

# VRP Problem Solving Based on Adaptive Dynamic Search Ant Colony Algorithm

Li Zheng<sup>1,\*</sup>, Zhiming He<sup>1</sup> and Wen Liang<sup>1</sup>

<sup>1</sup>College of information engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi, 341000, China

\*E-mail: zfighting517@163.com

**Abstract.** Based on ant colony algorithm to solve the defect analysis of VRP problem, an adaptive dynamic search ant colony algorithm (ADACO) is proposed. Firstly, the model is established and the combination parameters is experimentally configured. Secondly, the strategy of combining pseudo-random and adaptive transition probability are used to help the group choose a higher quality path. When the group is in a local predicament, the segmented setting of the pheromone intensity induces the group to break out of the predicament in time. Finally, multiple groups of experimental tests are performed on the "Jia-hui Fresh" cargo delivery case. The results show that, compared with the original algorithm, the ADACO algorithm has respectively improved 17.65%, 16.13% and 16.10% in terms of delivery cost, convergence algebra and CPU running time.

## 1. Introduction

Vehicle routing problem (VRP) refers to a group of vehicles can traverse a series of specific points that depart from a specified location [1]. The traditional methods to solve the problem often fail to achieve the desired results, such as precise algorithms [2] and heuristics [3]. Therefore, scholars combine AI with heuristic algorithms, such as simulated annealing (SA) [4], tabu search (TS) [5], genetic algorithm (GA) [6] and ant colony algorithm (ACO) [7].

Compared with other algorithms, ACO has a unique advantage in path finding. However, when the scale of the problem is large, the algorithm is liable to fall into a local dilemma, the search space cannot be further explored and developed, so that additional time overhead is also added [8,9]. As a result, scholars have made different levels of improvements. For example, Zhang Qiang et al. [10] combined artificial immune with ACO to solve the "last mile" problem of emergency food delivery. Zhang Wenbo et al. [11] eliminated the restrictions of the taboo table, and modified the final travel of the individual to achieve partial point traversal of the connected graph. Meanwhile, a temporary weight matrix was introduced to avoid repeatedly selecting paths with small weights. Thus, the path planning problem of multi-scenic spots is solved. Duan Ping et al. [12] modified the update rules of pheromone and evaporation coefficient, and adopted the boundary symmetric mutation strategy to improve the mutation performance. Although it improved the efficiency and quality of the solution.

The above improvement method mainly change part of the update rules or mix other algorithms. However, the number of pheromone, path heuristic function, product of pheromone number-heuristic function and cooperative behavior of search individuals are closely related in ACO, which seriously affects the convergence of the algorithm. Based on the above research and analysis, this paper proposes an adaptive dynamic search ant colony algorithm (ADACO). Firstly, the key parameters of



the combination is experimentally set, and the adaptive pseudo-random selection strategy is used to help the group reasonably choose the next transfer direction. In addition, the segmented setting of pheromone intensity can inspire the group to jump out of the dilemma in time. Finally, the algorithm before and after optimization are used to carry out multiple experiments on the case of cargo delivery. Multi-angle analysis of experimental results prove that ADACO is more efficient in solving the VRP problem, which greatly reduces the time and delivery cost.

## 2. VRP Model

### 2.1. Model assumptions

(1) The delivery vehicles are of the same specification and there is no slight error.

(2) Do not consider urban traffic jams.

(3) The delivery vehicles always run at a constant speed and the delivery cost within a unit distance is equivalent. Therefore, the driving distance can represent the delivery cost.

### 2.2. Symbol description

The relevant symbol descriptions included in the model are shown in Table 1.

