SIMULATION OF TRAVELLING SALESMAN PROBLEM

A THIRD YEAR PROJECT REPORT PROPOSAL

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF B.Sc. IN COMPUTATIONAL MATHEMATICS

BY

1. Samrajya Acharya (22617)
2. Abhay Sharma (22635)
3. Diya Shrestha (22638)

SCHOOL OF SCIENCE
KATHMANDU UNIVERSITY
DHULIKHEL, NEPAL

SUBMITTED TO:

Dr. Ganga Ram Phaijoo

Department of Mathematics

ABSTRACT

The Travelling Salesman Problem (TSP) is a classical combinatorial optimization NP-hard problem, which is simple to state but very difficult to solve. Although a global solution for the Travelling Salesman Problem does not yet exist, there are algorithms for an existing local solution [3].There are also necessary and sufficient conditions to determine if a possible solution does exist when one is not given a complete graph. This paper gives an introduction to the Travelling Salesman Problem with an algorithm to visualise the optimal distance between the nodes. Given a list of nodes, we find the shortest path that visits all nodes at once.

CONTENTS

INTRODUCTION

Travelling Salesman Problem (TSP) is a classical and most widely studied problem in Combinatorial Optimization. This problem belongs to a category referred to as NP-complete ie. if its solution can be guessed and verified in polynomial time [5]. It has been studied intensively in both Operations Research and Computer Science since the 1950s as a result of which a very large number of methods were studied to solve this problem. The idea of the problem is to find the shortest route of salesman starting from a given city, visiting n cities only once and finally arriving at the origin city [1]. The problem can be sketched on graphs with each city becoming a node. Assuming a complete weighted graph, edge lengths correspond to the distance between the attached cities [4].

The TSP occurs in countless forms with some applications of engineering that include Vehicle routing scheduling problems, integrated circuit designs, physical mapping problems, and constructing phylo-genetic trees.

Following are the constraints of a TSP problem:

- The total length of the loop should be a minimum.
- The salesperson cannot be at two different places at a particular time.
- The salesperson should visit each city only once.

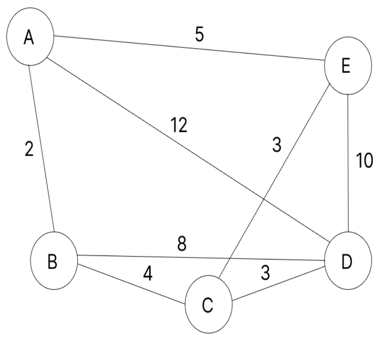

Figure 1: A graph with weights on its edges

According to the above figure, the problem lies in finding the shortest path to pass through all vertices at-least once.
For example, both Path1 (ABCDEA) and Path2 (ABCEDA) pass through all the vertices. However it can be seen that the total length of Path1 is 24, whereas the total length of Path2 is 31.

MOTIVATION

During the holidays, we friends were hanging out together and looking for some places to visit and enjoy. We ordered taxis from ride hailing service providers inside the valley and we all could find the map routes. Also, the map routes which were noticed too were uneven as terms of distance and travel cost, i.e; the distance vector would round up an entire location to reach a location that if travelled through another way would be viable in both terms of distance and travel expenses. So, to avoid such errors we came up with this project to optimise distance so that we get the shortest path and hence minimise the travel cost.

