

Research Article

Expected Length of the Shortest Path of the Traveling Salesman Problem in 3D Space

Hongtai Yang , Xiaoqian Lu , Xinan Zhou, Rong Zheng , Yugang Liu ,
and Siyu Tao 

School of Transportation and Logistics, National United Engineering Laboratory of Integrated and Intelligent Transportation, National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Institute of System Science and Engineering, Southwest Jiaotong University, Chengdu 610000, China

Correspondence should be addressed to Yugang Liu; liuyugang@swjtu.edu.cn

Received 1 October 2021; Revised 21 March 2022; Accepted 31 March 2022; Published 2 May 2022

Academic Editor: Zhixiang Fang

Copyright © 2022 Hongtai Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Finding the shortest path of the traveling salesman problem (TSP) is a typical NP-hard problem and one of the basic optimization problems. TSP in three-dimensional space (3D-TSP) is an extension of TSP. It plays an important role in the fields of 3D path planning and UAV inspection, such as forest fire patrol path planning. Many existing studies have focused on the expected length of the shortest path of TSP in 2D space. The expected length of the shortest path in 3D space has not yet been studied. To fill this gap, this research focuses on developing models to estimate the expected length of the shortest path of 3D-TSP. First, different experimental scenarios are designed by combining different service areas and the number of demand points. Under each scenario, the specified number of demand points is randomly generated, and an improved genetic algorithm and Gurobi are used to find the shortest path. A total of 500 experiments are performed for each scenario, and the average length of the shortest path is calculated. The models to estimate the expected length of the shortest path are proposed. Model parameters are estimated and k-fold cross-validation is used to evaluate the goodness of fit. Results show that all the models fit the data well and the best model is selected. The developed models can be used to estimate the expected length of the shortest path of 3D-TSP and provide important references for many applications.

1. Introduction

The traveling salesman problem (TSP) is a typical combinatorial optimization problem and is also NP-hard [1, 2]. It is explained as a salesman trying to find the shortest path (or minimum travel cost) through a given set of cities, where the travel distances (or travel costs) between cities are also given, under the requirement of visiting each city only once and returning to the starting city [3]. TSP has a wide range of applications in transportation, logistics, computer science, genetics, and other fields [4–9]. The traveling salesman problem in three-dimensional space (3D-TSP) is a generalization of the classic TSP. 3D-TSP refers to an executor, such as a drone that tries to find the shortest path in the three-dimensional space and returns to the starting point after visiting each of the given demand points. The difference between 3D-TSP and the classic TSP is that the coordinates

of the cities or demand points are no longer represented by two-dimensional coordinates but by three-dimensional coordinates. Many problems, such as the unmanned aerial vehicle (UAV) patrolling for inspection, could be regarded as a TSP in 3D space [10]. In these applications, it is important to know the expected length of the shortest path of the 3D-TSP under a given area and the number of demand points in order to determine the zoning strategy of the service area and the number of executors needed.

So far, many scholars have studied the expected length of the shortest path and proposed estimation models that adopt the number of cities and service area parameters as independent variables [11–15]. However, models of the expected length of the shortest path of 3D-TSP have not yet been studied. As a result, this study contributes to determining the influence of the number of demand points and the relevant parameters of the service area on the expected length of the

shortest path of 3D-TSP and establishing the estimation models of the expected length.

The remainder of this paper is structured as follows. Section 2 provides a review of the related literature on the algorithms to solve TSP and the estimation models of the expected length of the shortest path of TSP. Section 3 gives the formal definition of 3D-TSP, its objective function, and the improved genetic algorithm (GA) used in this paper. Section 4 describes the simulation experiment scenarios used in this paper, proposes the variables that can be used to construct the models, builds the models based on the results of the simulation experiment, and evaluates the goodness of fit of the models. Section 5 provides a summary of the entire study.

2. Literature Review

No research has yet been conducted on the expected length of the shortest path of the 3D TSP. Thus, the studies related to the expected length of the shortest path of the 2D TSP are reviewed.

Beardwood et al. [11] proposed that the expected length of the shortest path through N points in abounded plane region is always proportional to \sqrt{NA} when N is relatively large. In two-dimensional space, this result can be used to estimate the expected length of the shortest path of the 2D TSP. The specific function is provided as Model (1) in Table 1. In the model, L is the expected length of the shortest path of the 2D TSP, A is the area of the service area, N is the number of points in the area, and α is the constant of proportionality, which is not related to the shape of the area. After that, many scholars have estimated the value of α [16–23].

Based on the study of Beardwood et al., Daganzo [12] considered the influence of shape on the expected length of the shortest route and proposed Model (2), as shown in Table 1. In the model, Φ is an unknown constant, δ is the density of points ($\delta = N/A$), l is the width of the rectangle, N is the number of points, and A is the area of the service area. In the same year, Daganzo [13] proposed Model (3) to estimate the expected length when there is a fleet of vehicles operating in an area with heterogeneous demand, as shown in Table 1. In the model, α_1 is an unknown constant, A is the area of the subregion, N is the number of points in the subregion, D is the average straight-line distance from each point in the subregion to the depot, and C is the maximum number of times that a car can stop.

Chien [14] presented seven functions for the length of the shortest path of TSP on the basis of the actual locations of customers and the depot and conducted a large number of Monte Carlo experiments to determine the values of related parameters. The results show that when the service area and randomly distributed points in the area are provided, Model (4) in Table 1 performs the best and can obtain a highly accurate approximation of the expected length. In this function, α_1 and α_2 are unknown constants, RBAR is the average straight-line distance between the depot (origin) and customers, RECTAN2 is the smallest rectangular area that

TABLE 1: Proposed models in previous research.

| Model number | Author/s | Model |
|--------------|------------------|---|
| (1) | Beardwood et al. | $L \approx \alpha\sqrt{NA}$ |
| (2) | Daganzo | $L = \Phi (\delta l^2) \sqrt{NA}$ |
| (3) | Daganzo | $L \approx 2D (N/C) + \alpha_1 \sqrt{NA}$ |
| (4) | Chien | $L = \alpha_1 \text{RBAR} + \alpha_2 \sqrt{\text{RECTAN2}(N-1)}$ |
| (5) | Kwon et al. | $L = (\alpha_1 + \alpha_2 N + \alpha_3 S/N) \sqrt{NA}$ |
| (6) | Kwon et al. | $L = (\alpha_1 + \alpha_2 N + \alpha_3 S/N) \sqrt{NA} + \alpha_4 D$ |

covers all the customers, and N is the size of TSP (number of customers plus 1).

