A Chaotic Genetic Algorithm with Variable Neighborhood Search for Solving Time-Dependent Green VRPTW with Fuzzy Demand

Hao Fan 1, Xiaoxue Ren 2, Yueguang Zhang 2, Zimo Zhen 1 and Houming Fan 2,*

1 School of Transportation and Logistics, Dalian University of Technology, Dalian 116024, China
2 College of Transportation Engineering, Dalian Maritime University, Dalian 116026, China
* Correspondence: fhm468@dlmu.edu.cn

Abstract: Aiming at the time-dependent green vehicle routing problem with fuzzy demand, this paper comprehensively considers the dispatch costs, time window penalty costs, fuel costs, and the effects of vehicle travel speed, road gradient, and vehicle load on fuel consumption, a mixed integer programming model is formulated based on pre-optimization and re-optimization strategies. The traditional vehicle routing problems are modeled based on a symmetric graph. In this paper, considering the influence of time-dependent networks on route optimization, modeling is based on an asymmetric graph, which increases the complexity of the problem. In the pre-optimization stage, a pre-optimization scheme is generated based on the credibility measure theory; in the re-optimization stage, a new re-optimization strategy was used to deal with the service failure node. In order to solve this problem, we developed a chaotic genetic algorithm with variable neighborhood search, pseudo-randomness of chaos was introduced to ensure the diversity of initial solutions, and adaptive neighborhood search times strategy and inferior solution acceptance mechanism were proposed to improve the performance of the algorithm. The numerical results show that the model and algorithm we proposed are effective.

Keywords: vehicle routing problem; fuzzy demand; time-dependent; fuel consumption; asymmetric graph; chaotic genetic algorithm with variable neighborhood search

1. Introduction

In reality, due to the existence of objective and subjective factors, customer demand is often fuzzy and uncertain. Meanwhile, the traffic environment in a real logistics network also changes dynamically, so that vehicles cannot travel at their free speed, especially during rush hours. In recent years, environmental problems have become increasingly prominent. Controlling fuel consumption and carbon dioxide emissions is the key to protecting the environment. Therefore, the traditional mathematical model of the vehicle routing problem (VRP) is difficult to meet the requirements of fast and efficient distribution. The time-dependent green vehicle routing problem with time windows and fuzzy demand (TDGVRPTWF) proposed in this paper is a VRP expansion problem to optimize the vehicle route in the distribution network with fuzzy demand based on the previously collected traffic data and comprehensively considering the dispatch costs of vehicles, fuel costs and penalty costs on earliness and tardiness. TDGVRPTWF is the integration of VRP with fuzzy demand (VRPF), VRP with time windows (VRPTW), time-dependent VRP (TDRV), and green vehicle routing problem (GVRP), which is a more realistic road network environment and distribution requirements. Research on this problem has important theoretical and practical significance.

The existing literature on VRPF mostly used fuzzy sets, credibility, and other theories to define fuzzy demand and related constraints. Cao and Lai [1] developed a hybrid
intelligent algorithm to solve the fuzzy chance-constrained programming model established to minimize travel distance, which effectively optimizes the open vehicle distribution route under fuzzy demand. Kuo et al. [2] based on capacitated VRP, considering the fuzzy demand of customers, and taking the garbage collection system as the research object, established a CVRPFD model and designed an algorithm to solve it. Wang and Li [3] proposed a dynamic optimization strategy for VRPFD and developed a hybrid nondominated sorting genetic algorithm II (NSGA II) embedded fuzzy simulation for solving the model. Liu et al. [4] designed hybrid ant colony optimization to solve VRPFD and proposed a vehicle-coordinated strategy to reduce additional costs. Zhang and Fan [5] developed a variable neighborhood scatter search algorithm to solve VRPFD and proposed a novel real-time adjusted strategy. Li [6] formulated an optimization model based on the principle of pre-optimization and re-dispatch and proposed a new strategy to adjust the pre-optimization scheme in the second stage. Giallanza and Puma [7] considered the fuzziness of customer demand and carbon emissions during transportation, and a fuzzy chance-constrained programming model for 3E-FGVRP is established and solved by NSGA. Fan et al. [8] aimed at the time-dependent vehicle routing problem with fuzzy demand and time windows, established a two-stage optimization model, and developed an adaptive large neighborhood search algorithm (ALNS).

The models and algorithms for solving vehicle routing problems with time windows have been relatively mature. Hashimoto et al. [9] summarized the related research on VRP with hard and soft time windows, and emphatically introduced several penalty functions for violating time windows and related solving algorithms. Yu et al. [10] developed an improved branch-and-price (BAP) algorithm for heterogeneous fleet green VRPTW to optimize vehicle routing and minimize carbon emissions. To improve the performance of the algorithm, a multi-vehicle approximate dynamic programming (MVADP) algorithm and an integer branch method are introduced into the algorithm. Wang et al. [11] formulated a multi-objective optimization model for periodic VRPTW and service choice and developed an improved ant colony optimization algorithm to solve the proposed problem. Multi-adaptive particle swarm optimization [12] and hybrid estimation of distribution algorithm [13] are also used to solve VRPTW. With the development of intelligent transportation, we can collect more and more road network traffic data. Taniguchi and Shimamoto [14] established a mathematical model for VRP, in which the speed is time-varying. Mancini [15] used functional expressions to depict the change of vehicle speed. Haghani and Jung [16] studied TDVRP with dynamic changes in travel time and customer demand, and designed a GA to solve it. Hashimoto et al. [17] studied the time-dependent vehicle routing problem with time windows (TDVRPTW) and adopted an iterated local search algorithm to solve it. Hu et al. [18] proposed a robust optimization model and designed a two-stage algorithm to solve the VRPTW under travel time uncertainty. Figliozzi [19] improved the Solomon database and designed the Figliozzi database to compare different algorithms for solving TDVRPTW. Mu et al. [20] developed parallel-simulated annealing (SA) to solve TDVRPTW. The algorithm used four neighborhood search structures to optimize the initial solution, which improved the performance of the sequential SA.

In recent years, many scholars are interested in GVRP. Kuo [21] designed a SA to solve TDVRP. Liu et al. [22] assumed the carbon emissions are related to vehicle speed, load and type, and designed an improved ACO to solve it. Poonthalir and Nadarajan [23] studied GVRP with minimizing the routing costs and fuel consumption. Xu et al. [24] studied GVRP and developed a heuristic algorithm to solve it. Xiao and Konak [25] proposed a hybrid algorithm to solve GVRSP considering the effects of vehicle loads on emissions. Çimén and Soysal [26] considered the stochastic and time-dependent characters while estimating fuel consumption and emissions, and used a heuristic algorithm to solve TDGVRP.

The following points can be summarized by reviewing the existing research: (1) The research on VRPFD is mostly the innovation of algorithm and the research on return strategy after a service failure, ignoring the impact of rush hour, emergency, and other factors
on the traffic environment. (2) The existing literature on VRPFDFD mostly formulated optimization models aiming at the minimum distance, without considering the fuel costs and the effects of vehicle speed, vehicle load, and road gradient on fuel consumption. (3) Most research on the time-dependent vehicle routing problem uses the step function to depict the vehicle speed, causing a sudden change in speed change. Furthermore, all regard the demand of customers as certain and do not consider the fuzziness of customer demand in real life. (4) There is no research on VRP that comprehensively considers vehicle speed changes in real-time, fuel consumption changes during distribution, time window penalty, and fuzzy demand. Based on the above analysis, this paper takes TDGVRPTWFD as the research object:

- Considering the time dependence of vehicle speed and the relationship between fuel consumption and vehicle type, speed, load, and road gradient, based on fuzzy credibility theory, a fuzzy chance-constrained programming model is formulated to minimize the total cost to optimize the vehicle routing with fuzzy demand.
- Chaotic genetic algorithm with variable neighborhood search and re-dispatch strategy is developed to solve the proposed problem. In the algorithm, the Logistic chaotic map is used to generate the initial population, and an adaptive neighborhood search strategy and an inferior solution acceptance mechanism are proposed.