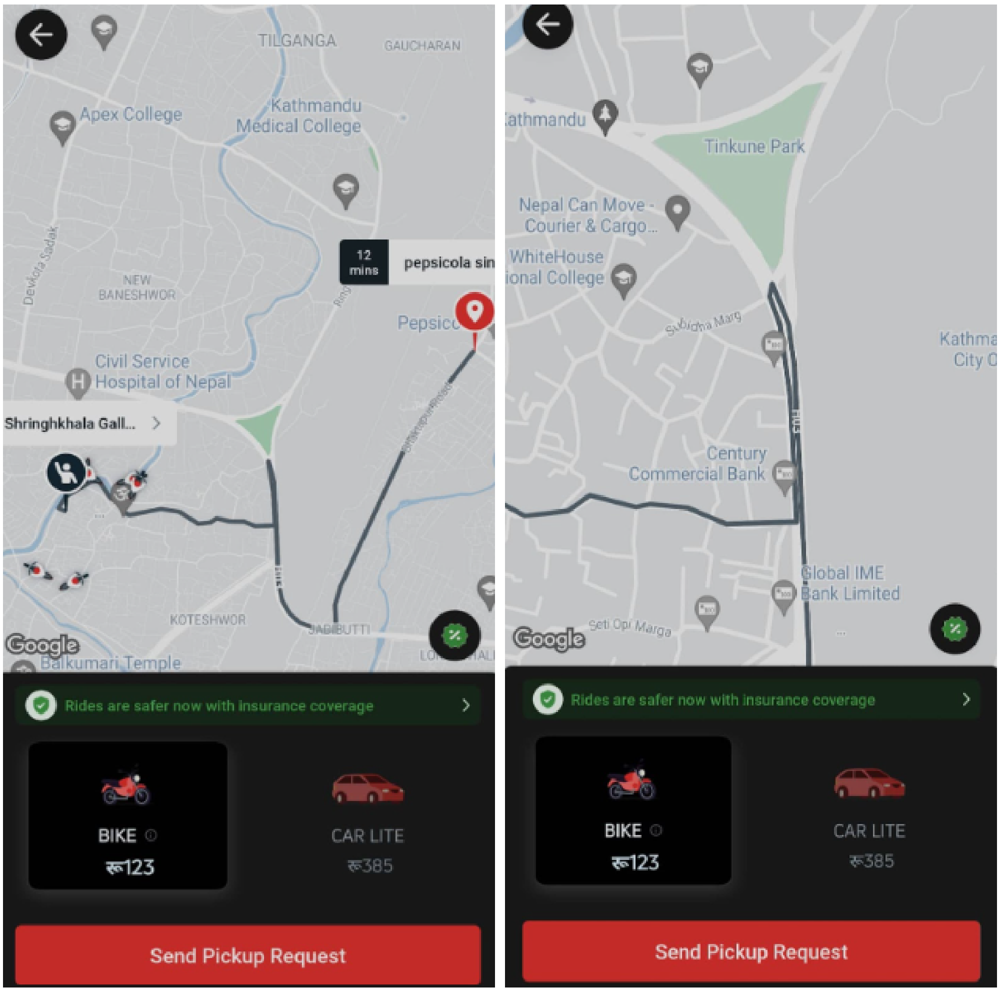

Figure 2: Screenshot of travel route

Figure implies the distance to be lengthy and increased travel cost but in practical, road is not so.

PROBLEM STATEMENT

A traveling salesman wishes to go to a certain number of destinations in order to sell objects. He wants to travel to each destination exactly once and return home taking the shortest total route.Each voyage can be represented as a graph G = (V, E) where each destination, including his home, is a vertex, and if there is a direct route that connects two distinct destinations then there is an edge between those two vertices. The traveling salesman problem is solved if there exists a shortest route that visits each destination once and permits the salesman to return home.

The traveling salesman problem can be divided into two types: the problems where there is a path between every pair of distinct vertices (no road blocks), and the ones where there are not (with road blocks). Though we are not all traveling salesman, this problem interests those who want to optimize their routes, either by considering distance, cost, or time. If one has four people in their car to drop off at their respective homes, then one automatically tries to think about the shortest distance possible. In this case, distance is minimised. If one is traveling to different parts of the city using the public transportation system, then minimizing distance might not be the goal, but rather minimizing cost.

In the first case, each vertex would be a person's home, and each edge would be the distance between homes. In the second case, each vertex would be a destination of the city and each edge would be the cost to get from one part of the city to the next. Thus, the Traveling Salesman Problem optimizes routes.

OBJECTIVES

- The goal of this site is to be an educational resource to help visualise, learn, and develop different algorithms for the travelling salesman problem in a way that's easily accessible
- As we apply different algorithms, the current best path is saved and used as input to whatever you run next. The order in which you apply different algorithms to the problem is sometimes referred to as the meta-heuristic strategy
- To understand the basic framework, current applications and future scope of the algorithm
- Learn to demonstrate the functionality of the sweep algorithm and site design using Java as the core programming language

LITERATURE REVIEW

Although the exact origins of the Travelling Salesman Problem are unclear, the first example of such a problem appeared in the German handbook Der Handlungsreisende - Von einem alten Commis - Voyageur for salesman travelling through Germany and Switzerland in 1832 as explained in [2]. This handbook was merely a guide, since it did not contain any mathematical language. People started to realise that the time one could save from creating optimal paths is not to be overlooked, and thus there is an advantage to figuring out how to create such optimal paths.

This idea was turned into a puzzle sometime during the 1800's by W. R. Hamilton and Thomas Kirkman. Hamilton's Icosian Game was a recreational puzzle based on finding a Hamiltonian cycle in the Dodecahedron graph.

The Travelling Salesman Problem was first studied in the 1930's in Vienna and Harvard as explained in [2]. Richard M. Karp showed in 1972 that the Hamiltonian cycle problem was NPcomplete, which implies the NP-hardness of TSP. This supplied a mathematical explanation for the apparent computational difficulty of finding optimal tours.