Kwon [15] constructed the regression model and neural network model of the expected length of the shortest path for 2D TSP and compared the results with the model proposed by Beardwood et al., Daganzo, and Robuste et al. [24]. The results show that the model built by Kwon performed best. Different from the previous model, Kwon's model also considers the effect of the aspect ratio S and S/N of the rectangular service area on the expected length of the shortest path. The specific functions of the regression models established by Kwon are shown in Models (5) and (6) in Table 1. In Models (5) and (6), α_1 , α_2 , α_3 , and α_4 are unknown constants; N is the number of points in TSP (the sum of the number of depot and all the customers); S is the ratio of the length to the width of the service area; A is the smallest rectangular area that covers all the points; D is the average straight-line distance from all the customers to the depot. This study primarily refers to Model (5) in Table 1 (proposed by Kwon) to construct our models.

When studying the expected length of the shortest path of the 3D TSP, we need to first solve the given 3D TSP in order to obtain the shortest path. The solution algorithm of 3D-TSP is similar to that of the classical TSP. The only difference is the calculation of the distance matrix. Therefore, this paper adopts the solution algorithm of TSP to find the optimal solution for 3D-TSP. Many scholars have studied various algorithms to solve TSP. Solution algorithms for TSP can be divided into two types: exact algorithm and heuristic algorithm [19]. For small-scale problems, exact algorithms can be used to solve them. For large-scale or medium-scale problems, exact algorithms cannot generate the optimal solution in a reasonably short time, and thus heuristic algorithms are needed to find the approximate optimal solution. Some commonly used algorithms [25–28] are shown in Table 2.

For the exact algorithm, the solver Cplex or Gurobi is usually used when the number of demand points is less than or equal to 100. For heuristics algorithm, GA is a random parallel search algorithm based on natural selection and genetics. It is an efficient method to find the optimal solution without relying on any initial data. Due to its simplicity, GA is widely used to solve TSP. However, it has a certain dependence on the selection of the initial population. As a result, GA is often improved by combining it with other heuristic algorithms. For example, Li et al. [2] proposed to combine GA with other algorithms such as greedy, hill

TABLE 2: Commonly used algorithms for solving TSP.

| Exact algorithm | Heuristic search algorithm |
|--------------------------------|----------------------------|
| Mixed integer programming | Genetic algorithm (GA) |
| Branch-and-bound search method | Simulated annealing |
| Cutting-plane method | Neural network algorithms |
| Dynamic programming | Ant colony optimization |

climbing (HC), and simulated annealing (SA) algorithm to improve GA to achieve better performance. Meng et al. [29] proposed a PBIL algorithm combining GA and competitive learning to explore better solutions. Later, Meng et al. [30] also proposed a variable neighborhood search (VNS) approach that utilizes a two-stage greedy initialization algorithm to generate an initial solution and employs a 2-opt method for local search. Xu et al. [31] proposed a biojective variable neighborhood search (BVNS) approach based on the VNS proposed by Meng et al. Kumar [32] proposed to improve the GA with a greedy algorithm to improve the quality of the initial population. Among them, the combination of GA and the greedy algorithm has generated good results within a reasonable time. Therefore, this paper uses Gurobi and GA combined with the greedy algorithm to obtain the shortest path of the 3D-TSP in different scenarios.

A randomly generated geometric TSP instance [3] assumes each point is independently and randomly generated from the service area based on the uniform distribution. The travel cost between two points is the Euclidean distance between the corresponding points. This study implements relevant experimental designs and randomly generates TSP examples in 3D space. In addition, TSPLIB contains more than 100 TSP instances and 85,900 cities [33]. It provides a testbed for the proposed algorithm.

3. 3D-TSP and Improved GA Algorithm

3.1. 3D-TSP. Let us give a formal definition of the 3D-TSP. Suppose that there is a drone (or performer of action) that needs to visit each of the N demand points once and come back to the origin. It can be formulated over a complete graph $G = (V, E)$, where vertex set $V = \{0, 1, 2, \dots, N-1\}$; and each edge in $(i, j) \in E, i \neq j$, is associated with a weight d_{ij} , which represents a visit cost (or distance) between two demand points i and j . The objective is to determine the route along which the drone departs from the origin, visits each point once, and returns to the origin with the lowest total travel cost.

The integer programming model of 3D-TSP is presented as follows [4]. Binary variable $x_{ij} = 1, i \neq j, i, j \in V$, if the drone passes through the edge (i, j) ; and otherwise, $x_{ij} = 0$. The objective function of the problem is

$$\min F = \sum_{i=0}^{N-2} \sum_{j=1}^{N-1} d_{ij} x_{ij}, \quad (1)$$

Subject to the following constraints: first, the drone departs from the demand point numbered 0 (origin) and returns to the origin, that is,

$$\sum_{j=1}^{N-1} x_{0j} = 1, \quad (2)$$

$$\sum_{i=0}^{N-2} x_{i0} = 1. \quad (3)$$

Each demand point except 0 must be visited by the drone exactly once, that is:

$$\sum_{j=1}^{N-1} x_{ij} = 1, \quad (4)$$

$$\sum_{i=0}^{N-2} x_{ij} = 1. \quad (5)$$

For each demand point, introduce a decision variable $\mu_i, \forall i \in V, \mu_i \geq 0$. μ_i is the access order of demand point i . For example, in $\mu_1 = 5$, it can be understood that point 1 is the fifth point visited from the starting point. M is a very large number. M should be an upper bound of $\mu_i - \mu_j + 1$. Some papers pointed out that the effect will be better if the tightest upper bound is taken, so we take $M = N$ [25]. Construct Miller–Tucker–Zemlin (MTZ) constraint to eliminate sub-tour, that is:

$$\mu_i - \mu_j + Nx_{ij} \leq N - 1, \forall i \in \{0, 1, 2, \dots, N-2\}, \quad (6)$$

$$j \in \{1, 2, \dots, N-1\}, i \neq j.$$

It should be noted that since the classical TSP is NP-hard [1, 2] and 3D-TSP is an extension of the classical TSP in the 3D space, 3D-TSP is also NP-hard. When all the points of the 3D-TSP fall onto the same plane, 3D-TSP turns into the classical TSP.

3.2. Improved GA Algorithm. Due to the complexity of 3D-TSP and the relatively large number of demand points in this study, it is necessary to use heuristic algorithms to solve the 3D-TSPs. As mentioned in the literature review, GA has a good global search ability and is easy to be combined with other algorithms for improvement. Combining the GA with the greedy algorithm can shorten the time to obtain the solution. It is achieved by using the greedy algorithm to improve the initial population of the GA since a high-quality initial population can accelerate the evolution of the GA to quickly reach a satisfactory optimal solution [2, 32]. Therefore, this paper chooses to use the GA combined with the greedy algorithm to find the shortest path of 3D-TSP.

The initial parameters include the population size M , the number of chromosome genes N (that is, the number of demand points), the number of iterations C , the crossover probability P_c , and the mutation probability P_m . The specific steps to perform the algorithm are as follows.