2. Problem and Mathematical Model

2.1. Problem Description

In the distribution area, \( G = (V, E) \) represents the complete directed graph, there are different types of roads, and the vehicle speed \( v = \{v_1, v_2, \ldots, v_n\} \) of each road increases or decreases smoothly. \( V = \{0\} \cup V_0 \) denotes the set of all nodes, 0 is the depot, \( V_0 = \{1, 2, \ldots, n\} \) represents the customer node set. \( E = \{(i, j) | i, j \in V\} \) is edge set, \( l_{ij} \) represents the distance between nodes \( i \) and \( j \), \( F_j \) is the fuel consumption of the vehicle from nodes \( i \) to and the fuel price is \( c_i \). Due to the symmetry of the graph, \( l_{ij} \) is the same as \( l_{ji} \) and \( F_j \) is different from \( F_i \) due to the asymmetry. \( k \) is any vehicle in the set of available delivery vehicles \( K = \{1, 2, \ldots, \varphi\} \), the capacity of each vehicle is \( Q \), \( c_i \) is the dispatch costs of a vehicle, the departure time of the vehicle from the depot is \( T^0_k \), and the maximum duration of a route is \( t \). The demand of each customer is uncertain before they are visited and is expressed by triangular fuzzy number \( d_j = (d_{j1}, d_{j2}, d_{j3}) \), \( d_{j1} \leq d_{j2} \leq d_{j3} \leq Q \), \( [ET, LT] \) is the time window of customer \( i \), penalty costs will be incurred when arriving at customers earlier than \( ET \) or later than \( LT \), the penalty costs per unit time for vehicles arriving earlier is \( c_i \), the penalty costs per unit time for vehicles arriving later is \( c_s \). \( T_i \) is the time when the vehicle \( k \) arrives at node \( i \), \( t_k^i \) is the service duration of the vehicle \( k \) at node \( i \). The decision variable \( x_{jk} \) indicates whether the vehicle \( k \) travels from \( i \) to \( j \), which is 1 and no is 0. The decision variable \( y_k \) indicates whether the customer \( i \) is served by the vehicle \( k \), and is 1 or 0.

2.2. Time-Dependent Function of Vehicle Speed

In the real world, vehicle speed changes continuously, such as the trigonometric function relation \( v(t) = \varphi \sin(\gamma t) + \delta \) (which \( \varphi, \gamma, \delta \) are related to road conditions) [24]. In this paper, the variation of vehicle speed on all-day roads is approximately expressed by a plurality of trigonometric function relations. The trigonometric expression between speed \( (v) \) and time \( (t) \) can be expressed as follows:
\[
\begin{align*}
v &= a_{\beta} \sin \left[ b_{\beta} \left( t - c_{\beta} \right) \right] + d_{\beta}, t \in \left[ T_{\beta}, T_{\beta+1} \right] \\
&\vdots \\
&= a_{\beta} \sin \left[ b_{\beta} \left( t - c_{\beta} \right) \right] + d_{\beta}, t \in \left[ T_{\beta}, T_{\beta+1} \right] \\
&= a_{\beta} \sin \left[ b_{\beta} \left( t - c_{\beta} \right) \right] + d_{\beta}, t \in \left[ T_{\beta}, T_{\beta+1} \right]
\end{align*}
\] (1)

Where \( a_{\beta}, b_{\beta}, c_{\beta}, d_{\beta}, \beta \) are related to road conditions.

The changing trend of vehicle speed on weekdays can be depicted through the research of Yang [27], as in Figure 1.

![Figure 1. Variation trend of vehicle speed.](image)

The whole day is divided into multiple periods, and each period has a different functional relationship between vehicle speed \( (v) \) and the time \( (t) \). \( T_{\beta} + T_{\beta}^{i} \) represents the time when the vehicle leaves node \( i \), \( T_{\beta} + T_{\beta}^{i} \in \left[ T_{\beta}, T_{\beta+1} \right] \). There are two conditions when the vehicle arrives at node \( j \). One is \( t_{j} \leq t_{j}^{i} \), the arrival time is within \( \left[ T_{\beta} + T_{\beta}^{i}, T_{\beta+1} \right] \). Otherwise, the vehicle needs to cross periods to arrive at node \( j \), that is \( t_{j} = \left( T_{\beta} + T_{\beta}^{i} - T_{\beta}^{i} \right) + t_{j}^{\beta+M} \), \( M \) represents the number of periods. Due to the asymmetry of the graph, \( t_{j} \) is different from \( t_{j}^{i} \).

### 2.3. Calculation of Fuel Consumption

This paper uses the MEET model [28] to calculate carbon emissions, we combined the research of Alinaghian and Naderipour [29] on 3.5 t-7.5 t vehicles to calculate the fuel consumption \( F_{y} \).

\[
F_{y} = \lambda \times \int_{T_{j}^{i} + \xi}^{T_{j}^{i}} \left[ (110 + 0.000375v^{3} + 8702 / v) \times GC \times LC \right] v dt 
\] (2)

In Equation (3), \( \lambda = 0.00043 \, L/g \), \( GC \) is the road gradient correction coefficient, \( LC \) is the load gradient correction coefficient. The calculations of \( GC \) and \( LC \) are:

\[
GC = \exp \left( (0.0059v^{2} - 0.0775v + 11.936) \xi \right)
\] (3)

\[
LC = 0.27 \xi + 1.061 + 0.0011 \xi^{2} \xi - 0.00235 \xi \omega - 0.33 / v \omega
\] (4)

In Equation (3), \( \xi \) is the road gradient in terms of percentage, in Equation (4), \( \omega \) is the loading rate.
In this paper, the fuzzy demand of the customer is considered, and the most probable value $d_i$ is selected as the demand of the customer $i$ to calculate the remaining capacity of the vehicle [7].

