Using the Improved Ant Colony Algorithm to Solve the Chinese TSP

Sun Jing, Bai Yan-ping, Hu Hong-ping and Lu Jin-na
Department of Mathematics, University of North of China, Shanxi Province, China
sunjinglove@126.com, baiyp@nuc.edu.cn

Abstract - In this paper, an improved mixed Ant Colony Algorithm is proposed. The introduced algorithm is based on the traditional ant colony system algorithm. At the beginning an initial result is constructed using the nearest neighbour method. Build on top of that, the result is improved using 2-opt partial search strategy. Only the best two colonies' global pheromones are updated which used the rank-based ant colony system idea. Then we used MATLAB to simulate the classic Chinese TSP problem the dimension of which is 31. The best result we achieved is 15377. This result surpasses all the other results we have ever known. Afterwards we used a method of counting the sum of the route edges to measure the population diversity of our algorithm. Then we compared the population diversity of our improved mixed algorithm and the base ACO algorithm. The result shows our algorithm has higher population diversity which gives us a theory support why our algorithm can achieve best result than ever known.

Index Terms - CTSP, Mixed ant colony algorithm, partial search strategy, population diversity

1. TSP AND ACO Algorithm Introduction

1.1 Brief Introduction of TSP

TSP is short for Travelling Sales Problem. It was first mentioned in a handbook in 1832 and an example tour through Germany and Switzerland was recorded, but it didn't get formulated until 1930. Since then it has become the most intensively studied optimization problem and a benchmark for the optimization algorithms. The TSP problem can be described as follow[1]: given a list of cities and the distance between any pair of the cities, if each city in the list must be visited once and only once and starting city be visited at last, in what sequence of city traverse can we get the shortest route?

TSP has been proved to be an NP-difficult problem in combinatorial optimization. And it's very important in theoretical computer science and operational research. Nowadays there are mainly three kinds of algorithms to solve it: precise algorithm, approximate algorithm and the heuristic algorithm [2,3]. In this paper, a mixed algorithm is introduced; it is based on the traditional ant colony system algorithm and the 2-opt partial search strategy [4,5] . A rank-based ant colony system is also used. With the MATLAB simulation, it can be shown to be an effective problem to solve the TSP problem.

1.2 ACO Algorithm

ACO was proposed by Marco Dorigo in his doctoral thesis in 1991[6].The original idea of ACO came from the observation of the exploitation of food resources among ants. An example of ants looking for food can be simulated as figure1. We assume that Nest-ABD-Food and Nest-ACD-Food have a length of 4 and 6 respectively, ants can move one unit distance within unit time, and none pheromone is left at the beginning [7].

t=0, 20 ants move from nest to A. They choose either the left path or the right path with the same possibility. So 10 ants turn right, the other 10 ants turn left.

t=4, the first 10 ants arrive at the destination food and starts to turn back.

t=5, the two groups of ants meet at D, at the same time , the density of the pheromone of path BD is the same as the density of the pheromone of path CD. So 5 of the 10 coming back ants turn left and 5 of them turn right.

t=8, the first 5 ants come back to the nest and 5 ants left on path AC and 5 ants left on path CD while the other 5 ants on BD.

t=9, the first 5 ants arrive at A again and they face the same left-right problem.

At this time, the density of AB is 20 which is a little higher than that of AC which is 15. More ants will choose the path AB. With the process continues, the gap between the pheromone of the two paths will increase and finally all the ants will choose the shortest path.

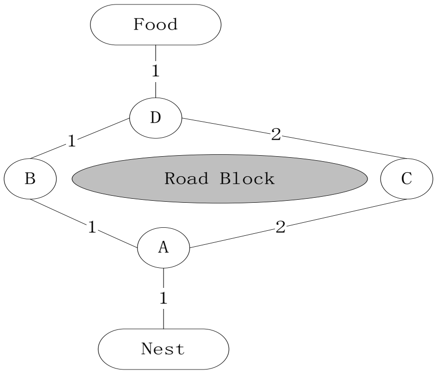

Fig.1 The Simulation of Ants Moving from Nest to Food

2. Improved Mixed Ant Colony Optimization Algorithm

ACO algorithm has been applied to many combinatorial optimization problems, such as VRP, protein folding, quadratic assignment and TSP. When solving the TSP problems, ACO always stop finding the best result at locally optimal solution. In order to find the best solution and to increase the possibility finding the best solution, we worked out three improvements on the basis of the ACO:

(1) 2-opt partial search strategy is used for the best solution and the second best solution in each iteration. 2-opt partial search strategy increase the possibility of finding the best solution. The reason we apply 2-opt partial search only for the best and the second best solution is that even we apply the 2-opt partial search for all the solutions, improvement is trivial compared with only applied to the best two.

