CS 301 - Assignment 5 - NP-Completeness Problem

Rebah Özkoç
29207

June 5, 2022

Definition Of The Problem

To show a problem is an NP-Complete problem, first we need to find the decision version of the given optimization problem. The given problem can be restated like this.

For a given graph G = (E,V), given set of nodes C and a given set of vertices S_c what is the shortest itinerary I_v from the given vertex s to vertex v from the set C that passes through every vertex exactly once in the set S_c for every vertex in the given set C?

We can transform this problem into a decision problem like this:

For a given graph G = (E,V), given set of nodes C, a given set of vertices S_c, and a given number k, is there an itinerary I_v from the given vertex s to vertex v that passes through every vertex exactly once in the set S_c of length of at most k for every vertices in the set C?

This question is the decision version of the given optimization problem.

This decision problem is not harder than the original problem because if we have solution to the original problem we can find answer of the decision problem version in polynomial time.

NP-Completeness Proof Of The Decision Problem

Membership

To show a problem belongs to the NP-Complete class, first we need to show that it belongs to NP set. NP problems are non-deterministically polynomial. For a given answer to the problem in this set, we can verify the correctness of the solution in polynomial time.

To show our decision problem belongs to NP class, we need to show that we can test the truthiness of a given answer in polynomial time.

Given shortest paths from s to each c in C, S_c and integer k for each shortest path, we need to check three things to validate the answer. First one is "Does the path contain all the vertices in S_c exactly once?", the second one is "Is the length of the path is smaller than the k?", and the last one is "Does the path start at s and ends at t?"

We can answer the first question by performing a linear search in the vertices of the answer for each vertex in the S_c. While performing the linear search we can count the occurrences to ensure that each vertex in S_c contains only once in the answer. This process takes polynomial time.

We can answer the second question by counting the number of edges in the answer and comparing it with k. This process takes polynomial time.

We can answer the last question by simply checking the starting and ending nodes of the answer and this process takes polynomial time.

As a result, we can test the truthiness of a given answer to our problem in polynomial time and this means that the problem belongs to NP class.

Hardness

If an NP-Complete problem P is polynomial time reducible to our problem Q, then we can say that our problem is as hard as problem P. This means that our problem satisfies the hardness condition of the NP-Complete class.

We can reduce the Hamiltonian Circuit problem which is an NP-Complete problem to our decision problem.

To reduce the Hamiltonian Circuit problem to our decision problem firstly we need to turn the graph G to a complete graph by adding edges between the vertices (u,v) that does not contain an edge in the original graph.

After adding new edges to the graph we need to change the weights of the edges. We can set the weight of the newly added edges to infinity and weight of the old edges to 1.

We can set C=s and $S_C = V - \{s\}$ By this way, we can find the path that visits all the cities exactly once.

We can select any vertex v as starting vertex because Hamiltonian Circuit is circular and we can start from any vertex without affecting the result.

As a result, if there exists a cycle in the graph that visits all the vertices with length $\leq$ v (number of nodes) then this means that the original graph has a Hamiltonian circuit. By this way we reduced the Hamiltonian circuit problem to our decision problem and solved a specific instance of our decision problem. This instance has C = s and $S_C = V - \{s\}$ and starts and ends at the same city.

We can conclude that without lost of generality our decision problem is as hard as the Hamiltonian Circuit Problem.

Thus, our decision problem belongs to NP-Complete class.