The current record for the largest Travelling Salesman Problem including 85,900 cities, was solved in 2006 as explained in [2]. The computers used versions of the branch-and-bound method as well as the cutting planes method (two seemingly elementary integer linear programming methods).

GANTT CHART

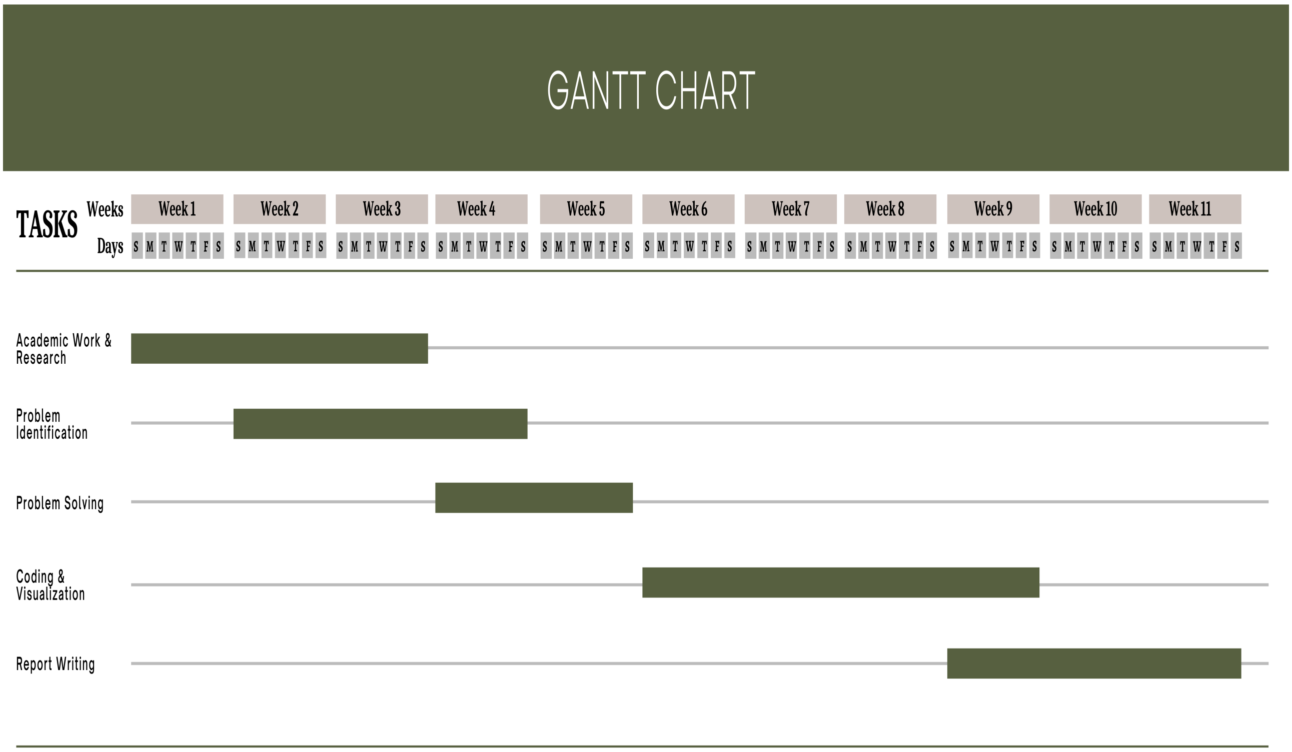

Figure 3: Gantt Chart

CONCLUSION

In this project, we will provide an overview of different approaches used for solving travelling salesman problem. The Travelling Salesman Problem (TSP) is a classical combinatorial optimization problem, which is simple to state but very difficult to solve. The problem is to find the shortest tour through a set of N vertices so that each vertex is visited exactly once [4]. This problem is known to be NP-hard, and cannot be solved exactly in polynomial time. Many exact and heuristic algorithms have been developed in the field of operations research (OR) to solve this problem. In this project, we will study approaches to solve the travelling salesman problem.

REFERENCES

[1] Tolga Bektas, *The multiple traveling salesman problem: an overview of formulations and solution procedures*, omega **34** (2006), no. 3, 209–219.

[2] Marco Dorigo and Luca Maria Gambardella, *Ant colony system: a cooperative learning approach to the traveling salesman problem*, IEEE Transactions on evolutionary computation **1** (1997), no. 1, 53–66.

[3] Steven G Krantz, *Essentials of mathematical thinking*, Chapman and Hall/CRC, 2017.

[4] Hamdy A Taha, *Operations research: an introduction*, vol. 790, Pearson/Prentice Hall Upper Saddle River, NJ, USA, 2011.

[5] J William, *Cook. in pursuit of the traveling salesman. mathematics at the limits of computation*, (2012).