**Table 1.** Explanation of related symbols.

Symbol	Description
$n$	Number of demand points, use $i$ and $j$ to represent different demand
$V$	A collection of delivery centers and various demand points
$S$	Any set of demand points
$m$	The total number of vehicles, where $k$ represents the vehicle number
$c_{ij}$	The delivery cost from demand point $i$ to demand point $j$
$L$	Maximum range of vehicle
$W$	The vehicle rated load limit
$q_i$	The quantity demanded at point $i$

### 2.3. Objective optimization function setting

The objective optimization function is defined as

$$\min Z = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ij} x_{ijk} \quad (1)$$

Other related constraints are defined as follows:

$$\sum_{i=1}^n q_i y_{ik} \leq W, \quad k = 1, 2, \dots, m \quad (2)$$

$$\sum_{i=0}^n \sum_{j=0}^n c_{ij} x_{ijk} \leq L, \quad k = 1, 2, \dots, m \quad (3)$$

$$\sum_{k=1}^m y_{ik} = 1, \quad i = 1, 2, \dots, n \quad (4)$$

$$\sum_{k=1}^m y_{ok} = m \quad (5)$$

$$\sum_{i=1}^n x_{ijk} = y_{jk}, \quad j = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (6)$$

$$\sum_{j=1}^n x_{ijk} = y_{jk}, \quad i = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (7)$$

$$x_{ijk} (x_{ijk} - 1) = 0, \quad i = 1, 2, \dots, n \quad j = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (8)$$

$$y_{ik}(y_{ik} - 1) = 0, \quad i=1,2,L,n \quad k=1,2,L,m \quad (9)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \geq 1, \quad \forall S \in V \quad k=1,2,L,m \quad (10)$$

Eq. (1) expresses the minimum total delivery cost. Eq. (2) indicates that the single delivery quantity of vehicle  $k$  does not exceed the rated load limit of vehicle. Eq. (3) indicates that the single driving distance of vehicle  $k$  is not more than the maximum limit distance. Eq. (4) indicates that the demand point  $i$  is only distributed by one vehicle. Eq. (5) indicates that a total of  $m$  vehicles participate in the delivery. Eq. (6) and Eq. (7) represents the relationship between two variables. Eq. (8) and Eq. (9) indicates that both variables satisfy the 0-1 constraint. Eq. (10) indicates that there is no loop phenomenon in the delivery.

### 3. ADACO Algorithm

#### 3.1. ADACO algorithm model

When solving large-scale VRP problems, the quality and efficiency of the solution are reduced due to the shortcomings of ACO. This paper analyzes its internal and external factors and makes some improvements based on the original algorithm framework.

##### (1) Improvement of transition probability $p_{ij}^k(t)$

This paper adopts the strategy of combining randomness with deterministic selection and adjusts the transition probability to be adaptive. In detail, the algorithm first generates a random number  $q \in [0,1]$  and determines the sizes of  $q$  and  $q_0$ . If  $q < q_0$ , choose the best edge according to Eq. (11), otherwise use the adaptive transition probability  $p_{ij}^k(t)$  in Eq. (12) to update

$$j = \begin{cases} \arg \max_{u \in J_k(i)} \{ [\tau_{iu}(t)]^\alpha [h_{iu}(t)]^\beta \}, & q < q_0 \\ \langle p_{ij}^k(t) \rangle, & \text{others} \end{cases} \quad (11)$$

In Eq. (11),  $q_0 = 1 - e^{-\frac{1}{nc}}$  ( $nc = 1, 2, \dots, \text{iterations}$ ),  $\text{iterations}$  is the maximum number of cycles.  $J_k(i)$  indicates the city that ant  $k$  is allowed to choose next.  $\tau_{ij}(t)$  represents the number of pheromones on the path  $(i, j)$  at time  $t$ .  $\eta_{ij} = 1/d_{ij}$  represents a heuristic factor.  $\alpha$  and  $\beta$  represent the weight ratio of pheromone and heuristic information respectively on the path.

$$\langle p_{ij}^k(t) \rangle = \begin{cases} \frac{[\tau_{ij}(t)]^\mu [\eta_{ij}(t)]^\eta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\mu [\eta_{iu}(t)]^\eta}, & j \in J_k(i) \\ 0, & \text{others} \end{cases} \quad (12)$$

In Eq. (12), as the number of iterations increase, the weight ratio between pheromone guidance  $\mu = \frac{4\alpha t}{\text{iterations}} + \alpha$  and heuristic factor  $\eta = \frac{2\beta t}{\text{iterations}} + \beta$  keep changing, realizing the self-adaptation of transfer probability.

