Research on Logistics Vehicle Routing Problem Based on Firework Algorithm

Shizheng Liu[1], Xinnan Ji[1], Xuan Chen[2]

[1]*College of Business and Commerce, Zhejiang Industry Polytechnic College, Shaoxing, China*

[2]*College of Design and Art Branch, Zhejiang Industry Polytechnic College, Shaoxing, China*

Abstract

Aiming at the shortcomings of long time consumption and high transportation cost caused by the lack of reasonable route planning in the distribution of logistics vehicles, a strategy for optimizing logistics vehicle paths using the Improved FireWorks Algorithm (IWFA) is proposed. First, the logistics distribution vehicle model based on time windows is described. Second, On the basis of the firework algorithm, the population initialization based on Bernouilli Shift chaos is used, and the non-core firework radius is optimized by the threshold to prevent individuals from falling into the local optimum. After each iteration, the locust algorithm is used to screen individuals to improve the quality of the solution. In the simulation experiment, compared with the FWA, PSO, and ACO algorithms, the algorithm in this paper has a better advantage in the performance comparison of the benchmark test function. In the comparison of the simulation logistics vehicle routing problem, the algorithm in this paper is in the unit completion time and unit transportation cost. The above has a good advantage.

Keywords: *Vehicle Routing Problem, Fire Works algorithm, time window*

I. Introduction

With the continuous development of E-commerce, the number of merchants and consumers using E-commerce platforms for commodity transactions has gradually increased, indirectly speeding up the development of the logistics industry. According to data [1], compared with developed countries, China's logistics industry has relatively low distribution efficiency and correspondingly higher distribution costs. The logistics cost of the whole society can even reach about 20% of domestic GDP. Therefore, reducing costs is the logistics industry important research direction. The Vehicle Routing Problem (VRP) [2] is a core component of logistics distribution, and is a key problem that continues to focus on computer, operations research and other disciplines. Its content is: multiple customer sites and centers in logistics distribution at the site, each customer has different needs for goods. There are multiple vehicles with a limited load capacity in the central site. These vehicles provide goods delivery services according to the corresponding customers who put forward their needs. We need to know the specific needs of each customer at the same time, under the condition of ensuring certain constraints, a reasonable vehicle travel route is planned, so VRP is a typical NP problem. The use of bionics algorithm to solve the NP problem is the best solution so far. Therefore, based on the existing VRP research results, this paper uses the optimization model of vehicle scheduling based on time window, and solves the model through the firework algorithm.

VRP has always been a hot and difficult point of research at home and abroad. So far, many VRP solutions have been proposed. This article is divided into three types according to the accuracy of the solution: traditional algorithms, classic heuristic algorithms and modern heuristics algorithm.

Traditional algorithms refer to algorithms that use their own mathematical knowledge and planning techniques to find the optimal solution, mainly mathematical evaluation methods in operations research, such as: Branch and Bound method, Dynamic Programming Approach, Cutting Planes Approach, Network Flow Approach Wait. Although these methods can derive the optimal path for the VRP problem, the amount of calculation will increase exponentially with the increase of the actual problem scale, and they are only suitable for small-scale or specific distribution problems. Branch and Bound method [3] is a search algorithm proposed by Laporte in 1986 for solving

ISSN: 0010-8189

integer programming problems. Dynamic Programming Approach [4] is an accurate algorithm proposed by R Bellman et al. in the early 1950s that can solve overlapping sub-problems, but the algorithm runs for a long time. Cutting Planes Approach [5] is a method proposed in 1958 that continuously introduces new linear constraints and continuously cuts the solved non-integer solutions to reduce the feasible region of the problem to achieve the best integer programming problem. Network Flow Approach [6] is an operation research optimization problem solving algorithm proposed by LR Ford in 1956. Its basic idea is to simplify distribution centers, customers, and transportation routes into points and lines in a directed graph to construct a the network model of the vehicle routing problem. In general, traditional algorithms use a strict mathematical method to obtain solutions that are better than swarm intelligence algorithms, but the solution time is longer, and the time will exponentially increase during operation, so accurate algorithms are not suitable for large-scale The VRP problem is only applicable to small-scale combinatorial optimization problems.

