M_1, M_2, etc. be an enumeration of coNP oracle TMs. We use the following algorithm:

1. First let $B_0 = \emptyset$.
2. For $i = 1, 2, 3, \ldots$ do
   (a) Choose n_i such that $\forall j < i$, $M_j^{B_{i-1}}$ does not ask about any strings of length n_i on input 0^{n_j} *on any computation path.* In addition, $M_i^{B_{i-1}}$ should run for at most $2^{n_i} - 1$ steps *on every path of computation* on input 0^{n_i}.
   (b) Simulate M_i with oracle B_{i-1} on input 0^{n_i}. (At this point, B_i has no strings of length n_i.)
   (c) If $M_i^{B_{i-1}}$ accepts, set $B_i = B_{i-1}$. (So $M_i^{B_i}$ has accepted 0^{n_i}, but $0^{n_i} \notin L_{B_i}$.
   (d) Else if $M_i^{B_{i-1}}$ rejects, then *choose some rejecting computation path,* and find w such that $|w| = n_i$ and $M_i^{B_{i-1}}$ on input 0^{n_i} did not ask w *on that path.* Then set
   $$B_i = B_{i-1} \cup \{w\}.$$
   (So we have $M_i^{B_{i-1}}$ rejecting 0^{n_i} on that path, but $0^{n_i} \in L_B$.)
3. Set $B = \bigcup_{i=0}^{\infty} B_i$.

Note that M_i^B is the same as $M_i^{B_i}$ for all i. $\Box$

3 Checking Polynomial Identities

In this sectoin, we consider the problem of checking whether the sum of two products of polynomials is identical to the product of another set of polynomials. Let $x_1, x_2, \ldots, x_n$ be variables, and $p_1, p_2, \ldots, p_k, q_1, q_2, \ldots, q_k, r_1, r_2, \ldots, r_k$ be polynomials on those variables. The problem is to determine whether

$$\prod_{i=1}^{k} p_i(x_1, x_2, \ldots, x_n) + \prod_{i=1}^{k} q_i(x_1, x_2, \ldots, x_n) - \prod_{i=1}^{k} r_i(x_1, x_2, \ldots, x_n) \equiv 0$$

To help us, we introduce Schwartz's Lemma.

Lemma 3 (Schwartz's Lemma) *Let $P(x_1, x_2, \ldots, x_n)$ be a polynomial of degree d. Then if $P \not\equiv 0$, then*

$$\Pr_{x_1, x_2, \ldots x_n \in S}[P(x_1, x_2, \ldots, x_n) = 0] \leq \frac{dn}{|S|}.$$

Proof: We use induction on n.

Base case $(n = 1)$: If $P \not\equiv 0$, then there are at most d zeroes in p. At most $d = d \cdot 1$ of them are in S.

Inductive step: Write

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

By hypothesis,

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

If $P_{d_1}(x_2, \ldots, x_n) \ne 0$, then $\Pr_{x_1}[P(x_1, \ldots, x_n)] \le \frac{d_1}{|S|}$. $\square$