

Received July 15, 2020, accepted August 24, 2020, date of publication August 28, 2020, date of current version September 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3020093

# An Improved Tabu Search Algorithm for the Stochastic Vehicle Routing Problem With Soft Time Windows

GUOMING LI<sup>1</sup> AND JUNHUA LI<sup>1</sup>

Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, Nanchang Hangkong University, Nanchang 330063, China

Corresponding author: Junhua Li (jhlee126@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61440049, Grant 61866025, and Grant 61866026; in part by the Natural Science Foundation of Jiangxi Province under Grant 20181BAB202025; and in part by the Superiority Science and Technology Innovation Team Program of Jiangxi Province under Grant 20181BCB24008.

**ABSTRACT** In real-world environments, vehicle travel and service time will be affected by unpredictable factors and present a random state. Because of this situation, this article proposes the vehicle routing problem with soft time windows and stochastic travel and service time (SVRP-STW). The probability distribution of vehicle travel and service time are introduced into the model, and a stochastic programming model with modification is established to minimize the distribution cost. An Improved Tabu Search algorithm (I-TS) based on greedy algorithm is proposed, in which adaptive tabu length and neighborhood structure are introduced; the greedy algorithm is used instead of the random methods to generate the initial solution. Experiments on different scale instances prove the effectiveness and superiority of the proposed algorithm.

**INDEX TERMS** Stochastic travel times, stochastic service times, soft time windows, stochastic programming model, tabu search.

## I. INTRODUCTION

With the rapid development of the distribution industry, the expanding scale of logistics distribution, and the enhancing complexity of urban transportation networks, some phenomena occur frequently, e.g., unreasonable distribution route planning, empty distribution vehicles, and unreasonable order allocation. These phenomena have led to the increasing cost of distribution, which seriously affects the operation and development of enterprises. The vehicle routing problem (VRP) is a hot issue in the field of operations research, which refers to that the distribution center plans the optimal path according to the distribution and demand of customers. The VRP can effectively reduce the cost of distribution and overcome the unreasonable phenomenon in the process of distribution. Therefore, the vehicle routing problem has been widely concerned by scholars.

In the actual distribution process, traffic accidents, weather changes, road congestion, and other factors will affect the original distribution plan, so it is more practical to study the stochastic vehicle routing problem (SVRP) [1]. Shahparvari and Abbasi [2] have established excellent mathematical

models for SVRP. The SVRP can be divided into two situations: (i) vehicle routing problem with stochastic demand (VRPSD) [3], [4], (ii) vehicle routing problem with stochastic time (VRPST) [5], [6].

This article investigates an extended situation of the VRPST problem, in which vehicle travel and service time are stochastic, and the vehicle is required to deliver the goods within the specified time to the customer [7]. This problem is usually termed the vehicle routing problem with time windows and stochastic travel and service time (SVRPTW). Time windows define the earliest and latest time that a vehicle may arrive at a customer and start service. In this article soft time windows are considered, i.e., the vehicle can arrive before the time window is opened or after the time window is closed, but it shall be given certain punishment. The SVRPTW problem involves a fleet of homogeneous vehicles stationed at a depot to serve different geographically scattered customers. In this process, the vehicle travel and service time are uncertain and cannot violate the vehicle capacity constraints.

The SVRPTW is an NP-hard combinatorial optimization problem. Exact algorithms can only be used to find solutions for small-and-medium scale instances. For large-scale instances, the exact algorithms cannot find solutions within an acceptable time [8]. Therefore, scholars and experts begin

The associate editor coordinating the review of this manuscript and approving it for publication was Maurice J. Khabbaz<sup>1</sup>.

to consider generating high-quality solutions in reasonable computational time and shift their research efforts to the application of heuristic or meta-heuristic approaches. Heuristic and meta-heuristic methods for VRP and its variants have emerged in recent years [9], [10]. Tabu search algorithm (TS) is a meta-heuristic algorithm that has strong local development ability and fast convergence speed. Many combinatorial optimization problems can be solved efficiently by TS. It has been used to solve the VRP and its variants, e.g., VRP [11], VRPMTW [12], HFVRP [13], ConVRP [14] and MCVRP [15].

In this context, a new stochastic programming model is proposed considering the stochastic vehicle travel and service time, and an improved meta-heuristic algorithm is proposed considering the high complexity of the SVRPTW problem. The main contributions of this article are:

1. The introduction of the vehicle routing problem with soft time windows and stochastic travel and service times (SVRP-STW).
2. A new mathematical model is established that better reflects the randomness of travel and service times.
3. The design and implementation of improved tabu search algorithm which includes specific components such as the generation of the initial solution, the setting of tabu length, and the setting of neighborhood structure.

The remainder of this article is organized as follows. Section II presents a literature review on the VRPST, VRPTW, and SVRPTW. Section III formally defines the SVRP-STW to be considered and develops a mathematical model. Section IV describes in detail the proposed I-TS and how to solve the SVRP-STW. Section V compares the proposed I-TS with other algorithms in the literature in various scale instances. Finally, conclusions are drawn in Section VI.

## II. LITERATURE REVIEW

### A. VRPST

For the VRPST, vehicle travel time or service time is a random variable subject to a probability distribution. VRPST is closer to real conditions and has attracted more and more attention.

Kuo *et al.* [16] develop a mathematical model for the VRPST problem using fuzzy theory, which fully considers the randomness of time. An improved fuzzy Ant Colony System (ACS) is proposed, which embeds a cluster insertion algorithm into the ACS algorithm. The results show that the proposed algorithm performs better than the other algorithms. Shi *et al.* [17] develop a stochastic programming model with recourse (SPR) for the VRPST problem. A Hybrid Genetic Algorithm (HGA) and stochastic simulation method are proposed.

For the vehicle routing problem with stochastic service and travel time, Gutierrez *et al.* [18] propose a Multi-Population Memetic Algorithm (MPMA). The algorithm uses a log-normal approximation to estimate the arrival time, taking into

account that the failure of previous customers to improve the estimation of the mean and variance of the arrival time. The results show that MPMA is superior to different methods in different target types, thus demonstrating its efficiency and flexibility.

### B. VRPTW

In the actual distribution process, customers will specify the delivery time of the goods, which is called the vehicle routing problem with time window (VRPTW). The VRPTW has effectively improved the level of vehicle service and attracts more research attention.