Modern heuristic algorithm is a swarm intelligence algorithm derived from the in-depth research in many fields such as natural biology and physics. The simulation results show that it has strong search competitiveness in solving complex combinatorial optimization problems, and more importantly, search the results are of high quality and provide more possibilities for the research of VRP issues. Such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Whale Optimization Algorithm (WOA), Grasshopper Optimization Algorithm (GOA) and fusion algorithms. Literature [7] proposes the use of genetic algorithm to solve the VRP problem. Experiments show that this algorithm has certain advantages over traditional methods; Literature [8] proposes to use particle swarm algorithm to solve the Capacitated Vehicle Routing Problem (CVRP) model. Experiments show that the minimum vehicle distance and maximum capacity provide services to users; Literature [9] uses Ant colony optimization to solve the vehicle routing problem. Simulation experiments show that this algorithm can save time and reduce transportation costs; Literature [10] proposed a scheme of using a hybrid whale algorithm for vehicle routing problem, which optimizes the transportation cost in a minimized way. Simulation experiments show that the algorithm has a good effect on the transportation cost; Literature [11] proposed a solution to the vehicle routing problem using the locust algorithm, which focuses on optimizing the number of vehicles and the shortest path as the target, and the locust algorithm is used to solve the problem. Simulation experiments show that this method has good results

II. Optimization of Logistics Delivery Vehicle Path Based on Time Window

This paper conducts research on the basis of the vehicle path description in Literature [12], setting the logistics distribution route optimization problem with time window can be described as: A distribution center has to complete the distribution tasks of N customers. The location and needs of the customers are determined. There are V vehicles that can participate in the distribution, and each vehicle has the same weight; each vehicle only needs to complete one route, starting with logistics distribution center. Complete the customer point service of the sub-line in order and finally return to the distribution center. Each customer puts forward restrictions on the time period of delivery, and the upper and lower bounds of the time window stipulate the earliest and latest service time of the customer point. When the service time deviates from the time window, the distribution center has to pay a certain penalty cost for this. In order to fully reflect the service quality and timeliness of logistics distribution and improve the customer experience of distribution, the model used in this section uses a hybrid time window to describe customer time constraints. But it has the following premise: there is only one distribution center, and each vehicle starts from the distribution center, and after completing the distribution task, it will eventually return to the starting point; the needs of each customer are determined and cannot exceed the carrying capacity of the vehicle; the carrying capacity of each vehicle is the same and known. The location of all customers is fixed and known; each customer can only be served by one car and the service must be completed at one time; all customer points must complete the service; each vehicle only completes one route. The symbolic expressions needed to build the model in this paper are as follows: c_{ij} refers to the unit transport cost for vehicle from customer i to customer j, q_i refers to the demand of customer i, d_{ij} refers to the distance between customer i and customer j, t_{ij} refers to

ISSN: 0010-8189

the transport time from customer i to customer j, x_{ijk} refers to the incident for vehicle k to travel from customer i to customer j, y_{ik} refers to that the transport task of i requires k vehicles, t_e refers to the earliest time prior to the time window, e_i refers to the earliest service time of customer i; l_i refers to the earliest service time of customer i; a_{ik} refers to the time for vehicle k to arrive at customer i, θ refers to the penalty coefficient of time cost, and $p_i(a_{iv})$ refers to time penalty coefficient. Therefore, the objective function established in this paper is as follows:

$$\min Z = \sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{v=1}^{v} C_{ij} d_{ij} x_{ijv} + \sum_{i=0}^{N} p_i(s_i) \tag{1}$$

