Shawn Koong

David Paner

CS170

21 November 2019

Design Doc

We are tasked with finding the most optimal route for driving and dropping off TAs. We are given an undirected graph $G = (L, E)$ where each vertex in L is a location, a starting location s, and a list H of homes that subsets L. Each edge weight (u, v) corresponds to the length of the road between locations u and v and the amount of energy expended is proportional to the length. Every unit of distance costs ⅔ units of energy driving and 1 unit of energy walking. Additionally, the car must start and end at S, and every TA has to return to their home in H. We want to find a route and sequence of drop-offs that minimizes the total amount of energy expenditure.

First Approach:

We decided to first find the shortest paths from each home to the starting location s using Dijkstra's algorithm. Then we construct a graph G' of the result from running Dijkstra's on G. Then if there exists some edge (u, v) between any two vertices from the shortest paths, we add those edges between the vertices to G' as well. From this, we will try to reduce the problem of finding the most optimal route in this graph to a variant of the Travelling Salesman Problem. We would solve this using the nearest neighbors algorithm, where our heuristic is visiting the nearest home from our start point and then the nearest home from the previous location and so on. Once we have our tour of minimal length we will follow this tour and drop off all of the TAs in order of the house vertices, thus completing the problem.

Second Approach:

This approach takes advantage of the fact that a solution to the vertex cover solution returns a set of vertices such that each edge of the original graph is incident to at least one vertex in the set. First we will greedily approximate the minimum vertex cover by starting with the set of edges E in G. We pick an arbitrary edge (u, v) from E and add that edge to our result. We then remove all other edges from E if they are incident to u or v. This result will be the approximation of the minimum vertex cover. From here, we then find the most optimal route among the vertices in the minimum vertex cover result starting from our starting point by reducing it to the Travelling Salesman Problem and using the nearest neighbors algorithm to find it. Our heuristic will be the vertex closest from our start point and then the nearest vertex after the previous. We then traverse this route in the car and drop off TAs at the vertex closest to their homes where they then walk home or at their home if their home happens to be part of the vertex cover solution.