2.4. Pre-Optimization Stage

2.4.1. Fuzzy Demand Measure

In this paper, the decision-maker preference value $\alpha \in [0,1]$ is introduced, and the credibility measure theory [30] is used to calculate the credibility of the remaining vehicle capacity to meet the demand of the next customer. Assuming that the vehicle $k$ departs from the depot with a full load, and after the vehicle visits the $m$th customer, the remaining load of the vehicle is $	ilde{Q}_m = Q - \sum_{i=1}^{m} d_i$. Since the customer’s demand is a triangular fuzzy number, $Q$ is also the triangular fuzzy number.

$$Q = \left[ q_{1,m}, q_{2,m}, q_{3,m} \right]$$

Based on the credibility theory, when the vehicle continues to visit the $(m+1)$th customer, the credibility that the demand of customer $e$ is less than the remaining vehicle capacity is:

$$Cr \left\{ d_{m+1} \leq \tilde{Q}_m \right\} = \left\{ (d_{1,m+1} - q_{3,m}, d_{2,m+1} - q_{2,m}, d_{3,m+1} - q_{1,m}) \right\}$$

$$= \begin{cases} 0, & d_{1,m+1} \geq q_{3,m}, d_{2,m+1} \geq q_{2,m}, d_{3,m+1} \geq q_{1,m} \\ 2(q_{3,m} - d_{1,m+1} + d_{2,m+1} - q_{2,m}), & d_{1,m+1} \leq q_{3,m}, d_{2,m+1} \geq q_{2,m}, d_{3,m+1} \geq q_{1,m} \\ 2(q_{2,m} - d_{2,m+1} + d_{3,m+1} - q_{1,m}), & d_{1,m+1} \leq q_{3,m}, d_{2,m+1} \leq q_{2,m}, d_{3,m+1} \geq q_{1,m} \\ 1, & d_{1,m+1} \leq q_{3,m}, d_{2,m+1} \leq q_{2,m}, d_{3,m+1} \leq q_{1,m} \end{cases}$$

$Cr \left( Cr \in [0,1] \right)$ represents the credibility of $d_{m+1} \leq \tilde{Q}_m$, the larger $Cr$ means the higher the credibility of the vehicle to meet the fuzzy demand of customers $e$. The fuzzy chance constraint is introduced and $\alpha$ is preset. When $Cr > \alpha$, the vehicle will continue to visit the next customer, otherwise, a new vehicle will be reassigned to start a new route. $\alpha$ is the critical value for the vehicle to visit the next customer, and it is also a reflection of the risk attitude of the decision-maker. When $\alpha = 1$, it indicates that the decision-maker is extremely conservative, and the vehicle is allowed to visit the next customer only when the remaining load of the vehicle can fully meet the demand of the customer. When $\alpha = 0$, it indicates that the decision-makers are extremely radical, no matter how small $Cr$ is, vehicles are required to continue to visit. It is generally believed that $\alpha \in (0,1]$.

2.4.2. Mathematical Model

The model of this paper is as follows:

$$\min C = c_1 \sum_{ij \in K} \sum_{l, k^l} x_{ijl} F_l + c_2 \sum_{ij \in K} \sum_{l} x_{ijl} + c_3 \sum_{i, j, k^l} \sum_{l} x_{ijkl} \max \{(ET_j - T_{ij}), 0\}$$

$$+ c_4 \sum_{i, j, k^l} \max \{(T_{ij} - LT_j), 0\}$$

s.t,

$$Cr \left\{ \sum_{j, l} x_{ijl} d_{j} \leq Q \right\} \geq \alpha, \forall k \in K$$

$$\sum_{j, l} x_{ijl} \leq |K|$$
\[
\sum_{i \in I} \sum_{k \in K} x_{ia} = \sum_{j \in J} \sum_{k \in K} x_{ja} = 1, \forall j \in V
\]

\[
\sum_{i \in I} \sum_{j \in J} x_{ia} = \sum_{j \in J} \sum_{k \in K} x_{ja} \leq 1, \forall k \in K
\]

\[
x_{ia} = 0, \forall i = j, \forall i, j \in V, \forall k \in K
\]

\[
\sum_{i \in I} x_{ia} = y_{ia}, \forall j \in V, \forall k \in K
\]

\[
\sum_{j \in J} x_{ja} = x_{ia}, \forall i \in V, \forall k \in K
\]

\[
\sum_{i \in I} \sum_{j \in J} t_{ai} x_{ia} + \sum_{j \in J} \sum_{k \in K} t_{ak} x_{ja} \leq t, \forall k \in K
\]

\[
T_a + t_{ia} + t_{aj} - M(1 - x_{ia}) \leq T_a, \forall k \in K, \forall i, j \in V
\]

\[
T_a + t_{ia} + t_{aj} + M(1 - x_{ia}) \geq T_a, \forall k \in K, \forall i, j \in V
\]

\[
x_{ia} \in \{0, 1\}, \forall i, j \in V, \forall k \in K
\]

\[
y_{ia} \in \{0, 1\}, \forall j \in V, \forall k \in K
\]

Equation (7) is objective function, which minimizes the sum of the dispatch costs, fuel costs and time window penalty costs. Equation (8) is a fuzzy capacity chance constraint, which ensures that the reliability of the remaining vehicle capacity meeting customer demand is higher than the preset confidence level. Equation (9) indicates that the number of vehicles used for routing optimization does not exceed the total number of vehicles. Equation (10) indicates each customer is served only once, and it is the balance constraint. Equation (11) indicates that each vehicle has only one route, and depart from the depot and return to the depot after the distribution task is completed. Equation (12) indicates that the same customer has no route connection. Equations (13) and (14) ensure that customers must have a route to connect with when they are visited by vehicles. Equation (15) restricts the maximum duration of vehicles. Equations (16) and (17) indicate that when the vehicle \( k \) travels from node \( i \) to node \( j \), the time when the vehicle arrives at node \( j \) is equal to the time when the vehicle arrives at node \( i \) plus the service time at node \( i \) and the travel time on the arc \((i, j)\), where \( M \) is an infinite positive number. Equations (18) and (19) are decision variables.

2.5. Re-Dispatch Strategy

In the actual distribution process, even if the vehicle load in the pre-optimization scheme meets the credibility test of all customers, there may be situations where the customer demand exceeds the remaining vehicle load. Therefore, it is necessary to re-optimize the distribution scheme for customers who have failed to visit.

In the research of vehicle routing problems with fuzzy demand, the methods of dealing with customers who have failed to visit are mostly failure node return strategy and pre-failure node return strategy [31-33].

(1) Failure node return strategy: As shown in Figure 2(b), when the vehicle fails to serve customer 3, the vehicle returns to the depot to be reloaded and then continues to visit customer 3 and its subsequent customers according to the pre-optimization scheme, the same is true for customer 7.

(2) Pre-failure node return strategy: As shown in Figure 2(c), Before the vehicle visits customer 3, the credibility that the remaining load of the vehicle meets the demand of customer 3 is pre-judged, and when the credibility is less than the preset value, the vehicle returns to the depot for loading and then visits according to the pre-optimized route.
For the above two strategies, the failure node return strategy must be inferior to the pre-failure node return strategy, because it is known from the triangle trilateral principle: \( l_{13} + l_{30} > l_{30} \). The pre-failure node return strategy has strict requirements on node selection, and it is difficult to select the appropriate return node. If the selection is not appropriate, the distribution costs will increase. Furthermore, adopting the failure node return strategy and the pre-failure node return strategy may cause more vehicles returning to the depot to reload and perform delivery services again, increasing the number of vehicles used. Since the customer demand is fuzzy, to avoid the vehicle failing to visit the customer again, the vehicle should be fully loaded or loaded according to the upper bound \( d_{30} \) of the triangular fuzzy number when leaving the depot again, which may lead to excessive goods left when the vehicle returns to the depot, which will have a bad effect to the goods and fuel costs. Given the shortcomings of the above strategy, this paper adopts the re-dispatch strategy \([34]\) for customers who fail in service, as shown in Figure 2(d). When the vehicle fails to meet the customer’s demand (the failure reason of customer 3) or fails to serve the customer through pre-judgment (the failure reason of customer 7), the vehicle returns to the depot, counts the unserved customers (customer 3, 4, 7) in all routes, and re-optimizes the distribution route according to the principle of the nearest neighbor method. In the re-optimization stage, the credibility of the customers to be visited by the vehicle is also checked, and the preference value \( \alpha \) is set at a higher level. At this time, there may also be customers who have failed in the service, and a failed node return strategy may be adopted for them. Due to the small scale of customers, the probability of customers who fail again in re-optimizing routes is extremely small.