Marinakis *et al.* [19] study the vehicle routing problem with time windows and a new variant of the Particle Swarm Optimization (PSO) algorithm is proposed. Three different adaptive strategies are used in the proposed Multi-Adaptive Particle Swarm Optimization (MAPSO) algorithm. The first adaptive strategy concerns the use of a Greedy Randomized Adaptive Search Procedure (GRASP) and the second adaptive strategy concerns the use of the Adaptive Combinatorial Neighborhood Topology. The algorithm is compared with other versions of PSO and has better results.

Several heuristics algorithms have been proposed to solve the VRPTW problem. Brandao [20] constructs an Iterated Local Search (ILS) algorithm for solving this problem and do numerical experiments with their instances. Molina *et al.* [21] propose an Ant Colony System (ACS) algorithm with local search for solving this problem. Perez-Rodriguez and Hernandez-Aguirre [22] propose an estimation of distribution algorithm for the problem. The algorithm uses the generalized Mallows distribution as a probability model to describe the distribution of the solution space. Experimental results show that the algorithm is competitive.

### C. SVRPTW

The vehicle routing problem with time windows and stochastic travel and service time (SVRPTW) is directly related to our questions. In recent years, it has begun to attract the attention of scholars.

Miranda and Conceicao [23] study the vehicle routing problem with hard time windows and stochastic travel and service time. They analyze that the arrival time distribution would be truncated at the earliest time windows, thus affecting the estimation of the arrival time of subsequent customers. To solve this problem, a specific statistical method is proposed to obtain the cumulative probability distribution of the vehicles over the customers, and a meta-heuristic algorithm based on Iterative Local Search (ILS) is proposed. A benchmark is used to evidence the superior performance and accuracy of the proposed method.

Several heuristics algorithms have been proposed to solve the SVRPTW problem. Tas *et al.* [24] propose a Tabu Search (TS) and an Adaptive Large Neighborhood Search (ALNS) for solving this problem and do numerical experiments for well-known problem instances. Gutierrez *et al.* [25] propose a Multi Population Memetic

Algorithm (MPMA) for solving this problem and do numerical experiments on modified instances proposed by Solomon [26] for the deterministic counterpart. Miranda *et al.* [27] propose a Multi-objective Memetic Algorithm (MMA) and a Multi-objective Iterated Local Search (MILS) for solving this problem. Experiments based on an adapted version of the 56 Solomon instances demonstrate the effectiveness of the proposed algorithms.

Although a large number of literatures have been published on VRPST and VRPTW problems, few types of researches have attempted to solve SVRPTW problems, and there is a lack of efficient solutions. Besides, this article differs from the existing work in the following aspects: (1) the vehicle travel time and service time are set as random variables subject to normal distribution at the same time, soft time window

TABLE 1. Variable definitions.

Sets	
$N$	set of vertexes to be served including depot
$V$	set of vertexes to be served excluding depot
$K$	set of vehicles
Parameters	
$c_{ij}$	cost for traveling from vertex $i$ to vertex $j$
$t_{ij}$	time required for traveling from vertex $i$ to vertex $j$
$d_{ij}$	distance between vertex $i$ and vertex $j$
$\delta_i$	service time at vertex $i$
$q_i$	demand of vertex $i$
$Q$	maximum vehicle capacity
$\lambda_w$	penalty cost of one unit of earliness
$\lambda_d$	penalty cost of one unit of delay
$E(W_{jk})$	expected earliness at vertex $j$ served by vehicle $k$
$E(D_{jk})$	expected delay at vertex $j$ served by vehicle $k$
$ S_k $	total number of vertexes served by vehicle $k$ including depot
Decision variables	
$x_{ijk}$	1, if vehicle $k$ travels from vertex $i$ to vertex $j$ , 0, otherwise
$y_{ik}$	1, if vertex $i$ is served by vehicle $k$ , 0, otherwise

TABLE 2. Constraints.

Constraints	
$\sum_{i \in N} x_{0ik} = 1 \quad \forall k \in K$	(2) each vehicle route must start from the depot
$\sum_{j \in N} x_{j0k} = 1 \quad \forall k \in K$	(3) each vehicle route must end at the depot
$\sum_{i \in N} x_{imk} - \sum_{j \in N} x_{mjk} = 0 \quad \forall m \in V, \forall k \in K$	(4) each vehicle must depart from a customer location after it visits the customer
$\sum_{k \in K} y_{ik} = 1 \quad \forall i \in V$	(5) each customer is served exactly once by exactly one vehicle
$\sum_{i \in N} \sum_{j \in N} q_i x_{ijk} \leq Q \quad \forall k \in K$	(6) for every vehicle route, the total demand cannot exceed the capacity of the vehicle
$\sum_{i \in N} \sum_{j \in N} x_{ijk} \leq  S_k  \quad \forall k \in K$	(7) for every vehicle route, no sub-loops are allowed

constraints are adopted, and a stochastic programming model with modification is proposed; (2) an improved tabu search algorithm is proposed.

### III. PROBLEM FORMULATION

In the SVRP-STW, vehicle travel and service time are unknown and the customers will specify the vehicle arrival time. The problem can be described as the depot planning the optimal distribution path according to the stochastic distribution of customers.

Let  $G = (N, A)$  be a complete digraph, where  $N = \{0, 1, 2, \dots, n\}$  is a set of nodes and  $A = \{(i, j): i, j \in N, i \neq j\}$  is a set of arcs. Each arc  $\{i, j\} \in A$  has an associated distance  $d_{ij}$ , the travel time  $t_{ij}$ , and the travel cost  $c_{ij}$ . Vertex 0 represents the depot,  $V = \{1, 2, \dots, n\}$  is a set of customers. Each customer  $i \in V$  has a non-negative demand  $q_i$ , service time  $\delta_i$ , and a time window  $[e_i, l_i]$ , where  $e_i$  is the start of the time window and  $l_i$  is the end of the time window. If the vehicle arrives at customer  $i$  before  $e_i$  or after  $l_i$ , it needs to be punished.  $K = \{1, 2, \dots, m\}$  is a set of vehicles, each vehicle  $k \in K$  has the same capacity  $Q$ . Both  $t_{ij}$  and  $\delta_i$  are random variables that subject to normal distribution.

Before presenting the mathematical formulations, we define the following notations in Table 1.