Step 1. Generate the digital codes (chromosomes) of M feasible solutions of 3D-TSP through the greedy algorithm as the initial population (the feasible solution set).

Step 2. Select the inverse of the total length of the path as the fitness function, which indicates the shorter the distance is, the better the fitness function is. The fitness of each individual is calculated by the fitness function.

Step 3. Use the wheel selection mechanism for the selection operation, eliminate individuals with low fitness, and select individuals with high fitness.

Step 4. Randomly select two individuals and perform the partial matching crossover. The specific steps are as follows: first randomly generate two intersection points, determine the matching areas between the two points, and exchange the two matching areas. If the code repetition occurs outside the matching area, it will be replaced based on the corresponding position in the matching area.

Step 5. Randomly select an individual, extract two genes from the individual code, and exchange their positions. In this way, variation of individual code is achieved.

Step 6. The selection, crossover, and mutation of individuals generate a new population for the next generation. After a new population is generated, it is evaluated and then selected, crossed, and mutated. These steps are repeated until the maximum number of iterations or the termination condition of the algorithm is reached. Finally, an approximate optimal solution is found.

The flowchart of the improved GA is as follows (Figure 1).

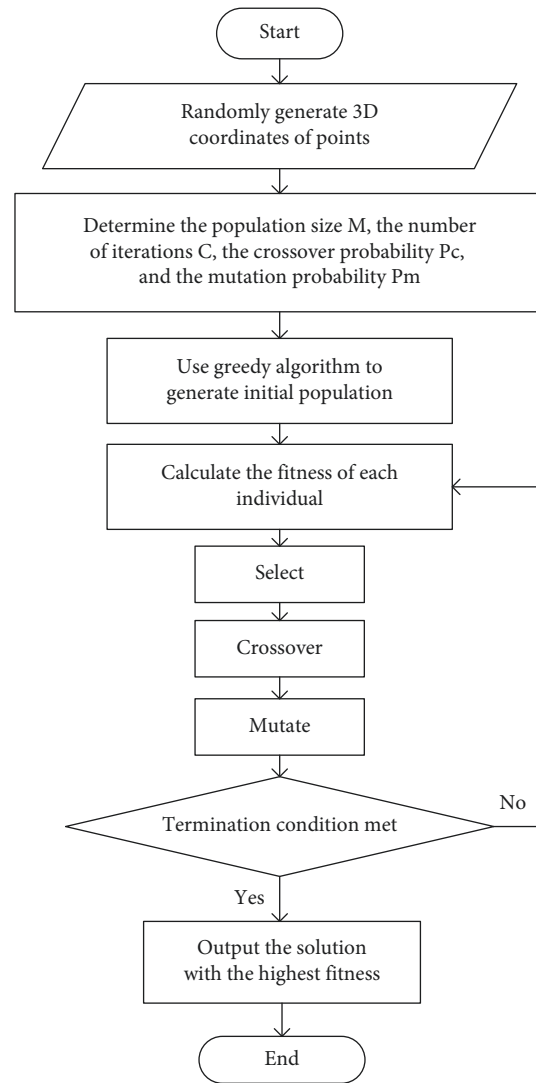


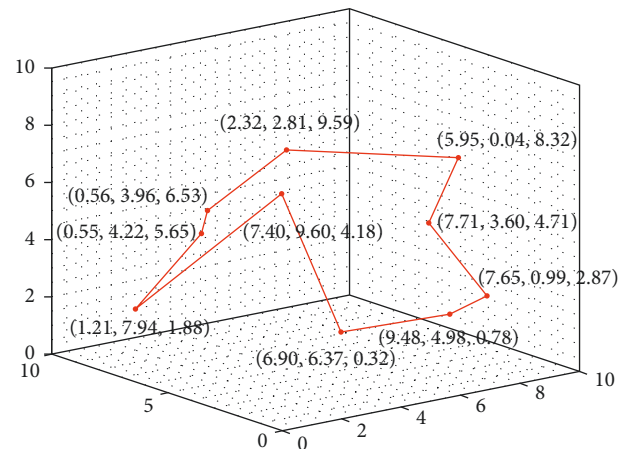
FIGURE 1: Flowchart of the improved GA for the population initialization.

3.3. Illustration. Figure 2 presents an example of the 3D-TSP. In this example, there are 10 demand points and the volume of the service area is 1000 with equal length, width, and height. The red line in the figure is the obtained shortest path. It can be seen from the figure that the length of the shortest path is 42.0285. The coordinates of each demand point are marked.

4. Experiments and Results

In this section, we first design the simulation experiment scenarios according to the previous literature [14, 15, 34], including the setting of the volume, shape, and the number of demand points of the service area. This section first compares the solution obtained by the GA combined with the greedy algorithm to the solution obtained by the solver Gurobi. The better solutions are selected for further analysis and modeling.

4.1. Settings of the Experimental Scenarios. The volume of the service area, the number of demand points, and the shape of the service area are three important factors that affect the expected length of the shortest path. We only consider the shape of the service area as a cuboid or cube. We assume that the longest side is length, the second longest side is width, and the shortest side is height. This study also assumes that the service area has three parameters: the



*The Shortest Path Length = 42.0285

FIGURE 2: The 3D figure of initial points and the shortest path of 3D-TSP for the experiment.

volume of the service area (V), the ratio of length to width ($S1$, the ratio of the longest side to the second longest side), and the ratio of length to height ($S2$, the ratio of the longest side to the shortest side). When V is fixed, the shape of the service area will change with the change in $S1$ and $S2$. For example, when $S1 = S2 = 1$, which means that the length, width, and height are equal, the shape of the service area is a cube.

A total of 500 (5-10-10) experimental scenarios are designed in this study. The range of V includes 1000, 8000, 27,000, 64,000, and 125,000. Under the premise of keeping V constant, let $S1$ and $S2$ increase from 1 to 4 at an interval of 1, such that the shape of the service area is gradually elongated from a cube to a cuboid. After the service area is set, demand points are randomly generated in each of the 3D service areas. The number of demand points increases from 10 to 100 at intervals of 10. The demand generation process is repeated under each scenario.

4.2. Algorithm Comparison. The computer environment is as follows: Windows 10 (64 bit), CPU is AMD R9-5900HX processor and baseline speed is 3.30 GHz. First of all, in order to verify the effectiveness of the improved GA, this paper selects 4 cases from the TSPLIB database with the number of cities less than or equal to 100. The solution results are shown in Table 3. The test results show that the improved GA algorithm used in this paper generates results that are very close to the solution given by the TSPLIB. The TSPLIB official website states that any solution that is close to or better than the solution they provide can be considered a good solution [33], which verifies the effectiveness of the algorithm.