3. Solution Methodology

Genetic algorithm (GA) originates from the computer simulation of biological systems. It is a stochastic global search and optimization method developed by imitating the biological evolution mechanism in nature. It draws on Darwin’s theory of evolution and Mendel’s theory of heredity. Its essence is an efficient, parallel and global search method, which can automatically acquire and accumulate knowledge about the search space in the search process, and adaptively control the search process to acquire the best solution. However, GA is easy to fall into local optimum. Variable neighborhood search algorithm (VNS) is an improved local search algorithm. It uses the neighborhood structure formed by different actions to search alternately and achieves a good balance between centrality and dispersity. The idea can be summarized as “change makes perfect progress”. Combining the advantages of GA and VNS, a chaotic genetic algorithm with variable neighborhood search (CGA_VNS) is designed to solve the TDGVRPTWFD. Figure 3 is the basic flow of CGA_VNS.
3.1. Generation of the Initial Population

The initial population of the genetic algorithm is mostly completely random. According to Darwin’s theory of evolution, most seemingly random species in biological systems evolve but all follow certain rules. Chaotic systems are characterized by internal randomness, boundedness, ergodicity and dependence on initial values. In this paper, the logistic chaotic map is used to generate an initial population to ensure the uniformity of initial population distribution. Equation (20) is the mathematical formula of the logistic chaotic map.

\[ x_{n+1} = r x_n (1 - x_n), n = 1, 2, \ldots, r \in (3.57, 4], r \in [0,1] \]  

In the chaotic equation \( y = f(x), x \in [a,b] \), if there is an initial value \( x_0 \in [a,b] \) that makes \( f(0) = x_0 \), then \( x_0 \) is called the fixed point of \( y \). In logistic chaotic mapping, \( x_0 = 0.25, 0.5, 0.75 \) is a fixed point, and Figure 4 shows the properties of the fixed point. From Figure 4, it can be seen that when the initial value \( x_0 \) is a fixed point, the system loses ergodicity. \( x_0 \in [0.1], x_0 \neq 0.25, 0.5, 0.75 \) generates pseudo-random numbers \( x_n \in [0.1] \) after many iterations. When \( r = 4, x_0 \in [0.1] \) is constant, and the system is in a completely chaotic state and has all the characteristics of a chaotic system. That is to say, the sequence generated by the initial condition \( x_0 = 0.3 \) is aperiodic and non-convergent.

Most of the initial populations of traditional genetic algorithms are completely randomly generated, while according to Darwin’s theory of evolution, most seemingly random species evolution in real biological systems follows certain rules—genetic theory. In order to better understand the advantages of chaos, this paper compares and analyzes chaotic phenomena and random phenomena. Although chaos and randomness appear to be unpredictable on the surface, the unpredictability of chaos is due to the lack of existing technology, which makes it difficult to overcome problems such as computing power or data collection. Long-term accurate prediction cannot be carried out, etc.; and the unpredictability of random phenomena is its essential property, which is congenital. In this paper, the internal randomness of chaos is used to ensure the diversity of initial solutions in genetic algorithm.
Figure 4. Logistic chaotic mapping fixed point.

The steps for generating the initial population are as follows:

1. Setting parameters: population size is \( \text{pop\_size} \), chromosome length is \( n \), logistic chaotic mapping coefficient is \( r \).
2. The initial values of \( \text{pop\_size} \) logistic chaotic maps are randomly generated.
3. The chaotic system of \( \text{pop\_size} \) groups (each group consists of \( n \) values) is calculated by Equation (20).
4. The initial population is generated according to the positions of \( n \) values in the chaotic system within their respective groups.

After the initial population is generated based on the logistic chaotic map, the nearest neighbor method is used to decode. Customers are divided into vehicles according to the initial sequence through vehicle load constraint and duration constraint tests. When the vehicle cannot meet the constraint, a new vehicle is dispatched to visit the customer and insert 0 before the customer who cannot meet the constraint, and so on until the last customer, and insert 0 at the beginning and end of the chromosome to complete decoding. As shown in Figure 5, assuming that the order of customers is 7-3-2-5-4-6-1-8, the first vehicle is dispatched to visit according to the order of customers. If the vehicle cannot meet the constraint when visiting customer 5, it will dispatch a new vehicle to visit customer 5, insert 0 in front of customer 5, and so on until customer 8 is visited, and insert 0 in front of customer 7 and after customer 8 to complete decoding. Finally, three routes are formed, i.e., vehicle1: 0-7-3-2-0; vehicle2: 0-5-4-6-1-0; vehicle3: 0-8-0.

![Decoding diagram](image)

Figure 5. Decoding diagram.

3.2. Fitness Function
The fitness function can be calculated as follows:

\[
f_R = \frac{1}{C_R}
\]

where \( C_R \) represents the objective function.

3.3. Select Operation

The selection operation combines the elite reservations with the roulette wheel selection. First, the roulette wheel selection is to calculate the selected probability. Secondly, individual \( A \) with the highest probability is stored in the temporary matrix. After completing the search, if there is no better solution, it calculates the similarity between individual \( A \) and the sub-generation individuals. Next, it replaces the individual with the highest similarity.

3.4. Evolutionary Operation

In accordance with Figure 6, it selects two parents from individuals randomly. Firstly, points \( i_{11}, i_{12}, i_{21}, \) and \( i_{22} \) are randomly generated, respectively, the part between \( i_{11} \) and \( i_{12} \) of the parent \( A \) is taken as the first segment of the sub-generation \( A_1 \) and the subsequent points of the sub-generation \( A_1 \) are related to the parent \( B \), i.e., the customer points between \( i_{11} \) and \( i_{12} \) in the parent \( B \) are eliminated first. In the elimination process, the position sequence of the customer points in the parent \( B \) is not changed, and then the eliminated customer points are arranged as the second segment of the sub-generation \( A_1 \) to form the sub-generation \( A_1 \) and the sub-generation \( B_1 \).

![Figure 6. The diagram of the order crossover operator.](image)

3.5. Local Search Strategy

Local search strategy is the core function of the algorithm, which determines the local search capability of the algorithm. In this paper, we adopt VNS algorithm applied to the local search strategy to strengthen the search capability.

3.5.1. Neighborhood Structure

1. Insert: Randomly select customers \( i \) and \( j \), and insert customer \( i \) after customer \( j \). As shown in Figure 7(a), customer 3 is inserted after customer 6.

2. Exchange: Randomly select customers \( i \) and \( j \), then their positions are exchanged. As shown in Figure 7(b), customer 3 and customer 6 are exchanged their locations.

3. 2-OPT: Randomly select customers \( i \) and \( j \), and the order of other customers among them are exchanged. As shown in Figure 7(c), the position of customer 3 is kept unchanged, and customers 4, 5, 7, and 6 are in reverse order.
Figure 7. Neighborhood structures.

3.5.2. Adaptive Mechanism and Solution Acceptance

We design an adaptive neighborhood search number strategy and a new solution acceptance rule to balance the breadth and depth required for evolution, so that the algorithm jumps out of the local optimum. The disturbance intensity required by the population is different at different stages of the algorithm iteration. The strategy of adaptive neighborhood search times in this paper is as follows:

\[ S_x = \tau_1 + \left\lfloor \tau_2 \cdot \left( \frac{\text{gen}}{\text{MAXGEN}} \right) \right\rfloor \]  

(22)

In Equation (22), \( S_x \) is the number of search times, \( \tau_1 \) is the minimum number of searches, \( \tau_2 \) is the adaptive number of searches, \( \left\lfloor \cdot \right\rfloor \) represents round down, \( \text{gen} \) is the current iteration number, \( \text{MAXGEN} \) is the maximum iteration number. It can be seen from Equation (22) that as the iteration number increases, the search times and depth gradually increase.

