Research on TSP Application Based on Improved Ant Colony Algorithm

Pan Zhao¹, Xiaoqin Ma^{1,+}, Xiaoling Yin¹

College of Mathematics and Computer Science, Chizhou University, Chizhou 247000, China

Abstract. In order to solve the shortcomings of traditional ant colony algorithm in solving traveling salesman problem (TSP), such as slow convergence speed and easy to fall into local optimum, an improved ant colony algorithm (IACO) is proposed. The algorithm uses k-nearest neighbor to influence the distribution of initial pheromones, applies roulette operator to urban transfer rules, and improves the pheromone updating strategy of ant colony to accelerate the convergence speed and improve the optimization ability of algorithm. Taking chn31 city problem as an example, the computer simulation results show that the improve dalgorithm is an optimization algorithm which can accelerate the convergence speed and improve the optimization ability, and is effective for solving TSP.

Keywords: Ant colony algorithm, Traveling Salesman Problem, Pheromone, Roulette

1. Introduction

TSP is a very classical combinatorial optimization problem [1]. It belongs to one of the most important problems in the field of optimization and has strong engineering application. It mainly seeks the optimal solution of discrete events through mathematical research. At present, it has been widely used in transportation, logistics distribution, computer network communication node setting, aircraft route arrangement, road network construction, integrated circuit wiring and other fields [2-3]. However, with the increase of the number of cities, the spatial and time complexity of solving problems will increase exponentially, which can not be solved by conventional methods. In recent years, with the rapid development of artificial intelligence, many researchers have proposed intelligent optimization algorithms such as ant colony, genetic, greedy, neural network and so on [4-9]. But these algorithms have their own advantages and disadvantages.

As a new bionic evolutionary algorithm, ant colony algorithm (ACO) was proposed by M. Dorigo scholars in the early 1990s [10]. The algorithm imitates pheromones generated by ants in the process of foraging to transmit information. Ant colony algorithm has the advantages of strong robustness and easy to combine with other algorithms. Although ant colony algorithm can evolve the path to the optimal solution through pheromone communication, when the number of ant colonies is large, the lack of pheromone in the initial stage of the algorithm results in a slow convergence rate, which makes it difficult to find a better path in a short time. In view of the shortcomings of the traditional algorithm, researchers at home and abroad have improved the algorithm in many ways. Although the convergence rate of the algorithm has been improved to a certain extent, the lack of diversity of ant population has led to the local optimum situation in solving TSP.

2. Description of TSP

⁺ Corresponding author. Tel.: +86 13866813085; fax: +.86 0566-2748769. *E-mail address*:czumxq@163.com.

The general description of the problem is: a traveler wants to visit n cities, and each city only visits once, and finally returns to the place of departure to find a shortest closed travel path [11]. The mathematical model is as follows:

Knowing n cities $V = \{v_1, v_2, \dots, v_n\}$, and the Euclidean distance $d_{ij} = (v_i, v_j)$ between any two cities v_i and v_j , where the v_i coordinates are (x_i, y_i) , then

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} , \qquad (1)$$

Seek a route with the shortest total path that passes through all cities in V only once. That is, to minimize D.

$$D = \sum_{i=1}^{n-1} d(v_i, v_i + 1) + d(v_n, v_1), \qquad (2)$$

Formula (2) is the model that needs to be optimized for designing the algorithm.

3. Proposed methods

3.1. K-nearest Neighbor Initial Pheromone Distribution

K-nearest neighbor is one of the simplest algorithms in data mining classification algorithm. Because of its simplicity, easy implementation and fast solution, this algorithm is used to construct the initialization travel path l_n of TSP. Applying k-nearest neighbor to solve the TSP problem, the algorithm process can be described as follows: in the first step, a certain city is identified as the starting city for constructing the travel path l_n ; in the second step, a city nearest to this starting city is selected among the cities that have not been visited, and the selected city is added to the constructed route l_n and serves as the next starting point; in the third step, the first two steps are repeated until all the cities are added to the constructed route l_n , and finally the starting city is used as the end of route l_n .

When solving the TSP problem, the traditional ant colony algorithm uses the initial pheromone set to a fixed constant. During the initial iteration of the algorithm, when the path distance d between two cities is shorter, the higher the pheromone concentration is, the greater the probability of being selected by other ants, and finally lead to the algorithm to choose the path non-optimal solution. Aiming at the shortcomings of the traditional method in the initial iteration of solving TSP problem, k-nearest neighbor is used to obtain the average value of the initial travel path to transform the distance between any two cities, and the transformed value is used as the initial pheromone distribution matrix $\tau_{ii}(0)$.

$$\tau_{ij}(0) = \begin{cases} n / (d_{ij} \cdot l_n), & i \neq j \\ 0, & others \end{cases}$$
(3)

