Application of Partial Ant Colony Algorithm on Path Planning

Lirui Yang, Mingxin Liu

The Rdfz Chaoyang Branch School, Beijing, 100020, China

Keywords: Ant colony algorithm, Optimization algorithm

Abstract: Ant colony algorithm is a method of solving the combinational optimization problems and it finds the optimal solution by simulating the process of ants searching food. This paper will discuss the basic principles of ant colony algorithm and its advantages and disadvantages. Based on the disadvantages of ant colony algorithm, namely, it comes to local optimal solution easily, this paper proposes update rule for new pheromone, which effectively improve the efficiency of algorithm. This paper also discusses the fields that the ant colony algorithm can be applied to and the benefits of the improved algorithm.

1. Introduction

Ant colony algorithm was first proposed by Italian scholars Dorigo, Maniezzo et, al.[6] in the 1990s. During the study of ant's feeding process, they found that the behavior of a single ant was relatively simple while the whole ant colony could produce some intelligent behaviors.[7] For example, the ant colony can find the shortest path to the food source in different environments, which is the reason that the ants in the ant colony can transmit information through an information mechanism. Later, the further research found that the ants will release a substance called “pheromone” on the path they pass. The ants in the ant colony can sense the “pheromone” and will follow the “pheromone” with high concentration. All the ants will leave “pheromone” on the path to form a positive feedback mechanism. After a period of time, the whole ant colony will reach the food source along the shortest path. This was first used to solve the TSP problem (Traveling Salesman Problem) and then was widely applied to various optimization and dispatching management problems. The algorithm rules include the following:

(1) The ants can perceive in a limited range and search food in “moving-perceiving” mode.

(2) The ants will leave pheromone during the movement, and the pheromone will disappear at a certain rate (i.e., evaporation rate).

(3) Ants will search food when they sense it in the sensing range; or they tend to move to places with pheromone of high concentration. Every ant will make mistakes in small probability.

(4) When there is no pheromone guides ants around them, they will follow the original motion direction initially. The ants who first leave the nest will pick the path randomly.

(5) The amount of pheromone released by the ants is negatively correlated with the distance between them and the food.

Most of the previous studies on ant colony algorithm are about the path planning based on ant colony algorithm (such as evacuation and material transportation problems in emergency situations) or different optimization solutions for ant colony algorithm. For example, Bao Wenjie [1] classifies “ant” and builds multiple ant colony algorithms based on different ideas to achieve optimization purpose and they have their own advantages and disadvantages; Qiu Lili[2] propose a method that the ant colony algorithm is applied to the mobile robot navigation technique. The robot finds a path that can reach the destination safely according to the external environment information. The way to solve the problem that the ant colony algorithm comes to local optimum is to combines the ant colony algorithm with immune algorithm and to get the optimal solution through the solution of nonlinear programming problem; At the initial stage of the operation, the ant colony algorithm cannot select the path according to the pheromone concentration but select the it randomly due to no pheromone in the path. So it is possible that the algorithm will stop in progress and it cannot calculate the next step. Or it is possible that the path selected randomly at the initial stage even is

  DOI: 10.25236/iccem.2021.039

not the part of the optimal path but the pheromone on the path can always remain at a higher concentration due to the random selection of the algorithm at the initial stage and the evaporation rate at a certain value, making the path obtained finally a simulated optimal path rather than an optimal path, that is, a certain part in the entire process can achieve the optimal solution but it not means that the entire process can have the optimal solution. The initial ant colony algorithm in this research mainly optimizes the improved calculation of the evaporation rate in the algorithm. In the optimization model, the evaporation rate is positively correlated with the pheromone concentration on the path. As the pheromone concentration increases, the evaporation rate is also faster. Therefore, there is no the situation that it is difficult to perform in the initial algorithm and the result of the algorithm is not the global optimal solution.

