# Ant Colony Optimization Routing Algorithm with Tabu Search

Masaya Yoshikawa, Kazuo Otani

***Abstract*—The route search problem is applied to various engineering fields. Many researchers study this problem. In this paper, we propose a new hybrid routing algorithm which combines Tabu search with Ant Colony Optimization. The proposed hybrid technique enables to find the shortest route including the blind alley. Experiments prove the effectiveness in comparison with conventional routing algorithm such as Dijkstra algorithm.**

***Index Terms*—Ant Colony Optimization, Tabu search, Routing algorithm, Blind alley**

## I. Introduction

The combinatorial optimization problem is applied to various engineering fields. There is the shortest path problem in one of such the combinatorial optimization problems. It can be classified into two categories. One is the problem for the searching on the grid as shown in Fig.1(a). The other is that for the searching on the graph as shown in Fig.1(b). Maze algorithm [1] which was proposed by Lee is the routing algorithm for the former category, and it is applied to VLSI CAD problems. On the other hand, Dijkstra algorithm [2] is the routing algorithm for the later category, and it is applied to route guidance problem such as car navigation system. These routing algorithms find the shortest path, whenever the path exists. However, they have the inherent processing time on the searching process.

Recently, many researchers study the meta-heuristics which can find the sub-optimal solution at short time. In these meta-heuristics, it is reported that Ant Colony Optimization (ACO) [3]-[6], which is inspired by feeding behavior of ants, shows the better capability than Genetic Algorithm (GA) [7]-[9] and Simulated Annealing (SA) [10] when it is applied to Traveling Salesman Problem (TSP).

In this paper, we propose a new routing algorithm using ACO. The proposed algorithm deals with the route searching problem represented by the graph representation. In the route searching problem, the moving destination has been limited unlike TSP. Therefore, there is a possibility that the ant agent is trapped in the blind alley. Thus, the proposed algorithm combines Tabu search algorithm with ACO to overcome this problem. Moreover, experiments compared with Dijkstra algorithm prove the validity of the proposed algorithm. This paper organizes as follows. Section 2 explains the mechanism of ACO and shows the related studies as preliminaries. Section 3 introduces the proposed algorithm and Section 4 shows experimental results. Finally, Section 5 summarizes and concludes this study.

This study was partially supported by the Ministry of Education , Science, Sports and culture in JAPAN. The authors would like to thank their supports.

Masaya Yoshikawa and Kaqzuo Otani are with Department of Information engineering, Faculty of Science and Engineering, Meijo University, Nagoya, JAPAN. (corresponding author to provide e-mail: evolution_algorithm@ yahoo.co.jp).

## II. Preliminaries

Ant Colony Optimization is the general name of the algorithm which is inspired by a behavior of feeding of ant. Almost all ACO algorithms are based on Ant System (AS) [11] which was proposed by Dorigo. Ant Colony System [3] is an algorithm which improved AS and it has better searching performance than AS. Therefore, we adopt ACS as a base algorithm. Hereafter, ACO indicates ACS in this paper. The searching on ACO utilizes two evaluations which consist of the static value and the dynamic one. The static evaluation is peculiar information of the target problem. Usually, a reciprocal number of the distance is adopted as the static evaluation value, when ACO is applied to the route searching problem such as TSP. On the other hand, the dynamic evaluation introduces pheromone amount.

Specifically, the optimization procedure of ACO is explained using an example of which ACO is applied to TSP.

First, the random number $q$ between from 0 to 1 is generated. Next, $q$ is compared with benchmark (parameter) $q_0$. When $q$ is smaller than $q_0$, the city that has the largest value of the product of the static evaluation and the dynamic one is selected as the next destination.

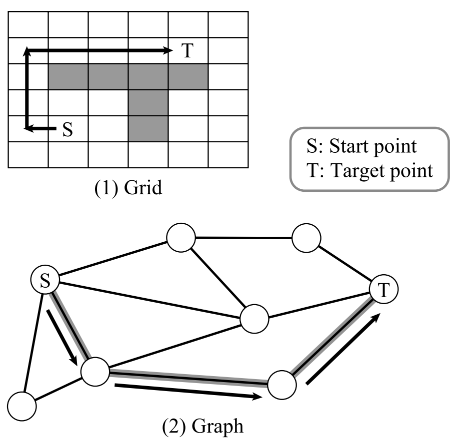


Fig.1 Example of the shortest path problem

ISBN: 978-988-18210-5-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Otherwise, ant $k$ in city $i$ selects the move to city $j$ according to probability $p^k$ and it is defined as follows.

$$p^k(i,j) = \frac{[\tau(i,j)][\eta(i,j)]^{\beta}}{\sum_{l \in n^k} [\tau(l,j)][\eta(l,j)]^{\beta}} \tag{1}$$

Where, $\tau(i,j)$ is a pheromone amount between city $i$ and city $j$, $\eta(i,j)$ is a reciprocal of the distance between city $i$ and city $j$, $\beta$ is a parameter which controls the balance between static evaluation value and dynamic one, and $n^k$ is a set of un-visit cities. Therefore, the selection probability is proportional to the product of the static evaluation and the dynamic one as shown in Fig.2.