The initial pheromone distribution matrix can be obtained by Equation (3), where n is the number of visited cities. l_n/n is the nearest neighbor average of the distance between any two neighboring cities in the initial tour route l_n . Therefore, the initial pheromone distribution matrix $\tau_{ij}(0)$ obtained from Equation (3) can weaken the pheromone concentration and compensate for the deficiencies of traditional algorithms. At the same time, since the initial travel route l_n is relatively stable and relatively close to the optimal solution path, it will not cause the final selected path to deviate from the optimal solution. In the initial stage of solving TSP problem, using k-nearest neighbor to initialize the pheromone distribution matrix and increase the diversity of solutions can effectively improve the deficiency that the algorithm is easy to fall into the local optimum due to the high initial pheromone concentration, which is beneficial to obtain the global optimum solution.

3.2. City Transfer Rules

Roulette algorithm is also called proportional selection operator which is often used in genetic algorithm. Therefore, roulette algorithm can be applied to city transfer rules with ant colony optimization. The probability of a city being selected is in direct proportion to its fitness. Applying roulette algorithm to ant colony optimization can drive the algorithm to a good direction and it is helpful to get optimal path. The

quantity of ants is set as m, and the fitness of ant i is f_i . The probability if ant i being selected to be inherited to the next generation is:

$$\rho_{ij}^{k}(t) = \begin{cases}
\frac{[\tau_{ij}(t)]^{\alpha} [\eta_{ij}(t)]^{\beta}}{\sum [\tau_{il}(t)]^{\alpha} [\eta_{il}(t)]^{\beta}} \cdot [\frac{f_{ij}}{\frac{p}{\sum j=1}}]^{r}, j \notin U_{k} \\
\frac{l \notin U_{k}}{0, \quad others}
\end{cases}$$
(4)

In the formula, q shows the number of subintervals the kth ant is in between the ith city and the i+1th city; f_{ij} is the fitness of solutions in the ith subinterval and the jth interval; $\sum_{j=1}^{r} f_{ij}$ is the sum of the solutions to ij in all q subintervals; γ is the heuristic factor of roulette.

3.3. Pheromone Updating

When searching paths with traditional algorithm, the searching is easy to be interfered by non-optimal path pheromones and falls into local optimum. To solve this problem, this research takes updating original pheromones as the foundation, releases more pheromones to the optimal paths of every iteration, and reduces part of pheromones for the worst paths. The improved pheromone updating formula is

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) + \rho(M_b \cdot \tau_{bij}(t) - M_w \cdot \tau_{wij}(t))$$
(5)

$$\Delta \tau_{ij}(t) = \sum_{k=1}^{m} \Delta \tau_{ij}^{k}(t)$$
(6)

$$\Delta \tau_{ij}^{k}(t) = \begin{cases} Q / L_{k}, & \text{the paths}(i,j) \text{ the kth ant passes in this cycle} \\ 0, & \text{others} \end{cases}$$
(7)

$$\Delta \tau_{bij}(t) = \begin{cases} Q / L_b, & (i,j) \in the \ optimal \ path \ of \ this \ iteration \\ 0, & others \end{cases}$$
(8)

$$\Delta \tau_{wij}(t) = \begin{cases} Q / L_W, & (i,j) \in the worst \ path \ of \ this \ iteration \\ 0, & others \end{cases}$$
(9)

In the formula, M_b and M_w are the quantity of ants which pass the optimal path and the worst path in this iteration respectively; L_b and L_w are the lengths of the optimal path and the worst path in this iteration respectively.

4. The Description of IACO

The process to solve problem TSP with improved ant colony optimization is shown as Fig. 1. Main steps are as follows:

Step 1: Initialize related parameters of ant colony optimization, such as m, α , β , γ , and ρ etc.

Step 2: Use k-nearest neighbor to initialize pheromone matrix randomly and calculate the fitness of ants.

Step 3: According to the rules of roulette algorithm, the kth ant transfers from city i to city j. City i is written in the taboo list U_k and get it updated.

Step 4: When all the ants finish travelling, record the quantity of ants and the lengths for the optimal path and the worst path in this iteration. Update the concentration of pheromone and the counter N_c for travel times.

Step 5: See if the iteration times N_c reach the setting value. If yes, stop the output of optimal path for the iteration. If no, clear the path table and continue the iteration until the setting value is reached.



Fig. 1: Process of Solving TSP with Improved Ant Colony Optimization.

5. Experimental Simulation

5.1. Setting Experimental Parameters

To verify the effectiveness of the algorithm, chn31 city problems are selected from TSPLIB for simulation in the computer. Key parameter settings are shown in the following table:

Table 1: Key Parameter Setting.

α	β	γ	ρ	Q	N_c
1.0	5.0	4	{0.2, 0.3, 0.4, 0.5, 0.7, 0.8}	110	200

5.2. Comparison of Simulation Results with the Algorithms

This thesis has compared and analyzed IACO, ACO, and MMAS [12]. Each algorithm has 10 simulation experiments.