2. Analysis of Advantages and Disadvantages of the Algorithm

Ant colony algorithm is a kind of bionic intelligent optimization algorithm which is extremely efficient in solving various optimization problems. After the comparison of ant colony algorithm with annealing algorithm (SA) and genetic algorithm (GA), Ma Lili [5] found that the speed of these three algorithms converge to the optimal solution differs greatly in the test of the same scale and the ant colony algorithm is most efficient.

Jiang Wenbo [3] has pointed out in Discussion on Local Optimal Solution Mechanism of Ant Colony Algorithm that with no randomness, the algorithm must come to the local optimal solution so the algorithm must have randomness. However, too strong randomness will cause great reduction in performance and decrease in convergence rate of the algorithm so the global optimal solution cannot be obtained. Therefore, the optimal solution and excellent performance can be obtained by balancing randomness and non-randomness.

3. Improvement 1 (Enhancing Randomness)

It is easy for the heuristic algorithm, greedy algorithm or local algorithm to cause local optimization, or it is unable to verify whether the optimal solution is global or only local, which is because the general algorithm focuses on local solution of the large systems or complex problems to reduce the amount of computation and algorithm complexity.

The random search is an effective method to get over the local optimal solution and for the problems with unclear mechanism, the more random the search of the solution is, the less likely it is to come to the local optimum.

The lack of randomness of the original ant colony algorithm is mainly because:

(1) The artificial ants operate based on the absolute probability so the randomness is low.

(2) The probability of information selection of artificial ants depends only on the concentration of pheromone and the heuristic value. Because of the simple update rule of algorithm pheromone and continuous increase in pheromone concentration after coming to the local optimum, it is impossible for the algorithm to “jump out” the local optimal solution.

In order to enhance the randomness, this paper will adjust the update rules of pheromone so that the pheromone concentration on each path tends to an average value in evaporation, that is, the residual rate of the pheromone with high concentration on the path is low while the residual rate of the place with the pheromone of low concentration is high.

The update method of the original ant colony algorithm pheromone is as follows:

$$\tau_{ij} = \tau_{ij} \cdot \rho + \sum_{k=1}^{m} \Delta \tau_{ij}^{k} \quad (1)$$

In formula (1), the update rules on all paths are the same. When coming to the local optimal solution, the pheromone will accumulate to be difficult to jump out of the local optimal solution. In order to avoid this case, less pheromone residual on the path with high pheromone concentration and more residual in the place with low concentration can help to increase the randomness of the algorithm. The update method of new pheromone is as follows:

$$\tau_{ij} = \tau_{ij} \cdot \rho^{(\frac{\tau_{ij}}{\bar{\tau}})^z} + \sum_{k=1}^{m} \Delta \tau_{ij}^{k} \qquad (2)$$

In the formula (2), $\bar{\tau}$ represents the average of the pheromones on all paths. Many pheromones over the average are evaporated and less pheromones below the average are evaporated.

4. Improvement 2 (Controlling Randomness)

The improvement above can enhance the randomness of the algorithm and avoid the algorithm falling into the local optimal solution. It is not enough to chance the update rules of pheromone to improve the algorithm for the global optimal solution.

In many experiments, we found another significant disadvantage of the ant colony algorithm is during the update of the optimal solution, the ant colony algorithm only compares the global sum, but the smaller global sum does not mean that the local path is better:

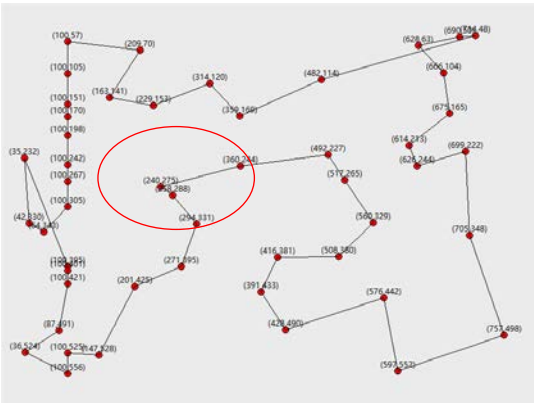


Fig.2 Ions of 50 Times

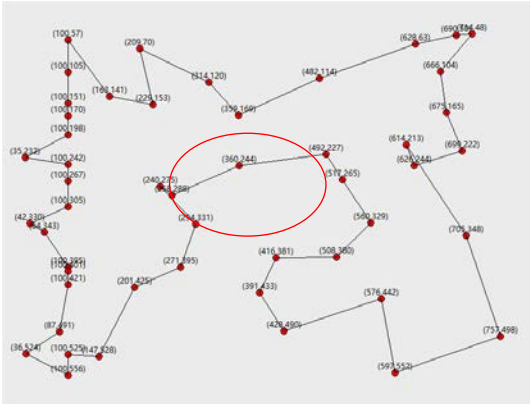


Figure 2 Iterations 150 Times

In the two figures above, Figure 1 is the result of the algorithm after 50 iterations with the total path as 3825 and Figure 2 is the result calculated after 150 iterations with the total path as 3703.

On the global path, the latter result is much better, which is obvious.

But this is not the case on some local paths, such as (294, 331); (240, 275); (258, 288); (360, 244) these four points. In the two solutions, the four points move slightly different. The beginning and end are the same but the two points in the middle move in different order. Although the global solution of the second solution is better, it is obviously that the motion of four points is not better that that of the solution 1.

To avoid this case, we will introduce a new algorithm, hereinafter referred to as the local ant colony algorithm. Its basic idea is to divide the path of the current optimal solution into several segments, hereinafter referred to as sub-segments. The sub-segments with fixed beginning and end are solved with ant colony algorithm, respectively, and are compared with the current optimal solution locally. The sub-segment with better solution will replace the original one or skip and calculate the next sub-segment.

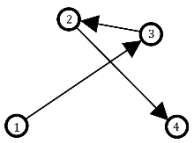


Fig.4 Gure 4

For example: Figures 3 and 4 show the two options for four points, 1, 2, 3, 4. When the beginning and end points are fixed to 1 and 4 respectively, there are only two types of options, $1 \to 2 \to 3 \to 4$ and $1 \to 3 \to 2 \to 4$. If the current one solution is divided into sub-segments of 4 points, the length of the two paths can be compared for each of the four points. The sub-segment with better solution will replace the original one and calculate the next sub-segment.

In the actual algorithm, sub-segments divided into 4 points are not enough. The global accuracy is not high, and the algorithm efficiency is reduced. However, if there are much more points taken, the local accuracy will be reduced. In order to find the most suitable number of nodes for sub-segments, we solve the number of nodes by using the ant colony algorithm 100 times randomly. The maximum number of iterations per solution is 150 and the correct rate is recorded:

For 10 nodes, the probability to find the optimal solution is 100%; For 20 nodes, the probability to the optimal solution is 86%; For 30 nodes, the probability to the optimal solution is 41%; For 40 nodes, the probability to the optimal solution is 13 %; For 50 nodes, the probability to the optimal solution is 0%.

