Handout 13: NP-Completeness

13.1 Polynomial-time reductions. A function $f\colon \Sigma^* \to \Sigma^*$ is *polynomial-time computable* if there exists a Turing decider with polynomial time complexity which accepts all input words w, and when entering q_a, has $f(w)$ on the tape. For two languages $A, B \subseteq \Sigma^*$, we say that A *polynomial-time reduces* to B, written $A \leq_p B$, if there exists a polynomial-time computable function $f\colon \Sigma^* \to \Sigma^*$ such that for all $w \in \Sigma^*$, we have $w \in A$ iff $f(w) \in B$. If $A \leq_p B$, then:

1. if $B \in$ P, then $A \in$ P.
2. if $B \in$ NP, then $A \in$ NP.
3. if $A \notin$ P, then $B \notin$ P.
4. if $A \notin$ NP, then $B \notin$ NP.

13.2 NP hardness and completeness. A language $A \subseteq \Sigma^*$ is NP-*hard* if for all languages $B \in$ NP, we have $B \leq_p A$. In other words, a problem A is NP-hard if A is as hard as any problem in NP. If $A \leq_p B$, then:

5. if A is NP-hard, then B is NP-hard.

The language A is NP-*complete* if (1) $A \in$ NP and (2) A is NP-hard. Note that if A is NP-complete, then $A \in$ P iff P = NP.

13.3 NP lower bounds. Suppose that we can establish an upper bound of NP for a problem A, but not an upper bound of P; that is, we find a deterministic polynomial-time verifier for A, but not a deterministic polynomial-time decider. Then we should try to prove that our failure to show that $A \in$ P is not due to our limitations, but that A is as hard as any problem in NP; that is, we should try to show that A is NP-hard. This is a *lower-bound* proof which matches the *upper-bound* proof that $A \in$ NP. To show that A is NP-hard, we polynomial-time reduce a known NP-complete problem B to A. Good candidates for B are 3SAT, HAMPATH, CLIQUE, and SUBSETSUM (especially if A is a problem involving numbers). The problem 3SAT is the restriction of SAT to boolean input formulas ϕ in 3cnf; that is, ϕ is a conjunction of "clauses," each clause being a disjunction of no more than 3 "literals," each literal being a boolean variable or the negation of a boolean variable.

13.4 Four paradigmatic NP-complete problems. We first note that SAT and 3SAT are NP-hard problems (the Cook-Levin theorem)[1]. All other NP-hardness results we conjecture by polynomial-time reductions. In particular, 3SAT $\leq_p$ HAMPATH[2] and 3SAT $\leq_p$ CLIQUE[3] and 3SAT $\leq_p$ SUBSETSUM[4]. (By the way, the problem COMPOSITES was recently shown to be in P.)

13.5 Partition. For a multiset $X = \{x_1, \ldots, x_n\}$ of numbers, let $\Sigma X = x_1 + \cdots + x_n$. Consider the problem PARTITION:

> Given a multiset X of natural numbers, does there exist a submultiset $Y \subseteq X$ such that $\Sigma Y = \Sigma(X \setminus Y)$.

To see that PARTITION is in NP, observe that Y is a polynomial-size witness. To see that PARTITION is NP-hard, we prove SUBSETSUM $\leq_p$ PARTITION:

[1] page 281 of M. Sipser's book – see SAT_3SAT.pdf handout
[2] page 290 of M. Sipser's book – see HAMPATH.pdf handout
[3] page 278 of M. Sipser's book – see CLIQUE.pdf handout
[4] page 296 of M. Sipser's book – see SubsetSum.pdf handout

> Given a multiset X of natural numbers and a natural number t, we need to construct in polynomial time a multiset X' of numbers such that (1) if there exists a submultiset $Y \subseteq X$ with $\Sigma Y = t$, then there exists a submultiset $Y' \subseteq X'$ with $\Sigma Y' = \Sigma(X' \setminus Y')$, and (2) if there exists a submultiset $Y' \subseteq X'$ with $\Sigma Y' = \Sigma(Y' \setminus X')$, then there exists a submultiset $Y \subseteq X$ with $\Sigma Y = t$.

We choose $X' = X \cup \{2t - \Sigma X\}$. For (1), choose $Y' = Y$; then $\Sigma Y' = \Sigma Y = t$ and $\Sigma(X' \setminus Y') = (\Sigma X) + (2t - \Sigma X) - t = t$. For (2), choose $Y = Y'$ if $(2t - \Sigma X) \notin Y'$, and $Y = (X' \setminus Y')$ if $(2t - \Sigma X) \in Y'$. In both cases, $\Sigma Y = (\Sigma X')/2 = ((\Sigma X) + (2t - \Sigma X))/2 = t$.