The improved GA and Gurobi are used to obtain the shortest path lengths of 3D-TSP. The computing time and quality of the two solutions are compared. We set the parameters of the improved GA as $M=200$, $C=10,000$, $P_c=0.7$, $P_m=0.1$. 500 simulation experiments were performed for each scenario, and the average value of the length of the shortest path of the 500 experiments was taken as the expected length of the shortest path in this scenario. To prove that 500 simulation experiments for each scenario are sufficient to accurately estimate the expected length of the shortest path, we plotted the relationship between the average length of the shortest path and the number of experiments (see Figure 3). As shown in the figure, the fluctuation is large when the number of simulations is less than 100. When the number of simulations is between 100 and 300, the fluctuation is smaller. When the number of simulations exceeds 300, the average value tends to be stable. To obtain accurate results, we set the number of simulation experiments to 500.

In each of the 50 scenarios with $N=10$, we compute the difference between the lengths of the shortest paths obtained by the two methods and divide the difference by the length obtained by Gurobi to get a percentage. Figure 4 shows the distribution of the percentage. Figure 4 shows that 90% of the difference is lower than 1%, and the biggest difference is 1.37%, which is also very small. As a result, the quality of the

TABLE 3: Comparison of the improved GA and TSPLIB.

| Instances | Optimal solution (improved GA) | Optimal solution (TSPLIB) |
|-----------|--------------------------------|---------------------------|
| ulysses22 | 75.27 | 74 |
| att48 | 33523.7 | 33,522 |
| eil76 | 545.25 | 538 |
| rand100 | 7920.8 | 7891 |

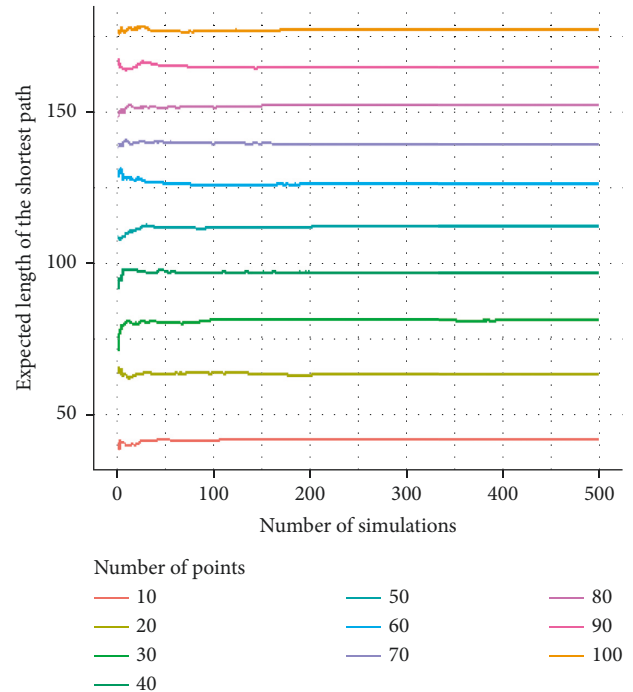


FIGURE 3: Relationship between the average length of the shortest path and the number of experiments.

solutions obtained by the two methods is not much different. At the same time, we found that the improved GA takes much less computing time than using Gurobi. Especially when the value of N is greater than 30, Gurobi cannot generate the results for the 500 scenarios in three days, while the improved GA can. As a result, this paper uses the solution obtained by the improved GA to further establish the estimation models.

4.3. Estimation Models of the Expected Length of the Shortest Path. We start by defining some variables that will be used in the estimation models. We then explore the relationship between these variables and the expected length of the shortest path and propose possible models for estimating the expected length. Finally, the goodness of fit of the proposed models is evaluated.

4.3.1. Variables for Building the Models. We obtain four variables (A , N , S , and L) from the previous literature [11–15]. These studies assume that the service area is a rectangle, where A is the area of the service area, N is the

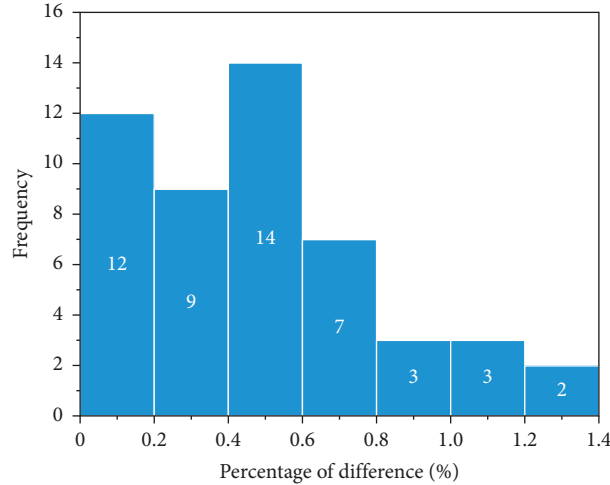


FIGURE 4: Distribution of the difference between the lengths of the shortest paths obtained by the two methods in percentage.

number of demand points in the service area, S is the ratio of the length of the service area to its width, and L is the expected length of the shortest path in the service area. Because this study focuses on the 3D space, the number of variables is expanded to five (V , N , $S1$, $S2$, and L), where V is the volume of the service area, $S1$ is the ratio of the length to the width, and $S2$ is the ratio of the length to the height.

4.3.2. Analysis of the Expected Length of the Shortest Path.

Before establishing the estimation models, it is necessary to analyze the relationship between the variables and L based on the experimental data. The results of the expected length of the shortest path for some of the 500 scenarios obtained using the improved GA are shown in Table 4. The demand points are randomly generated three-dimensional coordinates, as explained in Section 4.1. This table shows the change in the mean value of the shortest path length when different values of V , N , $S1$, and $S2$ are selected. As indicated in Table 4, the mean value of the shortest path length increases with an increase in V , N , $S1$, and $S2$. When V , $S1$, and $S2$ are fixed, the mean value of the shortest path length increases as N increases. When N , $S1$, and $S2$ are fixed, the mean value of the shortest path length increases as V increases. When V and N are fixed, the mean value of the shortest path length increases gradually as $S1$ and $S2$ increase.

We use the data when the shape of the service area is a cube as an example to demonstrate the relationship between the expected length of the shortest path and the volume of the service area and the number of demand points in the service area. We first present the relationship between the volume of the service area (V) and the expected length of the shortest path (L) under the different number of demand points (N) (Figure 5). As shown in Figure 5, when V does not change, L gradually increases as N increases. When N is constant, L gradually increases with an increase in V and there is a nonlinear relationship between the two variables. It can be seen from the figure that L and the cubic root of the volume of the service area ($\sqrt[3]{V}$) may have a linear

relationship. To construct better models, we also explore the relationship between L and $\sqrt[3]{V}$.

