Optimizing Vehicle Routes: A Solution to the Capacitated Vehicle Routing Problem with Time Windows

Aarthi Padmanabhan
CMSE 831 – Computational Optimization

College of Natural Science
MICHIGAN STATE UNIVERSITY

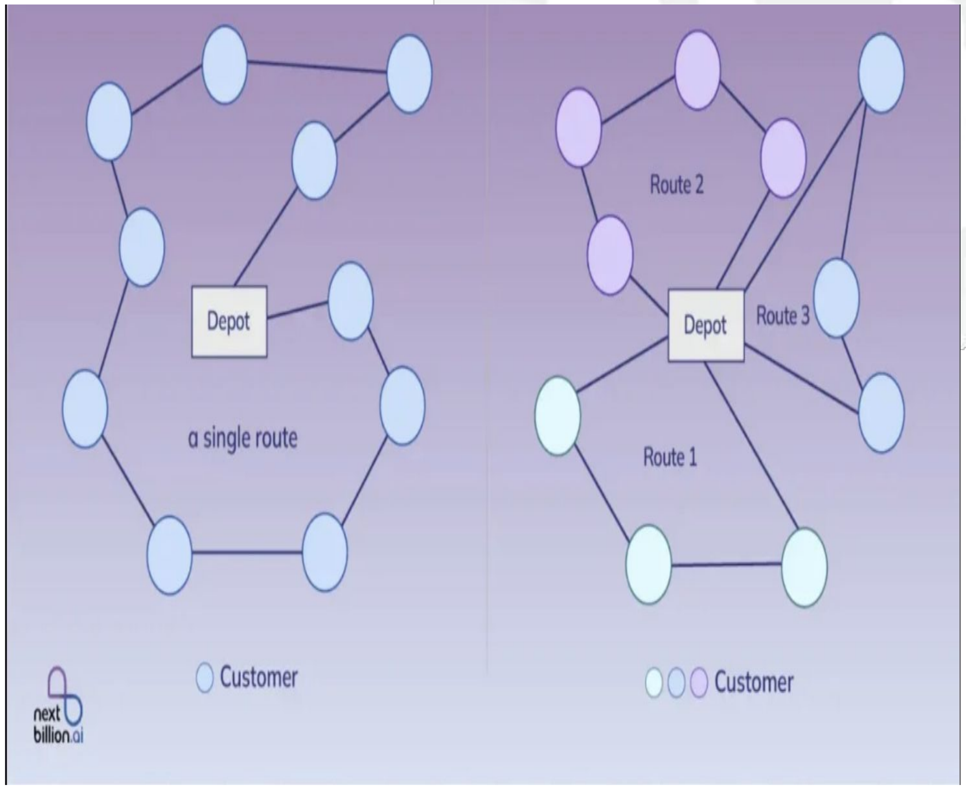
Introduction

- Efficient routing is essential for minimizing costs and improving service in logistics and transportation.

- **Capacitated Vehicle Routing Problem with Time Windows (CVRPTW):**
  A logistics optimization problem where vehicles must deliver goods to customers while adhering to constraints like limited vehicle capacity and specific delivery time windows.

- **Objective:** How can we design efficient vehicle routes to minimize total travel distance while meeting capacity and time window constraints?

- It enhances cost-efficiency, reduces environmental impact, ensures timely deliveries, and addresses real-world logistics challenges.



Dataset Overview

- **Dataset Sizes**: Three datasets with 20 nodes, 50 nodes, and 100 nodes.

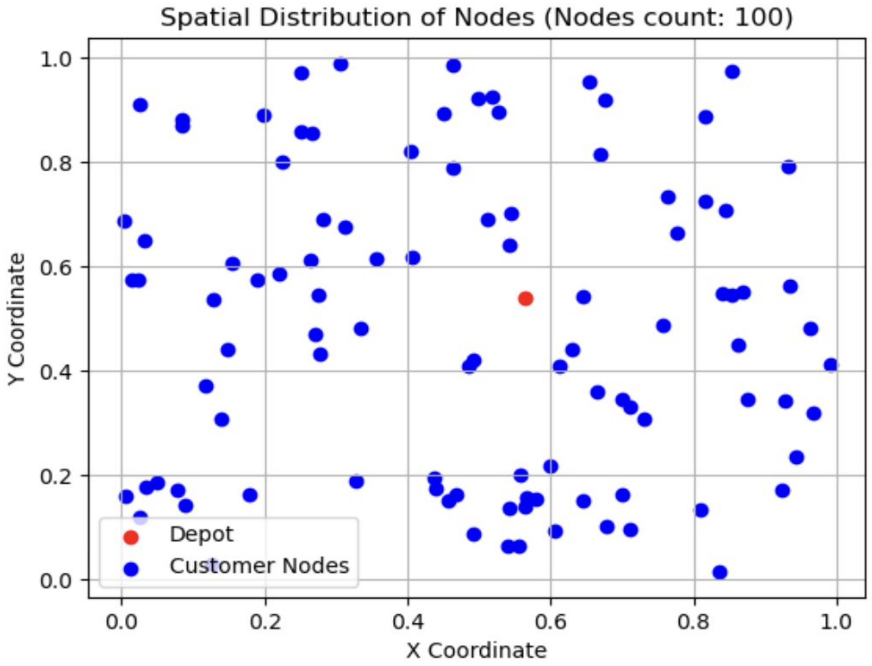
- **Key Fields**:
    - **Node Locations**: Coordinates of depot and customer nodes.
    - **Demands**: Customer requirements for delivery.
    - **Time Windows**: Service time constraints for each customer.
    - **Vehicle Capacity**: Maximum load a vehicle can carry.