When all ants have completed the traversal task, the update rule for the number of pheromones on each path is defined as

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (13)$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k \quad (14)$$

In Eq. (13),  $\rho$  ( $0 < \rho < 1$ ) is the pheromone evaporation ratio.  $\Delta\tau_{ij}^k$  represents the pheromone increase of ant  $k$  on the path  $(i, j)$ , which is usually defined by the Ant-Cycle model [9], as shown in Eq. (15).  $\Delta\tau_{ij}(t)$  represents the pheromone increase of  $o$  ants on the path  $(i, j)$ .

##### (2) Dynamic adjustment of pheromone intensity $Q_i$

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q_i}{L_k}, & \text{When ant } k \text{ passes the path } (i, j) \text{ in this iteration} \\ 0, & \text{others} \end{cases} \quad (15)$$

In Eq. (15), the setting of piecewise function  $Q_i$  is shown in Eq. (16).  $L_k$  represents the path length traveled by ant  $k$  in this iteration.

$$Q_i = \begin{cases} Q_1, & t < t_1 \\ Q_2, & t_1 \leq t < t_2 \\ Q_3, & t_2 \leq t \leq \text{iterations} \end{cases} \quad (16)$$

In Eq. (16), the segmentation of  $Q_i$  timely adjusted the pheromone intensity guidance on the path, which giving the algorithm the ability to jump out of local error.

### 3.2. Pseudo code of ADACO algorithm

According to the construction of the ADACO algorithm model in section 3.1, the pseudo code of the algorithm to solve the VRP problem is described as follows

---

#### Algorithm : ADACO algorithm for VRP problem

---

**Input:** Objective optimization function:  $\min Z$ ; Experimental parameters:  $\alpha, \beta, \rho, o, \text{iterations} \dots$ ; The

location information for warehouses and demand points; Demand at points.

**Output:** Global distribution plan and distribution costs.

nc = 1;

while nc <= iterations // Stop condition

  For  $k = 1 : o$  // Cycle to  $o$  ants

    While  $\tau_k = 0$

      For  $j = 1 : n$  // Traverse  $n$  points

        If  $\tau_k \not\subset j$  // tabu list  $\tau_k$  does not include point  $j$

          add point  $j$  to  $J_k(i)$  ;

        End

      End

      For  $j = 1 : J_k(i)$  //  $J_k(i)$  : list of demand points to be visited

        If vehicle remaining amount >= demand at the site to be visited

          select point  $j$  according to Eq. (11), calculate the loading distance, and update  $\tau_k$  and  $J_k(i)$  ;

        Else

          Invoke a new vehicle and increase  $m$  by 1, and continue traversing according to Eq. (11);

        End

      End

    End

      calculate the path length of the solution, which is the distribution cost;

  End

    Update the pheromone increment  $\Delta\tau_{ij}$  and global path pheromone  $\tau_{ij}$  on the path according to Eq. (11)

    to Eq. (16);

  nc = nc + 1;

  clear  $\tau_k$ ;

End

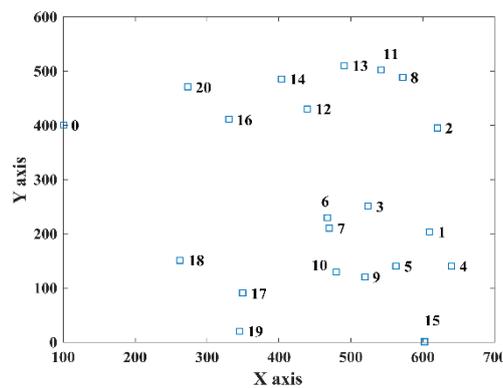
---

## 4. Experimental calculation and analysis

This paper takes the problem of goods delivery of "Jia-hui Fresh" in Zhengzhou as an experimental case. The warehouse supplies goods to various chain stores. In this case, the warehouse number is set to 0 and the position coordinates are (100, 400). The location coordinates and demand of the chain stores are shown in Table 2. The location coordinates are transformed into the XOY plane, as shown in Figure 1.