Equation (23) is the probability of new solution acceptance. \( x \) is a new solution.

\[ p = \begin{cases} 1, & C(x) \geq C(\hat{x}) \\ \exp \left( \frac{C(x) - C(\hat{x})}{\text{gen}} \right), & C(x) < C(\hat{x}) \end{cases} \]  

(23)

4. Instance Verification and Result Analysis

4.1. Algorithm Test

At present, there is no benchmark instance of TDGVRPFDTW. This paper selects instances of VRPTW and VRPFD to test the algorithm. The algorithm is programmed with MATLAB2018b in the Windows10 system. The proper setting of parameters has a great influence on the performance of heuristic algorithm. We analyze the function of parameters first. Next, we analyze the influence of parameters on solutions. Finally, we set the parameters through experiments. Setting the population size \( \_\_\_\_\text{pop size} \) and the maximum iteration number \( \_\_\_\_\text{MAXGEN} \) increases with the increase of the number of customers, and setting the adaptive parameters \( \tau_1 = 1, \tau_2 = \text{MAXGEN} / 8 \), and the local search repetition number \( \_\_\_\_\text{max_}N = 50 \) of each generation.

4.1.1. Instance Analysis of VRPTW

In this paper, the R1 set in Solomon benchmark instances is selected (instance source: http://www.cba.neu.edu/~msolomon/problems.htm, 2nd Aug 2022). Some have very tight time windows, while others have time windows that are hardly constraining. Set \( c_2 = 150 \), \( c_3 = c_1 = 1 \), the transportation cost between two nodes is \( C_{ij} \). Table 1 shows the best-known solution under hard time constraints and the operation results of the wasp colony
algorithm (WCA) [35] and CGA_VNS under soft time window constraints. \( N \) is the number of vehicles, \( \text{Trans\_cost} \) is the transportation cost, \( \text{P\_cost} \) is the time window penalty costs, \( \text{Total\_cost} \) is the total costs, \( \text{Gap} \) is the deviation between the optimal solution obtained by CGA_VNS and WCA, and \( \text{Ave\_gap} \) is the average value of \( \text{Gap} \).

The time complexity of CGA_VNS is \( O(\text{MAXGEN} \cdot \text{Max}_N \cdot \text{Sn} \cdot \text{Pop\_size} \cdot N) \). It takes about 2.5 min to solve the R1. Table 1 shows that in the results of the 10 benchmark instances, the minimum \( \text{Gap} \) is −21.38%, the maximum \( \text{Gap} \) is 5.48%, and \( \text{Ave\_gap} \) is −5.96%. Although the number of vehicles used in the results of R107, R108, and R108 optimized by CGA_VNS is more than that optimized by WCA, the time window penalty cost is greatly reduced, and the total cost is smaller. The effectiveness of CGA_VNS in solving VRPSTW can be verified by comparing the results.

Table 1. Comparison of CGA_VNS with other heuristics on VRPSTW instances.

| Instance | VRPHTW | WCA | CGA_VNS | Gap |
|----------|--------|-----|---------|-----|
|          | \( N \) | \( D \) | \( \text{Trans\_cost} \) | \( \text{P\_cost} \) | \( \text{Total\_cost} \) | \( \text{Trans\_cost} \) | \( \text{P\_cost} \) | \( \text{Total\_cost} \) | \( \text{Ave\_gap} \) |
| R101     | 19     | 1645.79 | 12 | 1564.49 | 721.20 | 4085.69 | 11 | 1347.89 | 1226.88 | 4224.77 | 3.40% |
| R102     | 17     | 1486.12 | 11 | 1376.37 | 760.00 | 3786.37 | 11 | 1336.07 | 841.29 | 3827.36 | 1.08% |
| R103     | 13     | 1292.68 | 9  | 1141.14 | 1173.76 | 3664.9 | 10 | 1163.59 | 417.57 | 3081.16 | −15.93% |
| R104     | 9      | 1007.24 | 8  | 1017.82 | 811.53 | 3029.35 | 9  | 936.02 | 476.41 | 2762.43 | −8.81% |
| R105     | 14     | 1377.11 | 10 | 1228.63 | 1055.70 | 3784.33 | 10 | 1172.32 | 1113.58 | 3785.9 | 0.04% |
| R106     | 12     | 1251.98 | 9  | 1127.40 | 1395.04 | 3872.44 | 9  | 1041.13 | 1116.56 | 3507.69 | −9.42% |
| R107     | 10     | 1104.66 | 8  | 1066.83 | 1451.95 | 3718.78 | 9  | 1015.70 | 557.93 | 2923.63 | −21.38% |
| R108     | 9      | 960.88  | 8  | 986.88  | 760.24 | 2947.12 | 9  | 946.00 | 198.18 | 2494.18 | −15.37% |
| R109     | 11     | 1194.73 | 9  | 1137.75 | 908.40 | 3396.15 | 10 | 1152.49 | 786.24 | 3438.73 | 1.25% |
| R110     | 10     | 1118.59 | 9  | 1100.66 | 553.29 | 3003.95 | 9  | 1091.81 | 726.84 | 3168.65 | 5.48% |

WCA runs with Microsoft Visual Studio 2005 in Windows XP, and the time complexity is \( o(m^2) \).

4.1.2. Instance Analysis of VRPF

VRPF is the basic problem of this paper. To test the performance of CGA_VNS to solve VRPF, improve the A part of the reference [36] to form a test set. Let \( \hat{d} = (0.75d, d \cdot 1.25d) \), the actual customer demand is \( d \). Table 2 shows the pre-optimization results of GA, VNS, and CGA_VNS under different preference values, and \( \alpha \) is the average time spent by the algorithm. Figure 8 shows the iterative process of solving A-n48-k7 by three algorithms and the optimal routing of CGA_VNS optimization when the decision preference value is 0.4. It can be seen from Table 2 and Figure 8(a) that the convergence speed of GA is slow, the time spent in one experiment is relatively long, and the solution quality is poor. Compared with VNS, CGA_VNS adds an order crossover operator. Although CGA_VNS takes a little longer time to run once than VNS, it increases its local search ability and has better solution quality, and obtains the best solution of all instances under different decision preference values. Therefore, CGA_VNS has better performance than GA and VNS.