Constraints:

$$\sum_{i=1}^{N} q_i y_{iv} \le Q, \left(v = 1, 2, \ldots, V\right) \tag{2}$$

$$\sum_{v=1}^{V} y_{ok} = V, \left(v = 1, 2, \ldots, V\right) \tag{3}$$

$$\sum_{v=1}^{V} y_{iv} = 1, \left(v = 1, 2, \ldots, N\right) \tag{4}$$

$$\sum_{v=1}^{V} y_{iov} = 1, \left(v = 1, 2, \ldots, V\right) \tag{5}$$

$$\sum_{i=0}^{N} x_{ijv} = y_{iv} \left(v = 1, 2, \ldots, V; i = 1, 2, \ldots, N\right) \tag{6}$$

$$\sum_{i=0}^{N} x_{ijv} = y_{jv} \left(v = 1, 2, \ldots, V; j = 1, 2, \ldots, N\right) \tag{7}$$

$$P_i(a_{iv}) = \begin{cases} \theta(e_i - a_{iv}), t_e \le a_{iv} < e_i \\ 0, e_i \le a_{iv} < l_i \\ \infty, else \end{cases} \tag{8}$$

$$x_{xijv} = \begin{cases} 1, \text{vehicle v from i to j} \\ 0, else \end{cases} \tag{9}$$

$$y_{iv} = \begin{cases} 1, \text{vehicle v from i to j} \\ 0, else \end{cases} \tag{10}$$

The objective function (1) indicates that the overall goal of optimization is to minimize the cost, including the

ISSN: 0010-8189

transportation cost of the driving path and the time cost penalty caused by the deviation from the time window; Constraint (2) ensures that the total demand allocated to each vehicle trip does not exceed the vehicle load Q; Constraint (3) ensures that all customers are served by V vehicles; Constraint (4) ensures that each customer is served by only one vehicle; Constraint (5) indicates that the distribution center is the starting and ending point of the vehicle journey; Constraints (6) and (7) indicate the relationship between x_{ijv} and y_{iv}; Constraint (8) is the penalty cost function that deviates from the time window; Constraints (9) and (10) indicate the value of incident x_{ijv} and y_{iv}.

III. Fire Works Algorithm

Tan and Zhu proposed Fire Works algorithm (FWA) [13] in 2010 based on the explosive form of Fire Works. It has the characteristics of randomness, locality and diversity. It is mainly used for problem optimization. Each Fire Works in FWA is regarded as a feasible solution in the optimization problem. According to the fitness function, the number of sparks generated by each Fire Works is calculated. The better the fitness value, the greater the number of sparks corresponding to Fire Works, and vice versa. And the mutation process in the explosion process of Fire Works guarantees the diversity of the population. FWA mainly consists of three parts: explosion operator, mutation operator and selection strategy.

3.1 Explosion operator

In the initialization of the algorithm, a preliminary evaluation of the fitness value corresponding to the location of the generated Fire Works is required. Fire Works (generally called the core Fire Works) with better fitness value can obtain more resources, and can generate more Fire Works in a small area and has a strong local search ability; Fire Works with poor fitness value (generally called non-core Fire Works) can only get less sparks, but has a certain global search capability. Therefore, the explosion radius of each Fire Works and the number of sparks produced are calculated according to the fitness values of other Fire Works in the population. The calculation of the explosion radius A_i and the number of explosion fireworks S_i is as follows:

$$A_i = \widehat{A} * \frac{f(x_i) - Y_{\min} + \varepsilon}{\sum_{i=1}^{N} (f(x_i) - Y_{\min}) + \varepsilon} \tag{11}$$

$$S_i = M \times \frac{y_{\max} - f(x_i) + \varepsilon}{\sum_{i=1}^{N} (y_{\max} - f(x_i)) + \varepsilon} \tag{12}$$

In the formula, $y_{\min} = \min(f(x_i))$ and $y_{\max} = \max(f(x_i))$ are the minimum and maximum fitness values in the current population. $\widehat{A}$ is used to adjust the explosion radius, M is a constant number used to set the size of fireworks, and ε is the minimum parameters of a machine used to avoid zero operation. In order to reduce the number of explosion sparks produced by Fire Works with good position values, the number of sparks needs to be limited:

$$\widehat{S}_i = \begin{cases} round(a * M), S_i < aM \\ round(b * M), S_i > bM, a < b < 1 \\ round(S_i), otherwise \end{cases} \tag{13}$$

In the formula, a,b are constant numbers, and $round(\bullet)$ is a rounding function.