- **Purpose**: Provides a realistic framework to test routing and scheduling algorithms for the CVRPTW.

Visualizations

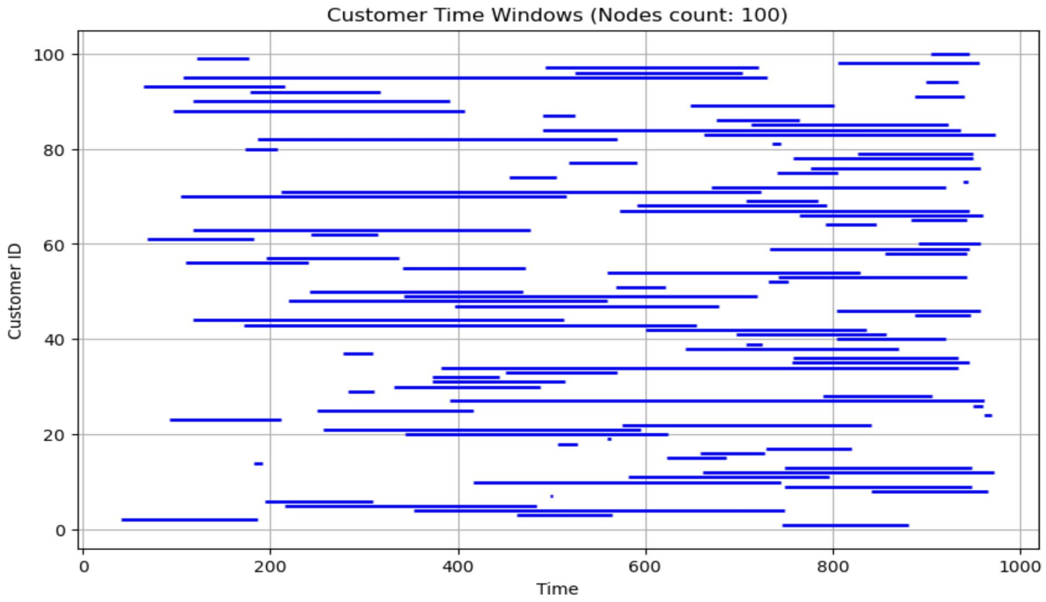
1. **Spatial distribution of nodes**

   - Highlights the scattered nature of customers, influencing route design.



2. Customer Time Windows

- Illustrates overlapping or tight time windows, increasing routing complexity.