Based on these variable definitions, the model for the problem can be described as follows:

$$\min Z = \sum_{i,j \in A} \sum_{k \in K} c_{ij} x_{ijk} + \sum_{j \in N} \sum_{k \in K} \lambda_w E(W_{jk}) + \sum_{j \in N} \sum_{k \in K} \lambda_d E(D_{jk}) \quad (1)$$

The objective function (1) is formed by three parts: the vehicle travel cost, the penalty cost for vehicles arriving earlier than the agreed time, and the penalty cost for vehicles arriving later than the agreed time. Where  $\lambda_w$  and  $\lambda_d$  represent the penalty cost for per unit earliness time and delay time of vehicle  $k$ ;  $E(W_{jk})$  and  $E(D_{jk})$  represent the expected earliness time and the expected delay time at customer  $j$  served by vehicle  $k$ .

The following are constraints:

**A. EXPECTATION AND VARIANCE OF THE ARRIVAL TIME**

$$a_{ik} = \sum_{i \in A_{ik}} t_{i-1,i} + \sum_{i \in B_{ik}} \delta_i \tag{8}$$

Eq.8 indicates the time when the vehicle  $k$  arrives at customer  $i$ , where  $A_{ik}$  represents all customer points the vehicle  $k$  has passed since it arrived at customer  $i$ ;  $B_{ik}$  represents all customer points the vehicle  $k$  has passed before it arrived at customer  $i$ . The difference between  $A_{ik}$  and  $B_{ik}$  is whether it includes the customer point  $i$  where the vehicle  $k$  is currently located.

Vehicle travel time and service time obey normal distribution. Due to the additivity of normal distribution, vehicle arrival time also obeys normal distribution. The expectation and variance of vehicle arrival time are as follows:

$$u_{ik} = \sum_{i \in A_{ik}} E(t_{i-1,i}) + \sum_{i \in B_{ik}} E(\delta_i) \tag{9}$$

$$\sigma_{ik}^2 = \sum_{i \in A_{ik}} Var(t_{i-1,i}) + \sum_{i \in B_{ik}} Var(\delta_i) \tag{10}$$

**B. ANALYSIS OF THE OBJECTIVE FUNCTION**

The last two parts of the objective function are the penalty cost for vehicles arriving earlier than the agreed time and the penalty cost for vehicles arriving later than the agreed time. Among them, the expected value of the vehicle’s early arrival and delayed arrival time can be calculated.

As the vehicle travel time, service time, and arrival time are all random variables subject to the normal distribution, the expectation of vehicle early arrival time can be calculated by the following process:

$$\begin{aligned} E(W_{jk}) &= (e_j - t) \cdot P(t \leq e_j) \\ &= e_j \cdot \phi\left(\frac{e_j - \mu_{jk}}{\sigma_{jk}}\right) - \frac{1}{\sqrt{2\pi}\sigma_{jk}} \cdot \int_{-\infty}^{e_j} t \cdot e^{-\frac{(t-\mu_{jk})^2}{2\sigma_{jk}^2}} dt \\ &= e_j \cdot \phi\left(\frac{e_j - \mu_{jk}}{\sigma_{jk}}\right) - \frac{1}{\sqrt{2\pi}} \\ &\quad \cdot \int_{-\infty}^{\frac{e_j - \mu_{jk}}{\sigma_{jk}}} (\sigma_{jk}x + \mu_{jk}) \cdot e^{-\frac{x^2}{2}} dx \\ &= (e_j - \mu_{jk}) \cdot \phi\left(\frac{e_j - \mu_{jk}}{\sigma_{jk}}\right) + \frac{\sigma_{jk}}{\sqrt{2\pi}} \cdot e^{-\frac{(e_j - \mu_{jk})^2}{2\sigma_{jk}^2}} \end{aligned} \tag{11}$$

Similarly, the expected value of vehicle delay arrival time is as follows:

$$\begin{aligned} E(D_{jk}) &= \mu_{jk} - l_j + (l_j - \mu_{jk}) \cdot \phi\left(\frac{l_j - \mu_{jk}}{\sigma_{jk}}\right) \\ &\quad + \frac{\sigma_{jk}}{\sqrt{2\pi}} \cdot e^{-\frac{(l_j - \mu_{jk})^2}{2\sigma_{jk}^2}} \end{aligned} \tag{12}$$

**IV. IMPROVED TABU SEARCH ALGORITHM TO SOLVE THE SVRP-STW**

The above research shows that SVRP-STW is an NP-hard problem with high complexity, and it cannot be solved by

the exact algorithm. Meta-heuristic algorithms can provide a more effective solution approach. Our algorithm is based on Tabu Search algorithm which has been applied successfully to solve various vehicle routing problems [28]. TS is a random heuristic search algorithm that uses an initial solution as a starting basis for seeking improved solutions by searching for different neighborhoods. Some factors that affect the performance of Tabu Search algorithm, such as the pros and cons of the initial solution, tabu length, etc. In this section, we give brief information about the improved greedy algorithm to generate an initial solution, the selection of tabu length, and the definition of neighborhoods.

**A. AN INITIAL SOLUTION**

In general, the initial solution can be obtained randomly or generated by heuristic algorithms such as a glowworm swarm optimization algorithm [29] and a genetic algorithm [30]. A better initial solution can help the algorithm search for the global optimal solution quickly. Here, the initial vehicle routes are constructed by using an improved greedy algorithm. A detailed description of the algorithm can be obtained from Shenmaier [31].

The Greedy algorithm proposed in this article is a simple and efficient insertion procedure that takes into account travel distance, residual capacity, and time window constraint. In short, the initial vehicle routes construction process can be summarized as follows. With the depot as the initial node, in each iteration, the customer with a narrow time window and close to the current node is preferred and inserted at the best feasible insertion place in the current route. When a route cannot admit any new customer due to capacity constraints, a new route is constructed. This is repeated until all customers are visited.

**B. TABU LENGTH**

To avoid falling into local optimum in the search process, some movements will be recorded in the tabu list. The recorded movements are called tabu object, and the tabu length refers to the time when the tabu object is forbidden. The tabu length decreases with the increase of the number of algorithm iterations. When the tabu length is zero, the recorded movement will be deleted from the tabu list.

Tabu length is a key factor in the performance of tabu search algorithm because it directly affects the search process and behavior. On the one hand, the algorithm is easy to fall into the local optimal solution with a smaller value of tabu length. On the other hand, the search time of the algorithm will become longer and the algorithm may eventually crash when the value of tabu length is set larger. Therefore, an adaptive tabu length is proposed in this article to improve the performance of tabu search algorithm. Specifically, tabu length is randomly selected from the tabu length interval at the beginning of each iteration. If the better solution is searched in the search process, the value of tabu length will be increased. Otherwise, the value of tabu length will be reduced.

