



# Optimal Path Planning for Unmanned Vehicles Using Improved Ant Colony Optimization Algorithm

Hongchang Deng<sup>1</sup> and Jing Zhu<sup>1,2</sup>(✉)

<sup>1</sup> College of Automation Engineering, Nanjing University of Aeronautics and Astronautics,  
Jiangsu 211100, China

drzhujing@nuaa.edu.cn

<sup>2</sup> Institute of Systems Engineering, Macau University of Science and Technology, Macau, China

**Abstract.** To prevent the locally optimal problem and slow convergence problem of unmanned vehicles (UVs) path planning, an improved ant colony algorithm is proposed by using a dynamic pheromone volatility coefficient. The best path is searched by selecting the appropriate pheromone volatility coefficient in ant colony algorithm, which has better searching ability, and converges to the optimal value quickly. The experimental results are illustrated to compare with other improved ant colony optimization algorithms to verify the effectiveness and efficiency of our proposed path planning method for UVs.

**Keywords:** Path planning · Ant colony optimization · Unmanned vehicle

## 1 Introduction

Path planning refers to find the best path from the beginning to the end in a specific environment under the premise of optimizing one or several performance indicators, and in other cases, it also refers to find the best path to accomplish reconnaissance, search and other tasks. There are many applications of path planning, including the movement of robot in industry, the reconnaissance and search task of unmanned aerial vehicles (UAVs) and unmanned vehicles (UVs) in military field, the movement of scavenging robot in the smart home, and so on. The methods of path planning on UAVs and UVs are various. In [1], the receding-horizon path planning method is proposed to realize positioning and autonomous search functions of manned aircraft with the use of multi-step planning for systems with limited sensor footprints. Coverage path planning, an energy-aware path planning algorithm, is mentioned in [2], in which the path covering all target points or other requirements is searched with little energy consumed in UVs. The collaborative path planning algorithm for target tracking is developed in [3], which makes use of dynamic occupied grids, Bayesian filters, just name a few, to enable the tracking movement of UAVs and UVs in urban environments. Although path planning has been studied for a long time, there are still some problems. For example, scholars only consider geometric constraints but do not pay attention to the characteristics and

practical significance of UVs and UAVs. The convergence speed and optimization result in path planning will also affect the application degree of the algorithm. And swarm intelligence bionic algorithms have achieved good results in this respect.

Up to today, scholars have developed sorts of advanced path planning algorithms on the basis of traditional optimization algorithms, including A\* algorithm [4, 5], roadmap algorithm (RA) [6, 7], cell decomposition method (CD) [8, 9], artificial potential field method (APF) [10, 11], to name but a few. However, with the change of search environment, the expansion of search space, and the passage of time, the computational cost and the demand of storage space of the classical traditional path planning algorithm will increase geometrically. To this end, researchers proposed swarm intelligence optimization algorithms, including the ant colony optimization algorithm (ACO) [12], neural network [13, 14], genetic algorithm (GA) [15, 16], cuckoo algorithm [15, 17], particle swarm optimization (PSO) [16, 18], and artificial bee colony algorithm (ABC) [19, 20], etc. ACO is a heuristic random search algorithm proposed in the 1990s [12]. When an ant colony is searching for some food, the pheromone [21] on the path will affect the ant's choice of path, and eventually form the best path from the nest to the food. However, ACO also has some obstacles when applied to path planning. For example, a small pheromone volatilization coefficient will reduce the randomness of the algorithm's search, while a large one will reduce the convergence speed. In addition, the convergence rate of ACO is slow and local best results are easy to appear. Therefore, international scholars have also improved ACO for these problems. In [22], the authors used ACO for UAV path planning while also meeting the requirement of obstacle avoidance. But when the number of obstacles is too large or the complexity is relatively large, the performance of the algorithm proposed by [22] will decrease. In [23], the idea of fuzzy logic (FL) is applied to ACO, using the rank-based ant system and virtual path length to realize the path planning of UVs. However, the calculation time of this method needs to be further reduced. Liu Guoliang and others [24] used ACO to design a UAV location-assignment method in the problem of multi-UAV formation path planning, and then adopted a new strategy to select the next target node to find the globally best path. However, the improved ant colony algorithm in [24] has not been tested in other application environments. Green Ant (G-Ant) [25] not only considers the path length of the vehicle but also considers the energy consumed during the driving of the vehicle in the path planning of the unmanned ground vehicle (UGV). But the route found by the green ants is not necessarily the path with the shortest energy consumption. In [26], authors proposed and designed a dynamic viewable method based on the local environment model, a new rule of ant colony state transfer, and a reverse eccentric expansion method to improve ACO to realize the unmanned surface vehicle (USV) in the static position and dynamic state. Know the path planning in the environment to avoid collisions. In [27], ACO was used to draw a digital map of the drone's mission environment, and a mathematical model of the drone's horizontal and vertical flight trajectory was established to simulate the flight trajectory of the drone's mission. We can see from the above description that scholars have improved ACO through various means, allowing ACO to show better performance in path planning and be applied in more fields. However, the research of path planning algorithms is still in the stage of solving problems such as

convergence speed, local optimization, unmanned vehicle modeling, dynamic environment, and path planning in emergencies. In addition, It should also consider the actual performance of the research object to improve the practicality of the algorithm.

