Computational Complexity — tutorial 3

NP-hardness

Lecture: P, NP and co-NP. Hardness and completeness. Turing and Karp reductions.
$P \subseteq NP$ and $P \subseteq$ co-NP. But it may be that P = NP (equivalently P = co-NP).
Some well-known NP-complete problems:

- NP Halting Problem. Given a non-deterministic Turing Machine $\mathcal{M}$, input w and an integer k (written in unary), is there an accepting run of $\mathcal{M}(w)$ which terminates in k steps?
- Boolean Satisfiability. Given a boolean formula φ over boolean variables $x_1, x_2, \ldots, x_n$, is it satisfiable for any values of $x_1, \ldots, x_n$? (That is, is $\exists_{x_1} \exists_{x_2} \ldots \exists_{x_n} \varphi(x_1, \ldots, x_n)$ true?)
- 3-CNF-SAT. Similar to above, but φ is of the form $\varphi = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_m$, and each ψ_i is a disjunction (OR) of 3 literals (x_i or $\neg x_i$). E.g., $\varphi = (x_1 \vee x_3 \vee \neg x_2) \wedge (x_1 \vee \neg x_3 \vee \neg x_4)$.

1. Does there exist a language L such that neither L nor the complement or L is semi-decidable? Can we show an explicit example of such L?

2. A language L is *recursively enumerable* (RE) if there exists a Turing machine enumerating all strings of L (e.g. writing them to some output tape).

(a) Prove that L is recursively enumerable iff it is semi-decidable.

(b) Let L be a language over unary alphabet, i.e. $L \subseteq \{1\}^\star$. Show that if L is infinite and semi-decidable, then there exists an infinite decidable subset of L.

3. Prove that the following computational problems are NP-complete:

(a) Given three boolean formulas $\varphi_1, \varphi_2, \varphi_3$, decide if at least two of them are satisfiable at the same time.

(b) Clique problem: given an undirected graph G and an integer k, does G contain a clique (a complete graph) on k vertices?

(c) Independent Set problem: given an undirected graph G and an integer k, does G contain an independent set of size k (that is, k vertices without any edges between them)?

(d) Vertex Cover problem: given an undirected graph G and an integer k, is there a subset A of k vertices of G such that each edge is incident to some vertex of A?

(e) $\{0,1\}$-Linear Programming problem (in a decision form): given an integer matrix $A \in \mathbb{Z}^{m \times n}$ and a vector $b \in \mathbb{Z}^m$, is there an **integer** vector $x \in \{0,1\}^n$ such that $Ax \leqslant b$?

($Ax \leqslant b$ is a set of linear inequalities of the form $a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leqslant b_i$.)

If x can be arbitrary $\mathbb{Z}^n$, then the problem is NP-complete, but this is non-trivial. Why?

Note: If x can be arbitrary $\mathbb{R}^n$, then the problem is in P, but the algorithms are hard.

(f) Subset Sum problem: given a multiset of n non-negative integers $\{a_1, \ldots, a_n\}$, and an integer A (target), is there a subset of the multiset whose sum is A?