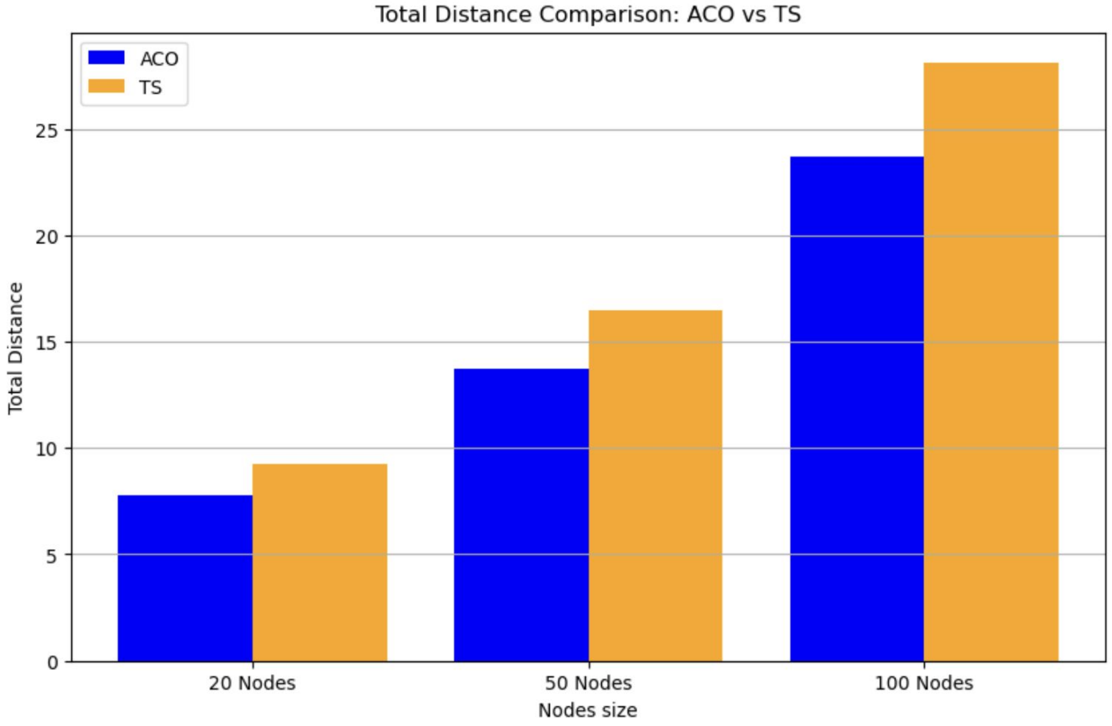
Methodologies

Optimization Algorithms Used:

1. **Ant Colony Optimization (ACO):**
   - Inspired by the foraging behavior of ants.
   - Uses **pheromone trails** and **probabilistic exploration** to construct efficient routes.
   - Parameters tuned: alpha (pheromone influence), beta (heuristic influence), rho (evaporation rate), and q (pheromone deposit factor).
2. **Tabu Search (TS):**
   - A metaheuristic optimization technique.
   - Explores the solution space using a **Tabu List** to avoid cycling back to recent solutions.
   - Parameters tuned: maximum iterations and Tabu List size.

Results

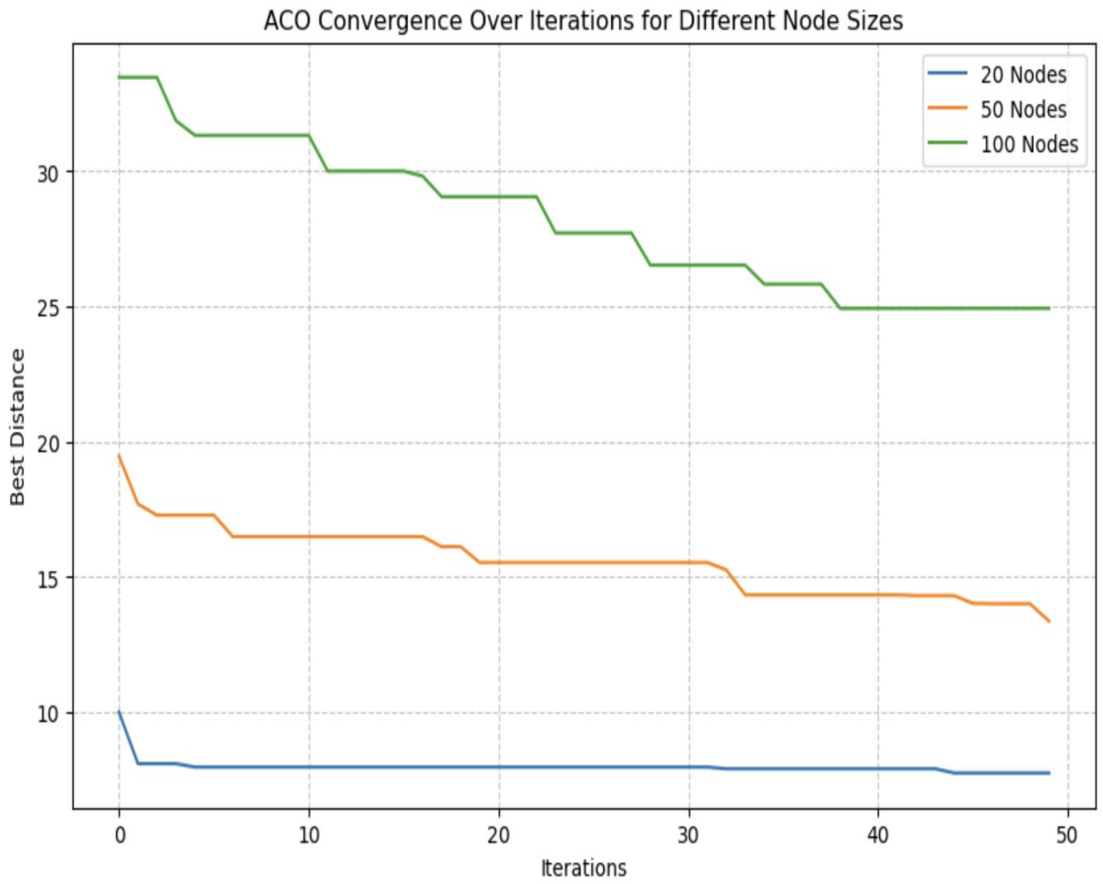
1. **Total Distance Comparison**
   - ACO consistently outperformed TS in minimizing total travel distance, with shorter distances observed across all dataset sizes.