The variation of pheromone volatilization coefficient is rarely considered in ACO algorithms. An improved ant colony method is proposed to change the pheromone volatilization coefficient. As such, the convergence speed is enhanced and the local optimum phenomena is largely avoided. We set up the search space by the Cartesian coordinate system, other than the raster map. The pheromone volatility coefficient changes along with the iteration times. In the beginning of the search, we use a relatively large pheromone volatilization coefficient. Afterwards, in the middle and late stages of a search, the pheromone volatilization coefficient turns to be small, to improve the searching accuracy.

The rest of this paper is organized as follows. Section 2 introduces the classical ant colony optimization algorithm and its application in path planning. Section 3 addresses the main results of this paper, including task environment modeling, improvement of pheromone volatilization coefficient, and the flow of improving ant colony optimization algorithm. Section 4 exhibits the comparison of experimental simulation with other UV path planning methods. Finally, Sect. 5 concludes the whole paper.

## 2 Path Planning Using ACO Algorithm

### 2.1 Classical Ant Colony Optimization Algorithm

In the biological world, when ants search for food [12, 28, 29], they secrete pheromones along the path that they traveled, as such clues are left for the ants behind them. Therefore, after a period of time, through the evaporation and accumulation of pheromones, a path with the largest pheromones from the ant nest to the object will be formed, which is also the optimal path. Ant colony algorithm uses artificial ants to simulate this process. Each artificial ant is placed at the starting point, and then the artificial ant independently selects the next target point according to the pheromone residue, path and heuristic information after evaporation. At time  $t$ , the probability  $p_{ij}^k(t)$  of ant  $k$  moving from target  $i$  to target  $j$  is.

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{S \in J_k(i)} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & j \in J_k(i) \\ 0, & otherwise \end{cases} \quad (1)$$

where,  $\alpha$  and  $\beta$  represent the relative importance of pheromones and heuristic factors, respectively;  $\tau_{ij}$  is the amount of pheromone between target points  $i$  and  $j$ ;  $\eta_{ij}$  is the heuristic information, representing the expectation extent of ants from the target point  $i$  to  $j$ , and  $\eta_{ij} = 1/d_{ij}$ , where  $d_{ij}$  is the distance between  $i$  and  $j$ .  $J_k(i) = \{1, 2, \dots, n\}$  is the set of target points that ant  $k$  is allowed to choose in the next step;  $tabu_k$  records the current target point that ant  $k$  has passed. When the path cost from target  $i$  to target  $j$  decreases, the state transition probability of the road segment will increase. Therefore, when the ant chooses the next moving target, it will be more inclined to choose target  $j$ .

When all ants traverse  $n$  targets once, the pheromone quantity on each path should be updated according to (2).

$$\tau_{ij}(t + n) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij} \tag{2}$$

where,  $\rho$  represents pheromone volatility coefficient;  $\Delta\tau_{ij}$  represents the pheromone increment between  $i$  and  $j$  in this iteration, which can be obtained as.

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \tag{3}$$

where,  $\Delta\tau_{ij}^k$  represents the amount of pheromone left between  $i$  and  $j$  by the  $k$  ant in this iteration. If the ant does not pass through two points  $i$  and  $j$ ,  $t$  is equal to zero.  $\Delta\tau_{ij}^k$  can be expressed by.

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{when ant } k \text{ passes } i \text{ and } j \text{ in this iteration} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where,  $Q$  is the positive constant, and  $L_k$  represents the length of the path traveled by the  $k$  ant in this iteration.