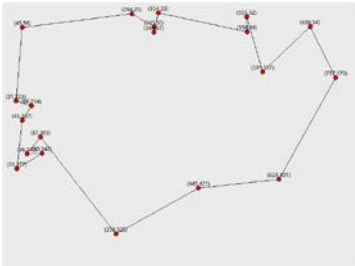


Fig.7

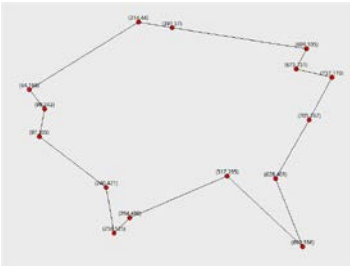


Figure 6 15 Nodes

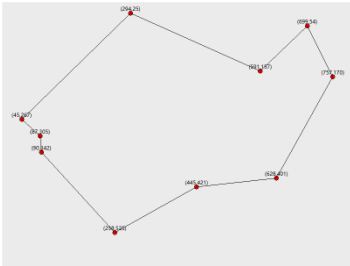


Figure 7 20 Nodes

In order to obtain higher algorithm efficiency, the optimal solution probability for the solution of sub-segments needs to be greater than 90%, and the number of nodes should be maximized. According to the experimental data, the number of nodes of sub-segments should be between 10 and 20. Fifteen nodes will be taken below.

5. Algorithm Implementation

In the paper Research and Application of Ant Colony Algorithm [4], Liu Jian Fang et, al. finds the optimal value range of each parameter through experiments. Pheromone residual factor $\rho = 0.2$; Number of ants $m =$ number of nodes; Information heuristic factor $\alpha = 2.5$; Expectation heuristic factor $\beta = 4.2$; Pheromone intensity $Q = 3050$.

□ Building the graph $G(V, E)$, V represents all nodes; E represents the path between nodes. The heuristic value $\eta_{ij} = \dfrac{1}{\tau_{ij}}$

□ Path construction and selection: Ants have some rules to choose a path. ①Ants will conduct proportional allocation according to the chosen probability of each path. ②Each ant will not take the route or pass through the node that has already passed. ③When the ants pass by any side ab, then the side ab will be added into the set. ④The probability of the kth ant to j node when node i selects the path.

$$p_{ij}^{k} = \begin{cases} \dfrac{S_{ij}}{\sum_{t \in N_i^k, w_k} \tau_{it}^{\alpha} \cdot \eta_{it}^{\beta}}, j \in N_i^k, j \notin W \\ 0, \qquad else. \end{cases}$$

□ Pheromone update. When the ants choose the route, the pheromone on each side will be updated:

$$\tau_{ij} = \tau_{ij} \cdot \rho^{(\frac{\tau_{ij}}{\tau})^{z}} + \sum_{k=1}^{m} \Delta \tau_{ij}^{k}$$

$$\Delta \tau_{ij}^{k} = \begin{cases} \dfrac{Q}{T \cdot (\frac{\tau_{ij}}{\tau})} Thekthantonthesideij \\ 0, else \end{cases}$$

Updating the optimal path: if Path $(k) \le S_{k-1}$, then:

S_k=Path (),

Otherwise: $s_k=s_{k-1}$

□ Defining the local ant colony algorithm:

Set the set s_i^j representin the set of all points from I to j;

Set the function $f\left(m,s_i^j,n\right)$:representing optimal solution obtained by 150 iterations in all elements in set s_i^j with m as the start point and n as the end point. Where, $m,n\in S_i^j$.

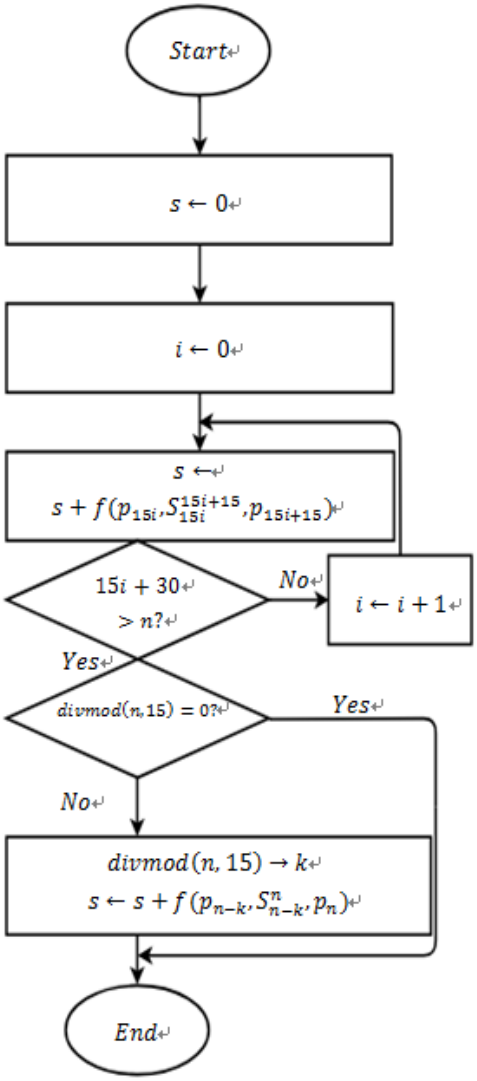
Insert local ant colony algorithm:

If: $s_k = s_{k-10}$

Then:

Insert local ant colony algorithm.

s_k Are ordered points:$p_1,p_2,p_3,...p_n$



□ Returning to step 2 and loop.

□ Initialization: $\tau_0 = 1$

6. Experimental Results

In order to verify the validity and feasibility of the algorithm, the simulation experiments are used for simulations and MATLAB is used for simulation. The simulation results of the ACO algorithm and the improved algorithm in this paper are compared and the results are analyzed. In the traditional ant colony algorithm, the core part of it is to simulate the transition probability selection behavior of the ant colony and to calculate the transition probability with the use of pheromone and heuristic function values. In the ant state transition process, the reciprocal of the

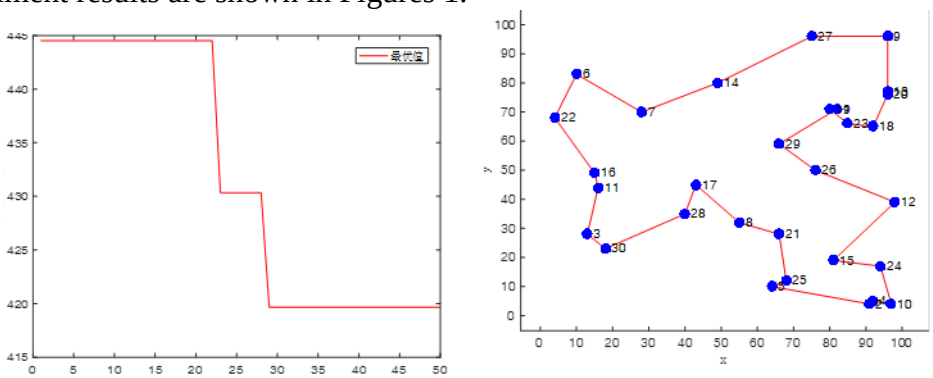
distance between the node and the target point is set as the heuristic information. What's more, under the complex path planning environment, the ant colony algorithm searches in a huge space. The pheromone concentration on the initial path is small, and the positive feedback information is not clear. Especially, the “blind search” in the process of random solution generates lots of local cross paths, which reduces the operation efficiency of the ant colony algorithm and makes it easy to come to the local optimum. When the search is proceeded to a certain extent, it is easy to cause stagnation behavior. The solutions found by all individuals are completely consistent and the further search cannot be proceeded, which is not conducive to finding a better solution. As to this problem, the improved algorithm in this paper introduces the idea of local ant colony algorithm to recalculate the part of the paths coming to the local optimal solution. Therefore, in the moving process of the ants, it guarantees that the paths will not come to the local optimum and be closer to the global optimal solution. At the same time, the ants will have directivity and the quality of the solution in a certain period of time can be improved.

The parameters set by the algorithm are shown in Table 1.

