Ant Colony Optimization for the Traveling Salesman Problem

A Project Summary

---

1. Introduction

- **The Traveling Salesman Problem (TSP):**
  - A classic optimization challenge: Find the shortest possible route that visits a set of cities exactly once and returns to the starting city.
  - NP-hard: Exact solutions become computationally infeasible for many cities.
- **Ant Colony Optimization (ACO):**
  - A metaheuristic inspired by the foraging behavior of real ants.
  - Ants deposit pheromones on paths; stronger pheromone trails attract more ants.
  - Effectively finds near-optimal solutions for complex problems like TSP.
- **ACO for TSP in this Project:**
  - Artificial ants build tours, choosing next cities based on pheromone levels and distance (heuristic).
  - Pheromone trails are updated: evaporation reduces old trails, deposition reinforces good tours.
  - Iterative process to converge on a short tour.

---

2. Project Objectives

- To implement an Ant Colony Optimization algorithm to solve a variant of the Traveling Salesman Problem.
- To analyze the performance of the ACO algorithm under different configurations:
  - **Number of Cities:** 10 cities vs. 20 cities.
  - **Number of Ant Agents:** 1, 5, 10, and 20 ants for each city set.
- To observe the development of the pheromone map and the optimal path over 50 iterations for each configuration.
- To compare results and draw conclusions about the impact of these parameters.

---

3. Methodology

- **ACO Parameters Used:**
  - Alpha (Pheromone Influence): 1.0
  - Beta (Heuristic Influence): 3.0
  - Rho (Pheromone Evaporation): 0.3
  - Q (Pheromone Deposit Factor): 100
  - Initial Pheromone: 0.1
  - Number of Iterations per run: 50
- **Distance Matrix Generation:**
  - Symmetric matrices with random integer distances between 3 and 50 (inclusive).
  - Generated ONCE for the 10-city set and ONCE for the 20-city set.
  - These fixed matrices were used for all ant agent configurations within their respective city sets.
- **Data Recording:**
  - Pheromone map state and current optimal path (and its length) recorded every 10 iterations (10, 20, 30, 40, 50).

---

4. Results: 10 Cities

- **Distance Matrix:** A 10x10 matrix was generated (details in the full report).
- **Best Tour Lengths Found (after 50 iterations):**
  - 1 Ant: 92
  - 5 Ants: 92
  - 10 Ants: 92
  - 20 Ants: 92
- **Key Observations (10 Cities):**
  - All of the ant configurations found the best optimal path but after different number of iterations.
  - The fastest configuration to find the best optimal path was 20 ants.
- **Pheromone & Path Progress:**
  - Pheromone maps showed increasing concentration on optimal paths over iterations.
  - Optimal path lengths generally decreased rapidly in early iterations.

(Detailed tables and pheromone values are available in the full report and raw data files.)

---

5. Results: 20 Cities

- **Distance Matrix:** A 20x20 matrix was generated (details in the full report).
- **Best Tour Lengths Found (after 50 iterations):**
  - 1 Ant: 177
  - 5 Ants: 161
  - 10 Ants: 164
  - 20 Ants: 161
- **Key Observations (20 Cities):**
  - 20. 5 ant configuration found the best solution (length 161).
  - Increasing ants generally improved solution quality for this larger problem except for 10 ants configuration.
  - ACO is stochastic and influenced by randomness, pheromone evaporation, and limited iterations, that explains why 5 ant configuration found the better route than 10 ants configuration.
- **Pheromone & Path Progress:**
  - Similar to 10-city case, pheromones concentrated on good paths.
  - The larger search space meant initial solutions were further from optimal.

(Detailed tables and pheromone values are available in the full report and raw data files.)

---

6. Analysis & Discussion

- **Impact of Ant Agent Count:**
  - Multiple ants consistently outperformed a single ant.
  - For 10 cities: 5 ants seemed optimal for efficiency and quality.
  - For 20 cities: More ants (20) yielded the best result, suggesting larger colonies are better for larger problems.
- **Impact of Problem Size (Number of Cities):**
  - 20-city problem was significantly harder (longer tours, longer convergence time).
  - ACO adapted, but efficiency of a fixed ant number decreased with problem size.
- **Comparison: 10 Cities vs. 20 Cities (Same Ant Count):**
  - The algorithm effectively handled both sizes, but the 20-city problem naturally resulted in longer paths and required more exploration (seen in convergence patterns).

---

7. Conclusion

- Ant Colony Optimization is an effective heuristic for the Traveling Salesman Problem.
- **Number of Ant Agents is Crucial:** More ants generally lead to better exploration and solution quality, especially for larger, more complex problems.
- **Problem Size Matters:** Larger problems require more computational effort (more ants and/or iterations) to find good solutions.
- **Key Findings:**
    - For 10 cities, 5 ants provided a good balance.
    - For 20 cities, 20 ants yielded the best tour.
    - The ACO algorithm successfully demonstrated pheromone-guided convergence towards good solutions within 50 iterations for all tested configurations.