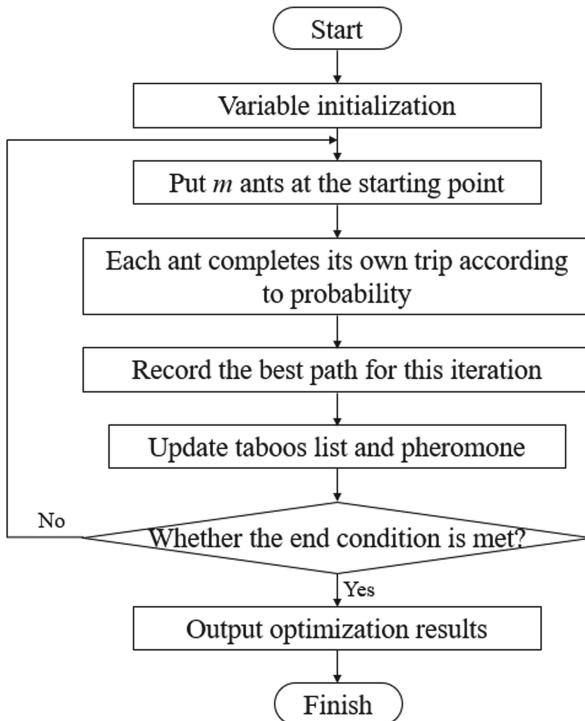


Fig. 1. Flow chart of basic ant colony optimization algorithm

The flow chart of the basic ACO is shown in Fig. 1. The process of path planning based on the basic ACO is briefly described as follows. In the initial time, the number

of search targets  $n$ , the number of ants  $m$ , the importance factor  $\alpha$  of pheromone, the importance factor  $\beta$  of heuristic information, the volatility coefficient  $\rho$ , pheromone slight  $Q$ , the initial iteration number  $iter$  and the maximum allowable iteration number  $iter_{max}$  are set. The target distance matrix, pheromone matrix, path distance matrix, optimal path recording matrix of each generation, and optimal path length recording vector of each generation are established. Then put the ants on the starting point of the driverless car. The ant chooses the next search target according to the target selection probability formula (1), and updates the ant taboo. When all the targets are visited and the ant returns to the starting position, the ant's search ends. Then the next ant searches until all the ants have finished the search. At this point, an iteration is completed, and the best path of the iteration is recorded. Then, according to formula (2) to formula (4), the pheromone on each path is updated, and the taboo list is cleared before the next iteration. The algorithm finds the best path before the end of iteration. So far, the basic ant colony algorithm has completed the whole optimization process.

### 2.2 Application of Basic Ant Colony Optimization Algorithms

ACO is essentially a parallel algorithm with a positive feedback mechanism and strong robustness. It has many applications, including traveling salesman problem (TSP), optimal tree problem, integer programming problem, general continuous optimization problem, vehicle routing problem (VRP), etc. Figure 2 shows the result of simple obstacle avoidance path planning for robots using the basic ant colony algorithm.

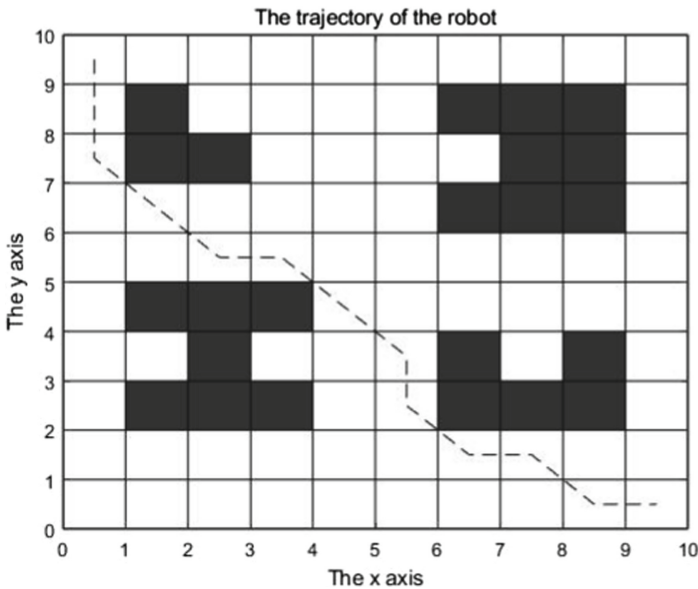


Fig. 2. Obstacle avoidance path diagram of mobile robot

In Fig. 2, the black area represents the obstacle, the white area represents the passable area, and the black dotted line refers to the moving trajectory of robot. In this simple

experiment, the robot mobile environment is constructed as a 10\*10 grid map. In the grid map, obstacles like “concave” or “L” will appear. This kind of obstacle is likely to lead artificial ants into a deadlock state, thus reducing the number of ants participating in the search and affecting the final search results. Therefore, we consider a completion method to solve the ant deadlock problem.

### 3 Path Planning Using Improved ACO Algorithm

#### 3.1 Task Environment Modeling

There are many kinds of search environment modeling in path planning, such as Cartesian coordinate system, raster map, probability path diagram, and so on. In the common raster maps, if ants encounter “concave” and “L” obstacles in the search process, ants are prone to the deadlock phenomenon, which affects the optimality of search results. Therefore, in view of the shortcomings of grid map, we use the Cartesian coordinate system to model the search environment, and represents the task points in the form of coordinates. The environment modeling is shown in Fig. 3.