On the basis of the expected length of the shortest path model ($L \approx \alpha \sqrt{NA}$) proposed by Beardwood et al. [11], we speculate that the expected length of the shortest path in 3D-TSP may have a linear relationship with $\sqrt{N} \sqrt[3]{V}$ or $\sqrt[3]{NV}$. Therefore, we use the data when the shape of the service area is a cube as an example to visualize the relationship between the variables in the form of a scatterplot. The scatterplot of the expected length of the shortest path and $\sqrt{N} \sqrt[3]{V}$ is shown in Figure 6. The scatterplot of the expected length of the shortest path and $\sqrt[3]{NV}$ is presented in Figure 7.

Figure 6 shows an evident linear relationship between the expected length of the shortest path (L) and $\sqrt{N} \sqrt[3]{V}$. By contrast, the linear relationship between L and $\sqrt[3]{NV}$ in Figure 7 is not that obvious.

In addition, this study calculates that when the volume of the service area is fixed to 1000, the ratio of L to $\sqrt{N} \sqrt[3]{V}$ and the ratio of L to $\sqrt[3]{NV}$ under different shapes are provided in Tables 5 and 6. As indicated in Table 5, when the volume of the service area (V) and the number of demand points (N) are fixed, the ratio increases as the shape of the service area is elongated (refer to any row in Table 5). When the shape and volume of the service area are fixed, the ratio increases with the number of demand points (refer to any column in Table 5). When the volume of the service area is fixed, the difference between ratios becomes smaller as the number of demand points increases. The conclusions drawn from Table 6 are similar to those drawn from Table 5.

4.3.3. *Model Results.* From the analysis above, we could build the following models:

$$L = \text{coefficient} \cdot \sqrt{N} \sqrt[3]{V}, \quad (7)$$

$$L = \text{coefficient} \cdot \sqrt[3]{NV}, \quad (8)$$

where the coefficient is a linear combination of the independent variables.

TABLE 4: Mean value of the shortest path length under different conditions.

| | Number of demand points in 3D TSP | | | | |
|------------------------------|-----------------------------------|----------|----------|----------|-----------|
| | $N = 20$ | $N = 40$ | $N = 60$ | $N = 80$ | $N = 100$ |
| $V = 1000, S1 = 1, S2 = 1$ | 63.64 | 96.93 | 126.3 | 152.1 | 177 |
| $V = 8000, S1 = 1, S2 = 1$ | 127.29 | 193.9 | 252.6 | 304.2 | 353.9 |
| $V = 27000, S1 = 1, S2 = 1$ | 190.9 | 290.5 | 378.5 | 456.3 | 530.9 |
| $V = 64000, S1 = 1, S2 = 1$ | 254.6 | 387.7 | 505.1 | 608.3 | 707.8 |
| $V = 125000, S1 = 1, S2 = 1$ | 318.2 | 484.4 | 630.8 | 760.5 | 884.9 |
| $V = 1000, S1 = 1, S2 = 2$ | 64.37 | 97.64 | 126.7 | 153.1 | 178.1 |
| $V = 1000, S1 = 1, S2 = 3$ | 66.33 | 99.29 | 128.3 | 154.5 | 179.5 |
| $V = 1000, S1 = 1, S2 = 4$ | 68.87 | 101.47 | 130.3 | 156.2 | 181.5 |
| $V = 1000, S1 = 2, S2 = 2$ | 64.56 | 97.72 | 126.9 | 152.8 | 177.9 |
| $V = 1000, S1 = 3, S2 = 3$ | 67.31 | 99.3 | 128.1 | 154.1 | 179.3 |
| $V = 1000, S1 = 4, S2 = 4$ | 70.41 | 100.92 | 129.3 | 155.8 | 180 |

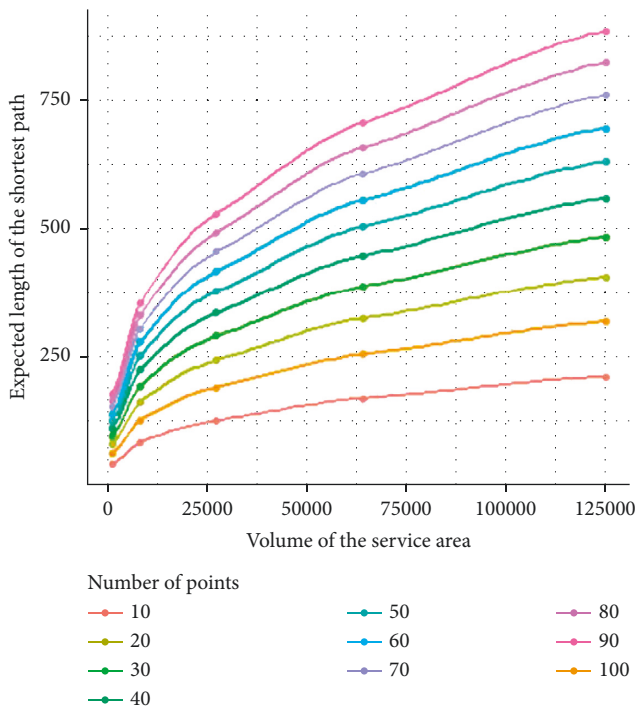


FIGURE 5: Relationship between the volume of the service area and the expected length of the shortest path under the different number of demand points ($s1 = 1, s2 = 1$).

This study establishes multiple linear regression equations based on the data obtained in the simulation experiments. Test and analyze the significance of the comprehensive linear effect of each independent variable on the dependent variable through RStudio, select the independent variable that only has a significant effect on the independent variable, and establish the regression equations. Based on correlation analysis, eight different models are considered and presented in Table 7. Models 1 and 2 are extensions of the models proposed by Beardwood [11]. Models 3–6 are extensions of the models proposed by Kwon [15]. In these models, $\alpha, \alpha_1, \alpha_2, \alpha_3,$ and α_4 are unknown parameters.