**Table 2.** Location coordinates and demand of chain stores.

Number	X axis	Y axis	Demand	Number	X axis	Y axis	Demand
1	610	203	3.67	11	542	502	3.29
2	620	395	3.58	12	440	430	3.18
3	524	250	3.15	13	491	510	3.36
4	640	140	3.46	14	404	485	3.54
5	563	140	3.77	15	602	0	3.46
6	467	230	2.89	16	330	410	3.38
7	470	210	3.08	17	350	90	3.56
8	572	488	3.24	18	263	151	3.18
9	519	120	3.43	19	345	20	3.49
10	480	130	3.62	20	273	470	3.67

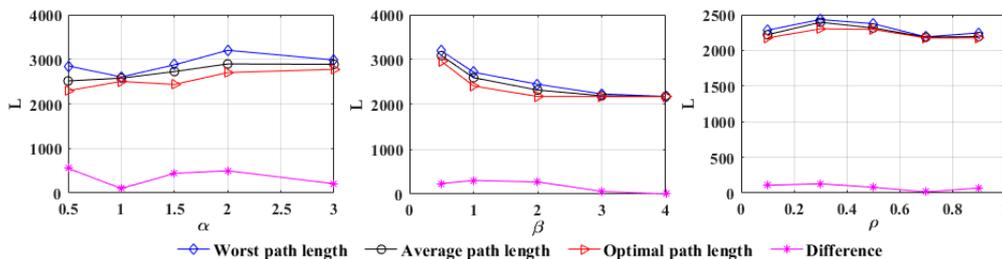


**Figure 1.** Distribution of warehouses and chain stores on XOY plane.

4.1. Parameter settings

In ACO, the combination configuration of  $\alpha$ ,  $\beta$  and  $\rho$  is the key to measure the algorithm performance and solution efficiency. On the TSP problem in this case, the combined parameters are experimentally set according to the rule that one parameter value is modified and the other two remain unchanged. The initial parameters are  $\alpha = 1$ ,  $\beta = 1$ ,  $\rho = 0.7$ , iterations = 100.

This experiment is divided into three groups, and each group is run for 10 times. The running results are shown in Figure 2. The worst, average, optimal path length and difference represent the maximum, average, minimum and the differences between maximum and minimum in the results of 10 runs. Therefore, the best combination parameter can be set to  $\alpha = 1$ ,  $\beta = 4$ ,  $\rho = 0.7$ .



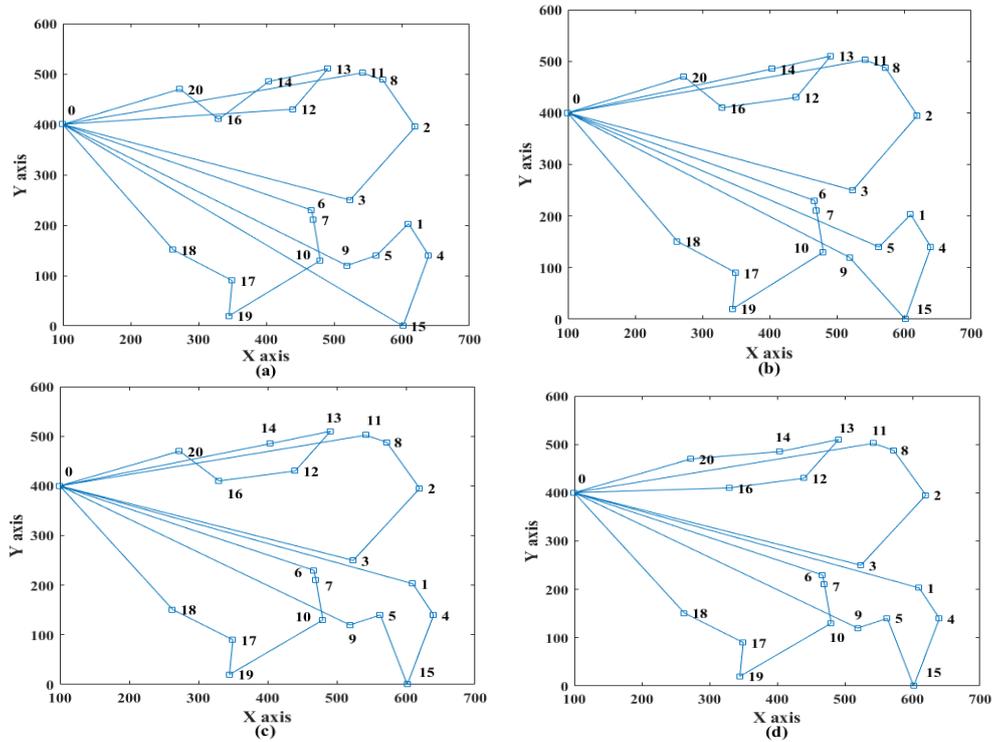
**Figure 2.** The effect of different configurations of  $\alpha$ ,  $\beta$  and  $\rho$ .

