6.045 Pset 5: NP-Completeness and More

Assigned: Wednesday, April 6, 2011
Due: Thursday, April 14, 2011

To facilitate grading, remember to solve each problem on a separate sheet of paper! Also remember to write your name on each sheet.

1. Let $EXACT4SAT$ be the following problem:

   - Given a Boolean formula φ, consisting of an AND of clauses involving exactly 4 distinct literals each (such as $(x_2 \vee \neg x_3 \vee \neg x_5 \vee x_6)$), decide whether φ is satisfiable.

   Show that $EXACT4SAT$ is NP-complete. You can use the fact, which we proved in class, that $3SAT$ is NP-complete.

2. Let $DOUBLESAT$ be the following problem:

   - Given as input a Boolean circuit C, decide whether there are *two or more* input assignments $x \in \{0,1\}^n$ such that $C(x) = 1$.

   Show that $DOUBLESAT$ is NP-complete.

3. Let G be an undirected graph with n vertices. Then a *Hamilton path* is a simple path in G that visits each vertex once (i.e., has n vertices and $n-1$ edges), while a *Hamilton cycle* is a simple cycle in G that visits each vertex once (i.e., has n vertices and n edges). Let $HAMPATH$ and $HAMCYCLE$ be the problems of deciding whether G has a Hamilton path and Hamilton cycle respectively, given G as input.

   (a) Show that if G has a Hamilton cycle, then G also has a Hamilton path.
   (b) Give an example of a graph G that has a Hamilton path but no Hamilton cycle.
   (c) Give a polynomial-time reduction from $HAMCYCLE$ to $HAMPATH$.
   (d) Give a polynomial-time reduction from $HAMPATH$ to $HAMCYCLE$.

   (Together, parts c and d imply that $HAMPATH$ and $HAMCYCLE$ are *polynomial-time equivalent*. Since $HAMCYCLE$ is a famous NP-complete problem, this immediately implies that $HAMPATH$ is NP-complete as well.)

4. In the *quadratic programming* ($QUADPROG$) problem, the input is a system of equalities and inequalities, each involving polynomials of degree at most 2 (with integer coefficients) in n real variables $x_1, \ldots, x_n$. The problem is to decide whether there exists an assignment to $x_1, \ldots, x_n$ that satisfies all the constraints simultaneously. As an example, the system

$$\begin{aligned} x_1 + x_2 &\leq 1 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \\ 4x_1x_2 &\geq 1 \end{aligned}$$

can be satisfied by setting $x_1 = x_2 = 1/2$, but if we replaced the last inequality by $x_1 x_2 \geq 1$, then the system would be unsatisfiable.

(a) Show that $QUADPROG$ is NP-hard, by reduction from any problem that was already proved NP-hard in class. [*Hint:* $3COLORING$ would be a good choice.]

(b) What is a difficulty in showing that $QUADPROG \in \mathsf{NP}$ (the other condition needed for $QUADPROG$ to be NP-complete)?

5. Suppose problem X is proved NP-complete, by a polynomial-time reduction that maps size-n instances of SAT to size-n^3 instances of problem X. And suppose that someday, some genius manages to prove that SAT requires $\Omega(c^n)$ time, for some constant $c > 1$. Then what can you conclude about the time complexity of problem X?

6. Recall that $\mathsf{EXP} = \cup_k \mathsf{TIME}\left(2^{n^k}\right)$, and that $\mathsf{NEXP} = \cup_k \mathsf{NTIME}\left(2^{n^k}\right)$. Just as it is a famous open problem whether $\mathsf{P} = \mathsf{NP}$, it is *also* an open problem whether $\mathsf{EXP} = \mathsf{NEXP}$. However, show that these problems are related in the following way: if $\mathsf{P} = \mathsf{NP}$, then $\mathsf{EXP} = \mathsf{NEXP}$ as well. [*Hint:* Given a language $L \in \mathsf{NEXP}$, can you come up with a different language $L' \in \mathsf{NP}$, such that deciding L in exponential time is equivalent to deciding L' in polynomial time? The trick of "padding" an input string with a bunch of trailing 1's will likely be helpful here.]

7. As we've discussed in class, *provable separations* of complexity classes are few and far between. In this problem, however, you'll prove a bizarre separation that happens to be known: P *does not equal the class of languages decidable in linear space.*

(a) Show that, if $\mathsf{SPACE}(n) \subseteq \mathsf{P}$, then $\mathsf{SPACE}\left(n^2\right) \subseteq \mathsf{P}$ also. [*Hint:* Use the "padding" trick, just like you did for problem 6.]

(b) Using part a, show that if $\mathsf{P} = \mathsf{SPACE}(n)$, then $\mathsf{SPACE}(n) = \mathsf{SPACE}\left(n^2\right)$. Conclude that $\mathsf{P} \neq \mathsf{SPACE}(n)$. [You can assume the Space Hierarchy Theorem.]

(c) From parts a and b, can you conclude that there exist languages decidable in polynomial time but not in linear space? Can you conclude that there exist languages decidable in linear space but not in polynomial time?