CMPT 710 - Complexity Theory: Lecture 4

Valentine Kabanets

September 11, 2003

1 Hierarchy Theorems

In this course, we will always use *proper* complexity functions $f(n)$. A function $f(n)$ is called *proper* if there is a TM M that, on input 1^n, outputs exactly $f(n)$ symbols and runs in time $O(f(n)+n)$ and space $O(f(n))$. The usual functions like $\log n$, $\sqrt{n}$, n^2, 2^n, $n!$ are proper. Also, if f and g are proper, then so are $f+g$, fg, $f(g)$, f^g, 2^g.

Lemma 1. *Let $f(n)$ be any proper complexity function. The language*

$$Halt_f = \{(M,x) \mid \text{ TM } M \text{ accepts } x \text{ in at most } f(|x|) \text{ steps}\}$$

is decidable in time $g(n) = (f(n))^3$.

Proof. We can construct a universal TM H with 3 tapes that does the following. Given an input (M,x), our TM H converts a (possibly multi-tape) TM M to an equivalent one-tape TM M'. If M accepts x in at most $f(|x|)$ steps, then the new TM M' will accept x in at most $(f(|x|))^2$ steps.

Next, H will simulate M' on x for at most $(f(|x|))^2$ steps. Each step of M' can be simulated by H in at most $|M'|$ steps (i.e., the size of the description of the TM M', which is some constant dependent on M'); this constant is at most $f(|x|)$. Thus, our entire simulation will take at most $(f(|x|))^3$ steps. (Note that we needed $f(n)$ to be a proper function in order to be able to simulate M for at most $f(n)$ steps!) □

Consider the language

$$Diag_f = \{M \mid \text{ TM } M \text{ does not accept input } M \text{ in at most } f(|M|) \text{ steps}\}$$

By Lemma 1, the language $Diag_f$ is in $\mathsf{Time}(g(2n))$.

Theorem 2. *$Diag_f$ is not in $\mathsf{Time}(f(n))$.*

Proof. The proof is virtually the same as the one showing that the language Diag (defined earlier) is undecidable. The details are left as an exercise. □

Hence, we have

Theorem 3 (Time Hierarchy). *For every proper complexity function $f(n) \geq n$,*

$$\mathsf{Time}(f(n)) \subsetneq \mathsf{Time}((f(2n))^3).$$

Similarly, we can prove

Theorem 4 (Space Hierarchy). *For every proper complexity function $f(n) \geq \log n$,*

$$\mathsf{Space}(f(n)) \subsetneq \mathsf{Time}((f(n)) \log f(n)).$$

2 Robust Time and Space Classes

"Robust" (intuitive notion): no reasonable changes to the model of computation should change the class; capable of classifying interesting problems.

Examples of robust classes:

1. L (contains Formula Value, integer multiplication and division),
2. P (contains circuit value, linear programming, max-flow),
3. PSPACE (contains 2-person games),
4. EXP (contains all of PSPACE).

3 Relationships among complexity classes

Theorem 5. $\mathsf{L} \subseteq \mathsf{P} \subseteq \mathsf{PSPACE} \subseteq \mathsf{EXP}$.

The middle inclusion ($\mathsf{P} \subseteq \mathsf{PSPACE}$) is trivial: in polynomial time, no TM can touch more than polynomial number of tape cells.

To prove the other two inclusions, we need the notion of a *configuration* of a TM. If at a given moment in time, a TM is in state q, and its tape contains symbols $\sigma_1\sigma_2 \ldots \sigma_i\sigma_{i+1} \ldots \sigma_m$, and its tape head is scanning the symbol σ_{i+1}, then the configuration of the TM at this moment in time is

$$C = \sigma_1\sigma_2 \ldots \sigma_i q \sigma_{i+1} \ldots \sigma_m.$$

(Note that the state name is immediately to the left of the symbol currently scanned by the TM.)

Start configuration (for a one-tape TM) on input x is $q_{start}x_1x_2 \ldots x_n$.

We say that a configuration C *yields* C' in one step if a TM in configuration C goes to configuration C' in one step, using its transition function.

We now define the *configuration graph* of a given TM M on input x:

- Nodes = configurations of M,
- Edges = $\{(C, C') \mid C \text{ yields } C' \text{ in one step}\}$

Easy Fact: The number of configurations of a 2-tape TM (with one read-only input tape and one work tape) is at most:

- n (input-tape head positions)
- $*f(n)$ (work-tape head positions)
- $*|Q|$ (state)
- $*|\Sigma|^{f(n)}$ (work-tape contents).

Note: the contents of the read-only input tape is not part of the configuration.

Proof of Theorem 5. For $f(n) = c \log n$, the number of configurations is at most

$$n * c \log n * c_0 * c_1^{c \log n} \leq n^{c_2},$$

a polynomial.

For $f(n) = n^c$, the number of configurations is at most

$$n * n^c * c_0 * c_1^{n^c} \leq 2^{n^{c_2}},$$

an exponential function.

To determine if a TM M accepts x,

1. construct the configuration graph of M on x;
2. check if q_{accept}-configuration is reachable from the start configuration (using, e.g., DFS); this can be done in time polynomial in the size of the graph.

Hence, for $f(n) = c \log n$, we need $\mathsf{poly}(n)$ time; and for $f(n) = n^c$, we need $\mathsf{poly}(2^{n^{c_2}})$ time. □

Theorem 6. $\mathsf{L} \subsetneq \mathsf{PSPACE}$ *and* $\mathsf{P} \subsetneq \mathsf{EXP}$.

Proof. We prove the first proper inclusion only; the second can be proved similarly. Note that $L \subseteq \mathsf{Space}(n)$ trivially. Now, by the Space Hierarchy Theorem, $\mathsf{Space}(n) \subsetneq \mathsf{Space}(n \log n)$. Finally, $\mathsf{Space}(n \log n) \subseteq \mathsf{PSPACE}$ trivially. □

As an immediate consequence, we obtain the following

Theorem 7. *Among the inclusions* $\mathsf{L} \subseteq \mathsf{P} \subseteq \mathsf{PSPACE}$ *at least one inclusion must be proper. Among the inclusions* $\mathsf{P} \subseteq \mathsf{PSPACE} \subseteq \mathsf{EXP}$ *at least one inclusion must be proper.*

Proof. Again we prove just the first part. Suppose that all inclusions are equalities. Then $\mathsf{L} = \mathsf{P} = \mathsf{PSPACE}$, contradicting Theorem 6. □

Major Open Questions:

- $\mathsf{L} \stackrel{?}{=} \mathsf{P}$
- $\mathsf{P} \stackrel{?}{=} \mathsf{PSPACE}$
- $\mathsf{PSPACE} \stackrel{?}{=} \mathsf{EXP}$

By Theorem 7, at least one of these three questions must have a negative answer. The bizarre fact is: *we do not know which one!* (even though we suspect that all of these questions have negative answers.)