Table 2. Comparison of CGA_VNS with GA and VNS.

| Instance & Algorithm | \( \alpha \) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 | t/s |
|---------------------|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|-----|
| CGA_VNS             |             | 711 | 724 | 768 | 770 | 802 | 818 | 851 | 877 | 902 | 902 | 7.3 |
| A-n36-k5            | VNS         | 712 | 724 | 779 | 770 | 811 | 831 | 866 | 899 | 902 | 906 | 6.2 |
| GA                  |             | 803 | 835 | 845 | 882 | 879 | 945 | 944 | 965 | 974 | 1007 | 18.21 |

\[\text{Ave\_gap} \]
A-n44-k7  CGA_VNS  852  872  898  933  939  965  1010  1031  1063  1093  12.0  VNS  864  872  904  944  949  978  1030  1035  1069  1108  9.5  GA  995  1079  1101  1131  1200  1140  1148  1224  
A-n48-k7  CGA_VNS  942  977  1038  1064  1111  1115  1179  1221  1240  1267  14.1  VNS  959  996  1040  1075  1109  1142  1224  1238  1243  1309  11.4  GA  1194  1212  1264  1258  1298  1301  1307  1401  1407  1490  25.8  
A-n55-k9  CGA_VNS  951  992  1012  1061  1082  1122  1163  1178  1233  1282  20.7  VNS  957  1015  1025  1068  1122  1157  1175  1237  1246  1279  17.1  GA  1222  1295  1312  1340  1375  1340  1337  1401  1407  1490  35.4  
A-n65-k9  CGA_VNS  1064  1106  1138  1185  1195  1247  1286  1313  1333  1380  26.8  VNS  1097  1096  1168  1223  1259  1305  1298  1373  1477  1487  24.2  GA  1484  1538  1508  1525  1615  1671  1741  1688  1745  1706  63.8  
A-n69-k9  CGA_VNS  1064  1106  1138  1185  1195  1247  1286  1313  1333  1380  26.8  VNS  1072  1121  1149  1241  1237  1310  1321  1350  1375  1390  25.6  GA  1255  1321  1483  1479  1582  1547  1554  1574  1610  1687  71.35

Table 3. Results of comparison between CGA_VNS and VNSS, VNTS.

| α   | VNSS | VNTS | CGA_VNS |
|-----|------|------|---------|
|     | N    | Best | Ave     | N    | Best | Ave      | N    | Best | Ave      | t/s | Gap   |
| 0.1 | 4    | 686  | 686     | 4    | 686  | 686      | 4    | 687  | 687      | 4.3 | 0.14% |
| 0.2 | 4    | 719  | 719     | 4    | 719  | 719      | 4    | 719  | 719      | 4.3 | 0 |
| 0.3 | 4    | 727  | 727     | 4    | 727  | 727      | 4    | 727  | 727      | 4.2 | 0 |
| 0.4 | 5    | 755  | 755.7   | 5    | 755  | 755      | 4    | 755  | 755      | 4.5 | 0 |

Figure 8. (a) Iterative process for solving A-n48-k7. (b) Delivery routing optimized by CGA_VNS.

In this paper, Zhang et al. [5] improved A-n32-k5 is selected to further verify the performance of the algorithm. The optimal solutions of variable neighborhood scatter search algorithm (VNSS) [5], variable neighborhood tabu search algorithm (VNTS) [6] and the CGA_VNS running 10 times are summarized in Table 3. N represents the number of vehicles used in the pre-optimization scheme, Best represents the best value of the pre-optimization scheme, Ave is the average of the results of 10 operations and Gap is the deviation between the best solution obtained by CGA_VNS and the best-known solution. t is the time spent in one experiment. It can be seen from Table 3 that the results obtained by CGA_VNS are not inferior to VNSS and VNTS except that α is 0.1, and the results obtained by CGA_VNS are relatively stable. The performance of the algorithm is further verified.
0.5 5 784 784 5 784 784 5 784 784 4.3 0
0.6 5 796 796 5 796 796 5 796 796 4.2 0
0.7 5 848 848 5 848 848 5 848 851.1 4.2 0
0.8 5 880 883.3 5 868 868 5 868 868 4.5 0
0.9 5 882 887.6 5 882 882 5 882 882.8 4.3 0
1.0 6 892 892.2 6 892 892 6 892 893.3 4.4 0

4.2. Numerical Experiment

At present, there are no benchmark instances of the TDGVRPTWFD. This paper improves the data of the top 50 customers of R101 in the Solomon database as the instance in this paper. The parameters are set as follows: \( d = (0.25d/25, d/25.1.25d/25) \), the customer’s time window is \([ET/20 + 0.5, LT/20 + 0.5]\), \(Q = 5t\), \(c_1 = 6.9\), \(c_2 = 500\), \(c_3 = 2\), \(c_4 = 3\), \(T_0 = 6:00\), \(t_0 = 0.5\). We consider three types of roads, urban main roads, secondary roads, and branch roads. In Figure 9, red and black represent main roads and secondary roads respectively. Undrawn routes are branch roads. Figure 10 is vehicle speeds.

Figure 9. Distribution road network.

Figure 10. The time-dependent function of speed.

4.2.1. Pre-Optimization Stage

To verify the performance of CGA_VNS to solve the problems in this paper, GA, VNS, and CGA_VNS were used to solve the instance in the pre-optimization stage and
the results were compared. Table 4 shows the pre-optimization results of the algorithm under different preference values, where \( \text{Best/N} \) represents the optimal value of the solution and the number of vehicles used, \( \text{Worst/N} \) represents the worst value of the solution and the number of vehicles used, \( \text{Ave} \) represents the average value of the solution results, and SD is the standard deviation of the solution results. Figure 11 shows the convergence process of the algorithm when \( \alpha = 0.4 \). Figure 12 is a pre-optimized delivery route solved by CGA_VNS when \( \alpha = 0.8 \).

| \( \alpha \) | GA | VNS | CGA_VNS |
|---|---|---|---|
|   | \( \text{Best/N} \) | \( \text{Worst/N} \) | \( \text{Ave} \) | \( \text{SD} \) | \( \text{Best/N} \) | \( \text{Worst/N} \) | \( \text{Ave} \) | \( \text{SD} \) | \( \text{Best/N} \) | \( \text{Worst/N} \) | \( \text{Ave} \) | \( \text{SD} \) |
| 0.1 | 4170.9/5 | 4198.0/5 | 4188.3 | 12 | 3995.0/5 | 4090.0/5 | 4045.6 | 39 | 3092.4/4 | 3120.4/4 | 3109.7 | 12 |
| 0.2 | 4011.1/5 | 4117.1/5 | 4066.7 | 43 | 3992.5/5 | 4025.5/5 | 4013.3 | 15 | 3067.4/4 | 3106.5/4 | 3091.7 | 17 |
| 0.3 | 3968.8/5 | 4320.8/5 | 4128 | 146 | 3981.4/5 | 4038.3/5 | 4010.7 | 23 | 3622.0/4 | 3640.8/5 | 3640 | 13 |
| 0.4 | 4700.8/6 | 4723.5/6 | 4713 | 9 | 3965.1/5 | 4604.3/6 | 4188.4 | 294 | 3644.8/5 | 3697.3/5 | 3673.9 | 22 |
| 0.5 | 4547.7/6 | 4599.2/6 | 4566.3 | 23 | 4472.0/6 | 4517.4/6 | 4491.6 | 19 | 4249.5/6 | 4311.4/6 | 4272.5 | 28 |
| 0.6 | 5025.6/7 | 5117.9/7 | 5073.8 | 38 | 4994.4/7 | 5042.8/7 | 5013.7 | 21 | 4797.3/7 | 4820.0/7 | 4807.2 | 9 |
| 0.7 | 5030.9/7 | 5058.0/7 | 5045.7 | 11 | 4891.8/7 | 5021.8/7 | 4953.6 | 53 | 4779.7/7 | 4863.1/7 | 4811.8 | 37 |
| 0.8 | 5067.0/7 | 5149.3/7 | 5108 | 34 | 5034.2/7 | 5175.3/7 | 5107.1 | 59 | 4832.3/7 | 4866.3/7 | 4848.8 | 14 |
| 0.9 | 5265.3/7 | 5455.7/7 | 5321.9 | 120 | 5338.9/7 | 5605.3/8 | 5522.6 | 130 | 4912.7/7 | 4958.3/7 | 4936.7 | 19 |
| 1 | 5654.5/8 | 5720.0/8 | 5702.1 | 34 | 5556.6/8 | 5642.0/8 | 5591.5 | 37 | 5402.1/8 | 5426.3/8 | 5409.3 | 12 |

Ave: 47

It can be seen from Table 4: (1) CGA_VNS obtains the pre-optimized optimal solution under all preference values and the standard deviation of the result is the smallest, which shows that CGA_VNS has higher solution quality and better solution stability than the other two algorithms. It can be seen from Figure 11 that CGA_VNS has a faster convergence speed. (2) Decision-makers’ preference value has a great influence on distribution costs. The costs of the pre-optimization solution increase with the increase of \( \alpha \), and the number of vehicles dispatched also increases, gradually increasing from 4 vehicles with \( \alpha \in \{0.1, 0.2\} \) to 8 vehicles with \( \alpha = 1 \).