**TABLE 3. Results obtained by the proposed I-TS under different values of tabu lengths.**

Instance	Tabu length											
	20		30		40		50		60		70	
	OS	AS	OS	AS	OS	AS	OS	AS	OS	AS	OS	AS
R101	1690.78	1727.12	1679.32	1687.68	1667.89	1683.59	1670.25	1686.41	1692.75	1708.60	1691.96	1697.52
C104	939.90	940.86	900.44	911.46	902.56	906.60	907.85	919.43	921.41	933.94	934.63	950.74
RC101	1685.92	1705.63	1668.32	1695.72	1683.91	1698.68	1675.43	1688.69	1715.15	1739.98	1730.69	1747.62
R201	1258.97	1262.28	1209.62	1218.41	1226.53	1229.05	1218.47	1226.02	1239.77	1253.79	1254.80	1270.09
C203	699.243	703.32	660.04	660.27	678.11	680.83	697.84	698.37	738.33	738.60	719.35	721.40
RC207	1103.42	1105.31	1063.81	1074.21	1041.41	1051.56	1082.83	1087.03	1134.17	1134.94	1143.12	1175.11

In this article, the left and right boundaries of the tabu length interval are determined by experimental comparison. Six instances were randomly selected from Solomon’s VRPTW benchmark and the experiment was conducted on the different values of tabu lengths. These results are summarized in TABLE 3. It is worth noting that the OS represents the optimal solution and the AS represents the average solution.

Based on the results presented in TABLE 3, when the tabu length is between 30 and 50, the performance of tabu search algorithm is optimal. Therefore, the left and right boundary of the tabu length interval are set to 30 and 50, respectively.

**C. NEIGHBORHOOD STRUCTURE**

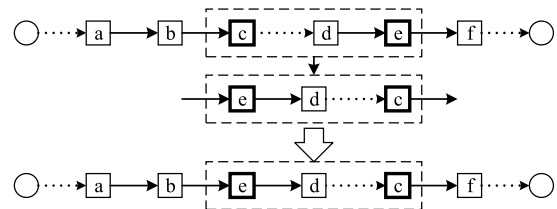
The neighborhood structure plays a key role in the performance of tabu search algorithm, since it determines the extent and the quality of the explored solution space. There are several general neighborhood structures, e.g., 2-opt, 3-opt, swap, cross, relocate and reverse, which have been widely applied to solve the combinatorial optimization problems. The improved tabu search algorithm proposed in this article comprises of four neighborhood moves: one specific neighborhood and three common neighborhoods (2-opt, swap and reallocate). At the beginning of each iteration, one of the four neighborhoods is randomly selected to explore solutions. If the improved tabu search algorithm stops searching in the neighborhood, it will continue to explore the next neighborhood. This is repeated until all neighborhoods are explored. The details of the neighborhood movements are as follows:

**Specific neighborhood:** Randomly select customer *i* to delete it from current path *l* and insert it into another path *k*. The path *k* must satisfy the capacity constraint and the customer *i* satisfies the time window constraint after inserting the path *k*, otherwise it will be inserted into the path *k* + 1.

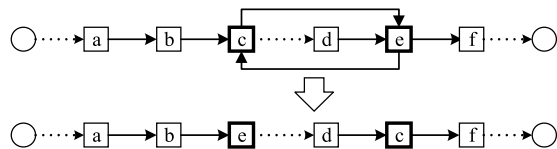
**2-opt:** Randomly select two customers in the path and flip the path between the two customers, which is indicated by FIGURE 1.

**Swap-operation:** Randomly select two customer points on the path and exchange positions, which is indicated by FIGURE 2.

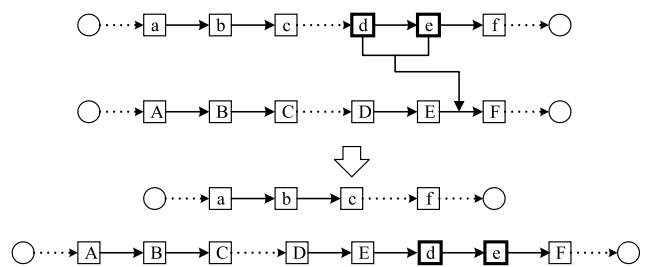
**Reallocate-operation:** Randomly select customers in the path and insert them anywhere in the other path, which is indicated by FIGURE 3.



**FIGURE 1. 2-opt.**



**FIGURE 2. Swap-operation.**



**FIGURE 3. Reallocate-operation.**

**D. IMPROVED TABU SEARCH ALGORITHM**

Tabu search algorithm (TS) is a modern heuristic algorithm that has flexible memory ability and aspiration criterion, so it is not easy to fall into the local optimum in the search process. It has been successfully applied to solve various complex combinatorial optimization problems [32]. Generally, TS uses a randomly generated initial solution as a starting basis for seeking improved solutions by searching for a certain neighborhood. It is improved with parallel computation and a combined neighborhood approach by Kiziloz and Dokeroglu [32]. In this article, the greedy algorithm is proposed to replace random methods to generate the initial solution, and adaptively adjust the tabu length and the type of neighborhood structure. The specific process of the improved tabu search algorithm proposed in this article is shown in FIGURE 4:

**Algorithm 1** Improved TS for stochastic VRPTW

---

```

1.  $s \leftarrow$  initial feasible route using the improved greedy
   algorithm
2.  $s_{best} \leftarrow s$ 
3.  $iterNum = 1, neighborNum = 1$ 
4. While  $iterNum < maxIterNum$  do
5.  $tabuLength = \text{round}(a+(b-a)*\text{rand}(1,1))$ 
6. While  $neighborNum < maxNeighborNum$  do
7.  $s' = \text{local search}(s)$   $s' \in N_l(s)$  ( $l$  represents
   neighborhood structure)
8. If  $f(s') < f(s_{best})$  Then
9.  $f(s_{best}) = f(s')$ 
10.  $s = s'$ 
11.  $tabuLength = tabuLength + 1$ 
12. Continue to search with  $N_l(s)$ 
13. Else
14.  $neighborNum = neighborNum + 1$ 
15.  $tabuLength = tabuLength - 1$ 
16. End if
17. End while
18.  $iterNum = iterNum + 1$ 
19. End while
20. Return  $s$ 

```

---

**FIGURE 4.** Pseudo-code of I-TS for stochastic VRPTW.**V. EXPERIMENTS AND RESULTS**

In this section, we demonstrate the effectiveness of proposed algorithm by testing it using well-known benchmark data sets. This section is divided into four sub-sections with different experiments. Section 5.1 describes the influence of variance changes of stochastic travel time and service time on the results of the algorithm. Section 5.2 shows the comparison results of the proposed algorithm and other algorithms in 25-customer instances. Section 5.3 describes the comparison results of the proposed algorithm and other algorithms in 50-customer instances. Section 5.4 shows the comparison results of the proposed algorithm and other algorithms in 100-customer instances. The improved tabu search algorithm was developed in a JDK8 environment and all experiments were tested in a computer with Window 10, 8GB RAM, I7 2.8GHz; the parameters related to proposed algorithm are summarized in TABLE 4.

