Advanced Theory of Computation
Takehome Exam Answersheet

Behdad Esfahbod

June 21, 2003

Problem 1

The proof for the first two parts is in [1].

For the last part, I cannot get the point, as the language $L = \{p^n q^n | n \geq 0\}$ is in $Time(n)$: A DTM with an input tape and a work tape, counts up the 0s, then counts down the 1s, and accepts if the result is zero. Counting from one to n takes $2n$ steps, with an amortized calculation, so $L \in Time(n)$.

Problem 2

I wonder if I have understood the problem well or not. From the r.e.-completeness of the problems, I just use the fact that they are infinite.

Consider two arbitrary such sets x and y. As these sets are recursive enumerable, there exist DTMs M_x and M_y which are enumerators for x and y respectively. So we define a bijection function f from x to y, so that f on input $a \in D_x$, simulates M_x, counting the outputs until it outputs a, say as its nth output. Then f simulates M_y until it outputs n strings, and grabs the nth output and returns it.

It is clear that the bijection is decidable. It also works for NP. It has nothing to do with the fact that it is not known if all NP-complete problems are polynomially isomorphic or not.

Problem 3

Assuming that by $DEXT$, you mean the same as $EXP = \cup_{c>0} DTIME(2^{cn})$. Quoting p. 31 of [2]:

> **Theorem 1** *$EXP \neq PSPACE$.*
>
> ***Proof.*** *From the Time Hierarchy Theorem, we get $EXP \subseteq DTIME(2^{n^{3/2}}) \subsetneq DTIME(2^{n^2})$. Thus, it suffices to show that if $PSPACE = EXP$ then $DTIME(2^{n^2}) \subseteq PSPACE$.*
>
> *Assume that $L \in DTIME(2^{n^2})$. Let \$ be a symbol not used in L, and let $L' = \{x\$^t : x \in L, |x| + t = |x|^2\}$. Clearly, $L' \in DTIME(2^n)$. So, by the assumption that $PSPACE = EXP$, we have $L' \in PSPACE$; that is, there exists an integer $k > 0$ such that $L' \in DSPACE(n^k)$. Let M be a DTM accepting L' with the space bound n^k. We can construct a new DTM M' that operates as follows.*
>
> > *On input x, M' copies x into a tape and the adds $|x|^2 - |x|$ \$'s.*
> > *Then, M' simulates M on $x\$^{|x|^2-|x|}$.*
>
> *Clearly, $L(M') = L$. Note that M' uses space n^{2k}, and so $L \in PSPACE$. Therefore, $DTIME(2^{n^2}) \subseteq PSPACE$, and the theorem is proven.* □

For the second part, we observe that the same technique does not work, as in $EXPSPACE$, n cannot be replace by n^2, or the problem would not remain inside $EXPSPACE$.

From the discussion on page 258 of the same book, it seems that it is not still known if $EXP \neq EXPSPACE$. By the way, it can be proved that if $EXP \neq EXPSPACE$ then there exists a tally set in $PSPACE - P$.

Problem 4

Definition 1 *A set $A \subset \Sigma^*$ is self-reducible if there exist a partial oredering $\prec$ on Σ^* and two polynomial-time TMs M_1 and M_2, such that the following hold:*

- *The question of whether $x \prec y$ can be decided in $p(|x| + |y)$ where p is a polynomial,*
- *For all inputs x, $M_1(x)$ generates a list $\langle y_1, y_2, \ldots, y_k \rangle$ with the property that $y_j \prec x$ for all $1 \leq j \leq k$ (if x is a minimal element of $\prec$, then $M_1(x)$ generrates an empty list); and*
- *$x \in A$ is and only if M_2 accepts $\langle x, \langle \chi_A(y1), \chi_A(y2), \ldots, \chi_A(y_k) \rangle \rangle$, where $\langle y1, \ldots, y_k \rangle$ is the list generated by m_1, and $\chi_A(y) = 1$ if $y \in A$ and $\chi_A(y) = 0$ if $y \notin B$.*

The best thing with this definition is that it tries to formalize our intuitive notion of self-reducibility. It says that a set problem A is self-reducible if given any case of the problem, if the solution to some smaller cases is known, this case can be solved easily. Smaller is defined as a partial order. Infact there is no constraint on the size of the smaller cases, they may be exponentially large, but what is important is that they are predecessors of the input, and the reducibility can cause no cycles.