After the regression models are determined, the experimental data is used to estimate the unknown parameters

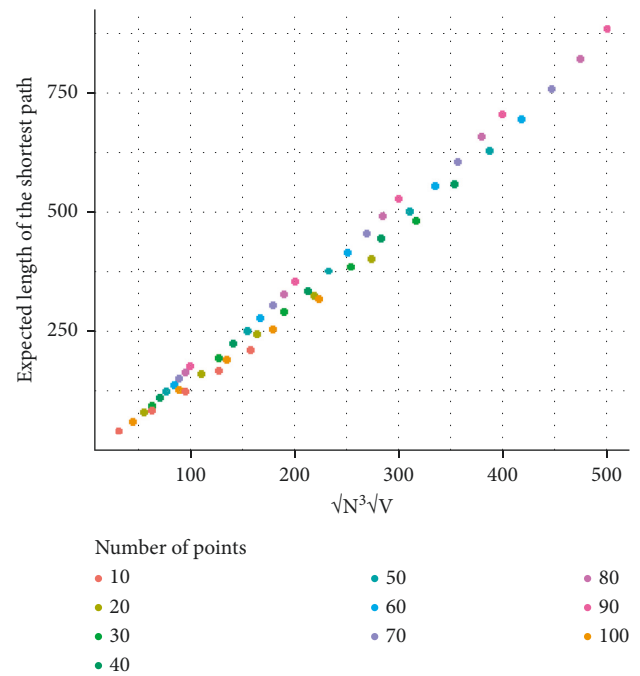


FIGURE 6: Scatterplot of the expected length of the shortest path and $\sqrt{N^3 V}$ ($s1 = 1, s2 = 1$).

of the models, and RStudio is used to analyze the significance of the coefficients of the fitted regression equations. The parameter estimation method used in this study is the least square method. Their adjusted R^2 and p values of each model are recorded. The results are summarized in Table 8. From Table 8, we can see that the R^2 of each model is very high, and the p value is very small. That means all the models fit the data well. Model 1 has the highest R^2 .

However, R^2 should not be the only evaluation measure of the model because the models could suffer from an overfitting problem. To deal with the possible overfitting problem, the k-fold cross-validation method is used [35]. The mean absolute error (MAE) and mean absolute percentage error (MAPE) of each model are calculated at the same time to compare the performance of these 6 models.

The k-fold cross-validation is a cross-validation method that can effectively compare models [35]. It first divides the

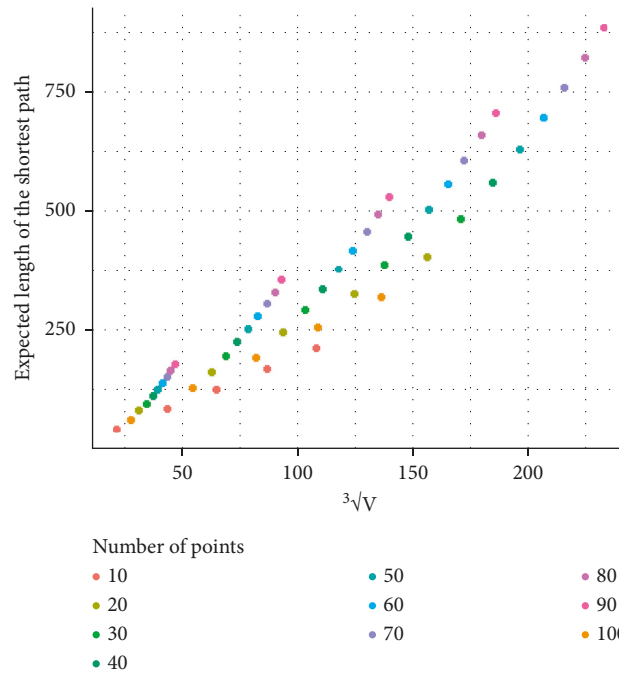


FIGURE 7: Scatterplot of the expected length of the shortest path and $\sqrt[3]{NV}$ ($s_1 = 1, s_2 = 1$).

TABLE 5: Ratio of L to $\sqrt{N}\sqrt[3]{V}$ under different conditions ($V = 1000$).

| | Service region shapes | | | | | | | | | |
|-----------|-----------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | S1 = 1 S2 = 1 | S1 = 1 S2 = 2 | S1 = 1 S2 = 3 | S1 = 1 S2 = 4 | S1 = 2 S2 = 2 | S1 = 2 S2 = 3 | S1 = 2 S2 = 4 | S1 = 3 S2 = 3 | S1 = 3 S2 = 4 | S1 = 4 S2 = 4 |
| $N = 10$ | 1.33 | 1.37 | 1.46 | 1.55 | 1.41 | 1.44 | 1.52 | 1.50 | 1.58 | 1.66 |
| $N = 20$ | 1.41 | 1.45 | 1.49 | 1.54 | 1.45 | 1.47 | 1.50 | 1.51 | 1.53 | 1.58 |
| $N = 30$ | 1.48 | 1.49 | 1.53 | 1.58 | 1.50 | 1.51 | 1.54 | 1.53 | 1.55 | 1.57 |
| $N = 40$ | 1.54 | 1.55 | 1.57 | 1.60 | 1.55 | 1.56 | 1.57 | 1.57 | 1.58 | 1.60 |
| $N = 50$ | 1.59 | 1.59 | 1.62 | 1.65 | 1.60 | 1.61 | 1.61 | 1.61 | 1.62 | 1.63 |
| $N = 60$ | 1.63 | 1.64 | 1.65 | 1.68 | 1.63 | 1.65 | 1.66 | 1.65 | 1.66 | 1.67 |
| $N = 70$ | 1.66 | 1.67 | 1.69 | 1.71 | 1.67 | 1.68 | 1.69 | 1.68 | 1.69 | 1.70 |
| $N = 80$ | 1.70 | 1.71 | 1.73 | 1.75 | 1.71 | 1.71 | 1.73 | 1.72 | 1.73 | 1.74 |
| $N = 90$ | 1.74 | 1.74 | 1.76 | 1.78 | 1.74 | 1.76 | 1.76 | 1.76 | 1.76 | 1.78 |
| $N = 100$ | 1.77 | 1.78 | 1.79 | 1.81 | 1.78 | 1.79 | 1.80 | 1.79 | 1.80 | 1.80 |

TABLE 6: Ratio of L to $\sqrt[3]{NV}$ under different conditions ($V = 1000$).

| | Service region shapes | | | | | | | | | |
|-----------|-----------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | S1 = 1 S2 = 1 | S1 = 1 S2 = 2 | S1 = 1 S2 = 3 | S1 = 1 S2 = 4 | S1 = 2 S2 = 2 | S1 = 2 S2 = 3 | S1 = 2 S2 = 4 | S1 = 3 S2 = 3 | S1 = 3 S2 = 4 | S1 = 4 S2 = 4 |
| $N = 10$ | 1.96 | 2.01 | 2.14 | 2.27 | 2.07 | 2.12 | 2.24 | 2.20 | 2.32 | 2.43 |
| $N = 20$ | 2.33 | 2.40 | 2.45 | 2.54 | 2.38 | 2.42 | 2.48 | 2.48 | 2.53 | 2.60 |
| $N = 30$ | 2.62 | 2.63 | 2.69 | 2.78 | 2.64 | 2.66 | 2.71 | 2.69 | 2.73 | 2.77 |
| $N = 40$ | 2.84 | 2.87 | 2.91 | 2.97 | 2.87 | 2.89 | 2.91 | 2.90 | 2.93 | 2.95 |
| $N = 50$ | 3.05 | 3.06 | 3.11 | 3.16 | 3.07 | 3.09 | 3.10 | 3.09 | 3.12 | 3.12 |
| $N = 60$ | 3.22 | 3.24 | 3.27 | 3.33 | 3.23 | 3.26 | 3.28 | 3.27 | 3.29 | 3.30 |
| $N = 70$ | 3.37 | 3.40 | 3.43 | 3.47 | 3.39 | 3.41 | 3.43 | 3.42 | 3.44 | 3.45 |
| $N = 80$ | 3.54 | 3.55 | 3.58 | 3.63 | 3.55 | 3.56 | 3.59 | 3.58 | 3.60 | 3.61 |
| $N = 90$ | 3.68 | 3.68 | 3.72 | 3.77 | 3.69 | 3.72 | 3.73 | 3.72 | 3.73 | 3.76 |
| $N = 100$ | 3.82 | 3.83 | 3.86 | 3.91 | 3.83 | 3.86 | 3.87 | 3.86 | 3.87 | 3.88 |