3.2 Mutation operator

In order to increase the diversity of the population, mutation operators are used to generate mutation sparks. This kind of mutation spark is Gaussian mutation spark, which is mainly to randomly select a Fire Works individual in the population, and perform Gaussian mutation operation in a certain dimension to get the formula shown in (14).

$$\widehat{x}_{ik} = x_{ik} \times e \tag{14}$$

In the formula, e is the Gaussian mutation operation of $N(1,1)$.

3.3 Selection strategies

In order to be able to select effective individuals in the current population and pass them to the next generation, after experiencing the above two operations, the algorithm selects a certain number of individuals as the next generation of Fire Works. The probability of selection is:

$$p(x_i) = \frac{R(x_i)}{\sum_{x_j \in K} x_j} \tag{15}$$

$$R(x_i) = \sum_{x_j \in K} d(x_i - x_j) = \sum_{x_j \in K} \| x_i - x_j \| \tag{16}$$

In the formula, $R(x_i)$ is the sum of the distances of all individuals in the current set of individual candidates except x_i. After the above method of operation, if the density of the individual is high, the probability of being selected around the individual will decrease.

IV. Vehicle Routing Based on Improved Fire Works Algorithm

4.1 Population initialization

The diversity of the initial population will greatly affect the convergence speed and accuracy of the swarm intelligence algorithm, while the bird swarm algorithm using random initialization methods cannot guarantee the diversity of the population. Chaos mapping has the characteristics of randomness, ergodicity and regularity. It has been widely used in the optimization of intelligent algorithms and has achieved good results. In order to make better use of the solution space, this paper introduces the improved Bernouilli with the best chaotic effect. Shift mapping for population initialization

$$x_{n+1} = (x_n + rand(0,1)) \bmod 2 \tag{17}$$

The specific steps of using the improved Bernouilli Shift chaos to generate the initial population are as follows:
Set the population size N, the dimension D and the maximum number of chaotic iteration steps K

ISSN: 0010-8189

$$
\begin{aligned}
&for\, i = 1\, to\, N\, do \\
&\quad for\;\; j = 1\, to\, D\, do \\
&\quad\quad for\; k = 1 to\; K\;\, do \\
&\quad\quad x_{k,j} = (x_{k-1,j} + rand(0,1)) \bmod 2 \\
&\quad\quad endfor \\
&\quad x_{i,j} = x_{\min,j} + x_{k,j} \times (x_{\max,j} - x_{\min,j}) \\
&\quad endfor \\
&endfor
\end{aligned}
$$

In the formula, $x_{k-1,j}$ represents the j-dimensional individual in the $k-1$-th iteration, $x_{k,j}$ represents the j-dimensional individual in the k-th iteration after the Bernouilli Shif chaotic map is used, and $x_{i,j}$ represents the i-th individual in the j-th dimension, and $x_{\max,j}$ and $x_{\min,j}$ are respectively Represents the upper and lower bounds in the j-dimensional space.

