Problem Set 1 Solutions

Due: Monday, September 11, 2023 at noon

Problem 1.1 [Hamiltonian Cycle Problem → Traveling Salesman Problem].

Recall from lecture that a *Hamiltonian cycle* in a graph is a cycle visiting every vertex exactly once. The Hamiltonian Cycle Problem is the NP-complete problem of deciding whether the input graph has a Hamiltonian cycle.

The Traveling Salesman Problem in Graphs is to decide, given a complete graph[1] with *positive integer* edge weights and given a target integer t, does there exist a cycle[2] in the graph that visits every vertex at least once and has total weight $\leq t$?

Prove that the Traveling Salesman Problem in Graphs is NP-complete. Remember that in order to prove a problem is NP-complete, you must show that it is both NP-hard and contained in NP. To prove NP-hardness, reduce from the Hamiltonian Cycle Problem.

Solution: (This first solution is intentionally more verbose than we'd normally write, and than we'd expect of you, so that you have a clear example of a fully formal reduction proof. In your solutions, make sure you don't omit any of the steps of the proof, but feel free to prove them more briefly.)

First, we show that the Traveling Salesman Problem in Graphs is contained in NP. Conceptually, this is straightforward: a witness will simply be a list of vertices visited by the cycle, and our algorithm needs only to add up the weights of the edges between consecutive vertices and check that the total weight is at most t. However, there is a subtle sticking point: we must ensure that for YES instances, there exists a witness *of polynomial size* (relative to the size of the original problem). Because vertices may repeat in the cycle, this is not obvious.

To address this point, we prove the following lemma:

Lemma. *Suppose (G, t) is an instance of Traveling Salesman Problem in Graphs, and suppose (G, t) has a solution. Then (G, t) has a solution of length at most n^2, where n is the number of vertices of G.*

Proof. Let C be a shortest possible solution, and let v be a vertex of G. Suppose v is visited k times in C, and let $S_1, S_2, \ldots, S_k$ be the sets of vertices visited between each pair of consecutive visits to v.

It must be the case that each S_i contains some vertex which is not present in the entire rest of the cycle. Otherwise, a shorter cycle could be obtained by deleting the part of C corresponding to S_i. This shorter cycle would have less total weight than C, so it would also be a solution. But that would contradict the fact that C was a shortest solution.

Thus $k \leq n$ and so each of the n vertices is visited at most n times, so the length of C is at most n^2. □

It follows that every YES instance has a witness of size n^2 vertices, which is polynomial in the size of the original problem.[3]

Now, we show that the Traveling Salesman Problem in Graphs is NP-hard. We reduce from the Hamiltonian Cycle Problem to the Traveling Salesman Problem in Graphs. Given an instance of

[1] A *complete graph* has an edge between every pair of distinct vertices.

[2] Throughout this class, a *cycle* is allowed to repeat vertices and/or edges; if it doesn't repeat vertices, the cycle is called *simple*.

[3] Here we're measuring the size of the witness in terms of the number of vertices; an even more detailed analysis would note that each vertex label can be written in $\log n$ bits, so the size of the witness would be $n^2 \log n$ bits.

Hamiltonian Cycle, namely a graph $G = (V, E)$ with $n = |V|$ vertices, our reduction outputs the following instance of TSP in Graphs:

- A complete graph G' on the same vertex set V, where each edge is labeled with weight 1 if it is in E, and with weight $n + 1$ otherwise.
- The target weight $t = n$.

Having described the reduction, we need to prove two properties about it. First, we need to show that our reduction is polynomial time. It takes $O(n)$ time to count the number n of vertices and $O(n^2)$ time to create the weighted complete graph G', so our reduction is certainly polynomial time. Second, we need to show that our reduction is correct—that is, the resulting TSP in Graphs instance (G', n) always has the same answer as the original Hamiltonian Cycle instance G. We will do this by showing that a solution to either instance implies the existence of a solution to the other instance (in both directions).

In one direction, suppose that there exists a solution to the Hamiltonian Cycle instance, that is, that G has a Hamiltonian cycle C. We can view C as a cycle in G' as well. Because C is Hamiltonian, it visits all n vertices exactly once. Thus C has length n and so it has weight n in G', as it uses only edges in E. Cycle C visits every vertex at least once and has weight $\leq n$, so it is also a solution to the TSP in Graphs instance.

In the other direction, suppose that there exists a solution to the TSP in Graphs instance, that is, a cycle C in G' visits every vertex at least once and has weight $\leq n$. Because C visits every vertex at least once, it has length $\geq n$.[4] On the other hand, C has weight $\leq n$ and thus length $\leq n$ (because all weights are positive integers). So the cycle C must have length exactly n, which means it visits every vertex exactly once. Finally, C must be a cycle in G because, if it included any edges not in E, then that edge alone would cause it to have weight at least $n + 1$ (because weights are nonnegative), a contradiction. Therefore, C is a Hamiltonian cycle on G, so it is also a solution to the Hamiltonian Cycle Problem instance.

We have demonstrated a correct polynomial-time reduction from Hamiltonian Cycle to TSP in Graphs. Because the Hamiltonian Cycle Problem is NP-hard, so is the Traveling Salesman Problem in Graphs.

[4] Recall that the *length* of a cycle is the number of edges in the cycle (counting repetitions), which is equal to the number of vertices in the cycle (counting repetitions).