**TABLE 4.** Parameters setting.

Parameters	The value
$maxIterNum$	2000
$neighborNum$	4
$a$	30
$b$	50
$\lambda_w$	0.1
$\lambda_d$	0.1

**A. DISCUSSION ON THE VARIANCE OF STOCHASTIC TRAVEL TIME AND SERVICE TIME**

Six types of VRPTW examples (C1 type, C2 type, R1 type, R2 type, RC1 type and RC2 type) of Solomon were used to test I-TS. In the process of applying the improved tabu search algorithm, the variance of vehicle travel time and service time was set as 10%, 20%, 30% and 40% of the mean respectively, and a comparative analysis was carried out. The algorithm was run independently for 5 times in each case, and the optimal solution and the average solution were compared to analyze the influence of randomness on the results. The experimental results are shown in FIGURE 5.

FIGURE 5 shows the optimal solution and average solution of each instance at different variance levels. For different instances, the greater the variance, the stronger the randomness of vehicle travel time and service time, and the worse the optimal solution and average solution obtained by the algorithm. Based on the results presented in FIGURE 5, the effect of the algorithm at different variance levels may be summarized as follows:

- For most instances, when the variance is set to 20%, 30% and 40% of the mean value, the optimal solution and average solution obtained by the algorithm are worse than the optimal solution and average solution obtained by the algorithm when the variance is set to 10% of the mean value.

In conclusion, the variance is taken to be 10% of the mean in all experiments.

**B. RESULTS FOR THE 25-CUSTOMER INSTANCES**

Based on Solomon's VRPTW benchmark (56 instances in total), select the first 25 customers to construct a new benchmark. Under this benchmark, the Genetic Algorithm (GA) [33], the Simulated Annealing algorithm (SA) [34], the Ant Colony Optimization algorithm (ACO) [35] and the proposed I-TS were compared. These results are summarized in TABLE 5.

**C. RESULTS FOR THE 50-CUSTOMER INSTANCES**

For the 50-customer instances of the Solomon's VRPTW benchmark, each instance is formed by selecting the first 50 customers in the corresponding instance. Under these instances, the Ant Colony Optimization algorithm (ACO) [35], the Simulated Annealing algorithm (SA) [34], the Tabu Search algorithm (TS) [36] and the proposed I-TS were compared. These results are summarized in TABLE 6.

**D. RESULTS FOR THE 100-CUSTOMER INSTANCES**

In the Solomon's VRPTW benchmark, there are totally 56 instances which can be divided into six categories: C1 type, C2 type, R1 type, R2 type, RC1 type and RC2 type. Among the C1, R1 and RC1 type instances, the vehicle capacity is small; among the C2, R2 and RC2 type instances, the vehicle capacity is large. Under the benchmark, the Ant Colony Optimization algorithm (ACO) [35], the Tabu Search algorithm (TS) [36], the Simulated Annealing

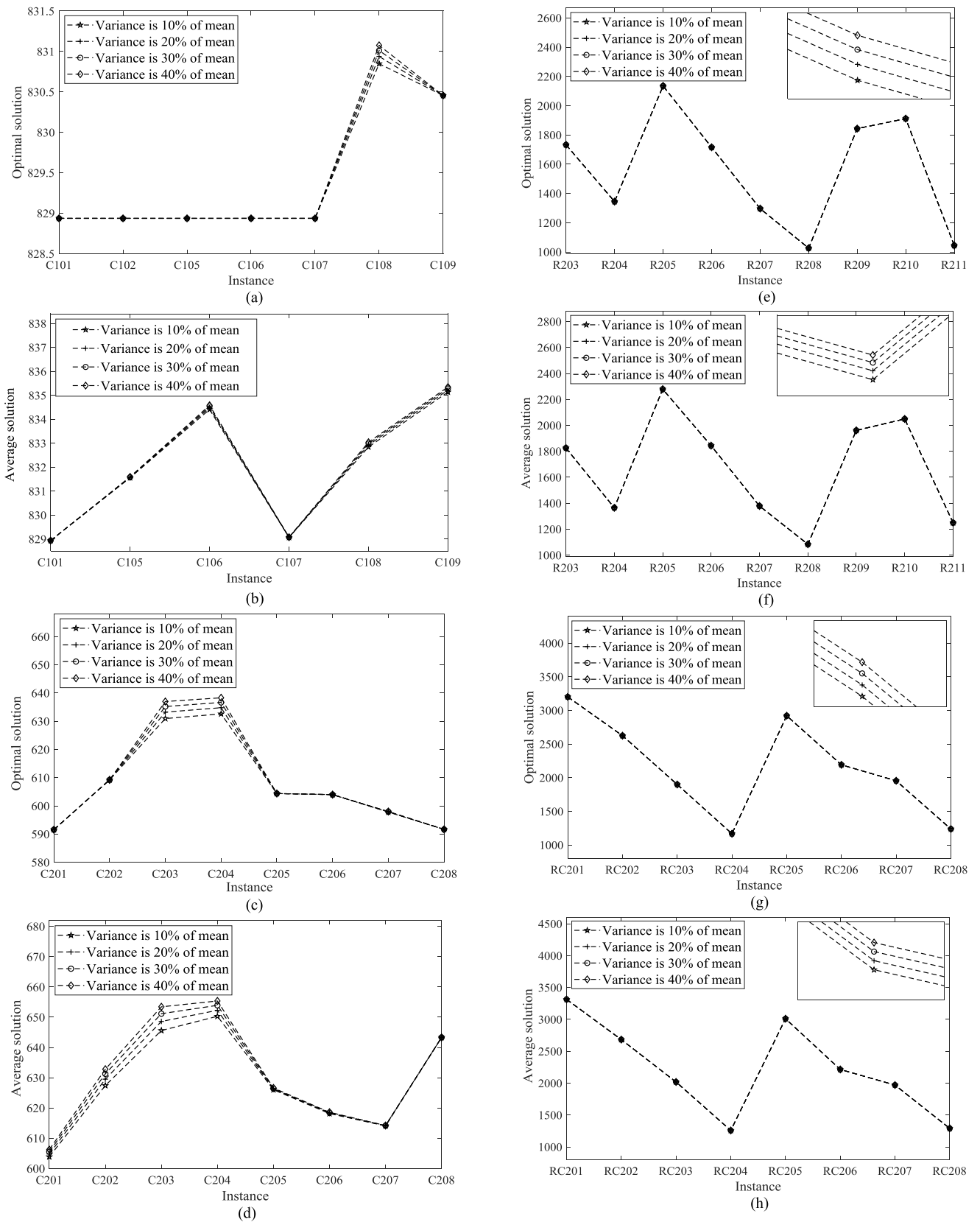


FIGURE 5. Differences between variance levels.