TABLE 7: Proposed regression models.

| Model number | Model |
|--------------|--|
| 1 | $L = \alpha\sqrt{N}\sqrt[3]{V}$ |
| 2 | $L = \alpha\sqrt[3]{NV}$ |
| 3 | $L = (\alpha_1 + \alpha_2\sqrt{N} + \alpha_3(S1/N + S2/N))\sqrt{N}\sqrt[3]{V}$ |
| 4 | $L = (\alpha_1 + \alpha_2\sqrt{N} + \alpha_3S1/N + \alpha_4S2/N)\sqrt{N}\sqrt[3]{V}$ |
| 5 | $L = (\alpha_1 + \alpha_2\sqrt{N} + \alpha_3(S1/N + S2/N))\sqrt[3]{NV}$ |
| 6 | $L = (\alpha_1 + \alpha_2\sqrt{N} + \alpha_3S1/N + \alpha_4S2/N)\sqrt[3]{NV}$ |

TABLE 8: Fitted models.

| Model number | Fitted model | R ² | p value |
|--------------|--|----------------|------------|
| 1 | $L = 1.6841\sqrt{N}\sqrt[3]{V}$ | 0.9969 | <2.2e - 16 |
| 2 | $L = 3.3183\sqrt[3]{NV}$ | 0.9813 | <2.2e - 16 |
| 3 | $L = (1.0611 + 0.0713\sqrt{N} + 0.4239(S1/N + S2/N))\sqrt{N}\sqrt[3]{V}$ | 0.9759 | <2.2e - 16 |
| 4 | $L = (1.0499 + 0.0724\sqrt{N} + 0.2168S1/N + 0.5966S2/N)\sqrt{N}\sqrt[3]{V}$ | 0.9807 | <2.2e - 16 |
| 5 | $L = (1.0539 + 0.2788\sqrt{N} + 0.5639(S1/N + S2/N))\sqrt[3]{NV}$ | 0.9960 | <2.2e - 16 |
| 6 | $L = (1.0384 + 0.2804\sqrt{N} + 0.2742S1/N + 0.8055S2/N)\sqrt[3]{NV}$ | 0.9964 | <2.2e - 16 |

TABLE 9: MAE and MAPE of the models.

| | MAE | | MAPE (%) | |
|---------|--------------|-------------|--------------|-------------|
| | Training set | Testing set | Training set | Testing set |
| Model 1 | 18.17 | 18.19 | 6.40 | 6.41 |
| Model 2 | 45.16 | 45.31 | 16.95 | 16.99 |
| Model 3 | 0.01 | 0.01 | 0.72 | 0.73 |
| Model 4 | 0.01 | 0.01 | 0.64 | 0.65 |
| Model 5 | 0.02 | 0.02 | 0.83 | 0.84 |
| Model 6 | 0.02 | 0.02 | 0.78 | 0.79 |

dataset into k groups of mutually exclusive subsets of the same size. Then, it selects one subset each time as the testing set, leaving $k-1$ subsets as the training set. In this manner, k groups of training and testing sets are obtained. The operation is performed k times. The MAE and MAPE are used to evaluate the model performance. The average values of the measures are calculated for each model. The value of k in this study is set to 10. The results are provided in Table 9. The MAE and MAPE of the testing set are larger than those of the training set, and such findings are reasonable.

Among the 6 models, Model 1 and Model 2 are the simplest models. The MAE and MAPE of Model 1 are smaller than those of Model 2. Model 1 is selected as the representative of the simple models. Models 3 to 6 are more complex and fit the data better in terms of MAE and MAPE. Among them, Model 4 has the smallest MAE and MAPE, followed by Model 3. As a result, Model 4 is selected as the representative of the complex models.

5. Conclusions

In this paper, we intend to study the expected length of the shortest path of the 3D-TSP. By changing the volume, shape, and number of demand points within the service area, 500 experimental scenarios were designed. The improved GA and the solver Gurobi are used to obtain the shortest path of 3D-TSP in each scenario. The results of the two methods are compared and analyzed. Though the Gurobi could generate

better results, it takes much longer computing time. Especially when the number of demand points is large, it becomes infeasible to obtain the results in an acceptable time. The shortest path obtained by the improved GA is very similar to that obtained by the Gurobi. As a result, the shortest path obtained by the improved GA is used in this study for further analysis and modeling. Regression models of the expected length of the shortest path are proposed based on data analysis and previous studies. The coefficients of models are fitted using RStudio. The k -fold cross-validation is applied to evaluate the model performance based on MAE and MAPE. All the models proposed in this study fit the training and testing datasets well. The results indicate that Model 1 can accurately estimate the expected length of the shortest path of the 3D-TSP and has a simple formulation. For a more complex model formulation, Model 4 could generate better prediction accuracy.

The results of this study can be used to estimate the expected length of the shortest path of 3D-TSP. Although this study uses linear regression to model the expected length of the shortest path of the 3D-TSP, machine learning methods, such as neural networks, are also worth exploring. The limitation of this study is as follows. Firstly, we obtained the solution using the improved GA. The solution may not be globally optimal. Better algorithms could be explored in the future to make sure the global optimal solution is obtained. Secondly, this study is based on experiments designed under ideal conditions. The applicability of the proposed models in practice should be further verified.

Data Availability

The data used to prove the validity of the algorithm of this study are available from <https://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was funded by the National Natural Science Foundation of China (Grant nos. 71704145, 71831006, and 51774241), China Postdoctoral Science Foundation, and Sichuan Youth Science and Technology Innovation Research Team Project (Grant nos. 2019JDTD0002 and 2020JDTD0027).

