# Routing Algorithm for Ocean Shipping and Urban Deliveries

Project nº2 for the course of Algorithms Design

by:

- Guilherme Santos up202105090
- Alexandre Lopes up202207015
- Tomás Silva up202108698

# Backtracking Algorithm

Backtracking is a general algorithmic technique that incrementally builds candidates for the solutions and abandons a candidate ("backtracks") as soon as it determines that this candidate cannot possibly be completed to a valid solution.

The algorithm starts at the initial node (e.g., node 0) and explores all possible paths by visiting each city exactly once and returning to the starting node. The algorithm uses a recursive function to explore all possible routes. If a route doesn't lead to a solution (i.e., it doesn’t visit all cities or isn't the shortest path), the algorithm backtracks to try another path.

To optimize the algorithm we used a memoization table, what this does is that when it goes to check a result it verifies if the result has already been calculated, if yes it gets the result from the table. We also used pruning, what this does is that it prunes paths that are guaranteed not to lead to an optimal solution. Finally we implemented a base case that if all vertices have been visited that specific solution would terminate.

It guarantees the best solution every time.

All these optimizations lead to having a time complexity of O(n^2*2^n), instead of O(n!).

# Triangle Approximation Heuristic

The triangular approximation heuristic leverages the triangular inequality to ensure a 2-approximation solution for the TSP.

The algorithm starts by constructing a MST of the graph using Prim's algorithm. This MST connects all nodes with the minimum total edge weight and no cycles. After that we perform a preorder walk (depth-first traversal) of the MST starting from node 0. Finally we Convert the traversal into a Hamiltonian cycle by shortcutting repeated visits while maintaining the triangular inequality.

This algorithm leads to a time complexity of O((V+E)log(V)), where V is the number of vertices and E the number of Edges.

# Other Heuristics : Christofides

The Christofides algorithm is an approximation algorithm for solving the TSP. It guarantees a solution within a factor of 1.5 of the optimal tour length.

The algorithm starts by constructing an MST, after that it finds a minimum weight perfect matching. It then combines the MST and the Matching by adding the edges from both to form an Eulerian multigraph. Finally, it converts the Eulerian Circuit to an Hamiltonian Cycle by shortcutting repeated vertices while maintaining the triangle inequality, and it calculates the total cost.

In this exercise we realized that using a hash map for the vertex set would be more optimal so we adopted that strategy.

# Other Heuristics : Nearest Neighbour

The nearest Neighbour algorithm is a simple and intuitive heuristic for solving the TSP. We start at node 0 and repeatedly visit the nearest unvisited node until all nodes are visited. It does not guarantee an optimal solution but often provides a good approximation quickly.

# Our Results

| | Backtracking | Triangular | Christofides | Nearest Neighbour |
|---|---|---|---|---|
| Duration (sec) | 2.17e+03 | 1.17e-02 | 6.52e-03 | 3.04e-04 |
| Min. Cost | 228387.7 | 364937.2 | 364937.5 | 300951.6 |

These tests were performed fully connected graphs with 25 edges.

The duration is an approximation as the result will not always be the same.