**TABLE 5. Comparison of the proposed I-TS with other algorithms on 25-customer instances.**

Instance	GA		SA		ACO		I-TS		Time(Sec.)
	OS	AS	OS	AS	OS	AS	OS	AS	
R104.25	503.686	522.102	417.960	445.040	442.296	461.095	438.874	<b>442.163</b>	1.15
R107.25	488.532	499.166	457.340	464.924	473.917	481.583	<b>449.220</b>	<b>453.302</b>	1.09
R110.25	484.349	491.905	464.130	477.482	453.042	460.538	459.490	466.276	1.03
R111.25	494.412	502.224	454.090	460.320	460.133	467.047	<b>452.924</b>	<b>455.472</b>	1.09
R112.25	476.016	486.927	436.820	458.696	403.490	409.485	403.768	<b>404.296</b>	1.08
RC104.25	350.486	383.739	384.790	388.048	307.936	346.020	324.233	<b>326.190</b>	1.06

**TABLE 6. Comparison of the proposed I-TS with other algorithms on 50-customer instances.**

Instance	ACO		SA		TS		I-TS		Time(Sec.)
	OS	AS	OS	AS	OS	AS	OS	AS	
C102.50	402.868	425.818	398.060	440.988	419.830	433.135	<b>389.584</b>	<b>413.754</b>	3.49
C103.50	405.987	438.090	397.240	426.202	404.130	422.499	<b>381.163</b>	<b>420.531</b>	3.41
C104.50	400.731	433.023	404.780	442.894	385.364	404.323	400.517	420.470	3.25
C105.50	402.154	424.039	396.680	417.532	396.010	427.179	<b>381.863</b>	<b>381.863</b>	3.63
C107.50	383.610	406.685	398.010	422.154	394.515	449.890	<b>373.810</b>	<b>373.810</b>	3.41
C108.50	388.471	416.679	396.350	416.302	413.215	452.285	<b>364.977</b>	<b>366.651</b>	3.34
C109.50	378.904	405.754	371.020	390.034	368.959	421.353	<b>363.247</b>	<b>364.760</b>	3.27
R106.50	958.340	968.852	841.060	864.724	811.199	841.877	840.885	845.142	3.32
R107.50	854.394	878.377	736.440	757.678	764.621	781.818	<b>734.737</b>	<b>745.385</b>	3.35
R108.50	718.852	737.544	629.910	646.56	640.661	678.664	638.141	<b>643.460</b>	3.25
R109.50	931.737	941.785	824.750	843.766	802.489	855.411	822.123	<b>837.203</b>	3.41
R110.50	798.627	819.062	747.250	752.754	770.687	790.243	<b>738.719</b>	<b>744.739</b>	3.41
R112.50	708.356	719.664	660.830	686.908	640.022	655.416	658.799	662.105	3.52
R208.50	542.726	562.406	561.696	603.471	648.422	666.192	669.276	692.960	3.60
RC104.50	606.525	617.398	565.020	605.586	562.703	580.200	<b>561.978</b>	<b>568.520</b>	3.20
RC106.50	949.871	981.032	730.510	759.454	851.052	867.626	777.954	812.562	3.40
RC108.50	638.398	641.282	621.360	648.352	769.806	812.562	<b>617.097</b>	<b>622.135</b>	3.21
RC204.50	512.848	519.359	707.485	758.654	524.764	571.991	668.858	684.964	3.28

**TABLE 7. Comparison of the proposed I-TS with other algorithms on 100-customer instances.**

Instance	ACO		TS		SA		I-TS		Time(Sec.)
	OS	AS	OS	AS	OS	AS	OS	AS	
C101	1035.736	1095.905	828.937	828.937	828.937	873.108	<b>828.937</b>	<b>828.937</b>	12.31
C105	1232.364	1248.115	828.937	851.959	862.370	963.812	<b>828.937</b>	<b>831.561</b>	14.93
C106	1206.036	1231.830	829.711	851.265	878.900	942.808	<b>828.937</b>	<b>834.409</b>	12.61
C107	1283.488	1322.327	828.937	904.984	904.150	966.502	<b>828.937</b>	<b>829.067</b>	12.34
C108	1162.571	1199.763	872.349	926.619	935.950	967.256	<b>830.849</b>	<b>832.844</b>	12.54
C109	1222.784	1262.173	873.533	916.469	975.240	1030.976	<b>830.456</b>	<b>835.137</b>	12.28
C201	816.163	862.311	634.155	671.313	591.557	691.704	<b>591.557</b>	<b>603.879</b>	17.11
C202	855.714	892.463	626.663	659.399	783.418	858.618	<b>609.224</b>	<b>627.411</b>	16.51
C205	666.521	688.388	611.524	644.236	670.058	785.307	<b>604.377</b>	<b>626.034</b>	15.55
C206	634.503	669.592	615.186	650.402	754.084	842.721	<b>603.994</b>	<b>618.145</b>	16.77
C208	684.065	714.551	601.723	633.295	772.360	844.855	<b>591.671</b>	<b>643.316</b>	15.06
R104	1330.814	1364.238	1079.007	1106.420	1045.650	1055.466	1073.358	1097.578	13.24
R108	1270.738	1288.029	1006.655	1034.074	1017.410	1022.476	<b>1004.369</b>	<b>1016.113</b>	13.04
R112	1270.406	1283.982	1024.265	1064.500	1002.650	1020.674	1013.520	1025.754	12.31
R208	1002.322	1024.805	819.509	839.512	912.819	946.809	1025.858	1083.247	16.20
R211	1055.760	1097.463	925.706	966.865	1084.490	1117.472	1043.735	1248.884	18.89
RC104	1417.134	1435.799	1191.848	1231.757	1222.620	1247.684	<b>1190.453</b>	<b>1213.514</b>	12.55
RC107	1614.689	1649.215	1351.017	1378.626	1302.110	1329.500	1328.371	1342.716	12.97
RC108	1443.811	1479.778	1209.629	1243.994	1200.280	1236.266	<b>1189.728</b>	<b>1208.539</b>	12.72
RC208	1115.148	1150.269	922.804	957.131	1115.360	1185.804	1237.108	1290.488	17.68

algorithm (SA) [34] and the proposed I-TS were compared. These results are summarized in TABLE 7.