References

- [1] C. H. Papadimitriou, "The Euclidean travelling salesman problem is NP-complete," *Theoretical Computer Science*, vol. 4, no. 3, pp. 237–244, 1977.
- [2] J. Li, M. Zhou, Q. Sun, X. Dai, and X. Yu, "Colored traveling salesman problem," *IEEE Transactions on Cybernetics*, vol. 45, no. 11, pp. 2390–2401, 2015.
- [3] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*, Princeton University Press, Princeton, NJ, USA, 2011.
- [4] A. P. Punnen, "The traveling salesman problem: applications, formulations and variations," *Combinatorial Optimization*, vol. 12, pp. 1–28, 2007.
- [5] H. Yang, Z. Zhang, W. Fan, and F. Xiao, "Optimal design for demand responsive connector service considering elastic demand," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2476–2486, 2021.
- [6] J. Huo, H. Yang, C. Li, R. Zheng, L. Yang, and Y. Wen, "Influence of the built environment on E-scooter sharing ridership: a tale of five cities," *Journal of Transport Geography*, vol. 93, Article ID 103084, 2021.
- [7] H. Yang, G. Zhai, X. Liu, L. Yang, Y. Liu, and Q. Yuan, "Determinants of city-level private car ownership: effect of vehicle regulation policies and the relative price," *Transport Policy*, vol. 115, pp. 40–48, 2022.
- [8] H. Yang, G. Zhai, L. Yang, and K. Xie, "How does the suspension of ride-sourcing affect the transportation system and environment?" *Transportation Research Part D: Transport and Environment*, vol. 102, Article ID 103131, 2022.
- [9] H. Yang, J. Huo, R. Pan, K. Xie, W. Zhang, and X. Luo, "Exploring built environment factors that influence the market share of ridesourcing service," *Applied Geography*, vol. 142, p. 102699, 2022.
- [10] P. K. Donta, B. S. P. Rao, T. Amgoth, C. S. R. Annavarapu, and S. Swain, "Data collection and path determination strategies for mobile sink in 3D WSNs," *IEEE Sensors Journal*, vol. 20, no. 4, pp. 2224–2233, 2020.
- [11] J. Beardwood, J. H. Halton, and J. M. Hammersley, "The shortest path through many points," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 55, no. 4, pp. 299–327, 1959.
- [12] C. F. Daganzo, "The length of tours in zones of different shapes," *Transportation Research Part B: Methodological*, vol. 18, no. 2, pp. 135–145, 1984.
- [13] C. F. Daganzo, "The distance traveled to VisitNPoints with a maximum of CS tops per vehicle: an analytic model and an application," *Transportation Science*, vol. 18, no. 4, pp. 331–350, 1984.
- [14] T. W. Chien, "Operational estimators for the length of a traveling salesman tour," *Computers & Operations Research*, vol. 19, no. 6, pp. 469–478, 1992.
- [15] O. Kwon, B. Golden, and E. Wasil, "Estimating the length of the optimal TSP tour: an empirical study using regression and neural networks," *Computers & Operations Research*, vol. 22, no. 10, pp. 1039–1046, 1995.
- [16] D. M. Stein, "An asymptotic, probabilistic analysis of a routing problem," *Mathematics of Operations Research*, vol. 3, no. 2, pp. 89–101, 1978.
- [17] D. M. Stein, "Scheduling dial-a-ride transportation systems," *Transportation Science*, vol. 12, no. 3, pp. 232–249, 1978.
- [18] E. Bonomi and J.-L. Lutton, "The N-city travelling salesman problem: statistical mechanics and the Metropolis algorithm," *SIAM Review*, vol. 26, no. 4, pp. 551–568, 1984.
- [19] P. Jaillet, "A priori solution of a traveling salesman problem in which a random subset of the customers are visited," *Operations Research*, vol. 36, no. 6, pp. 929–936, 1988.
- [20] D. S. Johnson, L. A. McGeoch, and E. E. Rothberg, *Asymptotic Experimental Analysis for the Held-Karp Traveling Salesman Bound*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1996.
- [21] W. Krauth and M. Mézard, "The cavity method and the travelling-salesman problem," *Europhysics Letters*, vol. 8, no. 3, pp. 213–218, 1989.
- [22] J. Lee and M. Y. Choi, "Optimization by multicanonical annealing and the traveling salesman problem," *Physical Review A*, vol. 50, no. 2, pp. R651–R654, 1994.
- [23] A. G. Percus and O. C. Martin, "Finite size and dimensional dependence in the euclidean traveling salesman problem," *Physical Review Letters*, vol. 76, no. 8, pp. 1188–1191, 1996.
- [24] F. Robuste, C. F. Daganzo, and R. R. Souleyrette, "Implementing vehicle-routing models," *Transportation Research Part B: Methodological*, vol. 24, no. 4, pp. 263–286, 1990.
- [25] R. Purkayastha, T. Chakraborty, A. Saha, and D. Mukhopadhyay, "Study and analysis of various heuristic algorithms for solving travelling salesman problem-a survey," *Advances in Intelligent Systems and Computing*, vol. 1112, pp. 61–70, 2020.
- [26] K. Chaudhari and A. Thakkar, "Travelling salesman problem: an empirical comparison between aco, PSO, ABC, FA and GA," *Emerging Research in Computing, Information, Communication and Applications*, vol. 906, pp. 397–405, 2019.
- [27] I. S. Donbosco and U. K. Chakraborty, "Comparing performance of evolutionary algorithms - a travelling salesman perspective," in *Proceedings of the 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 182–187, Noida, India, January 2021.
- [28] A. H. Halim and I. Ismail, "Combinatorial optimization: comparison of heuristic algorithms in travelling salesman problem," *Archives of Computational Methods in Engineering*, vol. 26, no. 2, pp. 367–380, 2017.
- [29] X. Meng, J. Li, M. Zhou, X. Dai, and J. Dou, "Population-based incremental learning algorithm for a serial colored traveling salesman problem," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 2, pp. 277–288, 2018.
- [30] X. Meng, J. Li, X. Dai, and J. Dou, "Variable neighborhood search for a colored traveling salesman problem," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1018–1026, 2018.
- [31] X. Xu, J. Li, and M. Zhou, "Bi-objective colored traveling salesman problems," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021.
- [32] A. Kumar, "Improved genetic algorithm to solve small scale travelling salesman problem," in *Proceedings of the International Conference on Intelligent Computing and Control*

Systems (Iciccs 2020), pp. 516–520, IEEE, Madurai, India, May 2020.

- [33] G. Reinelt, “TSPLIB-A traveling salesman problem library,” *ORSA Journal on Computing*, vol. 3, no. 4, pp. 376–384, 1991.
- [34] H. Yang, X. Liang, Z. Zhang, Y. Liu, and M. M. Abid, “Statistical modeling of quartiles, standard deviation, and buffer time index of optimal tour in traveling salesman problem and implications for travel time reliability,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 12, pp. 339–347, 2020.
- [35] T. Fushiki, “Estimation of prediction error by using K-fold cross-validation,” *Statistics and Computing*, vol. 21, no. 2, pp. 137–146, 2011.