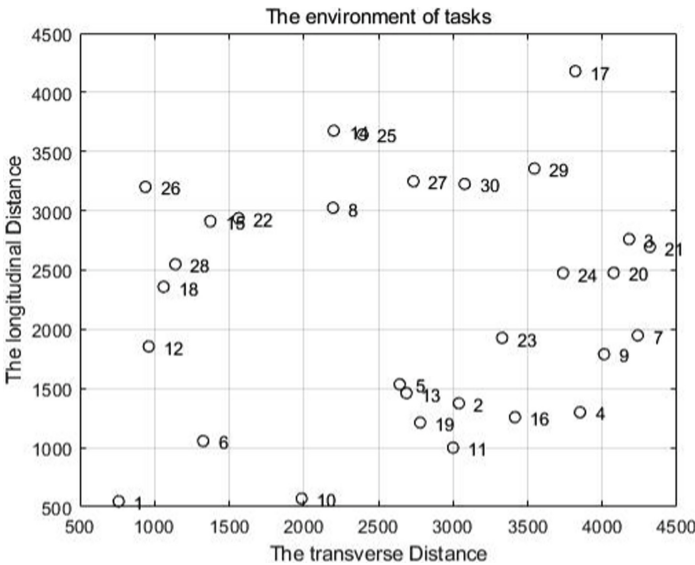


Fig. 3. Search environment modeling schematic

According to the task point coordinates to be searched, the task environment is constructed as a plane Cartesian coordinate system of  $(4500-500) * (4500-500)$ , as depicted in Fig. 3. The X-axis and Y-axis in the figure represent the transverse distances and longitudinal distances between any two task points respectively, in meters. The task point coordinates are composed of the two, and each task point is labeled, as shown in the black circle in Fig. 3. Treat the driverless car as a particle, search all mission points in the environment map and return to the starting point.

As for the deadlock phenomenon of ants in the grid map, we consider a fence method to solve this problem. In the process of driving, UVs often encounter obstacles of various shapes. We consider using straight line segments to enclose the obstacle into a polygon, as shown in Fig. 4. The black areas represent obstacles, and the black dotted line segments are straight line segments surrounding the obstacle. To avoid affecting the optimization results, we should make the area of the polygon as small as possible, and avoid “concave”-shaped and “L”-shaped edges. In this way, in the process of ant search, deadlock phenomenon can be effectively avoided, and the optimization accuracy is improved as well.

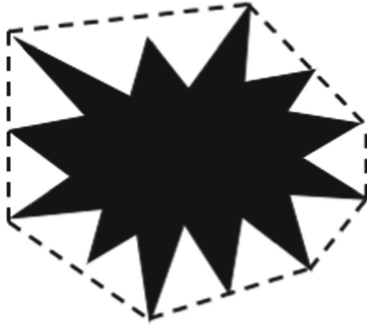


Fig. 4. Diagram of obstacle handling

### 3.2 Improvement of Pheromone Volatilization Coefficient

The pheromone volatilization coefficient of the classical ACO is a small constant. As such, when using the basic ACO for path search, the residual pheromone amount in the search after the initial pheromone volatilization is large. For the next iteration ants with a larger impact on target selection, they are more inclined to choose the path of the pheromone, consequently leading to the reduction of search range and search randomness. Thus, the locally optimal solution is made. To solve the issues of slow convergence and local optimal in classical ACO, we change the pheromone volatilization coefficient in this paper. The improved expression of the pheromone volatilization coefficient is.

$$\rho(iter) = \begin{cases} \left(1 - \frac{iter}{iter_{max}}\right)\rho(iter - 1), & \text{if } iter \left\lfloor \left[\frac{iter_{max}}{b}\right] \right\rfloor \text{ and } \rho > \rho_{min} \\ \rho_{min}, & \text{otherwise} \end{cases} \quad (5)$$

where  $iter$  represents the current number of iterations and  $iter_{max}$  represents the maximum number of iterations;  $\rho_{min}$  represents the minimum pheromone volatility coefficient. Parameter of  $b$  is an adjustable positive parameter with a value range of 3 to  $\sqrt{iter_{max}}$ , and the specific value of  $b$  is determined according to the maximum number of iterations.

In (5), the pheromone volatility coefficient varies with iterations. With the increase of iteration,  $\rho(iter)$  decreases from large to small until it decreases to the minimum value. In this improvement, the parameter of  $b$  is used to divide the whole iteration process into two parts. In the early part of the iteration,  $\rho(iter)$  varies with iterations; the second part is the middle and late part of the iteration, and the pheromone volatilization coefficient takes its minimum value. The value range of parameter  $b$  is determined according to the maximum number of iterations, and the maximum value is  $\sqrt{iter_{max}}$ . When the value of  $\frac{iter_{max}}{b}$  is not an integer, let  $\lfloor \frac{iter_{max}}{b} \rfloor$  be the integer that is less than or equal to  $\frac{iter_{max}}{b}$  and close to  $\frac{iter_{max}}{b}$ .