It is known from the simulation experiments that the optimal solution value of traditional ant colony optimization is 15602km, its optimal path is shown as Fig. 2; The optimal solution value of MMAS is 15496km, its optimal path is shown as Fig. 3; The optimal solution value of improved ant colony optimization is 15381km, its optimal path is shown as Fig. 4. According to the comparison result of the three algorithms in Table 2, better optimal path value is obtained through improved ant colony optimization.



Fig. 2: Optimized Path of Ant Colony Optimization.



Fig. 3: Optimized Path of MMAS.



Fig. 4: Optimized Path of Improved Ant Colony Optimization.

Test sugables	optimal path length (km)				
lest number	ACO	MMAS	IACO		
1	15838	15722	15602		
2	15602	15602	15475		
3	15602	15496	15381		
4	15602	15496	15381		
5	15602	15496	15381		
6	15602	15496	15381		
7	15722	15838	15381		
8	15602	15496	15381		
9	15838	15722	15495		
10	15840	15838	15602		
mean length	15685	15620	15446		
the shortest length	15602	15496	15381		

Table 2: Result Comparison on Simulation Experiments with Three Algorithms

Fig. 5 is the comparison on paths in the iteration with the three algorithms. It can be seen that the improved ant colony optimization enjoys faster searching speed in initial iteration calculation. It keeps the variety in later period and the iteration times in searching for optimal path is fewer. The experiment result shows that ant colony optimization not only improves the convergence speed, but also improves the shortage of being easy to fall into local optimum with the traditional algorithm. It is effective to solve TSP.



Fig. 5: The Comparison on Paths in the Iteration.

6. Conclusion

To avoid the defects of low convergence speed and being easy to fall into local optimum in TSP, this research brings up an improved ant colony optimization, which uses k-nearest neighbor to influence the initial pheromone distribution and ensures the variety of ant colonies to get overall optimum. It applies roulette operator to city transfer rules and drives the algorithm to a better direction to improve the quality to get the optimal path. Meanwhile, the updating strategy for ant pheromones are improved to accelerate the convergence speed and optimization capacity. Computer simulation is done for chn31 problems. According to the simulation result, the improvements for ant colony optimization is effective.

7. Acknowlegments

This work is supported by the Natural Science Research in Colleges and University of Anhui Province, China (KJ2015A290, KJ2017A579); the Application course construction project of Chizhou University, China (2017XYYKC01); the Natural Science Research Program of Chizhou University, China (CZ2018ZR04); the Supporting Program for Excellent Young Talents in Colleges and Universities of Anhui Province in 2019, China (gxyq2019110).

8. References

- [1] Flood Merrill M. The traveling-salesman problem. Operation research. 1956, 4(1): 61-75.
- [2] Michalewicz Z, Fogel D B. How to solve it: Modern Heuristick. Berlin Heidelberg: Springer. 2000: 58-78.
- [3] HE Qing, WU YiLe, XU TongWei. Application of improved genetic simulated annealing algorithm in TSP optimization. *Control and Decision*. 2018, 33(02): 219-225.
- [4] LI Qing, ZHANG Chao, CHEN Peng, et al. Improved ant colony algorithm based on particle swarm optimization. *Control and Decision*. 2013, 28(6): 873-878.
- [5] Tsai C F, Tsai C W, Tseng C C. A new hybrid heuristic approach for solving large traveling salesman problem. *formation Sciences*. 2004, 166(1/2/3/4): 67-81.
- [6] Dorigo M, Di C G, Gambardella L M. Ant algorithms for distributed discrete optimization. *Artificial Life*. 1999, 5(2): 137-172.
- [7] Tsai C F, Tsai C W, Yang T. A Modified multiple-searching method to genetic algorithms for solving traveling salesman problem. *2002 IEEE Int Conf on Systems, Man, and Cybernetics.* 2002: 1-6.
- [8] Zhang X D, Chen J, Luo Z, et al. An Improved Greedy Algorithm with Information of Edges' Location for Solving the Euclidean Traveling Salesman Problem. *Chinese Journal of Computers*. 2013, 36(4): 836-850.
- [9] Cochrane E M, Beasley J E. The co-adaptive neural network approach to the euclidean travelling salesman problem. *Neural Networks*. 2003, 16(10): 1499-1525.
- [10] Dorigom M, Maiezzo V, Colorni A. Ant system: Optimization by a colony of cooperation agents. *IEEE Transaction on Systems Man and Cybernetics: Part B.* 1996, 26(1): 29–41.
- [11] Dantzig G, Johnson S. Solution of a large-scale traveling-salesman problem. *Operations Research*. 2010, 2(4): 393-410.
- [12] Stutzle T, Hoos H H. Max-min ant system. Future Generation Computer Systems. 2000, 16(8): 889-914.