Graduate Computational Complexity Theory **CMU 15-855*, Fall 2017**

Homework 2
Due: 10:00am, Tuesday September 19

---

1. **(Almost-Everywhere Time Hierarchy Theorems.)**

   (a) The standard (Deterministic) Time Hierarchy Theorem we considered in class shows that if $T(n)$ is time-constructible and $t(n)\log t(n) = o(T(n))$ then there is a language $L \in \mathsf{TIME}(T(n))$ such that $L \notin \mathsf{TIME}(t(n))$. If we unpack the definition of $L \notin \mathsf{TIME}(t(n))$, it means this:

   $$\text{for any TM } M \text{ with running time } O(t(n)), \quad \exists x \; M(x) \neq L(x), \tag{1}$$

   Here we're abusing notation a little by writing $L(x)$ for the answer to the question $x \overset{?}{\in} L$. Actually, if you inspect the proof of the theorem, it showed something stronger:

   $$\text{for any TM } M \text{ with running time } O(t(n)), \quad \exists^{\infty} x \; M(x) \neq L(x), \tag{2}$$

   where the symbol $\exists^{\infty}$ means "there exists infinitely many" (or synonymously, "infinitely often").[1] Show that even if you didn't remember the proof of the THT, you could deduce (2) in a purely "black-box" fashion from (1). (You may assume that $t(n) \geq n$.)

   (b) Similarly show that you can deduce the following in a purely "black-box" fashion:

   $$\text{for any } M \text{ deciding } L\text{, and any } C, \quad \exists^{\infty} x \; M(x) \text{ takes } > Ct(|x|) \text{ time steps.} \tag{3}$$

   (c) Arguably even (2) is pretty weak. Here is an upgraded statement that one might desire:

   $$\text{for any TM } M \text{ with running time } O(t(n)), \quad \forall^{\infty} x \; M(x) \neq L(x), \tag{4}$$

   where the symbol "$\forall^{\infty}$ means "for all but finitely many x" (or synonymously, "almost everywhere"). Show that (4) is provably too much to hope for.

   (d) Here is an upgrade of (3):

   $$\text{for any } M \text{ deciding } L\text{, and any } C, \quad \forall^{\infty} x \; M(x) \text{ takes } > Ct(|x|) \text{ time steps.} \tag{5}$$

   This *can* be achieved, but the proof is much harder (it took 13 years after the original THT). Short of that, you are asked to prove a weaker statement in this problem.

   Say that a language A is in the class i.o.-P if there is a polynomial-time Turing Machine M that computes A correctly for infinitely many input lengths (i.e., $A \cap \{0,1\}^n = L(M) \cap \{0,1\}^n$ for infinitely many n). Prove that there is a language $L \in \mathsf{EXP}$ that is not in i.o.-P.

2. **(Superiority.)** Do Exercise 3.4 in Arora–Barak.[2]

---

[1] In fact, the proof kind of needed to show this, to take care of the fact that you need to diagonalize against all $O(t(n))$ running times.

[2] Of course, you may assume $n^{1.1}$ is time-constructible.

3. **(Awesome circuit lower bounds from depth-3 circuit lower bounds.)** Suppose $f : \{0,1\}^n \to \{0,1\}$ can be computed by a circuit of logarithmic depth $c_1 \log n$ and linear size $c_2 n$. The goal of this problem is to show that f can also be computed by a depth-3 circuit of subexponential size, namely $2^{O(n/\log\log n)}$.[3] In fact, you should be able to make the depth-3 circuit an OR of CNFs, where each CNF has at most $2^{O(n^{.01})}$ clauses, and where the circuit has the additional property that on all inputs, at most one of the CNFs outputs True.

   By the way, this result shows that to get a superlinear circuit lower bound for log-depth circuits (which would be awesome), "all" you have to do is get an essentially-fully-exponential circuit lower bound for depth-3 circuits. Later in the class we will show that depth-3 circuits require size $2^{\Omega(\sqrt{n})}$ to compute the Parity function $f(x) = \sum_i x_i \bmod 2$. Close, but no cigar.

   (a) In the log-depth, linear-size circuit for f, show that it is possible to "cut" $O(n/\log\log n)$ wires, leaving a collection of subcircuits each of which depends on at most $O(n^{.01})$ inputs. (Hint: an earlier homework problem.)

   (b) Complete the proof — i.e., the construction of the depth-3 circuit for f. (Hint: consider "enumerating" all possible values for the cut wires.)

[3] As per usual conventions, in the log-depth linear-size circuit, we assume the allowed gates are NOT and fan-in-2 AND/OR, whereas in the depth-3 circuit we assume the allowed gates are NOT and unbounded-fan-in AND/OR. Also, NOT gates are not counted toward depth in constant-depth circuits.