INFO 6205 Spring 2023 Final Project

Traveling Salesman

Team member 1: Subbu Manickam

Team member 2: Manikanta Reddy Thikkavarapu

Introduction:

Finding the shortest path between a number of points and places that must be visited is the goal of the algorithmic problem known as the "traveling salesman problem" (TSP). The cities a salesperson might visit are the points in the problem statement.

Aim:

The goal of the issue is to determine the shortest path that travels through each listed city exactly once before turning around and returning to the starting location.

The issue is named after the analogy of a salesman who must visit several towns and wants to travel the fewest amount of distance possible while doing so. The issue has significant applications in network routing, logistics, and other fields.

The TSP is an NP-hard problem, which means that it is computationally impractical to find an accurate solution for large instances of the problem. However, there are a variety of approximation algorithms and heuristics that can be applied to quickly find effective answers.

Approach:

The traveling salesman problem (TSP) can be solved in a number of ways, including accurate and heuristic methods.

Exhaustively examining every potential solution, exact approaches seek to identify the best answer to the issue. These techniques include dynamic programming, cutting-plane algorithms, and branch-and-bound algorithms. These techniques, nevertheless, are computationally expensive and might not be practical for widespread cases of the issue.

Even though the solution is not always optimal, heuristic approaches try to find a good one in a fair period of time. These techniques consist of Nearest neighbor, Insertion, Simulated annealing, and Genetic Algorithms

In our project, we are going with the Simulated annealing approach.

Simulated annealing approach: This is a probabilistic algorithm that starts with an initial solution and iteratively makes small random changes to the solution. The changes are accepted or rejected based on a probability function that gradually decreases over time, similar to the process of annealing in metallurgy.

Program:

Data Structures & Classes: Data structures and classes used in our project are mentioned below:

1. Optimization classes: AntColonyOptimization, simulated annealing, ThreeOptForSA, ThreeOptSwapOptimization, and TwoOptSwapOptimization.
2. Entity class: Point
3. Preprocess class: CsvReader
4. Algorithm classes: Christofides, GreedyAlgorithm, and MinimumSpanningTree