4.2 Threshold optimization of non-core fire works radius

According to the FWA algorithm, when Fire Works explodes, it will generate a lot of sparks in a certain area around it. The range of the spark is the radius of Fire Works, the core Fire Works is the Fire Works individual with the smallest fitness value in the Fire Works population, and the non-core Fire Works radius is shown in formula (18). Obviously, this method will find that the radius of Fire Works with the smallest fitness value is very small, almost close to 0, thus wasting resources. In order to avoid this situation, the minimum radius threshold is set, and its manifestation is as follows:

$$
A_{i,k} = \begin{cases} A_{\min,k}, if\;\, A_{i,k} < A_{\min,k} \\ A_{i,k}, otherwise \end{cases} \tag{18}
$$

The threshold value $A_{\min,k}$ is not fixed because the algorithm will produce dynamic effects in the iterative process, so it adopts a non-linear decrement method.

$$
A_{j,k}(t) = ((A_{start} + A_{end}) / 2 - d_{\max}) \times e^{\frac{t}{d_{\max}}} \tag{19}
$$

In the formula, $d_{\max}$ is the maximum number of function evaluations, t is current function evaluation times, A_{start} and A_{end} are the initial explosion radius and final radius value.

4.3 Selection strategy based on locust algorithm

In the FWA algorithm, excellent Fire Works individuals can be passed to the next generation population during each iteration. The FWA algorithm mainly selects the individual with the smallest fitness value, and the remaining individuals adopt a random state. Obviously, this selection strategy has certain drawbacks, although in a certain iteration, individuals are selected according to the small fitness value. However, from a global perspective, it may not be possible to ensure that the selected individual is better than the remaining random state individual. At the same time, the remaining individuals adopt the random state, which is not conducive to the birth of the algorithm's global optimal solution as a whole. In response to this situation, this article introduces the Grasshopper optimization algorithm (GOA), which is divided into two parts: exploration and development. The exploration part corresponds to the larval stage of the locust, and the development part corresponds to the adult stage of the locust. In the larval stage, the behavior of the locust swarm jumping movement mainly performs a global search; in the adult stage, the locust moves in a small area, which is conducive to local search. When a locust population reproduces, forages and migrates, its individual position will be affected by the interaction force of the population, gravity and wind. Therefore, the mathematical model is used to describe its position update behavior as follows:

$$X_i = r_1 S_i + r_2 G_i + r_3 A_i \tag{20}$$

In the formula, X_i refers to the position of the i locust, S_i refers to the influence of the i locust from other locusts, G_i is the influence of the i locust from gravity, A_i is the influence of the i locust from wind, and r_1, r_2, r_3 refer to random numbers in [0,1], in which

$$S_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} s(d_{ij})\widehat{d}_{ij} \tag{21}$$

In the formula, $d_{ij} = | x_j - x_i |$ refers to the distance between locusts i, j . $\widehat{d}_{ij} = \dfrac{x_j - x_i}{d_{ij}}$ refers to the unit vector from locust i to locust j , s refers to the influencing function of locust from other locusts. The expression is as follows:

$$s(r) = f e^{\frac{-r}{l}} - e^{-r} \tag{22}$$

In formula (22), when $s(r)$ is greater than 0, the locusts attract each other. Therefore, the value of r is called attraction domain. When $s(r)$ is smaller than 0, the locusts repel each other. Therefore, the value of r is called exclusion domain; when $s(r)$ is 0, the locusts are neither attracted nor repelled, so r is a comfortable distance. In addition, f and l are attraction strength parameter and scale parameter. Their values affect the attractive domain, repulsive domain and moderate distribution distance. Generally, the value of l is 1.5, and the value of f is 0.5.

$$G_r = -g\widehat{e}_g \tag{23}$$

$$A_i = u\widehat{e}_w \tag{24}$$

In formula (23), g is the gravitational constant, $\widehat{e}_g$ is unit vector pointing to the center of the earth; in formula (24), u refers to wind direction constant, $\widehat{e}_w$ refers to wind direction unit vector. Therefore, the location of individual locusts is updated as follows:

$$X_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} s(| x_j - x_i |)\frac{x_j - x_i}{d_{ij}} - g\widehat{e}_g + u\widehat{e}_w \tag{25}$$

Although formula (25) is used to simulate the locust population, it is considered from the perspective of practical