### 3.3 The Flow of Improved Ant Colony Algorithm

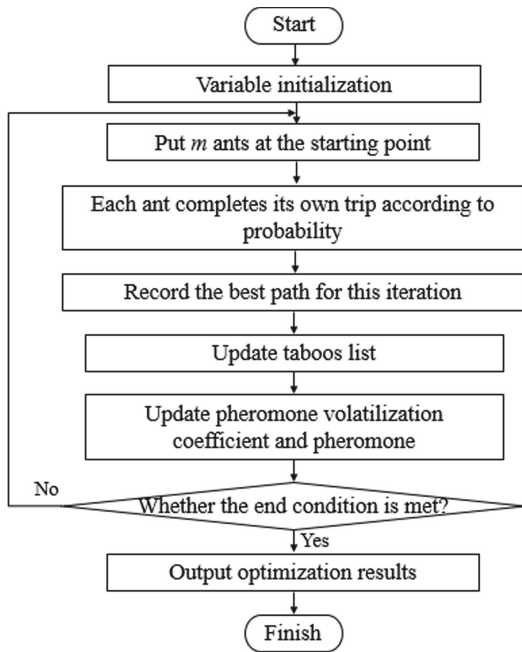


Fig. 5. The flow chart of improved ant colony optimization algorithm

The flow chart of the improved ant colony algorithm is shown in Fig. 5. The process of path planning realized by the improved ACO is briefly described as follows: at the initial time, the number of search targets  $n$ , the number of ants  $m$ , the importance factor of pheromone  $\alpha$ , the importance factor of heuristic information  $\beta$ , the minimum value of



pheromone volatility coefficient  $\rho_{min}$ , the initial value of pheromone volatility coefficient  $\rho$ , the light value of pheromone  $Q$ , the initial number of iterations  $iter$  and the maximum allowable number of iterations  $iter_{max}$  were set. The target distance matrix, pheromone matrix, path distance matrix, optimal path recording matrix of each generation, and optimal path length recording vector of each generation are established. Then put the ants on the starting point of the driverless car. The ant chooses the next search target according to the target selection probability formula (1), and updates the ant taboo. When all the targets are visited and the ant returns to the starting position, the ant's search ends. Then the next ant searches until all the ants have finished the search. At this point, an iteration is completed, and the best path of the iteration is recorded. Firstly, the pheromone fluctuation coefficient is updated according to formula (5), and then the pheromone on each path is updated to formula (4) according to the global pheromone update formula (2), and the next iteration is started after the tabu list is cleared. The algorithm finds the best path before the end of iteration.

## 4 The Experimental Results

To illustrate the effectiveness of the improved algorithm proposed in this paper and to improve the convergence speed of the algorithm, this section uses MATLAB software to conduct experimental verification. We compare it with the basic ACO and other improved ACO based on regulating pheromone volatility [30–33]. The basic idea of an adaptive ant colony algorithm is: after each iteration, the current optimal solution is obtained and retained. When the issue scale becomes large, because of the existence of  $\rho$ , the pheromones of paths that have never been searched gradually dwindle or even disappear. Thus, this will reduce the globality of the algorithm. When  $\rho$  is too large, the probability of the previously searched path being selected here is very high. And this will also affect the globality of the algorithm. Therefore, it is necessary to adaptively change the value of  $\rho$ . The adaptive formula is shown in (6).

$$\rho(t) = \begin{cases} 0.95\rho(t-1), & \text{If } 0.95\rho(t-1) \geq \rho_{min} \\ \rho_{min}, & \text{Otherwise} \end{cases} \quad (6)$$

To simplify the experiment, the analysis and experimental simulation are carried out based on a two-dimensional plane. The initial parameters of the algorithm are as follows: the number of search targets  $n = 30$ ; the number of ants  $m = 50$ ; pheromone importance factor  $\alpha = 1$ ; heuristic information importance factor  $\beta = 5$ ; pheromone intensity  $Q = 100$ ; the minimum value of pheromone volatilization coefficient  $\rho_{min} = 0.1$ . The maximum number of iterations allowed is  $iter_{max} = 100$ . Figures 6, 7 and 8 show the optimal path obtained by the basic ACO, improved ACO and adaptive ACO, respectively. Figure 9 shows the convergence curve comparison of these methods and

