ReflexActInt2_A01645033

The Kruskal algorithm was applied to determine the optimal way to connect all neighborhoods with fiber optic cables, minimizing the total wiring cost. It constructs a Minimum Spanning Tree (MST) by adding edges in increasing order of weight while avoiding cycles until all nodes are connected. Kruskal was selected because of its efficiency and its simplicity when implemented with the union find data structure, which detects cycles in almost constant time. Its time complexity is O(E log E), where E is the number of edges. In this context, Kruskal allowed for the design of a low-cost, fully connected network, ensuring communication between all neighborhoods with the minimum possible amount of cable.

To find the shortest possible route visiting each neighborhood exactly once and returning to the starting point, the Nearest Neighbor heuristic was used. This algorithm selects the closest unvisited node at each step until all nodes are visited.

Although it does not guarantee the optimal global solution, it is extremely fast and practical, especially for large datasets. Its time complexity is O(n2), since it compares each node with every other node. In this project, it served as an efficient approximation for modeling delivery routes for mail or advertisements, achieving a balance between computational efficiency and route quality. It also illustrates how local optimization (choosing the nearest next node) can yield strong results in realistic scenarios without excessive computational cost.

The Edmonds-Karp algorithm was used to calculate the maximum flow of information between a source node and a sink node in a directed network, considering capacity constraints. It is an optimized version of the Ford-Fulkerson algorithm, using Breadth-First Search (BFS) to find the shortest augmenting paths in the residual graph.

Edmonds-Karp guarantees a polynomial time complexity of O(VE2), which makes it more predictable than the original Ford-Fulkerson method. In this project, it modeled how data flows through the network and helped identify bottlenecks caused by limited capacity or interference. The algorithm’s iterative update of residual capacities provided an accurate representation of how network throughput adapts under congetions.

Finally, the Voronoi diagram was applied to analyze the geographical distribution of service centers. This computational geometry technique partitions the plane into regions, where each region contains all points closer to one generator point than to any other. The Voronoi diagrams rely on Euclidean distance and are often generated via Delaunay triangulation, with an average time complexity of O(n log n). In the project, they provided a clear visualization of coverage areas, helping determine which exchange should serve a new customer based on proximity.