![Figure 11. Convergence process diagram (\( \alpha = 0.4 \)).](image-url)
4.2.2. Re-Optimization Stage

Assuming that the actual demand of the served customers is \( d_1 = d/25 \), Table 5 shows the re-optimization results corresponding to the best pre-optimization results under different preference values. Strategy 1 is the failure node return strategy, strategy 2 is the pre-failure node return strategy, and strategy 3 is the optimization strategy in this paper. \( \text{Pre-}\)best is the best pre-optimization result, \( \text{E-}\)cost is the extra costs of the pre-optimization scheme due to route failure, \( T-\)cost is the total distribution costs after re-optimization, \( \text{Re-}\)best is the lowest total costs of the three strategies. Table 5 shows that when \( \alpha \in [0.1, 0.4] \) the pre-optimization scheme needs to be re-optimized, the extra costs of the distribution scheme solved by the strategy in this paper are lower than the other two strategies. Table 6 gives detailed adjustment results when \( \alpha = 0.4 \).

Table 5. Results of different re-optimization strategies.

| \( \alpha \) | Pre-Best | Strategy 1 | Strategy 2 | Strategy 3 | Re-Best |
|------|---------|------------|------------|------------|---------|
| 0.1  | 3092.4  | 536.8      | 3629.2     | 466.3      | 3558.7  |
| 0.2  | 3067.4  | 514.6      | 3582.0     | 460.1      | 3527.5  |
| 0.3  | 3622.0  | 544.6      | 4166.6     | 496.8      | 4118.8  |
| 0.4  | 3644.8  | 594.5      | 4239.3     | 533.1      | 4177.9  |
| 0.5  | 4249.5  | 0          | 4249.5     | 0          | 4249.5  |
| 0.6  | 4797.3  | 0          | 4797.3     | 0          | 4797.3  |
| 0.7  | 4779.7  | 0          | 4779.7     | 0          | 4779.7  |
| 0.8  | 4832.3  | 0          | 4832.3     | 0          | 4832.3  |
| 0.9  | 4912.7  | 0          | 4912.7     | 0          | 4912.7  |
| 1    | 5402.1  | 0          | 5402.1     | 0          | 5402.1  |

Table 6. Re-optimization result when \( \alpha = 0.4 \).

| Pre-Optimization Scheme | Strategy 1 | Strategy 2 | Strategy 3 |
|-------------------------|------------|------------|------------|
| V1: 0-27-31-7-11-32-33-3-29-24-25-26-0; | V1: 0-27-31-7-11-32-33-3-29-24-0-25-26-0; | V1: 0-27-31-7-11-32-33-3-29-24-0-25-26-0; |
| V2: 0-28-12-4-39-23-22-21-0; | V2: 0-28-12-4-39-23-0-22-21-0; | V2: 0-28-12-4-39-23-0-22-21-0; |
| V3: 0-18-8-47-36-49-19-20-9-35-34-0; | V3: 0-18-8-47-36-49-19-20-9-0-35-34-0; | V3: 0-18-8-47-36-49-19-20-9-0-35-34-0; |
| V4: 6-5-17-45-46-48-10-30-50-1-0; | V4: 6-5-17-45-46-48-10-30-50-0-50-1-0; | V4: 6-5-17-45-46-48-10-30-50-0-50-1-0; |
| V5: 0-40-2-41-15-43-42-14-44-38-16-37-0; | V5: 0-40-2-41-15-43-42-14-44-38-16-37-0; | V5: 0-40-2-41-15-43-42-14-44-38-16-37-0; |
| V1: 0-25-13-22-21-26-0; | V1: 0-25-13-22-21-26-0; | V1: 0-25-13-22-21-26-0; |
Based on the above analysis, the essence of the three strategies is to maintain a certain degree of balance between risk and costs. The strategy in this paper is better than strategy 1 and strategy 2 on the whole.

5. Conclusions

In reality, due to the existence of objective and subjective factors, customer demand is often fuzzy and uncertain. Meanwhile, the dynamic change of traffic information will also affect the formulation of the distribution scheme. For TDGVRPTWFD, this paper comprehensively considers the influence of the dispatch costs, fuel costs, and time window penalty costs on the total costs, and establishes a fuzzy chance-constrained optimization model with time-varying vehicle speed and time window. In the pre-optimization stage, a pre-optimization scheme is generated based on the credibility measure theory. In the re-optimization stage, route adjustments will be made for customers who have failed in service. The chaotic genetic algorithm with variable neighborhood search is developed to solve the problem. Numerical results show that the model and algorithm are effective. The following conclusions are obtained:

(1) The fuzzy chance-constrained optimization model of TDGVRPTWFD has fully considered the fuzziness of customer demand in the actual distribution process, further expanding and deepening the VRP research.

(2) The model considers the time-dependent vehicle speed and the influence of vehicle speed, load, and road gradient on fuel consumption. After considering the above factors, the model becomes more complex and closer to the actual situation.

(3) The proposed algorithm uses logistic chaotic mapping to generate initial solutions, which ensures the diversity of initial solutions. Adaptive neighborhood search times strategy and inferior solution acceptance mechanism improve the performance of algorithm.

(4) In the pre-optimization stage, the distribution costs of the pre-optimization scheme increase with the increase of the preference value of decision-makers, and the number of vehicles dispatched also increases. In the re-optimization stage, the more conservative the decision-makers tend to be, the lower the extra costs they pay and the smaller the number of vehicles they dispatch.

The research in this paper takes into account the fuzziness of customer demand and the dynamic changes in the traffic environment, which is more in line with the actual distribution situation. The research results not only further enrich the research on VRP-related theories, but also provide a scientific basis for the distribution optimization decision-making of logistics enterprises.

Author Contributions: H.F. (Hao Fan) and X.R. provided the core idea of this paper, collected and analyzed the data. Y.Z. and Z.Z. wrote the manuscript. H.F. (Houming Fan) contributed to the conceptualization of this paper, and the provision of valuable comments. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Special Project of National Emergency Management System Construction of Chinese National Funding of Social Science, Grant number 20VYJ024.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Cao, E.; Lai, M. The open vehicle routing problem with fuzzy demands. *Expert Syst. Appl.* **2010**, *37*, 2405–2411.

2. Kuo, R.J.; Zulvia, F.E.; Suryadi, K. Hybrid particle swarm optimization with genetic algorithm for solving capacitated vehicle routing problem with fuzzy demand—A case study on garbage collection system. *Appl. Math. Comput.* **2012**, *219*, 2574–2588.