ISSN: 0010-8189

application. Gravity factors are usually not considered, and the wind direction is determined to point to the target position. Therefore, finding the best individual locust position is the optimization solution as shown in formula (26).

$$X_i^d = c\left\{\sum_{\substack{j=1\\j\neq i}}^{N} c\frac{ub_d - lb_d}{2} s(|x_j^d - x_i^d|)\frac{x_j - x_i}{d_{ij}}\right\} + \hat{T}_d \tag{26}$$

$$c = c_{\max} - t\times\frac{c_{\max} - c_{\min}}{T_{\max}} \tag{27}$$

In formula (26), ub_d and lb_d correspond to the upper bound and lower bound of locust i at the d -dimensional variable, and $\hat{T}_d$ is the target position of locust group. In formula (27), c is the decrease factor. On the one hand, it is used to balance the global search and local development capabilities, and on the other hand it is used in the repelling domain and the attractive domain. t is the current iteration times, $c_{\max}$ and $c_{\min}$ are the maximum value and minimum value.

4.4 Algorithm steps

Step 1: Initialize the path optimization model, vehicle capacity (vehicles, load, speed), algorithm parameters (population size, etc.);

Step 2: Obtain customer data (coordinates, needs, time windows);

Step 3: Set the objective function $\min Z = \sum_{i=0}^{N}\sum_{j=0}^{N}\sum_{v=1}^{v} C_{ij} d_{ij} x_{ijv} + \sum_{i=0}^{N} p_i(s_i)$

Step 4: Call the optimized FireWorks algorithm to solve the problem;

Step 5: Terminate the iteration and output the path optimization results.

V. Simulation Experimental

The experiment is divided into two parts. The first part verifies the performance of the algorithm, and the second part verifies the effect of this algorithm in path planning. The hardware environment selects the CPU as a Core i5 processor, the memory is 4GDDR3, the hard disk capacity is 1000G, the software environment is Windows7, and the simulation software is Matlab2012.

5.1 Algorithm performance comparison

This paper chooses four benchmark test functions as shown in Table 1 for comparison. The algorithm in this paper is compared with the FWA algorithm, and each algorithm is run 50 times under the condition of 10, 50, and 100 dimensions. The results of comparing the optimal value and variance are shown in Table 2 to Table 5. It is found from the table that the algorithm in this paper has certain advantages compared to the FWA algorithm in terms of the optimal value and variance results in different dimensions. This shows that compared with the FWA algorithm, the algorithm in this paper has undergone population initialization, threshold optimization and individual screening after that, the performance of the algorithm has been significantly improved, laying a good foundation for subsequent vehicle path planning

Table 1 Benchmark functions

Function name	Range
$\min f_1(x)=\sum_{1}^{n} x^2$	[-100,100]
$\min f_2(x)=\sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	[-100,100]
$\min f_3(x)=\sum_{i=1}^{n} ([x_i+0.5])^2$	[-100,100]
$\min f_4(x)=\max(abs(x_i))$	[-100,100]

Table 2 Optimization f_1 results of two algorithm pairs

Dimension	Algorithm	Optimal value	Variance
10	FWA	2.983e-11	4.913e-12
	IFWA	1.431-12	2.732e-14
50	FWA	4.327e-13	5.127e-12
	IFWA	3.147e-14	8.142e-13
100	FWA	4.713e-12	4.191e-12
	IFWA	1.212e-15	1.421e-16

Table 3 Optimization f_2 results of two algorithm pairs

Dimension	Algorithm	Optimal value	Variance
10	FWA	3.713e-12	1.183e-11
	IFWA	6.134-15	5.631-12
50	FWA	7.516e-17	4.116e-11
	IFWA	3.231e-19	3.148e-13
100	FWA	4.163e-16	3.713e-12
	IFWA	7.314e-20	1.184e-14

