

A Study of Heuristic Solutions to the Traveling Salesman Problem

Vandan Amin

Student ID: 44006979

Department of Computer Science, Central Washington University
CS465/565 - Scientific Computing

Dr. Donald Davendra

November 9, 2024

Contents

0.1	Introduction	1
0.2	Simulated Annealing	1
0.2.1	Algorithm Overview	1
0.2.2	Advantages	1
0.2.3	Disadvantages	2
0.2.4	Industrial Applications	2
0.3	Genetic Algorithm Solver	2
0.3.1	Algorithm Overview	2
0.3.2	Advantages	2
0.3.3	Disadvantages	3
0.3.4	Industrial Applications	3
0.4	Conclusion	3
0.5	References	4

Abstract

The Traveling Salesman Problem (TSP) is a well-known combinatorial optimization problem with applications in various industries, including logistics, manufacturing, and telecommunications. This report presents an overview of heuristic solutions used to solve the TSP, focusing on practical and easy-to-implement approaches. Specifically, the study includes an analysis of two different solvers: Simulated Annealing and Genetic Algorithm. The report evaluates the performance of these solvers based on their efficiency, scalability, and practicality for solving real-world TSP instances.

0.1 Introduction

The Traveling Salesman Problem (TSP) is one of the most studied combinatorial optimization problems in the field of operations research and computer science. Given a list of cities and the distances between each pair, the objective of TSP is to find the shortest possible route that visits each city exactly once and returns to the starting point. TSP has numerous practical applications, including route planning, circuit board manufacturing, and supply chain optimization. This report explores two popular heuristic approaches to solving the TSP: Simulated Annealing and Genetic Algorithm, highlighting their strengths, limitations, and use cases in industry.

0.2 Simulated Annealing

Simulated Annealing (SA) is a probabilistic heuristic used to find approximate solutions to optimization problems like TSP. It is inspired by the annealing process in metallurgy, where a material is heated and then slowly cooled to decrease defects and find a low-energy state.

0.2.1 Algorithm Overview

The Simulated Annealing algorithm begins with an initial solution and iteratively makes small changes to the current solution. The new solution is accepted with a probability that depends on the difference in cost between the current and new solutions and a parameter called temperature, which gradually decreases over time.

Mathematical Description:

Let $f(x)$ be the cost function evaluating the quality of solution x , and let T be the temperature. The algorithm proceeds as follows:

1. Start with an initial solution x_0 and an initial temperature T_0 .
2. Set the current solution $x = x_0$ and temperature $T = T_0$.
3. Repeat until the stopping criterion is met (e.g., a sufficiently low temperature or a maximum number of iterations):
 - (a) Generate a new solution x' by making a small random change to x .
 - (b) Calculate the change in cost $\Delta f = f(x') - f(x)$.
 - (c) If $\Delta f < 0$, accept the new solution ($x = x'$).
 - (d) Otherwise, accept x' with probability $e^{-\Delta f/T}$.
 - (e) Decrease the temperature T according to a cooling schedule.

0.2.2 Advantages

- Simple to implement and can escape local optima.
- Capable of finding good approximate solutions for large problem instances.
- Flexible and can be adapted for various problem types.

0.2.3 Disadvantages

- Does not guarantee an optimal solution.
- Performance depends on the cooling schedule and parameter tuning.
- May require a long runtime to converge to a good solution.

0.2.4 Industrial Applications

Simulated Annealing is used in industries where finding a near-optimal solution is acceptable, and the problem space is large. For example, it is used in telecommunications for network optimization and in manufacturing for job scheduling.

0.3 Genetic Algorithm Solver

The Genetic Algorithm (GA) is a heuristic approach inspired by the process of natural selection. It is commonly used to find approximate solutions to complex optimization problems like TSP.

0.3.1 Algorithm Overview

Genetic Algorithms work by simulating the process of evolution. A population of potential solutions (individuals) is generated, and each individual is evaluated based on a fitness function. The best individuals are selected to create offspring through crossover and mutation, with the goal of improving the quality of solutions over successive generations.

Mathematical Description:

Let $P(t)$ represent the population at generation t , and let $f(x)$ be the fitness function evaluating the quality of solution x . The GA proceeds as follows:

1. Initialize a population $P(0)$ of potential solutions.
2. Evaluate the fitness of each individual in $P(0)$ using $f(x)$.
3. Repeat until a stopping criterion is met (e.g., a maximum number of generations):
 - (a) Select individuals from $P(t)$ to form a mating pool.
 - (b) Apply crossover and mutation operators to generate offspring.
 - (c) Form the new population $P(t + 1)$ from the offspring.
 - (d) Evaluate the fitness of each individual in $P(t + 1)$.

0.3.2 Advantages

- Capable of finding good approximate solutions in a reasonable amount of time.
- Scalable to larger problem instances compared to exact methods.
- Flexible and can be adapted to various problem constraints and objectives.

0.3.3 Disadvantages

- Does not guarantee an optimal solution.
- Performance depends on the choice of parameters, such as population size and mutation rate.
- May require significant computational resources for large populations or many generations.

0.3.4 Industrial Applications

Genetic Algorithms are widely used in industries where finding a near-optimal solution quickly is more important than finding the exact optimal solution. For instance, logistics companies use GAs to optimize delivery routes, reducing fuel consumption and improving delivery times.

0.4 Conclusion

The Traveling Salesman Problem is a challenging optimization problem with many real-world applications. The choice of a suitable solver depends on the specific requirements of the problem instance. Simulated Annealing is a simple and easy-to-implement method that provides good approximate solutions, making it suitable for large problem instances where escaping local optima is important. On the other hand, Genetic Algorithms are more suitable for larger instances where a good approximate solution is needed quickly. Understanding the strengths and limitations of each approach is crucial for selecting the right method for solving TSP in practice.

0.5 References

1. Applegate, D. L., Bixby, R. E., Chvátal, V., Cook, W. J. (2006). The Traveling Salesman Problem: A Computational Study. Princeton University Press.
2. Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., Shmoys, D. B. (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. Wiley.
3. Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press.
4. Mitchell, M. (1998). An Introduction to Genetic Algorithms. MIT Press.
5. Laporte, G. (1992). The Traveling Salesman Problem: An Overview of Exact and Approximate Algorithms. *European Journal of Operational Research*, 59(2), 231-247. [https://doi.org/10.1016/0377-2217\(92\)90138-Y](https://doi.org/10.1016/0377-2217(92)90138-Y)