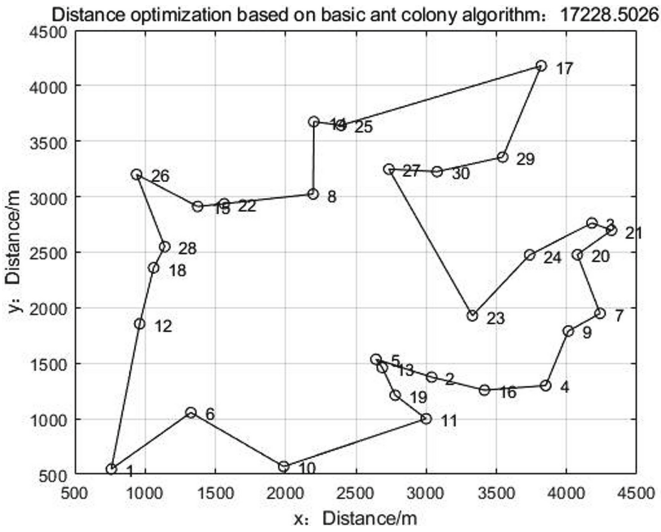


Fig. 6. Path results of basic ant colony optimization algorithm

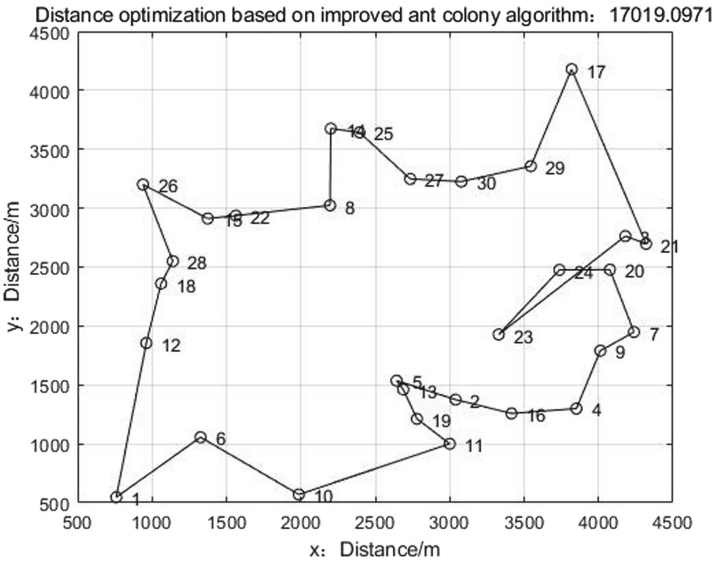


Fig. 7. Path results of improved ant colony optimization algorithm

those in [31–33]. It can be seen from Fig. 9 that the proposed improved ACO algorithm has the highest convergence speed among all. The path length comparisons are as shown in Table 1, from which we can find that our proposed result has the shortest path length, as well.