Table 4 Optimization f_3 results of two algorithm pairs

Dimension	Algorithm	Optimal value	Variance
10	FWA	2.983e-17	1.713e-15
	IFWA	1.431-19	1.621e-16
50	FWA	2.316e-20	3.421e-17
	IFWA	2.137e-21	5.218e-19
100	FWA	3.663e-26	7.135e-22

	IFWA	6.913e-28	5.294e-25

Table 5 Optimization f_4 results of two algorithm pairs

Dimension	Algorithm	Optimal value	Variance
10	FWA	8.183e-13	9.183e-14
	IFWA	9.431e-14	2.831e-16
50	FWA	6.116e-10	3.426e-11
	IFWA	7.231e-12	8.217e-13
100	FWA	3.663e-20	6.163e-11
	IFWA	2.724e-22	8.117e-16

5.2 Comparison of vehicle logistics path effects

Taking 10 logistics distribution points in a certain city as the research object, simulate them on a two-dimensional plane. The simulation coordinates of the 10 logistics distribution points are shown in Table 6, and the coordinates of the center point are (4, 5), in order to Highlight the effect of the algorithm in this paper in path planning, set the vehicle speed between two logistics points to be constant, the size of the time window is 1 second, the number of vehicles is 3, and they run different routes, as shown in Table 7. The unit transportation cost is 5 yuan, and the unit cost of customer satisfaction penalty due to deviation from the time window is 50 yuan. The comparison algorithms are FWA and PSO algorithms respectively.

Table 6 Simulated coordinates of 10 points

No.	Coordinate	No.	Coordinate
1	(1,1)	6	(5,3)
2	(2,6)	7	(4,9)
3	(4.2)	8	(7,7)
4	(7,4)	9	(3,2)
5	(9,1)	10	(9,6)

Table 7 Delivery routes and vehicles

Route	Delivery order
Route 1	center point→1→4→8→9→2→3→10
Route 2	center point→5→7→6→2
Route 3	center point→4→9→3

Table 8 shows the results of the unit delivery time of the three algorithms on the 3 routes. From the overall effect of the three routes, the IFWA algorithm has a better advantage than the FWA and PSO algorithms. Route 1 is a relatively complicated route. The vehicle path needs to pass through 6 points in total. Compared with the FWA algorithm, the IFWA algorithm saves about 23% in delivery time and about 52.9% compared to the PSO algorithm. Although there are only 4 delivery points in route 2, but the distance between them is relatively long. Compared with the FWA algorithm, the IFWA algorithm saves about 26.7% of delivery time and about 67.7% compared to the PSO algorithm. There are only 3 points in route 3. Therefore, the overall time consumed by the three algorithms is at the smallest, the IFWA algorithm saves about 38% of the delivery time compared to the FWA algorithm, and about 70% compared to the PSO algorithm. This shows that the unit delivery time of the IFWA algorithm on three different routes is significantly better than the FWA algorithm and the PSO algorithm. This shows that the performance of the algorithm has been improved after population initialization, threshold optimization and individual screening, so the unit delivery time is the least.

Table 8 Comparison of delivery time

Algorithm	Route	Delivery Time (S)

ISSN: 0010-8189

IFWA	1	1.7
	2	1.5
	3	1.3
FWA	1	2.1
	2	1.9
	3	1.8
PSO	1	2.7
	2	2.5
	3	2.3

According to the IFWA algorithm, the cost result of optimizing the distribution route is that the unit length of route 1 is 77.16, the unit length of route 2 is 15.45, and the distance of route 3 is 12.81. The total unit length of this distribution task is 105.42, and the transportation cost is 527.1 yuan. Therefore, the total distribution cost is 577.1 yuan. According to the FWA algorithm to optimize the cost of the distribution route, the result is that the unit length of route 1 is 80.23, the unit length of route 2 is 18.75, and the unit length of route 3 is 14.27. The total unit length of this delivery task is 113.25, so the total delivery cost is 616.25 yuan. The cost result of the PSO algorithm to optimize the distribution route is that the unit length of route 1 is 87.12, the unit length of route 2 is 25.15, the unit length of route 3 is 19.32, and the total unit length of this delivery task is 131.6, so the total delivery cost is It is 658 yuan, which can verify that the vehicle routing proposed by the IFWA algorithm in this paper has obvious advantages in terms of distribution costs.