(2) After the 2-opt partial search, we rank the results once again and do the overall pheromone update for the best and the second best solution. The weight of updating the pheromone is 1 and 0.5. Formula (1) gives the value of$\Delta\tau_{ij}$.

$$\Delta\tau_{ij} = \begin{cases} \frac{1}{Lbest}, if\,(i,j) \in Best \\ 0.5 * \frac{1}{Lbest2}, if\,(i,j) \in SBest \end{cases} \tag{1}$$

In the formula above, Best means the best solution and *SBest* means the second best solution. *Lbest* stands for the length of the best solution and *Lbest2* stands for the length of the second best solution. In order to widen the search scope, we did the update for the second best solution. In order to keep the competitive advantage of the best solution, the weight of the second best solution is smaller than 1 and assigned to be 0.5 which is the best choice for the TSP problem after sufficient experiment.

(3) In this algorithm, maximum and minimum pheromone is adopted so as to prevent the algorithm from stopping at the locally optimal solution[8]. In every iteration, the maximum and the minimum pheromones are related to the path length, the value of which is shown in formula (2):

$$\Delta\tau_{\max} = \begin{cases} \frac{1}{1-\rho} * \frac{1}{Lbest}, if\,(i,j) \in Best \\ \frac{1}{1-\rho} * \frac{1}{Lbest2}, if\,(i,j) \in SBest \end{cases} \tag{2}$$

In the formula above, the same meaning applies to *Best*, *SBest*, *Lbest*, *Lbest2* as in fomula (1). The minimum pheromone is assigned to one percent of the maximum pheromone: $\Delta\tau_{\min} = 0.01 * \Delta\tau_{\max}$.

3. Simulation in Matlab

The main procedures of the MATLAB program of the improved mixed ACO Algorithm to solve the TSP problem are designed as below:

Step 1. Initialize the global parameters: $\alpha=1$, $\beta=2$, $\rho=0.75$, MaxNc=5000, M=20 and all the other global parameters used in the algorithm.

Step 2. At the beginning of the iteration, decide whether current iteration number is larger than MaxNc, if not, current iteration continues, else goes to step 13.

Step 3. Initialize the parameters used in current iteration. Place M ants at M random starting cities. Start current iteration. Different ants may be placed at a same city.

Step 4. In the current iteration, decide for each ant whether it has visited all the cities. If there are still cities left, select one city for each ant to visit using the Pseudo-random proportion rule, else go to step 7.

Step 5. Place the selected city in the forbidden matrix. Cities in the forbidden matrix will never be selected again. At the beginning, the forbidden matrix only includes the starting city.

Step 6. Do the local pheromone update for the path (i,j) formed by the current city i and the selected city j.

Step 7. When all the cities have been visited, take the starting city as the next target city. Ants return to the starting city, thus an entire route is formulated, then update the pheromone for the last path.

Step 8. Calculate the length of the route for each ant. And find the best solution and the second best solution among all the ants.

Step 9. Execute the 2-opt partial search method for the best and the second best solutions.

Step 10. Compare the best solution achieved from step 9 with current best solution, if the best solution from step 9 is better than current best solution, assign the current best solution to the best solution achieved from step 9, or else discard the best solution of this iteration.

Step 11. Update the global pheromones for the best and the second best solution achieved in step 9 with a constraint from $\Delta\tau_{\min}$ to $\Delta\tau_{\max}$. If the pheromone is larger than $\Delta\tau_{\max}$, then assign $\Delta\tau_{\max}$ to the path, similarly, if the pheromone is smaller than $\Delta\tau_{\min}$, then assign $\Delta\tau_{\min}$ to the path.

Step 12. Using the visit sequence matrix to calculate the quantity of the route ants passed by in the iteration and save the value for further analysis. Go to step 2.

Step 13. Output the results of this experiment.