According to the experiment scale, other parameters can be set as follows: the number of chain stores  $n = 20$ ; The number of ants  $o = 100$ ; The iteration interval  $t_i (i = 1, 2) = 30, 60$ ; The pheromone intensity.

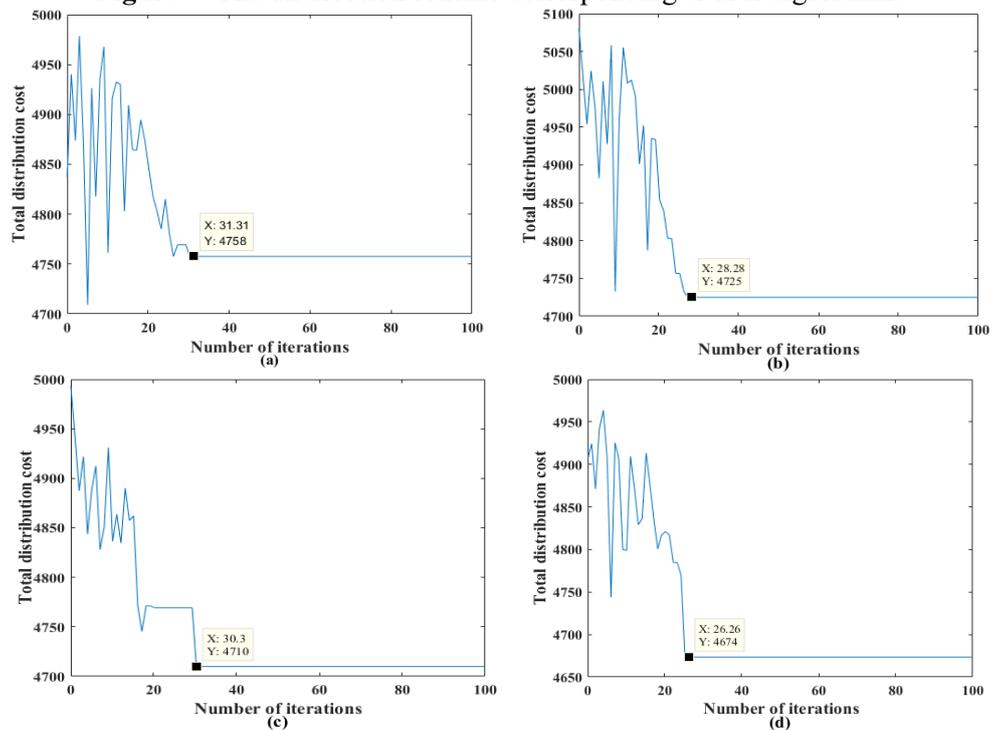
4.2. Analysis of experimental results

In this section, the algorithm before and after optimization are tested and analyzed comprehensively.