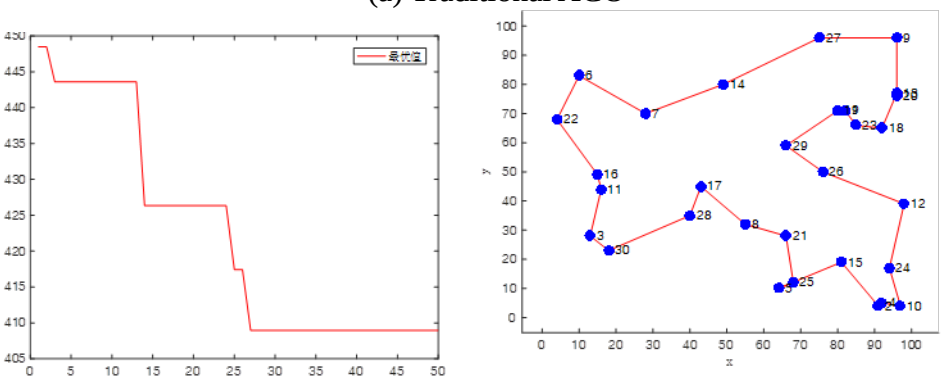
Table 1 the Parameters Set by the Algorithm

Parameter settings	Traditional algorithm	Algorithm in this paper
MaxIter: ant colony iterations	50	50
Size: number of ants	50	50
Alpha : pheromone factor	3	3
Beta : heuristic value factor	4	4
Rho:pheromone recovery coefficient	0.1	0.1
Q:pheromone enhancement factor	1	1

The experiment results are shown in Figures 1.



(a) Traditional ACO

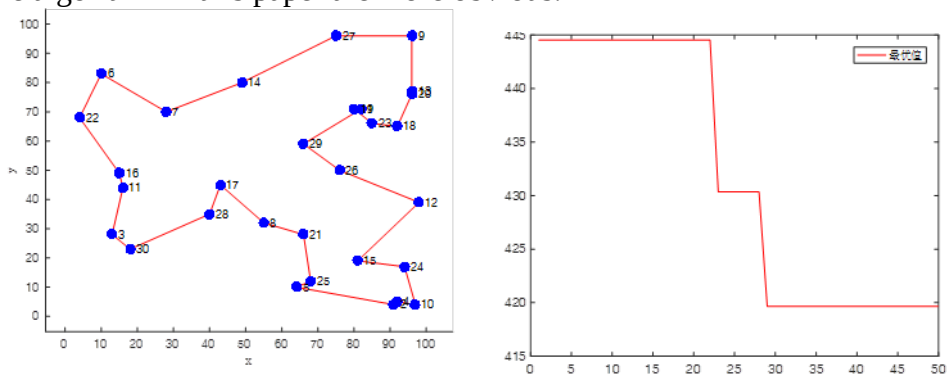


(b) Partial ACO

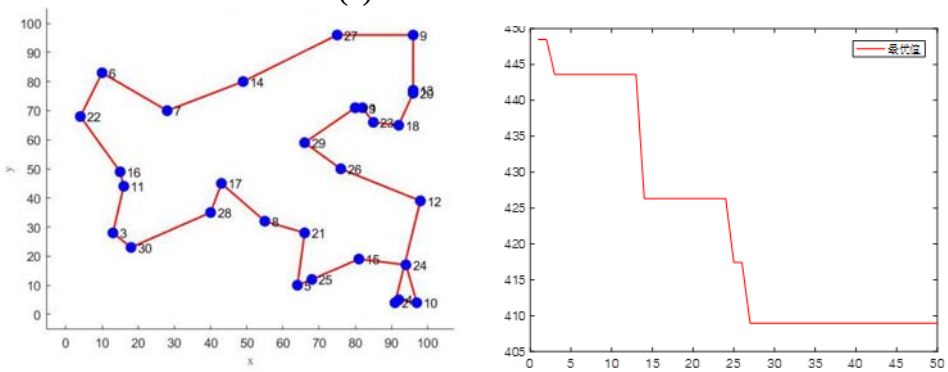
Fig.1 Experiment Results

Figure 2 shows the comparison between the traditional ant colony algorithm and the improved algorithm in this paper. From the figures, it can be seen that under the situation that the number of iterations is the same, the path of the traditional ant colony algorithm is 419.652868324472 while the path of the improved ant colony algorithm is 408.966538517999. On the local path, the improved algorithm in this paper is improved, and the path selection is optimized, preventing the