4. Experiment on the Chinese TSP and result comparison

The Chinese TSP problem is a classic TSP problem. Many Chinese scholars have studied this problem for many years since 1991. Professor Fan Jin gave a solution for this problem in his book [9]. The problem consists of 31 capital cities of China. The city sequence in our paper is:

1. Lhasa 2.Beijing 3. Shanghai 4.Tianjin 5. Shijiazhuang 6.Taiyuan 7. Hohhot 8.Shenyang 9.Changchun 10.Harbin 11.Xi'an 12.Lanzhou 13.Yinchuan 14. Xining 15. Urumchi 16.Ji'nan 17.Nanjing 18. Hangzhou 19. Hefei 20.Nanchang 21.Fuzhou 22.Taibei 23.Zhengzhou 24.Wuhan 25.Changsha 26.Guangzhou 27.Nanning 28.Haikou 29.Chengdu 30.Guiyang 31.Kunming

The coordinate matrix of the cities is [10]:

Location = [1304 2312; 3639 1315; 4177 2244 ; 3712 1399 ; 3488 1535; 3326 1556 ; 3238 1229; 4196 1004; 4312 790; 4386 570 ;3007 1970; 2562 1756; 2788 1491; 2381 1676; 1332 695 ; 3715 1678; 3918 2179; 4061 2370; 3780 2212; 3676 2578; 4029 2838; 4263 2931;3429 1908; 3507 2367; 3394 2643; 3439 3201; 2935 3240; 3140

3550; 2545 2357; 2778 2826;2370 2975]

We used MATLAB to simulate and solve the Chinese TSP problem. We assigned the following parameters: M=20， α=1， β=2， ρ=0.75. And the best solution we got is 15377.7113 km and this is the best solution known by now and the runtime is 123.194s.

The sequence of the city visited of the best solution we achieved is:

Beijing-Shijiazhuang-Taiyuan-Hohhot-Yinchuan-Lanzhou-Xining-Urumchi-Lhasa-Chengdu-Kunming-Guiyang-Naning-Haikou-Guangzhou-Changsha-Nanchang-Fuzhou-Taibei-Hangzhou-Shanghai-Nanjing-Hefei-Wuhan-Xi'an-Zhengzhou-Ji'nan-Tianjin-Shenyang-Harbin-Beijing.

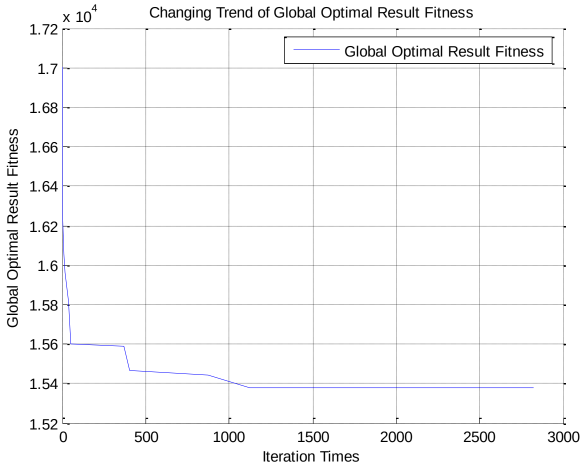

Fig.2 the Changing Trend of Global Optimal Result Fitness

We also obtained figures from our program. Figure 2 shows us the changing process of the best solution while Figure 3 shows us the best visiting sequence of the Chinese TSP problem. Figure 3 looks a little different from the Chinese Map because the measurements of Figure 3 are not longitude and latitude. In fact, Figure 3 will look exactly the same as the Chinese Map if it is rotated 180 degrees from the bottom up.

