Homework 7

Walker Bagley

April 18, 2024

1 Problem Definitions

1. VERTEX-COVER: Given a graph $G = (V, E)$, find a subset $V' \subseteq V$ such that every edge goes to or from a node in V', that is, $\forall(u, v) \in E$, $u \in V' \vee v \in V'$. To optimize, we look for a vertex cover containing the fewest nodes, for which the decision problem becomes whether or not G has a vertex cover of size k.
2. TSP: Given a graph of cities (nodes) and the corresponding costs of travel between them, find a tour that starts and ends at the same city and visits every city once. For optimization, we want to find the tour with minimum cost. Then the decision problem becomes $\langle G, c, k\rangle$ where $G = (V, E)$ is a complete graph, c is a function on $V \times V \to \mathbb{N}$, $k \in \mathbb{N}$, and G admits a TSP tour with a cost no greater than k.
3. SUBSET-SUM: Given a finite set of positive integers S and a target sum t, we want to determine if there exists a subset $S' \subseteq S$ such that the sum of elements in S' is equal to t. That is $\sum_{s\in S'} s = t$.
4. Subgraph Isomorphism: Two graphs are isomorphic if they have the same number of vertices and edges and are structurally equivalent. So, given two graphs G_1 and G_2, the subgraph isomorphism problem asks if we can find a subgraph of G_2 which is isomorphic to G_1.

2 3-CNF-SAT $\leq_P$ CLIQUE

To reduce 3-CNF-SAT to CLIQUE, we must convert a 3-CNF-SAT expression which is satisfiable to a graph that admits a clique and one that is unsatisfiable to a graph that does not admit a clique. A 3-CNF-SAT expression with k clauses looks like $C_1 \wedge C_2 \wedge \ldots \wedge C_k$. Each clause will consist of 3 literals, so the general form is $C_r = l_1^r \vee l_2^r \vee l_3^r$. We construct the graph $G = (V, E)$ by doing the following: for each clause C_r, create three nodes v_1^r, v_2^r, v_3^r and add them to V. Then add an edge $\forall i, j \forall r, s$, (v_i^r, v_j^s) into the graph if $i \neq j$ and the literals l_i^r, l_j^s are not negations of each other. These steps will yield a graph that admits a clique of length k iff the source 3-CNF-SAT expression is satisfiable.

3 Example

The set partition problem (SP) asks if given a set S of n integers, one can split S into two disjoint subsets S_1, S_2 which both sum to the name number.

Let's prove SP is NPC by reduction from SUBSET-SUM.

First, we must show that SP is NP. We consider the certificate of S, which we can partition into S_1 and S_2. Then we can check if S_1, S_2 are disjoint in polynomial time as well as if $S_1 \cup S_2 = S$ in polynomial time. Then we can sum the elements of each in polynomial time and compare the sums in linear time. So, SP can be solved in polynomial time and is therefore NP.

Now, we will show that SUBSET-SUM $\leq_P$ SP. Consider a SUBSET-SUM certificate $\langle S, t\rangle$ which is satisfiable, that is, there is a subset of S which sums to t. Then our SP problem will consist of a set $S' = S \cup \{2s-t, s+t\}$ where s is the sum of all elements in S. Suppose the subset of S that sums to t in the SUBSET-SUM problem is called R. Then we partition S' as follows: one partition consists of $S_1' = R \cup \{2s-t\}$ and the other consists of $S_2' = (S - R) \cup \{s + t\}$. We know the sum of R is equal to t, so the sum of S_1' is $t + 2s - t = 2s$. Also, the sum of $S - R$ is $s - t$, so the sum of S_2' is $s - t + s + t = 2s$. Clearly the sums of these two partitions are equal. Also, the two new elements we add to S to generate S' add to $2s - t + s + t = 3s$, and we know S_1', S_2' will each sum to $2s$ if the SUBSET-SUM problem is satisfiable, so these two new elements must be in separate partitions of S'. If there is not a solution to the SUBSET-SUM problem, then the two partitions of S' will not be equal to $2s$. So, we have reduced SUBSET-SUM to SP.