CS218 Design and analysis of algorithms

Jan-Apr 2025

Exercises: NP, NP-completeness

Independent set: given a graph and a number k, is there a set of k vertices such that there is no edge between them?

Vertex Cover: given a graph and a number k, is there a set S of k vertices which covers all edges (i.e., every edge in the graph has at least one end point in S).

Set Cover: Suppose we are given a collection of subsets of a set U, say $S_1, S_2, \ldots, S_r \subseteq U$. Can we select k of these subsets such that their union is U?

Directed Hamiltonian cycle: Given a directed graph with n vertices, is there a (directed) cycle of length n? By definition, a cycle cannot have repeated vertices.

- 1. Which of the following decision problems do you think are in NP?
 - Given a graph G and a number k, is the minimum vertex cover size in G equal to k?
 - We are given a directed graph, a source vertex, a destination vertex, and a number k. Is k the minimum number vertices required to be deleted to disconnect source from destination?
 - Given two C++ programs, are they equivalent (do they have same input output behavior)?
 - Given two C++ programs, are they different (is there an input where they give different outputs)?
 - Given a fully quantified Boolean formula (every variable is quantified with either ∃ or ∀ in some order), is it true?
- 2. Reduce the independent set problem to the vertex cover problem.
- 3. Reduce vertex cover problem to the set cover problem.
- 4. Suppose the following version of knapsack problem is NP-complete. Given a set of integer weights w_1, w_2, \ldots, w_n and target weights W_1, W_2 , is there a subset S of the weights whose sum is between W_1 and W_2 , i.e., $W_1 \leq \sum_{i \in S} w_i \leq W_2$?

Using this fact, prove that the following load balancing problem is NP-complete. Given a set of integer loads t_1, t_2, \ldots, t_n and a target makespan T, is there a way to distribute all the loads to two machines so that the maximum load on any machine is at most T?

5. Integer programming: Given a set of linear inequalities in variables x_1, x_2, x_n , decide if there is an integer solution satisfying all of them simultaneously. For example the set

$$0 \le x_1, x_2 \le 1$$
$$x_1 - x_2 \ge 0.$$

has an integer solution (1,0) (and also (1,1), (0,0)).

Show that Integer Programming is NP-hard. You can try a reduction from SAT to this problem. Given a CNF Booelan formula ϕ , you need to generate a set S of linear inequalities and prove that ϕ is satisfiable if and only if the set S has an integer solution.

- 6. Reduce SAT problem to Directed Hamiltonian cycle problem.
- 7. Consider the 3-coloring problem, where we are given a graph and we have to decide if we can color the vertices of the graph with 3 colors, such that any two neighboring vertices get different colors. Reduce this problem to the independent set problem.