Aiming at the two aspects of improvement, the adaptive transfer ant colony method (AACO) and the dynamic directed ant colony algorithm (DACO) are respectively obtained, and the other two groups are ACO and ADACO. Figure 3 and Figure 4 respectively draw the corresponding delivery scheme, total delivery cost and convergence algebra of the four algorithms. Table 3 lists the results of each of the four algorithms running 10 times.



**Figure 3.** The distribution scheme corresponding to four algorithms.



**Figure 4.** Distribution costs of the four algorithms.

**Table 3.** The results of 10 runs of four algorithms.

Number	ACO			AACO			DACO			ADACO		
	Cost	Algebra	CPU									
1	4744	35.35	21.44	4729	29.29	17.10	4710	31.31	20.38	4674	34.34	20.31
2	4750	31.31	22.11	4725	28.28	19.66	4714	30.30	15.22	4689	28.28	18.23
3	4753	32.32	23.81	4725	34.34	18.06	4710	33.33	17.83	4680	30.3	21.43
4	4750	55.56	20.37	4725	30.30	27.46	4710	30.30	24.35	4674	29.29	18.46
5	4744	31.31	21.57	4729	46.46	20.63	4710	32.32	20.48	4689	28.28	16.51
6	4747	33.33	18.76	4729	33.33	19.50	4714	34.34	23.44	4689	30.30	17.12
7	4758	31.31	19.86	4740	30.30	21.14	4714	30.30	18.31	4689	35.35	16.31
8	4757	34.34	19.70	4729	31.31	24.54	4713	33.33	17.27	4674	26.26	20.25
9	4747	35.35	25.63	4734	29.29	18.80	4715	32.32	19.61	4674	33.33	16.06
10	4758	31.31	23.51	4732	28.28	19.08	4713	31.31	21.02	4674	34.34	17.29
Best value	4744	31.31	18.76	4725	28.28	17.11	4710	30.30	15.22	4674	26.26	16.06
Average value	4750.8	35.15	21.68	4729.7	32.12	20.20	4712.3	31.92	19.79	4680.6	31.01	18.20
Worst value	4758	55.56	25.63	4740	46.46	24.55	4715	37.37	24.35	4689	35.35	21.43

Combining the chart and table, it is clearly seen that only the routes of vehicles 1 and 3 are different in the four algorithms, but the total delivery cost are slightly different. Figure 3a shows that the vehicle 1 in ACO traverses No. 20, No. 16 and then turns to No. 14, contrary to the axiom "the shortest line between two points". In addition, Table 3 shows that the delivery cost of ACO is mainly concentrated around 4750, but Figure 4a shows that in the 5th generation, the bottleneck is expected to be broken and close to 4710. While the convergence algebra also shows ups and downs and is not stable. Therefore, it can be seen that ACO has a certain improvement space. Table 3 also shows that the delivery cost of AACO and DACO are relatively stable and tend to be around 4729 and 4710 with a certain probability. In terms of convergence algebra, the former is irregular, and the latter is mainly stable in the 30-34 generation. Figure 3b and Figure 3c show that only the route of vehicle 3 is slightly different between the two algorithms, but the difference of the total delivery costs is 15. This difference fully demonstrates that the route of vehicle 3 in DACO is desirable. Figure 4c shows that the delivery cost of DACO has a significant jump in the 30th generation, which exactly confirms the feasibility of interval setting of pheromone intensity to guide the group to escape from the predicament in time. Although AACO and DACO have improved slightly in all aspects compared to ACO, they are far from enough. By combining the two algorithms effectively, ADACO is obtained by absorbing the advantages of both algorithms. Comparing Figure 3c and Figure 3d, the two algorithms differ only in the driving route of vehicle 1, but ADACO is no longer limited to the constraint of 4700 and is reduced to 4674 with a 50% probability, and the optimal convergence algebra has reached 26.26. From a macro perspective, the CPU runtimes of the four algorithms are uneven, without any regularity.

Through the vertical and horizontal comparison of the running results in Table 3, the improvement strength of the three improved algorithms over ACO are obtained, as shown in Table 4.