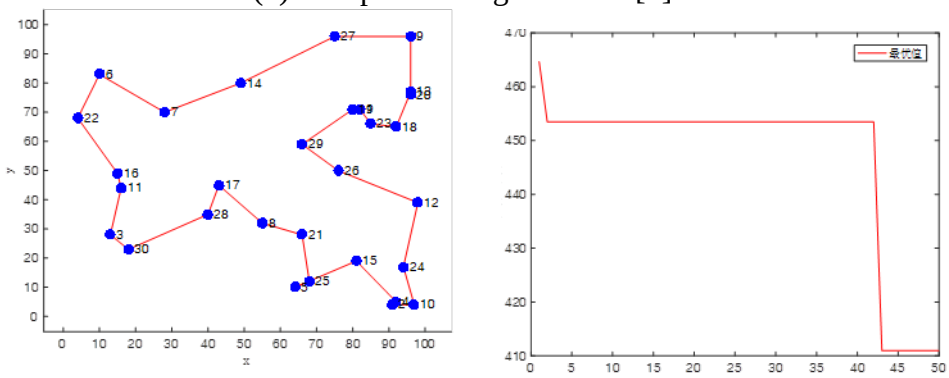
algorithm from falling into local optimum. With the increase of the number of iterations, the advantages of the algorithm in this paper are more obvious.



(a) Traditional ACO



(b) Comparison Algorithm in [2]



(c) Partial ACO

Fig.2 The Comparison between the Traditional Ant Colony Algorithm and the [illegible]ved Algorithm

The above is the comparison of the algorithm in this paper, the traditional algorithm and the comparison algorithm [2] and the results obtained by statistically calculating each algorithm are shown in Table 2.

Table 2 the Results Obtained by Statistically Calculating Each Algorithm

Algorithms	Length of optimal path	Nodes	Iterations	Operation time
Traditional ACO	420.3958	30	50	5.4735631s
Partial ACO	407.4252	30	50	3.8143829s
Comparison ACO[2]	410.9815	30	50	4.8384326s

7. Conclusions

The traditional ant colony algorithm has the problem to come to the local optimum due to the long search time and as to this problem, this paper proposes the local ant colony algorithm based on the traditional ant colony algorithm. Considering that the pheromone is added in the way of local path again on part of the path, the length of the optimal path is used for another convergence to

improve the efficiency of the algorithm. In the moving process, the ants depend largely on the existence of pheromone. The algorithm in this paper can update the pheromone efficiently so the ants can find the correct path more accurately. In terms of the solution of the shortest path, the local ant colony algorithm has good performance. In order to make the algorithm better applied in practice, next we will discuss the practical applications in the environment with obstacles.

References

[1] Wenjie Bao. Improvement and optimization of polymorphic ant colony algorithm's applications in TSP problems [D], Zhejiang Normal University 2016

[2] Qiu Lili. Research on path planning of mobile robot based on improved ant colony algorithm [D], DongHua University, 2015

[3] Jiang Wenbo. To Investigate the Mechanism of Ant Colony Algorithm to Solve the Local Optimal [J], Intelligent Computer and Applications, 2014, 4(03):53-54+59.

[4] Liu Jian Fang. Research and Application of Ant Colony Algorithms [D], Chong Qing University, 2015

[5] Ma Lili. Talking About Ant Colony Algorithms [J], CaiZhi, 2008:141-142

[6] Dorigo M, Stützle T. Ant colony optimization: overview and recent advances [M]//Handbook of metaheuristics. Springer, Cham, 2019: 311-351.

[7] Dorigo M, Birattari M. Ant colony optimization[M]. Springer US, 2010.