CS301 - HW4

Tahir Turgut - 27035

May 23, 2021

(a) *Object:* Given the graph, determine if there is an itinerary from Istanbul to i that cost less than k_i, where i is every node in the graph -except Istanbul.

Iterating the below process for each vertex will solve the problem:

Input:

A graph $G = (V, E), c : VxV \rightarrow Z^+$, where cost information is stored inside edges

A set $R = \{v_1, v_2, ..., v_n\}$, where v_i is a vertex in graph that has to be included to itinerary to v,

A positive integer k

Output:

Yes: if there is a path to vertex v exist that costs less than k,

No: if there is not a path to vertex v exist that costs less than k

(b) *Theorem:* "Finding Itinerary problem" is NP-Complete.

Membership:

Guess an itinerary for a specific vertex (city), v,

Check whether every city in the R appears in the itinerary exactly once: $O(n)$, where n is the number of nodes in the path,

Also, check whether any city in the itinerary appears more than once: $O(n)$, where n is the number of nodes in the path,

After those checks, if the itinerary is valid, find the cost of the road by adding all lengths: Search for a road in edges: $O(E)$, there n roads in the itinerary: $n \cdot O(E)$ where n is the number of nodes in the path,

Finally, checks whether that sum is minimum or not: $O(1)$

In total: $O(n) + O(n) + O(E \cdot n) + O(1) = O(n \cdot E)$

Repeat above steps for each vertex, v in the graph (except Istanbul itself): $V \cdot O(n \cdot E)$

Since n can be V at most: $O(V^2 \cdot E)$

Above process can be handled in polynomial time. Hence, problem is in NP.

Hardness:

Reduce *Travelling salesman problem (TSP)* to Find itinerary problem,

To find TSP of graph $G = (V, E)$, every node has to be visited, and start and end nodes has to be same.

In Finding Itinerary problem, $G' = (V', E')$ can be same as G with an additional fictional city that has 0 cost edge to starting node. Ending node can be set to that node, and R can be set to the all of the nodes except starting and ending node.

Also, in both of those problems, every node has to be visited only once.

Set R as all other nodes: $O(|V|)$

Finally, k of TSP can be set as k' of FIP.

FIP have to visit all of the nodes exactly once in the R, which is all of the nodes in G. Also, ending vertex has no cost to starting vertex, specified. Thus, it will return YES, if the path that going through every node in the graph G exactly once and reaches to fictional city -which has no distance to starting city- is shorter than $k' = k$. This has to be also YES in TSP as well, since it is required to visit every node in the graph G exactly once and return to starting vertex. If path is longer, both will return NO as output.

Above reduction can be handled in polynomial time. Hence, problem is NP-Complete.