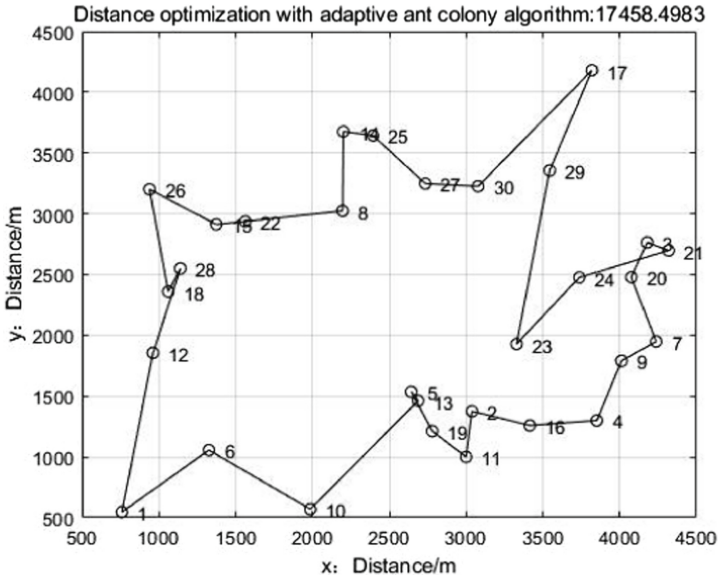


Fig. 8. Path results of adaptive ant colony optimization algorithm

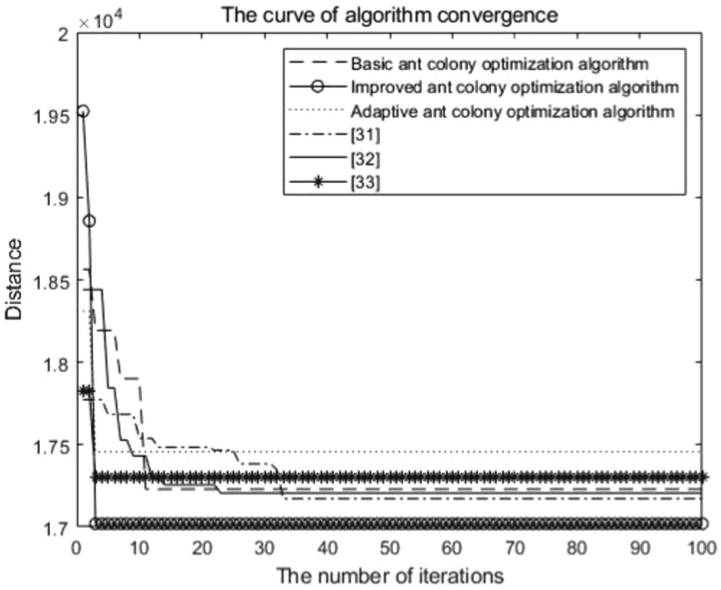


Fig. 9. Comparison diagram of algorithm convergence speed

**Table 1.** The path length of the experimental results

The algorithm name	The path length of the search results
Basic ant colony algorithm	17228.5206 m
<b>Improved ant colony algorithm</b>	<b>17019.0971 m</b>
Adaptive ant colony algorithm	17458.4983 m
Reference [31]	17171.1176 m
Reference [32]	17204.7807 m
Reference [33]	17301.4208 m

## 5 Conclusions

Ant colony algorithm is widely used in path planning, whereas, there still are unsolved problems, such as slow convergence speed, local optimization in real applications. To this end, this paper proposes an improved ACO algorithm on UV path planning with high convergence speed and global optimization ability by using a time-varying pheromone volatilization coefficient. The iterative process consists of two parts. In the beginning paragraph, the pheromone volatilization coefficient decreases from a large value along with iteration times. In the second part, the pheromone volatilization coefficient remains at a small value and gradually reduced. There is still a lot of room for improvement. In our next work, we shall consider the constraints of the actual working environment and the performance of UV itself to enhance the applicability of the ACO algorithm.

**Acknowledgements.** This work was fund by the Science and Technology Development Fund, Macau SAR (File no. 0050/2020/A1).

## References

1. Tisdale, J., Kim, Z., Hedrick, J.K.: Autonomous UAV path planning and estimation. *IEEE Robot. Autom. Mag.* **16**(2), 35–42 (2009)
2. Di Franco, C., Buttazzo, G.: Energy-aware coverage path planning of UAVs. In: 2015 IEEE International Conference on Autonomous Robot Systems and Competitions, pp. 111–117. IEEE, Vila Real (2015)
3. Yu, H., Meier, K., Argyle, M., Beard, R.W.: Cooperative path planning for target tracking in urban environments using unmanned air and ground vehicles. *IEEE/ASME Trans. Mechatron.* **20**(2), 541–552 (2015)
4. Yang, R., Cheng, L.: Path planning of restaurant service robot based on a-star algorithms with updated weights. In: 2019 12<sup>th</sup> International Symposium on Computational Intelligence and Design (ISCID), pp. 292–295. IEEE, Hangzhou (2019)
5. Zhang, Z., Tang, C., Li, Y.: Penetration path planning of stealthy UAV based on improved sparse a-star algorithm. In: 2020 IEEE 3<sup>rd</sup> International Conference on Electronic Information and Communication Technology (ICEICT), pp. 388–392. IEEE, Shenzhen (2020)