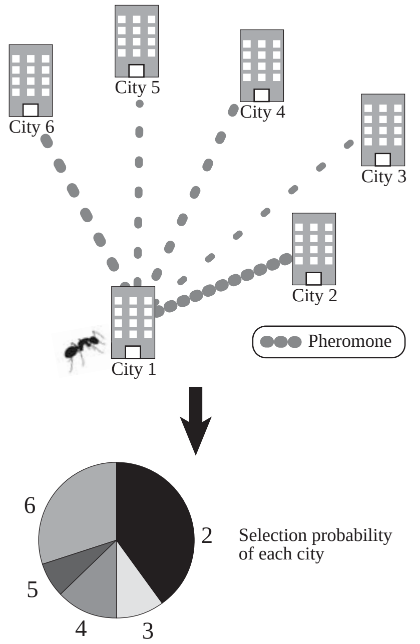


Fig.2 Example of the selection probability

Regarding the dynamic evaluation, a pheromone amount on each route between cities is calculated by using two pheromone update rules. One is local update rule and the other is global update rule. The local update rule is applied to the route which is selected by equation (1), and it is defined as follows.

$$\tau(i,j) \leftarrow (1-\psi)\tau(i,j) + \psi\tau_0 \tag{2}$$

Where, $\psi$ is a decay parameter in local update rule, $\tau 0$ is the initial value of pheromone. Thus, the local update rule adds the pheromone to the selected route, when the ant moves. The global update rule adds pheromone to the best route (the completed route) of all routes. The best route usually indicates the shortest route. The global update rule is defined as follows.

$$\tau(i,j) \leftarrow (1-\rho)\tau(i,j) + \rho\Delta\tau(i,j)$$

$$\Delta\tau(i,j) = \begin{cases} 1/L^+ & \text{if } (i,j) \in T^+ \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Where, $\rho$ is s a decay parameter in the global update rule, $T^+$ is the best route, and $L^+$ is the distance of the best route.

Examples of the routing algorithm using ACO have been reported by X.Fan *et al.*[12], Y.Zhang *et al.*[13], and others. X.Fan *et al.* proposed the hybrid ACO algorithm. They introduced the algorithm which combined dynamic local search with ACO, and it is applied to the ship pipe route design problem in three-dimensional space. Y.Zhang *et al.* proposed the two improvement techniques for ACO algorithm. They applied these techniques to the route optimization and simulation results show the effectiveness of the proposed technique.

However, no studies have ever seen, to our knowledge, the routing algorithm which is combined Tabu search with ACO.

## III. Hybrid Routing Algorithm

It differs from general TSP when ACO is applied to the route searching problem, and there are the following two problems: (1) the moving destination has been limited, and (2) the ant argent is trapped in the blind alley. Regarding the problem (1), the proposed algorithm solves it by separately handling the un-visit node, the visited node and the moving candidate node, respectively. In addition, regarding the blind alley of the problem (2), the proposed algorithm overcomes it by adopting Tabu search. Here, the situation of which the ant agent is trapped in the blind alley represents that the ant agent has not reached the final destination and no moving candidate node exists. In other words, the current node of which the ant agent stays has only the connections to the visited nodes. Specifically, the situation of which the ant agent is trapped in the blind alley is explained using Fig.3. In this example, the ant agent moved in order of a, b, c, d and e, and is trapped in the blind alley.

In the proposed algorithm, first of all, the ant agent returns to the previous node. That is, the ant agent moves to node d, and the node e is set on taboo. Thus, the node e is excluded by ant agent's moving destination candidate as shown in Fig.4.

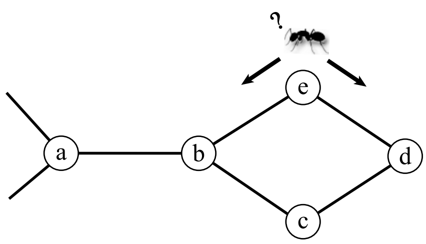


Fig.3 Example of the situation of which the ant agent is trapped in the blind alley

ISBN: 978-988-18210-5-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

However, the node d has no moving candidate node as well as the previous situation. In this case, the node d has only the visited node and the taboo node. Therefore, the ant agent returns to the previous node as well as the previous processing regarding the taboo operation as shown in Fig.5.