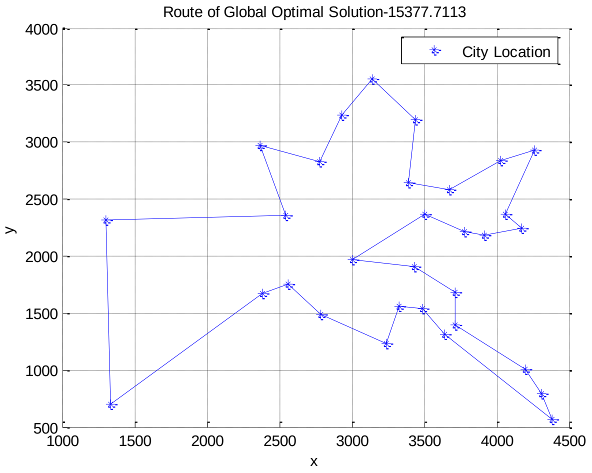

Fig.3 the Route of Global Optimal Solution of the Mixed ACO

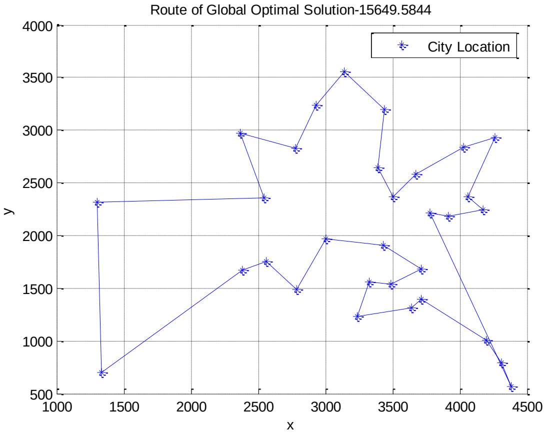

Fig.4 the Route of Global Optimal Solution of the base ACO

In order to demonstrate the efficiency of our program, we compared the improved mixed ACO with the base ACO. The length of the best solution of the base ACO is 15649.5844 km, and the runtime is 79.7789s. Although the run time of the base ACO is smaller than our improved mixed ACO algorithm, we got a much more precise result than the base ACO. Figure 4 shows the best solution of the base ACO. More detailed statistic comparing two algorithms can be found in table 1.

TABLE I the Comparison of the Base ACO with the Mixed ACO Solving the Chinese TSP

Algorithm	Average path (km)	Average time (s)	Shortest path （km）	runtime （s）
Base ACO	15931.68	97.40052	15649.5844	79.7789
Improved mixed ACO	15391.99	169.0869	15377.7113	123.194

In document 11, the author found a solution with length 15404 km using improved ACO, and compared with this solution, our solution is smaller by 0.18 percent.

5. Population Diversity Analysis

The population diversity is the basis of finding the optimal route, without which we can only achieve partial optimal result instead of the global optimal result because the program would stop too early.

There are two traditional methods to measure the population diversity. One is to calculate the variance of each individual's fitness value and the other is to compare the visiting sequences of each individual ant. The first method is not very accurate sometimes, e.g. literature 12 shows us a TSP with the dimension 75.The route length is almost the same, but actually the two routes are quite different. Figure 5 shows us the two different routes. Moreover the second method is too complex. When the dimension of the city is very large, it will take very long time to compare the visiting sequence of the ants. In document 13, the author introduced a new method: using the sum number of the route edges of each iteration to measure the population diversity. And we can draw the conclusion from the paper that the larger the sum number of the route edges is, the higher the population diversity is. And in this paper we will use this method to count the population diversity.

As we used 2-opt partial search strategy in the algorithm, we calculate the sum of the route edges after the 2-opt partial search in each iteration. At the initializing phase, we initialize the *visitMatrix* to be an zero matrix with rank the dimension of the cities. We use the visiting sequence matrix(the final visiting sequence of the route of each ant) to fill the *visitMatrix*. Suggest in some row of the sequence matrix we have the visiting sequence i j k, we and one for row i column j of the *visitMatrix*, and for the sequence j k, we add one for row j column k of *visitMatrix*. Meanwhile in order to count the numbers in an easy way, we only fill the upper half of *visitMatrix*, e.g. if i>j then we will only add one for row j column i for *visitMatrix*. In each iteration, the maximum element of *visitMatrix* should be M, which equals the number of the ants, and when the element is M, it presents all the ants choose this edge in the route; this route is definitely in the final optimal route.

In the instance of CTSP, we used the method mentioned above to measure the population diversity in both the improved mixed algorithm and the base ant colony algorithm. We ran 10 times for both of the two algorithms, and in each process, the iteration times is 5,000. After 10 times experiment, we recorded the sum of the edges in each iteration. More detailed statistic comparing two algorithms can be found in table 2.