**Table 4.** The improvement strength of the three improved algorithms over ACO.

Category 1	Category 2	ACO	AACO	DACO	ADACO
Cost	Best value	—	0.4%	0.72%	1.48%
	Average value	—	0.44%	0.81%	1.48%
	Worst value	—	0.38%	0.9%	1.45%
Algebra	Best value	—	9.77 %	3.23%	16.13%
	Average value	—	8.62%	9.19%	11.78%
	Worst value	—	16.38%	32.74%	36.38%
CPU	Best value	—	8.8%	18.87%	14.39%
	Average value	—	6.83%	8.72%	16.05%
	Worst value	—	4.21%	4.99%	16.39%

## 5. Conclusion

After analyzing the internal and external factors of ACO's shortcomings, this paper not only experimentally sets the key combination parameters, but also properly adjusts the transition probability and pheromone strength. Therefore, an adaptive dynamic search ant colony algorithm (ADACO) is proposed. In addition, the algorithm before and after the optimization are tested multiple times on test case. The results strongly show that ADACO has strong global optimization ability in solving VRP problem, which saves time and greatly reduces vehicle delivery cost.

The algorithm in this paper is applicable to the logistics industry represented by "CAINIAO STATION". The purpose is to use the least number of cars, travel the shortest distance and reasonably complete the delivery service. In order to meet the convenience of human social life and diversified needs, the delivery and pick-up service industry will be the next research direction, which is carried out simultaneously and increases the time window.

## Acknowledgement

This work has been supported by National Natural Science Foundation of China (No. 41562019), Scientific Research Fund of Jiangxi Provincial Education Department, China (No. GJJ170518), and Special Foundation of Postgraduate Innovation of Jiangxi province, China(No. YC2018-S331).

## References

- [1] Kim G, Ong Y S, Heng C K, et al. 2015 City vehicle routing problem (city VRP): A review[J]. *IEEE Transactions on Intelligent Transportation Systems*, 16(4): 1654-1666.
- [2] Salavati-Khoshghalb M, Gendreau M, Jabali O, et al. 2019 An exact algorithm to solve the vehicle routing problem with stochastic demands under an optimal restocking policy[J]. *European Journal of Operational Research*, 273(1): 175-189.
- [3] Dixit A, Mishra A, Shukla A. 2019 Vehicle routing problem with time windows using meta-heuristic algorithms: a survey. *Harmony Search and Nature Inspired Optimization Algorithms*. pp: 539-546.
- [4] Vincent F Y, Redi A A N P, Hidayat Y A, et al. 2017 A simulated annealing heuristic for the hybrid vehicle routing problem[J]. *Applied Soft Computing*, 53: 119-132.
- [5] Qiu M, Fu Z, Eglese R, et al. 2018 A Tabu Search algorithm for the vehicle routing problem with discrete split deliveries and pickups[J]. *Computers & Operations Research*, 100: 102-116.
- [6] Mohammed M A, Ghani M K A, Hamed R I, et al. 2017 Solving vehicle routing problem by using improved genetic algorithm for optimal solution[J]. *Journal of computational science*, 21: 255-262.
- [7] Dorigo M, Stützle T. 2019 Ant colony optimization: overview and recent advances. *Handbook of Metaheuristics*. pp: 311-351.
- [8] Mirjalili S. 2019 Ant Colony Optimisation. *Evolutionary Algorithms and Neural Networks*. pp: 33-42.
- [9] Deng W, Xu J, Zhao H. 2019 An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem[J]. *IEEE Access*, 7: 20281-20292.
- [10] Zhang Q, Xiong S. 2018 Routing optimization of emergency grain distribution vehicles using the immune ant colony optimization algorithm[J]. *Applied Soft Computing*, 71: 917-925.
- [11] Zhang W, Gong X, Han G, et al. 2017 An improved ant colony algorithm for path planning in one scenic area with many spots[J]. *IEEE Access*, 5: 13260-13269.
- [12] Duan P, Yong A I. 2016 Research on an improved ant colony optimization algorithm and its application[J]. *International Journal of Hybrid Information Technology*, 9(4): 223-234.