Complexity Theory

Supervision 2

4. NP-completeness

1. Show that the identity function is a poly-time reduction, and composition of poly-time reductions is a poly-time reduction.
2. A problem in NP is called NP-*intermediate* if it is neither in P nor NP-complete.
   a) Are there any problems that are known to be NP-intermediate?
   b) Research and briefly summarise *Ladner's theorem*.
3. Suppose that a language $L_1 \subseteq \Sigma_1^*$ is polynomial-time reducible to a language $L_2 \subseteq \Sigma_2^*$ with the underlying function $f : L_1 \leq_P L_2$. Prove or disprove the following claims, or state if the answer is unknown and explain why:
   a) If $L_2 \leq_P L_1$, then f is a bijection.
   b) If f is a bijection, then $L_2 \leq_P L_1$.
   c) If f is a bijection, then L_2 is in NP.
   d) If f is a bijection and L_1 is in NP, then L_2 is in NP.
   e) If L_1 is NP-complete, then $L_2 \leq_P L_1$.
   f) If L_2 is NP-complete, then $L_2 \leq_P L_1$.

5. NP-complete problems

1. Given a graph $G = (V, E)$, a set $C \subseteq V$ of vertices is called a *vertex cover* of G if, for each edge $(u, v) \in E$, either $u \in C$ or $v \in C$. That is, each edge has at least one end point in C. The decision problem V-COVER is defined as:

   Given a graph $G = (V, E)$, and an integer K, does G contain a vertex cover with K or fewer elements?

   a) Show a polynomial time reduction from IND to V-COVER.
   b) Use a) to argue that V-COVER is NP-complete.
2. The problem of *four-dimensional matching*, 4DM, is defined analogously with 3DM:

   Given four sets, W, X, Y and Z, each with n elements, and a set of quadruples $M \subseteq W \times X \times Y \times X$, is there a subset $M' \subseteq M$ such that each element of W, X, Y and Z appears in exactly one tuple in M'?

   Show that 4DM is NP-complete.

3. Given a graph $G = (V, E)$, a source vertex $s \in V$ and a target vertex $t \in V$, a *Hamiltonian path* from s to t in G is a path that begins at s, ends at t, and visits every vertex in V exactly once. We define the decision problem $\mathrm{HamPath}$ as:

   Given a graph $G = (V, E)$ does G contain a Hamiltonian path?

   a) Give a poly-time reduction from the Hamiltonian cycle problem to $\mathrm{HamPath}$.

   b) Give a poly-time reduction from $\mathrm{HamPath}$ to the Hamiltonian cycle problem.

   c) Consider the following, modified statement of the Hamiltonian path problem:

      Given a graph $G = (V, E)$ and vertices $s, t \in V$,
      does G contain a Hamiltonian path from s to t?

      Explain how this differs from the problem above, and comment on whether your reductions in parts a) and b) can be simplified for this version.

4. We know from the Cook–Levin Theorem that every problem in NP is reducible to SAT. The proof worked for a general nondeterministic Turing-machine, but for some problems it is easy to give an explicit reduction. Describe how to obtain, for any graph $G = (V, E)$, a Boolean expression φ_G such that φ_G is satisfiable if and only if:

   a) G is 3-colourable.

   b) G contains a Hamiltonian cycle.

   Hint: By analysing the search space of the problem, determine a collection of Boolean variables that can encode the relevant properties of the graph (cf. $S_{i,q}$, $T_{i,j,\sigma}$ and $H_{i,j}$ in the Cook–Levin Theorem proof). Give the constraints on the variables which are required to make the encoded graph an instance of the given problem (cf. expressions (1)-(7) in the CLT proof). Combine these with the constraints that would decide whether a potential instance is a member of the problem or not (cf. expression (8) in the CLT proof).

5. An instance of a *linear programming* problem consists of a set $X = \{ x_1, \ldots, x_n \}$ of variables and a set of constraints, each of the form

   $$\textstyle\sum_{1 \leq i \leq n} c_i x_i \leq b,$$

   where each c_i and b is an integer.

   The 0-1 Integer Linear Programming Feasibility problem $\mathrm{01\text{-}ILP}$ is defined as follows:

   Given an instance of a linear programming problem, determine whether there is an assignment of values from the set $\{ 0, 1 \}$ to the variables in X so that substituting these values in the constraints leads to all constraints being simultaneously satisfied.

   Prove that this problem is NP-complete.

6. *Self-reducibility* refers to the property of some problems in $L \in \mathrm{NP}$, where the problem of finding a witness for the membership of an input x in L can be reduced to the decision problem for L. This question asks you to give such arguments in three specific instances.

a) Show that, given an oracle (i.e. a black box) for deciding whether a formula φ over a set of variables $\mathcal{V} = \{x_1, x_2, \ldots, x_n\}$ is satisfiable, there is a polynomial-time algorithm that gives a variable assignment which satisfies a formula over $\mathcal{V}$.

b) Show that, given an oracle for deciding whether a given graph $G = (V, E)$ is Hamiltonian, there is a polynomial-time algorithm that, on input G, outputs a Hamiltonian cycle in G if one exists.

c) (Harder) Show that, given an oracle for deciding whether a given graph G is 3-colourable, there is a polynomial-type algorithm that, on input G, produces a valid 3-colouring of G if one exists.

Optional exercises

1. The problem E3SAT is defined as follows:

   Given a set of clauses, each clause being a disjunction of exactly three distinct literals and containing exactly three distinct variables, determine whether it is satisfiable.

   Prove that E3SAT is NP-complete. *Hint:* introduce new variables to the set by adding a tautological clause.

2. We use $x; 0^n$ to denote the string that is obtained by concatenating the string x with a separator ; followed by n occurrences of 0. If $[M]$ represents the string encoding of a non-deterministic Turing machine M, show that the following language is NP-complete:

   $$S = \{ [M]; x; 0^n \mid M \text{ accepts } x \text{ in } n \text{ steps} \}$$

   Hint: Rather than attempting a reduction from a particular NP-complete problem, it is easier to show this from first principles, i.e. construct a reduction for any NDTM M and polynomial bound p.