**E. ANALYSIS OF RESULTS**

TABLES 5, 6 and 7 give comparisons between the I-TS and the comparing algorithms in different scale instances. OS and AS give the best and average results obtained by algorithms, respectively; Time(Sec.) gives the computational time

(in seconds) of the I-TS. Bold numbers indicate that the solutions obtained by I-TS are better than the results obtained by all comparing algorithms.

As can be seen, the best solutions obtained by I-TS that are better than the best results obtained by all comparison algorithms for instances R107.25 and R111.25 in TABLE 5. And the average solutions obtained by I-TS are better than the average solutions obtained by all comparison



algorithms for instances R104.25, R107.25, R111.25, R112.25, and RC104.25 in TABLE 5.

From TABLE 6, we can see that the best solutions are obtained by I-TS are better than the best solutions obtained by other three comparison algorithms in 10 out of the 18 tested instances. And the average solutions are obtained by I-TS are better than the average solutions obtained by other three comparison algorithms in 12 out of the 18 tested instances. As is shown in TABLE 7, the performance of I-TS is better than ACO, TS, and SA in 14 out of the 20 tested instances. In general, the results obtained by the proposed I-TS method are good in terms of solution quality, robustness, and computational time, when compared with the state-of-the-art algorithms from literatures in different scale instances. The results demonstrate the effectiveness and superiority of the proposed algorithm.

## VI. CONCLUDING REMARKS AND FURTHER RESEARCH

This article approaches the vehicle routing problem with soft time windows and stochastic travel and service time. Since vehicle travel and service time are random variables, a stochastic programming model with modification is established. The tabu search is improved according to the characteristics of the problem. The improved greedy algorithm is used to generate the initial solution, and an adaptive tabu length and neighborhood structure are designed. Meanwhile, the influence of the variance of random variables such as vehicle travel time on the objective function is analyzed through experiments. The greater the variance, the greater the randomness of the problem, and the larger the objective function.

In the process of generating the initial solution, some constraints are considered, e.g., capacity constraint and soft time window constraint. Therefore, the initial solution obtained by the proposed algorithm has a low penalty cost for violating the constraints, which plays a key role in generating the optimal solution by the algorithm.

Most algorithms first determine the vehicle travel and service time in the process of searching for the optimal solution, which will generate a large correction cost. The proposed algorithm in this article fully considers the randomness of vehicle travel and service time, and obtains the expected earliness time and the expected delay time according to its probability density function and distribution function. When the vehicle travel and service time change randomly, the proposed algorithm has a low probability of generating a large correction cost. Based on the above analysis, we can see that the proposed algorithm has a significant effect on solving the vehicle routing problem with soft time windows and stochastic travel and service time.

To demonstrate the effectiveness of the I-TS algorithm, an experimental study has been carried out in 25-customer instances, 50-customer instances, and 100-customer instances respectively from Solomon's VRPTW benchmark and compared to other meta-heuristics algorithms. Experimental results show the effectiveness of the proposed algorithm for

solving the SVRP-STW problem, and the proposed algorithm has strong optimization ability and high robustness.

Future researches mainly focus on three aspects: (1) introducing stochastic demand into the problem; (2) making the variables in the problem subject to probability distributions such as gamma distribution, exponential distribution, and logarithmic normal distribution; (3) designing more efficient algorithms, such as constructing multi-objective optimization algorithms.

## REFERENCES

- [1] E. Berhan, B. Beshah, D. Kitaw, and A. Abraham, "Stochastic vehicle routing problem: A literature survey," *J. Inf. Knowl. Manage.*, vol. 13, no. 03, Sep. 2014, Art. no. 1450022.
- [2] S. Shahparvari and B. Abbasi, "Robust stochastic vehicle routing and scheduling for bushfire emergency evacuation: An Australian case study," *Transp. Res. A, Policy Pract.*, vol. 104, pp. 32–49, Oct. 2017.
- [3] A. M. Florio, R. F. Hartl, and S. Minner, "Optimal *a priori* tour and restocking policy for the single-vehicle routing problem with stochastic demands," *Eur. J. Oper. Res.*, vol. 285, no. 1, pp. 172–182, Aug. 2020.
- [4] L. Bertazzi and N. Secomandi, "Faster rollout search for the vehicle routing problem with stochastic demands and restocking," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 487–497, Oct. 2018.
- [5] D. Guimarans, O. Dominguez, J. Panadero, and A. A. Juan, "A simheuristic approach for the two-dimensional vehicle routing problem with stochastic travel times," *Simul. Model. Pract. Theory*, vol. 89, pp. 1–14, Dec. 2018.
- [6] A. Lombard, S. Tamayo-Giraldo, and F. Fontane, "Vehicle routing problem with roaming delivery locations and stochastic travel times (VRPRDL-S)," *Transp. Res. Procedia*, vol. 30, pp. 167–177, Jan. 2018.
- [7] X. Li, P. Tian, and S. C. H. Leung, "Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm," *Int. J. Prod. Econ.*, vol. 125, no. 1, pp. 137–145, May 2010.
- [8] G. Andreatta, M. Casula, C. De Francesco, and L. De Giovanni, "A branch-and-price based heuristic for the stochastic vehicle routing problem with hard time windows," *Electron. Notes Discrete Math.*, vol. 52, pp. 325–332, Jun. 2016.
- [9] J. Brandão, "A memory-based iterated local search algorithm for the multi-depot open vehicle routing problem," *Eur. J. Oper. Res.*, vol. 284, no. 2, pp. 559–571, Jul. 2020.
- [10] M. Okulewicz and J. Mańdziuk, "A metaheuristic approach to solve dynamic vehicle routing problem in continuous search space," *Swarm Evol. Comput.*, vol. 48, pp. 44–61, Aug. 2019.
- [11] H. Jia, Y. Li, B. Dong, and H. Ya, "An improved tabu search approach to vehicle routing problem," *Procedia Social Behav. Sci.*, vol. 96, no. 6, pp. 1208–1217, Nov. 2013.
- [12] S. Belhaiza, P. Hansen, and G. Laporte, "A hybrid variable neighborhood tabu search heuristic for the vehicle routing problem with multiple time windows," *Comput. Oper. Res.*, vol. 52, pp. 269–281, Dec. 2014.
- [13] Y. Meliani, Y. Hani, S. L. Elhaq, and A. E. Mhamedi, "A developed tabu search algorithm for heterogeneous fleet vehicle routing problem," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 1051–1056, 2019.
- [14] C. D. Tarantilis, F. Stavropoulou, and P. P. Repoussis, "A template-based tabu search algorithm for the consistent vehicle routing problem," *Expert Syst. Appl.*, vol. 39, no. 4, pp. 4233–4239, Mar. 2012.
- [15] P. V. Silvestrin and M. Ritt, "An iterated tabu search for the multi-compartment vehicle routing problem," *Comput. Oper. Res.*, vol. 81, pp. 192–202, May 2017.
- [16] R. J. Kuo, B. S. Wibowo, and F. E. Zulvia, "Application of a fuzzy ant colony system to solve the dynamic vehicle routing problem with uncertain service time," *Appl. Math. Model.*, vol. 40, nos. 23–24, pp. 9990–10001, Dec. 2016.
- [17] Y. Shi, T. Boudouh, and O. Grunder, "A Home health care routing problem with stochastic travel and service time," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 13987–13992, Jul. 2017.
- [18] A. Gutierrez, L. Dieulle, N. Labadie, and N. Velasco, "A multi-population algorithm to solve the VRP with stochastic service and travel times," *Comput. Ind. Eng.*, vol. 125, pp. 144–156, Nov. 2018.