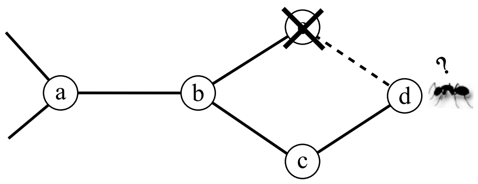


Fig.4 Example of the situation of which the node e is excluded by ant agent's moving destination candidate

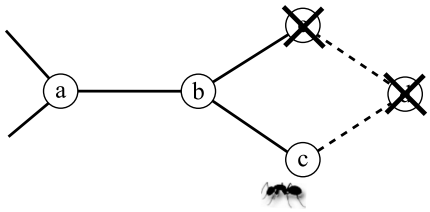


Fig.5 Example of the situation of which the ant agent returns to the previous node

These processing are repeated until the moving destination candidate's node appears. The concrete procedure is as follows.

*Step1:* Select the start node and the destination node (target node).

*Step2:* Select the node using equations (1), (2), and (3).

*Step3:* Move to the selected node and mark the current node as the visited node.

*Step4:* Taboo operation is executed when no moving candidate node exists.

*Step5:* Repeat from Step2 to Step4 until the ant agent reaches to the final destination.

*Step6:* Update the pheromone value.

*Step7:* Repeat from Step2 to Step6 until the generations are terminated.

Thus, the proposal algorithm enables ACO to be applied to the route searching problem including the blind alley, by combining Tabu search with ACO.

## IV. Experiments And Discussions

To evaluate the proposed algorithm, we conduct several experiments. First, we compare the proposed algorithm with Dijkstra algorithm to evaluate the searching performance. In the experiments, 8 kinds of start point and 8 kinds of the destination points are adopted. That is, these comparisons are conducted on 56 routes in total. Comparison results are shown in Figs. 6 and 7. The proposed algorithm found the optimal solution (found the shortest path) in the almost all cases as shown in comparison with Fig.7.

Next, we evaluate the escape performance of the proposed algorithm when it is trapped in the blind alley. Fig.8 shows the example of the graph for this experiment. Experimental result is shown in Fig.9. The proposed algorithm escaped form the blind alley which consists of node 2, node 5 and node 6.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 6 | 2 | 3 | 4 | 7 | 9 |
| 2 | 1 | 0 | 5 | 3 | 2 | 3 | 8 | 10 |
| 3 | 6 | 5 | 0 | 8 | 3 | 2 | 3 | 5 |
| 4 | 2 | 3 | 8 | 0 | 5 | 6 | 11 | 7 |
| 5 | 3 | 2 | 3 | 5 | 0 | 1 | 6 | 8 |
| 6 | 4 | 3 | 2 | 6 | 1 | 0 | 5 | 7 |
| 7 | 7 | 8 | 3 | 5 | 6 | 5 | 0 | 2 |
| 8 | 9 | 10 | 5 | 7 | 8 | 6 | 2 | 0 |

(1) Result of 1st trial

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 6 | 2 | 3 | 4 | 7 | 9 |
| 2 | 1 | 0 | 5 | 3 | 2 | 3 | 8 | 10 |
| 3 | 6 | 5 | 0 | 8 | 3 | 2 | 3 | 5 |
| 4 | 2 | 3 | 8 | 0 | 5 | 6 | 11 | 7 |
| 5 | 3 | 2 | 3 | 5 | 0 | 1 | 6 | 8 |
| 6 | 4 | 3 | 2 | 6 | 1 | 0 | 5 | 7 |
| 7 | 7 | 8 | 3 | 5 | 6 | 5 | 0 | 2 |
| 8 | 9 | 9 | 5 | 7 | 8 | 6 | 2 | 0 |

(2) Result of 2nd trial

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 6 | 2 | 3 | 4 | 7 | 9 |
| 2 | 1 | 0 | 5 | 3 | 2 | 3 | 8 | 10 |
| 3 | 6 | 5 | 0 | 8 | 3 | 2 | 3 | 5 |
| 4 | 2 | 3 | 8 | 0 | 5 | 6 | 5 | 7 |
| 5 | 3 | 2 | 3 | 5 | 0 | 1 | 6 | 8 |
| 6 | 4 | 3 | 2 | 6 | 1 | 0 | 5 | 7 |
| 7 | 7 | 8 | 3 | 5 | 6 | 5 | 0 | 2 |
| 8 | 9 | 10 | 5 | 7 | 7 | 6 | 2 | 0 |