3. Wang, J.; Li, B. Dynamic management of vehicle routing problem with time windows and fuzzy demands based on the multi-objective optimization. *Chin. J. Manag.* **2013**, *10*, 238–243.

4. Liu, C.S.; Kou, G.; Huang, F.H. Vehicle coordinated strategy for vehicle routing problem with fuzzy demands. *Math. Probl. Eng.* **2016**, *9*, 9071394.

5. Zhang, X.N.; Fan, H.M. Optimization and real-time adjustment for vehicle routing problem with fuzzy demand. *Shanghai Jiao tong Univ*. **2016**, *50*, 123–130.

6. Li, Y.; Fan, H.M.; Zhang, X.; Yang, X. Two-stage variable neighborhood tabu search for the capacitated vehicle routing problem with fuzzy demand. *Sys. Eng. Theory Pract.* **2018**, *38*, 522–531.

7. Gallanza, A.; Li Puma, G. Fuzzy green vehicle routing problem for designing a three echelons supply chain. *J. Clean Prod.* **2020**, *259*, 120774.

8. Fan, H.; Li, D.; Kong, L.; Ren, X. Optimization for time dependent vehicle routing problem with fuzzy demand and time windows. *Con. Theory Appl.* **2020**, *37*, 950–960.

9. Hashimoto, H.; Yagiura, M.; Imahori, S.; Ibaraki, T. Recent progress of local search in handling the time window constraints of the vehicle routing problem. *Ann. Oper. Res.* **2010**, *8*, 221–238.

10. Yu, Y.; Wang, S.H.; Wang, J.W.; Huang, M. A branch-and-price algorithm for the heterogeneous fleet green vehicle routing problem with time windows. *Transp. Res. Pt. B Methodol.* **2019**, *122*, 511–527.

11. Wang, Y.; Wang, L.; Chen, G.C.; Cai, Z.Q.; Zhou, Y.Q.; Xing, L.N. An improved ant colony optimization algorithm to the periodic vehicle routing problem with time window and service choice. *Swarm Evol. Comput.* **2020**, *55*, 100675.

12. Marinakis, Y.; Marinaki, M.; Migdalas, A. A multi-adaptive particle swarm optimization for the vehicle routing problem with time windows. *Inf. Sci.* **2019**, *481*, 311–329.

13. Perez-Rodriguez, R.; Hernandez-Aguirre, A. A hybrid estimation of distribution algorithm for the vehicle routing problem with time windows. *Comput. Ind. Eng.* **2019**, *130*, 75–96.

14. Taniguchi, E.; Shimamoto, H. Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times. *Transp. Res. Pt. C-Emerg. Technol.* **2014**, *12*, 235–250.

15. Mancini, S. Time dependent travel speed vehicle routing and scheduling on a real road network: The case of Torino. *Transp. Res. Procedia* **2014**, *3*, 433–441.

16. Haghani, A.; Jung, S. A dynamic vehicle routing problem with time-dependent travel times. *Comput. Oper. Res.* **2005**, *32*, 2959–2986.

17. Hashimoto, H.; Yagiura, M.; Ibaraki, T. An iterated local search algorithm for the time-dependent vehicle routing problem with time windows. *Discret. Optim.* **2008**, *5*, 434–456.

18. Hu, C.; Lu, J.; Liu, X.; Zhang, G. Robust vehicle routing problem with hard time windows under demand and travel time uncertainty. *Comput. Oper. Res.* **2018**, *94*, 139–153.

19. Figliozi, M.A. The time dependent vehicle routing problem with time windows: Benchmark problems, an efficient solution algorithm, and solution characteristics. *Transp. Res. Pt. E-Logist. Transp. Res.* **2012**, *48*, 616–636.

20. Mu, D.; Wang, C.; Wang, S.C.; Zhou, S.C. Solving TDVRP based on parallel-simulated annealing algorithm. *Comput. Int. Manuf. Syst.* **2015**, *21*, 1626–1636.

21. Kuo, Y.Y. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Comput. Ind. Eng.* **2010**, *59*, 157–165.

22. Liu, C.S.; Kou, G.; Zhou, X.C.; Peng, Y.; Sheng, H.F.; Alsaadi, F.E. Time-dependent vehicle routing problem with time windows of city logistics with a congestion avoidance approach. *Knowl.-Based Syst.* **2020**, *188*, 104813.

23. Poonthalir, G.; Nadarajan, R. A fuel efficient green vehicle routing problem with varying speed constraint (F-GVRP). *Expert Syst. Appl.* **2018**, *100*, 131–144.

24. Xu, Z.; Elomri, A.; Pokharel, S.; Mutlu, F. A model for capacitated green vehicle routing problem with the time varying vehicle speed and soft time windows. *Comput. Ind. Eng.* **2019**, *137*, 106011.

25. Xiao, Y.; Konak, A. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. *Transp. Res. Pt. E-Logist. Transp. Res.* **2016**, *88*, 146–166.

26. Ç imen, M.; Soysal, M. Time-dependent green vehicle routing problem with stochastic vehicle speeds: An approximate dynamic programming algorithm. *Transp. Res. Part D-Transp. Environ.* **2017**, *54*, 82–98.

27. Yang, H. Research of Urban Recurrent Congestion Evolution based on Taxi GPS Date. Doctoral Thesis, Harbin Institute of Technology, Harbin, China, 2018.

28. Hickman, J.; Hassel, D.; Joumard, R.; Samaras, Z.; Sorenson, S. MEET Methodology for Calculating Transport Emissions and Energy Consumption; Technical Report DG VII; Rue de la Loi 200, 1049 Brussels, Belgium: European Commission: Brussels, Belgium, 1999.

29. Alininghian, M.; Naderipour, M. A novel comprehensive macroscopic model for time-dependent vehicle routing problem with multi-alternative graph to reduce fuel consumption: A case study. *Comput. Ind. Eng.* **2016**, *99*, 210–222.

30. Cao, E.B.; Lai, M.Y. A hybrid differential evolution algorithm to vehicle routing problem with fuzzy demands. *J. Comput. Appl. Math.* **2009**, *231*, 302–310.
31. Yang, W.; Mathur, K.; Ballou, R.H. Stochastic vehicle routing problem with restocking. *Transp. Sci.* 2000, 34, 99–112.
32. Xie, B.L.; An, S.; Guo, Y.H. Multi-tour optimization policy for stochastic vehicle routing problem. *Syst. Eng. Theory Prac.* 2007, 27, 167–171.
33. Zhao, Y.; Li, C.; Zhang, J.; Lu, Y.; Wang, W. Novel algorithm for multi-objective vehicle routing problem with stochastic demand. *Comput. Integr. Manuf. Syst.* 2012, 18, 523–530.
34. Fan, H.M.; Liu, P.C.; Liu, H.; Hou, D.K. The multi-depot vehicle routing problem with simultaneous deterministic delivery and stochastic pickup based on joint distribution. *Acta Auto Sinica* 2021, 47, 1–15.
35. Yang, J.; Ma, L. Wasp Colony Algorithm for Vehicle Routing Problem with Soft Time Windows. *Appl. Res. Comput.* 2010, 29, 67–70+61.
36. Augerat, P.; Belenguer, J.M.; Benavent, E.; Corberan, A.; Rinaldi, G. *Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem;* Rapport de Recherche-IMAG; Universite Joseph Fourier: Grenoble, France, 1998; 495.