Table 9 shows the comparison of user satisfaction at 10 distribution points. A total of 50 users are selected for statistical surveys. From the table, it is found that the use of IFWA algorithm for vehicle logistics distribution can be recognized by more users, followed by the adoption of FWA algorithm and PSO algorithm. The reason is that the IFWA algorithm can effectively save time and reduce user waiting time in long-path distribution

Table 9. User satisfaction of different routes

Algorithm	Route	Satisfaction (%)
IFWA	1	87
	2	84
	3	85
FWA	1	79
	2	78
	3	80
PSO	1	70
	2	71
	3	72

VI. Conclusion

Logistics vehicle path planning has always been the focus of research in the logistics industry. When the vehicle path planning is unreasonable, it will cause excess time and transportation costs. This paper proposes an improved firework algorithm for vehicle path planning. It describes a logistics distribution vehicle model based on time windows. Based on the firework algorithm, it uses chaotic reverse-Cauchy distribution, through threshold optimization, and uses the locust algorithm. Screening and other measures have improved the performance of the firework algorithm. The algorithm is used in simulation experiments to optimize the three distribution routes in a city, saving time and transportation costs. The next step will be to study the research of vehicle scheduling under complex backgrounds.

References

[1] Statistics Bureau of the People's Republic of China. China Statistical Yearbook. Beijing: China Statistics Press, 2018.

[2] P. Toth and D. Vigo. The vehicle routing problem. Society for Industrial and Applied Mathematcs, 2002.

[3] Berger, J.M. Bourjolly, G. Laporte. Branch-and-bound algorithms for the multi-product assembly line balancing problem. European Journal of Operational Research, 1992, 58(2): 215-222.

[4] R. Bellman. Dynamic programming and modern control theory. Proc. internat. congr. math, 1965, 37(1): 65-82.

[5] R.E. Gomory. Outline of an algorithm for integer solutions to linear programs. Bulletin of the American Mathematical Society, 1958, 64(5): 275-278.

[6] L.R. Ford, D.R. Fulkerson. A suggested computation for maximal multi-commodity network flows. Management Science, 1958, 5(1): 97-101.

[7] B.M. Baker, M.A. Ayechew. A genetic algorithm for the vehicle routing problem. Computers & Operations Research, 2003, 30(5): 787-800.

[8] S.R. Venkatesan, D. Logendran, D. Chandramohan. Optimization of capacitated vehicle routing problem using PSO. International Journal of Engineering Science and Technology (IJEST), 2011, 3(10): 7469-7477.

[9] J.E. Bell, P.R. McMullen. Ant colony optimization techniques for the vehicle routing problem. Advanced engineering informatics, 2004, 18(1): 41-48.

[10] S.K. Dewi, D.M. Utama. A New Hybrid Whale Optimization Algorithm for Green Vehicle Routing Problem. Systems Science & Control Engineering, 2021, 9(1): 61-72.

[11] V. Soto-Mendoza, I. García-Calvillo, E. Ruiz-y-Ruiz, et al. A Hybrid Grasshopper Optimization Algorithm Applied to the Open Vehicle Routing Problem. Algorithms, 2020, 13(4): 96-115.

[12] Z.H. Fu and C.S. Liu. Research on the multi-depot vehicle routing problem based on joint distribution mode. Computer Engineering and Applications, 2021, 1-12.

[13] Y. Tan, Y. Zhu. Fireworks algorithm for optimization International conference in swarm intelligence. Springer, Berlin, Heidelberg, 2010: 355-364.

ISSN: 0010-8189