

# Revised Time Complexity Analysis of Solving TSP with Genetic Algorithms Including the Number of Cities

## 1 Introduction

The Traveling Salesman Problem (TSP) seeks the shortest path visiting each of a set of cities exactly once and returning to the origin city. Genetic Algorithms (GAs) offer a heuristic approach to find approximate solutions for the TSP by mimicking natural evolutionary processes.

## 2 Genetic Algorithm Overview

Genetic Algorithms address optimization problems by evolving a population of candidate solutions through selection, crossover (recombination), and mutation over multiple generations, aiming to improve the fitness of solutions.

## 3 Time Complexity Analysis Including the Number of Cities ( $N$ )

The time complexity of applying Genetic Algorithms to solve the TSP is influenced by several key factors, including the number of cities ( $N$ ), population size ( $P$ ), number of generations ( $G$ ), and the complexities of the crossover ( $C(N)$ ), mutation ( $M(N)$ ), and selection ( $S(N)$ ) operations.

Given these parameters, we can express the overall time complexity as:

$$O(G \cdot (P \cdot (S(N) + P \cdot (C(N) + M(N))))) \quad (1)$$

### 3.1 Detailed Explanation

- The selection operation's complexity,  $S(N)$ , may depend on evaluating the fitness of each solution, which in turn relies on computing the total distance of a path through  $N$  cities.

- The complexities of the crossover and mutation operations,  $C(N)$  and  $M(N)$ , are functions of  $N$  because they involve manipulating sequences representing paths through the  $N$  cities.
- Each generation involves selecting solutions based on fitness, generating new solutions through crossover, and applying mutations, leading to the multiplication by  $G$ .

## 4 Impact of the Number of Cities ( $N$ )

- The direct involvement of  $N$  in the complexities of  $C(N)$ ,  $M(N)$ , and  $S(N)$  highlights how the number of cities affects the algorithm's performance. Specifically, operations that manipulate or evaluate paths become more complex as  $N$  increases.
- The encoding of solutions and the calculation of fitness values, essential for the selection process, inherently depend on  $N$ , emphasizing its role in the GA's computational demands.

## 5 Conclusion

Incorporating  $N$  into the time complexity analysis of Genetic Algorithms for solving the TSP provides a more comprehensive understanding of the factors influencing the algorithm's efficiency. The revised notation underscores the significance of the number of cities, revealing its impact on the complexity of selection, crossover, and mutation operations. This analysis demonstrates the importance of considering  $N$  when evaluating the performance and scalability of Genetic Algorithms in the context of the TSP.