- [19] Y. Marinakis, M. Marinaki, and A. Migdalas, "A multi-adaptive particle swarm optimization for the vehicle routing problem with time windows," *Inf. Sci.*, vol. 481, pp. 311–329, May 2019.
- [20] J. Brandão, "Iterated local search algorithm with ejection chains for the open vehicle routing problem with time windows," *Comput. Ind. Eng.*, vol. 120, pp. 146–159, Jun. 2018.
- [21] J. C. Molina, J. L. Salmeron, and I. Eguia, "An ACS-based memetic algorithm for the heterogeneous vehicle routing problem with time windows," *Expert Syst. Appl.*, vol. 157, Nov. 2020, Art. no. 113379.
- [22] R. Pérez-Rodríguez and A. Hernández-Aguirre, "A hybrid estimation of distribution algorithm for the vehicle routing problem with time windows," *Comput. Ind. Eng.*, vol. 130, pp. 75–96, Apr. 2019.
- [23] D. M. Miranda and S. V. Conceição, "The vehicle routing problem with hard time windows and stochastic travel and service time," *Expert Syst. Appl.*, vol. 64, pp. 104–116, Dec. 2016.
- [24] D. Taş, N. Dellaert, T. van Woensel, and T. de Kok, "The time-dependent vehicle routing problem with soft time windows and stochastic travel times," *Transp. Res. C, Emerg. Technol.*, vol. 48, pp. 66–83, Nov. 2014.
- [25] A. Gutierrez, L. Dieulle, N. Labadie, and N. Velasco, "A multi population memetic algorithm for the vehicle routing problem with time windows and stochastic travel and service times," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1204–1209, 2016.
- [26] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Oper. Res.*, vol. 35, no. 2, pp. 254–265, Apr. 1987.
- [27] D. M. Miranda, J. Branke, and S. V. Conceição, "Algorithms for the multi-objective vehicle routing problem with hard time windows and stochastic travel time and service time," *Appl. Soft Comput.*, vol. 70, pp. 66–79, Sep. 2018.
- [28] W. Zhu, H. Qin, A. Lim, and L. Wang, "A two-stage tabu search algorithm with enhanced packing heuristics for the 3L-CVRP and M3L-CVRP," *Comput. Oper. Res.*, vol. 39, no. 9, pp. 2178–2195, Sep. 2012.
- [29] M. Marinaki and Y. Marinakis, "A glowworm swarm optimization algorithm for the vehicle routing problem with stochastic demands," *Expert Syst. Appl.*, vol. 46, pp. 145–163, Mar. 2016.
- [30] A. S. Tasan and M. Gen, "A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries," *Comput. Ind. Eng.*, vol. 62, no. 3, pp. 755–761, Apr. 2012.
- [31] V. V. Shenmaier, "A greedy algorithm for some classes of integer programs," *Discrete Appl. Math.*, vol. 133, nos. 1–3, pp. 93–101, Nov. 2003.
- [32] H. E. Kiziloz and T. Dokeroglu, "A robust and cooperative parallel tabu search algorithm for the maximum vertex weight clique problem," *Comput. Ind. Eng.*, vol. 118, pp. 54–66, Apr. 2018.
- [33] M. A. Mohammed, M. K. Abd Ghani, R. I. Hamed, S. A. Mostafa, M. S. Ahmad, and D. A. Ibrahim, "Solving vehicle routing problem by using improved genetic algorithm for optimal solution," *J. Comput. Sci.*, vol. 21, pp. 255–262, Jul. 2017.
- [34] Ş. Birim, "Vehicle routing problem with cross docking: A simulated annealing approach," *Procedia Social Behav. Sci.*, vol. 235, pp. 149–158, Nov. 2016.
- [35] Y. Li, H. Soleimani, and M. Zohal, "An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives," *J. Cleaner Prod.*, vol. 227, pp. 1161–1172, Aug. 2019.
- [36] M. Qiu, Z. Fu, R. Eglese, and Q. Tang, "A tabu search algorithm for the vehicle routing problem with discrete split deliveries and pickups," *Comput. Oper. Res.*, vol. 100, pp. 102–116, Dec. 2018.



**GUOMING LI** was born in Jiangxi, China, in 1996. He received the B.S. degree from Jiangxi Science and Technology Normal University, China, in 2018. He is currently pursuing the M.S. degree with Nanchang Hangkong University, China. His main research interests include system optimization and scheduling.



**JUNHUA LI** was born in Jiangxi, China, in 1974. He received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, China, in 2009. He is currently a Professor with Nanchang Hangkong University. His research interests include evolutionary computation and intelligent control.

• • •