6. Cao, Y., Han, Y., Chen, J., Liu, X., Zhang, Z., Zhang, K.: A tractor formation coverage path planning method based on rotating calipers and probabilistic roadmaps algorithm. In: 2019 IEEE International Conference on Unmanned Systems and Artificial Intelligence (ICUSAI), pp. 125–130. IEEE, Xi'an (2019)
7. Ravankar, A.A., Ravankar, A., Emaru, T., Kobayashi, Y.: HPPRM: hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots. *IEEE Access* **8**, 221743–221766 (2020)
8. Gonzalez, R., Kloetzer, M., Mahulea, C.: Comparative study of trajectories resulted from cell decomposition path planning approaches. In: 2017 21<sup>th</sup> International Conference on System Theory, Control and Computing (ICSTCC), pp. 49–54. IEEE, Sinaia (2017)
9. Lupascu, M., Hustiu, S., Burlacu, A., Kloetzer, M.: Path planning for autonomous drones using 3D rectangular cuboid decomposition. In: 2019 23<sup>rd</sup> International Conference on System Theory, Control and Computing (ICSTCC), pp. 119–124. IEEE, Sinaia (2019)
10. Chen, M., Zhang, Q., Hou, L.: Improved artificial potential field method for dynamic target path planning in LBS. In: 2018 Chinese Control and Decision Conference (CCDC), pp. 2710–2714. IEEE, Shenyang (2018)
11. Chen, Z., Xu, B.: AGV path planning based on improved artificial potential field method. In: 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), pp. 32–37. IEEE, Shenyang (2021)
12. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. -Part B (Cybern.)* **26**(1), 29–41 (1996)
13. Luo, M., Hou, X., Yang, J.: Multi-robot one-target 3D path planning based on improved bioinspired neural network. In: 2019 16<sup>th</sup> International Computer Conference on Wavelet Active Media Technology and Information Processing, pp. 410–413. IEEE, Chengdu (2019)
14. Wang, J., Chi, W., Li, C., Wang, C., Meng, M.Q.-H.: Neural RRT\*: learning-based optimal path planning. *IEEE Trans. Autom. Sci. Eng.* **17**(4), 1748–1758 (2020)
15. Wang, J., Shang, X., Guo, T., Zhou, J., Jia, S., Wang, C.: Optimal path planning based on hybrid genetic-cuckoo search algorithm. In: 2019 6<sup>th</sup> International Conference on Systems and Informatics (ICSAI), pp. 165–169. IEEE, Shanghai (2019)
16. Tong, Y., Zhong, M., Li, J., Li, D., Wang, Y.: Research on intelligent welding robot path optimization based on GA and PSO algorithms. *IEEE Access* **6**, 65397–65404 (2018)
17. Wang, W., Tao, Q., Cao, Y., Wang, X., Zhang, X.: Robot time-optimal trajectory planning based on improved cuckoo search algorithm. *IEEE Access* **8**, 86923–86933 (2020)
18. Liu, X., Gu, Q., Yang, C.: Path planning of multi-cruise missile based on particle swarm optimization. In: 2019 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), pp. 910–912. IEEE, Beijing (2019)
19. Li, X., Huang, Y., Zhou, Y., Zhu, X.: Robot path planning using improved artificial bee colony algorithm. In: 2018 IEEE 3<sup>rd</sup> Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 603–607. IEEE, Chongqing (2018)
20. Tian, G., Zhang, L., Bai, X., Wang, B.: Real-time dynamic track planning of multi-UAV formation based on improved artificial bee colony algorithm. In: 2018 37<sup>th</sup> Chinese Control Conference (CCC), pp. 10055–10060. IEEE, Wuhan (2018)
21. Bonabeau, E., Dorigo, M., Theraulaz, G.: Inspiration for optimization from social insect behavior. *Nature* **406**(6), 39–42 (2000)
22. Chen, J., Ye, F., Jiang, T.: Path planning under obstacle-avoidance constraints based on ant colony optimization algorithm. In: 2017 IEEE 17<sup>th</sup> International Conference on Communication Technology (ICCT), pp. 1434–1438. IEEE, Chengdu (2017)
23. Song, Q., Zhao, Q., Wang, S., Liu, Q., Chen, X.: Dynamic path planning for unmanned vehicles based on fuzzy logic and improved ant colony optimization. *IEEE Access* **8**, 62107–62115 (2020)

24. Liu, G., Wang, X., Liu, B., Wei, C., Li, J.: Path planning for multi-rotors UAVs formation based on ant colony algorithm. In: 2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS), pp. 520–525. IEEE, Chongqing (2019)
25. Jabbarpour, M.R., Zarrabi, H., Jung, J.J., Kim, P.: A green ant-based method for path planning of unmanned ground vehicles. *IEEE Access* **5**, 1820–1832 (2017)
26. Wang, H., Guo, F., Yao, H., He, S., Xu, X.: Collision avoidance planning method of USV based on improved ant colony optimization algorithm. *IEEE Access* **7**, 52964–52975 (2019)
27. Li, Z., Han, R.: Unmanned aerial vehicle three-dimensional trajectory planning based on ant colony algorithm. In: 2018 37<sup>th</sup> Chinese Control Conference (CCC), pp. 9992–9995. IEEE, Wuhan (2018)
28. Kumar, P., Dwivedi, R., Tyagi, V.: Fuzzy ant colony optimization based energy efficient routing for mixed wireless sensor network. In: 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 1–7. IEEE, Ghaziabad (2019)
29. Khaled, A., Farid, K.: Mobile robot path planning using an improved ant colony optimization. *Int. J. Adv. Robot. Syst.* **15**(3), 1–7 (2018)
30. Gambardella, L.M., Dorigo, M.: Solving symmetric asymmetric TSPs by ant colonies. In: Proceedings of the IEEE Conference on Evolutionary Computation, pp. 622–627. IEEE, Nagoya (1996)
31. Liu, T., Yin, Y., Yang, X.: Research on logistics distribution routes optimization based on ACO. In: 2020 5<sup>th</sup> International Conference on Information Science, Computer Technology and Transportation (ISCTT), pp. 641–644. IEEE, Shenyang (2020)
32. Liu, Y., Hou, Z., Tan, Y., Liu, H., Song, C.: Research on multi-AGVs path planning and coordination mechanism. *IEEE Access* **8**, 213345–213356 (2020)
33. Li, J., Zhang, J.: Global path planning of unmanned boat based on improved ant colony algorithm. In: 2021 4<sup>th</sup> International Conference on Electron Device and Mechanical Engineering (ICEDME), pp. 176–179. IEEE, Guangzhou (2021)