Smart systems and computational intelligence

mariam mohamed abdelmohsen kinawy 230540

mariam mohamed foaad 2305372

mariam gerges 2305397

menna yasser 2305399

Vehicle Routing Problem (VRP) Optimization Report

This application is developed to optimize vehicle routing for delivery trucks under weight constraints using three algorithms:

- Ant Colony Optimization (ACO)
- Greedy Algorithm
- Nearest Neighbor Algorithm

The system reads a city-to-city distance matrix and customer orders with destination and weight, then assigns orders to trucks based on a weight limit and optimizes the delivery routes for each truck.

System Description

Data Input

- distance.csv provides a symmetric matrix of distances between cities.
- order_small.csv contains order IDs, destination cities, and weights.

Core Functionalities:

- Order grouping based on truck capacity.
- Route optimization using user-selected algorithm.
- Calculation of total delivery distance and cost.

Algorithms Implemented

1. Ant Colony Optimization (ACO)

- Nature-Inspired Metaheuristic:
- Mimics how ants find the shortest paths using pheromones.
- Parameters Used:
- n_ants=10, n_best=3, n_iterations=30, decay=0.1, alpha=1, beta=2
- Strengths:
- Suitable for complex, large-scale route problems.
- Can escape local optima.
- Weaknesses:
- Slower due to iterative nature.
- Requires parameter tuning.

2. Greedy Algorithm

Deterministic Heuristic:

Always chooses the nearest unvisited city from the current location.

Strengths:

- Fast and simple.
- Good baseline performance.
-

Weaknesses:

- Can get stuck in suboptimal paths.
- Doesn’t backtrack or consider future choices.

Performance Comparison

Criteria	ACO	Greedy
Accuracy	High (better routes)	Moderate
Speed	Slower	Very fast
Scalability	Good for large problems	Limited in complex cases
Route Quality	More optimal paths found	Often longer total route
Robustness	Handles complexity well	Sensitive to input order
Execution Time	Measured in seconds	Measured in milliseconds

Example Output (Sample Truck):

- ACO Route: Depot → City A → City B → City C → Depot
- Greedy Route: Depot → City B → City A → City C → Depot
- ACO Distance: 8450 meters
- Greedy Distance: 9030 meters
- Execution Time ACO: 0.75 seconds
- Execution Time Greedy: 0.02 seconds

VRP using ACO / Greedy / Nearest Neighbor

Enter Max Truck Weight (kg):

230000

Choose Algorithm:

ACO

Run Optimization

Number of Trucks Used: 1

==========

Truck_1

Orders: A140109, A140112, A140112, A140112, A140112, A190223, A190225, A190226, A190226, A190226

Total Weight: 192800.00 kg

Route: City_24 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_24

Distance: 3603730.00 meters

Execution Time: 0.1367 seconds

Total Delivery Distance for All Trucks: 3603730.00 meters

Total Estimated Delivery Cost: 1801865.00 currency units

VRP using ACO / Greedy / Nearest Neighbor

Enter Max Truck Weight (kg):

230000

Choose Algorithm:

Greedy

Run Optimization

Number of Trucks Used: 1

==========

Truck_1

Orders: A140109, A140112, A140112, A140112, A140112, A190223, A190225, A190226, A190226, A190226

Total Weight: 192800.00 kg

Route: City_24 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_54 -> City_53 -> City_24

Distance: 3603730.00 meters

Execution Time: 0.0000 seconds

Total Delivery Distance for All Trucks: 3603730.00 meters

Total Estimated Delivery Cost: 1801865.00 currency units

Conclusion

- Use ACO when:
- Accuracy and route optimality are a priority.
- Problem size is large.
- You're willing to trade execution time for better performance.
- Use Greedy when:
- Fast results are needed.
- Simpler or small-scale problems are being solved.
- A rough solution is acceptable.