The cons are that there is no restriction on the size of the reduced cases, and none on the length of the chain of reduced cases that one most solve to solve the given cases. This makes the definition of no use in complexity theory, as forces no constraint on the time and space of the problems in this class. A restricted case of this definition which bound the size of reduced parts and the length of reduced chains to a polynomial is known is $tt - self - reducibility$.

Problem 5

There is a definition known as *weak* $EXPTIME$ hierarchy[3], this definition is known to collapse:

$$\begin{array}{l} EXPH = \cup_i \Sigma_i^{exp}, \text{ where:} \\ \Sigma_0^{exp} = EXPTIME \\ \Sigma_i^{exp} = NEXPTIME(\Sigma_i^p), \text{ for } i > 0 \\ \Pi_i^{exp} = co - \Sigma_i^{exp} \end{array}$$

but another way is to define the whole hierarchy recursively:

$$\begin{array}{l} EXPH = \cup_i \Sigma_i^{exp}, \text{ where:} \\ \Sigma_0^{exp} = EXPTIME \\ \Sigma_i^{exp} = NEXPTIME(\Sigma_{i-1}^{exp}), \text{ for } i > 0 \\ \Pi_i^{exp} = co - \Sigma_i^{exp} \\ \Delta_i^{exp} = EXPTIME(\Sigma_{i-1}^{exp}) \end{array}$$

This definition, known as the *strong* exponential hierarchy, is known to collapse[4], but I could not find the article. The following discussion is based on the later definition.

These propositions are simple to show:

- $A \in NEXPTIME(A)$,
- $A \in EXPTIME(B) \Rightarrow A \in NEXPTIME(B)$,
- $EXPTIME^A = EXPTIME^{\bar{A}}$.

The following is immediately followed from above:

$$\Sigma_k^{exp} \cup \Pi_k^{exp} \subseteq \Delta_{k+1}^{exp} \subseteq \Sigma_{k+1}^{exp} \cup \Pi_{k+1}^{exp}$$

Lemma 1 *Let $k \geq 0$.*

- *If $A, B \in \Sigma_k^{exp}$, then $A \cup B, A \cap B \in \Sigma_k^{exp}$, and $\bar{A} \in \Pi_k^{exp}$.*
- *If $A, B \in \Pi_k^{exp}$, then $A \cup B, A \cap B \in \Pi_k^{exp}$, and $\bar{A} \in \Sigma_k^{exp}$.*
- *If $A, B \in \Delta_k^{exp}$, then $A \cup B, A \cap B, \bar{A} \in \Delta_k^{exp}$.*

Proof. *Same as the proof for the similar lemma for PH in page 80 of [2].* □

Problem 6

The class H_n defines an upper bound on the gap between Σ_n and Σ_{n+1}, and L_n defines a lower bound on that.

$$H_n = \{A \in NP | \Delta_{n+1} \subseteq \Delta_n(A)\}$$
$$L_n = \{A \in NP | \Delta_n(A) \subseteq \Delta_n\}$$

Problem 9

Definition 2 *For any function $t(n)$, we define $RSPACE(t(n))$ as the set of languages that there exists a PTM whose space is bounded by $t(n)$ and accept probability greater than $\frac{1}{2}$. Then*

$$PPSPACE = \bigcup_{k>0} RSPACE(n^k).$$

It can be shown that $PPSPACE$ is the same class as $PSPACE$. The idea is the same as when showing $DPSPACE = NPSPACE$: Given a PTM M for a problem in $PPSPACE$, one can create a DTM M', which simulates M for each sequence of random bits, and computes the accept probability, and accepts iff the accept probability is greater than $\frac{1}{2}$. The new machine M' still needs polynomial space.

References

[1] John Hopcroft and Jeffry D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.

[2] Ding-Zhu Du and Ker-I Ko. *Theory of Computational Complexity*. John Wiley & Sons, Inc., 2000.

[3] Dawar, Gottlob, and Hella. Capturing relativized complexity classes without order. *ZEITSCHR: Mathematical Logic Quarterly (formerly Zeitschrift fuer Mathematische Logik und Grundlagen der Mathematik)*, 44, 1998.

[4] L. A. Hemachandra. The strong exponential hierarchy collapses. *Proceedings 19-th Annual ACM Symposium on Theory of Computing*, 1987.