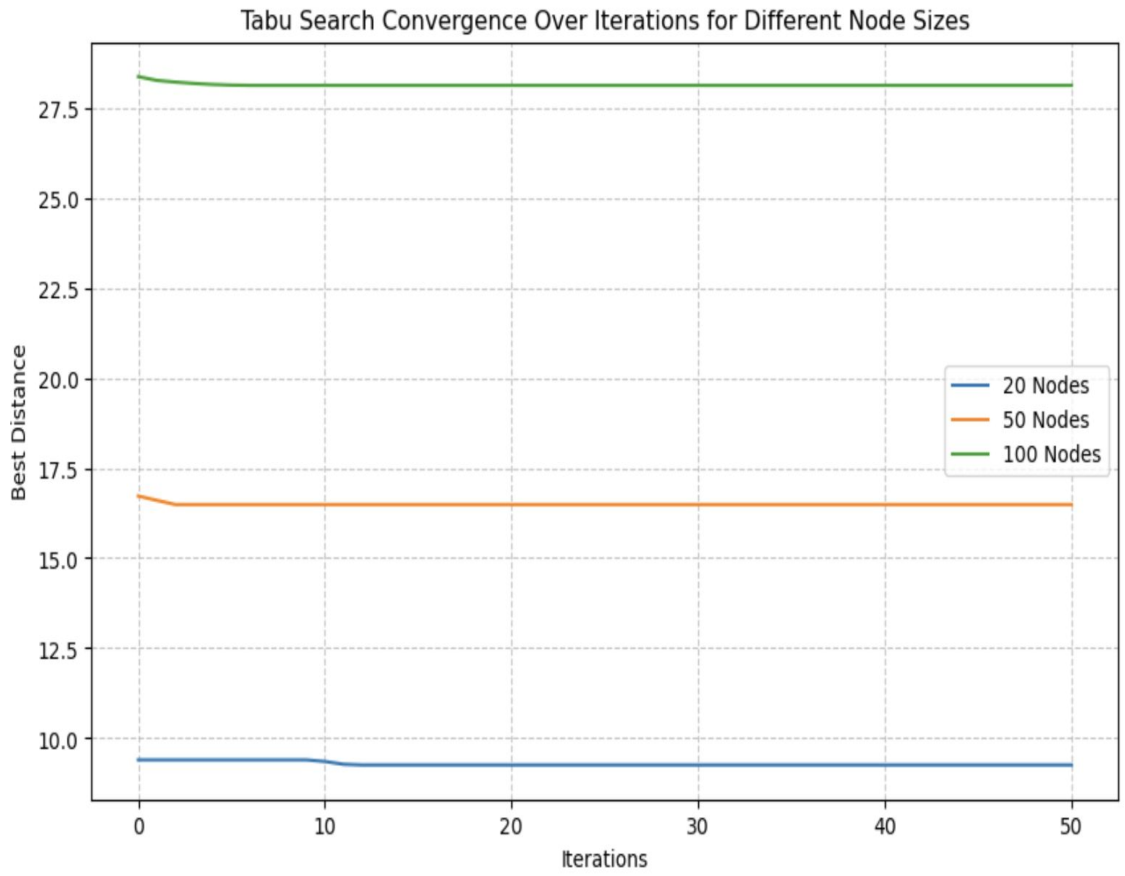


2. **Number of Routes Comparison**
   - ACO generated fewer routes than TS, showcasing its efficiency in clustering nodes and optimizing vehicle usage.

Results

3. Convergence visualizations





Conclusion & Future Work

- ACO outperformed TS in minimizing travel distances but required higher computational effort.
- TS converged faster with simpler implementation but produced slightly longer routes.
- Both algorithms scaled effectively, with ACO achieving better optimization quality.

Future Work

- Explore hybrid approaches combining ACO and TS.
- Test scalability on larger datasets (500+ nodes).
- Incorporate real-world constraints like traffic and dynamic demands.