(3) Result of 3rd trial

Fig.6 Results of the proposed algorithm

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 6 | 2 | 3 | 4 | 7 | 9 |
| 2 | 1 | 0 | 5 | 3 | 2 | 3 | 8 | 9 |
| 3 | 6 | 5 | 0 | 8 | 3 | 2 | 3 | 5 |
| 4 | 2 | 3 | 8 | 0 | 5 | 6 | 5 | 7 |
| 5 | 3 | 2 | 3 | 5 | 0 | 1 | 6 | 7 |
| 6 | 4 | 3 | 2 | 6 | 1 | 0 | 5 | 6 |
| 7 | 7 | 8 | 3 | 5 | 6 | 5 | 0 | 2 |
| 8 | 9 | 9 | 5 | 7 | 7 | 6 | 2 | 0 |

Fig.7 Result of the Dijkstra algorithm

ISBN: 978-988-18210-5-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

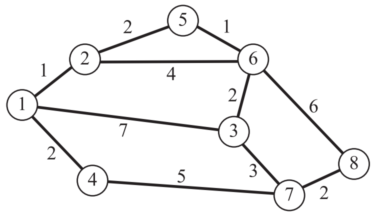


Fig.8 Example of the graph for the experiment of escape from the blind alley

1 –> 2 –> 6 –> 5 –> 6 –> 3 –> 7
(1) Case 1

1 –> 2 –> 5 –> 6 –> 3 –> 7 –> 8 –> 7 –> 4
(2) Case 2

Fig.9 Examples of escape from the blind alley

## V. Conclusion

In this paper, we proposed a new routing algorithm for the route search problem on the graph representation. The proposed algorithm combined Tabu search with ACO. This hybrid technique was able to find the shortest route when the blind alley existed in the map. Experiments proved the effective searching performance compared with Dijkstra algorithm.

Future work includes the experiments using large scale data. We will also apply the propose routing algorithm to the dynamic route search problem..

## References

[1] R.Hentschke, J.Narasimhan, M.Johann, R.Reis, "Maze Routing Steiner Trees With Delay Versus Wire Length Tradeoff", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol.17, Issue 8, pp.1073-1086, 2009.

[2] D.C.Lee, "Proof of a modified Dijkstra's algorithm for computing shortest bundle delay in networks with deterministically time-varying links", IEEE Communications Letters, Vol.10, Issue 10, pp.734-736, 2006.

[3] M.Dorigo, L.MGambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem", IEEE Transactions on Evolutionary Computation, vol.1, No.1, 1997, pp.53-66.

[4] M.Birattari, P.Pellegrini, M.Dorigo, "On the Invariance of Ant Colony Optimization", IEEE Transactions on Evolutionary Computation, Vol.11, Issue 6, pp.732-742, 2007.

[5] M.Yoshikawa, H.Terai, "Hardware Architecture of Pheromone-balance Aware Ant Colony Optimization", Proc. of International Conference on Genetic and Evolutionary Methods (GEM), pp.135-139, 2008.

[6] M.Yoshikawa, T.Nagura, "Adaptive Ant Colony Optimization Considering Intensification and Diversification", Proc. of International MultiConference of Engineers and Computer Scientists 2009, pp.200-203, (2009-3)

[7] Holland, J.: Adaptation in Natural Artificial Systems, the University of Michigan Press(Second edition; MIT Press), 1992.

[8] M.Yoshikawa, H.Yamauchi, H.Terai: Dedicated Hardware For Hybrid Evolutionary Computation, Ao, Sio-Iong; Huang, Xu; Wai, Ping-kong Alexander (eds.), Trends in Communication Technologies and Engineering Science, Springer, Netherlands, Chapter 12, pp.151-161, 2009

[9] M.Yoshikawa, H.Terai, "Car Navigation System based on Hybrid Genetic Algorithm", Proc. of World Congress on Computer Science and Information Engineering, pp.62-65, 2009

[10] Meijuan Gao, Jingwen Tian, "Path Planning for Mobile Robot Based on Improved Simulated Annealing Artificial Neural Network", Proc. of International Conference on Natural Computation, Vol.3, pp.8-12, 2007.

[11] M.Dorigo, V.Maniezzo, A.Colorni, "Ant system: optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol.26, Issue 1, pp.29-41, 1996.

[12] Xiaoning Fan, Yan Lin, Zhuoshang Ji, "The Ant Colony Optimization for Ship Pipe Route Design in 3D Space", Proc. of World Congress on Intelligent Control and Automation, Vol.1, pp.3103-3108, 2006.

[13] Yi Zhang, Zhi-li Pei, Jin-hui Yang, Yan-chun Liang, "An Improved Ant Colony Optimization Algorithm Based on Route Optimization and Its Applications in Travelling Salesman Problem", Proc. of IEEE International Conference on Bioinformatics and Bioengineering, pp.693-698, 2007.

ISBN: 978-988-18210-5-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)