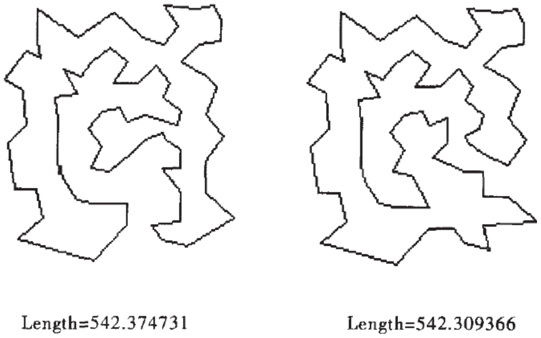

Fig. 5 an Example of the Incorrectness of Variance Method

TABLE Ⅱ the Data after 10 Times Program Run for the Two Algorithms

Algorithm	Average sum of the edges	Max sum of the edges	Min sum of the edges
The improved algorithm	122	150	92
Base ACO	108	142	73

As shown in the table above, the improved algorithm has better result in average sum of the edges, max sum of the edges, and min sum of the edges. And this table explains from the data level why our improved algorithm can achieve better results than the base ACO, the reason is we have higher population diversity and we are more likely to find the global best result.

6. Population Diversity Analysis

On the basis of ACO Algorithm, we proposed an improved mixed ACO which used the 2-opt partial search strategy. Through the simulation with MATLAB, we can conclude that this algorithm works very well on TSP problems and we achieved very good solution for the classic Chinese TSP problem with the length 15377 km which is shortest ever known. We studied the population diversity of the algorithm with the idea of the sum of the quantity of the route edges. Then we compared the diversity of our improved mixed ACO with the base ACO, the result showed us that the population diversity of our improved mixed ACO is much higher than that of base ACO's. And that's why our improved mixed algorithm can achieve better results. However, compared to the base ACO, we did the 2-opt search, the runtime is a little longer, and this is also our striving direction.

Acknowledgment

Thanks for my tutor, Professor Bai. She helped me a lot in everyday research and study. I learned a lot from her, without her help I couldn't have written this paper. Thanks for my boyfriend, he helped me a lot in everyday life, supported me on my back.

References

[1] Gao Hai-chang, Feng Bo-qin, Zhu Li. Reviews of the Meta-heuristic Algorithms for TSP. Control and Decision,vol.21,No.3,pp. 241-242, 2006

[2] Chen Yin, Xu Hong-mei. Application of Hybrid Algorithm in Vehicle Routing Problem. Computer Simulation, vol.29,no.25,pp. 357-358,2012

[3] Yang Ya-ping. Research on Modeling and Algorithm Based on Vehicle Routing Problem. Computer Development & Applications, vol.25, no.12, pp. 962-963, 2012

[4] Liu Xia, Yang Chao.Min-max vehicle routing problem based on ant colony algorithm. Journal of PLA University of Science and Technology (Natural Science Edition), vol.13, no.3. pp.340 -341, 2012

[5] Zhang Yu-dong, Wu Le-nan, Wei Geng. Geng. Comparison on solving TSP via intelligent algorithm. Computer Engineering and Applications,vol.45,no.11, pp. 11-15,2009

[6] Colorni A, Dorigo M, ManiezzoV, et al. Distributed optimization by ant colonies//Proceedings of the 1st European Conference on Artificial Life, pp. 134-142, 1991

[7] Ma Liang, Zhu Gang, Ning Ai-bing. Ant colony optimization algorithm. Bei Jing: Science Press, 2007, pp. 20-21.

[8] Thomas Stutzle, Holger Hoos. The MAX-MIN Ant System and Local Search for the TraVeling Salesman Problem. Indianapolis, Indiana, USA, Proceedings of the IEEE International Conference on EVolutionary Computation, pp. 3-13,1997

[9] Jin Fan, Fan Jun-bo, Tan Yong. Neural Network Computer. Chengdu: Southwest Jiaotong University Press, 1991, pp. 94-104,375-376

[10] Li Ming-hai, Xing Gui-hua.Resolution of Chinese Travling Salesman Problem in Matlab. Mirocomputer Applications, vol25,no.2, pp. 221-222, 2004

[11] Cong Shuang, Jia Ya-jun. Traveling Salesman an Problem Based on integration of evolution strategies and Ant Colony Algorithm. Control Engineering of China, vol.18,no.1,pp. 84-85, 2001

[12] Chen Xian-fu. The theoretic research of genetic optimization .Anhui: University of Science and Technology of China,1996

[13] Jiang Lei, Chen Xian-fu. A new method for measuring the population diversity of TSP. Microelectronics & Computer,vol.21,no.8, pp. 10-12,2004