

Research Article

Path Planning of Multirotor UAV Based on the Improved Ant Colony Algorithm

Duo Qi ^{1,2}, Zhihao Zhang ^{1,2} and Qirui Zhang ³

¹Aviation Swarm Technology and Operational Application Laboratory, Air Traffic Control and Navigation College, Air Force Engineering University, Xi'an, China

²Shaanxi Province Laboratory of Meta-Synthesis for Electronic and Information System, Xi'an, China

³Unit 93525 of PLA, Rikaze City, Xizang Province, China

Correspondence should be addressed to Zhihao Zhang; 408691252@qq.com

Received 29 November 2021; Revised 18 April 2022; Accepted 23 May 2022; Published 9 June 2022

Academic Editor: Ruthber Rodriguez Serrezuela

Copyright © 2022 Duo Qi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Path planning is an important part of the unmanned aerial vehicle (UAV) to realize its autonomous capabilities. Aiming at the shortcomings of the traditional ant colony algorithm-based trajectory planning method, which has slow convergence speed and easy to fall into the local optimum, a path planning method based on the improved ant colony algorithm is proposed. First, a dynamic adjusting factor is added into the heuristic function to improve the directivity of path selection and search speed. Then, the state transition strategy is improved to solve the problem of slow convergence in the initial stage and easy to fall into local optimum in the later stage. Finally, the path inflection point is smoothly optimized through the cubic B-spline curve. Simulation results show that the improved ant colony algorithm can quickly converge to the optimal path and well adapt to the flight requirements of multirotor UAV.

1. Introduction

With the continuous progress of science and technology, especially the continuous development of microelectronic technology and microelectromechanical technology, UAV has been widely used in military and civil fields. Compared with large UAVs, small UAVs have a broader application space in the fields of police, rescue and disaster relief, traffic management, news media, aerial photography, geological survey, environmental evaluation, pipeline inspection, and so on. Among them, the multirotor small UAV, as a typical representative, has developed particularly rapidly in recent years and received the focus of researchers.

Path planning is one of the key technologies for UAV to realize autonomous flight. A large number of scholars have done research on it and achieved a lot of results. In fact, path planning is to find an optimal flight path from initial point to target point under certain constraints. Common planning methods include classical algorithms such as the A^* (A -star) algorithm [1, 2], artificial potential field method [3, 4], and

optimal control method [5, 6], as well as intelligent algorithms such as the genetic algorithm [7, 8], ant colony algorithm [9, 10], and particle swarm optimization algorithm [11].

The classical algorithm is generally only suitable for simple spatial path planning. In the face of complex problems, the difficulty of solving them increases exponentially with the increase of dimension, which has certain limitations. With the development of computer and big data technology, intelligent algorithms have more and more obvious advantages of good robustness, positive feedback mechanism, and self-organization. Taking the ant colony algorithm as an example, theoretically, it can find the optimal solution of the path, which is suitable for path planning in complex environment. However, in practice, we must consider the problems of how to shorten search time, speed up convergence, and jump out of local optimal. Li et al. [12] introduced a weight factor into the probability transfer function and improved the updating way of information. The simulation results showed that the convergence speed of

the improved algorithm is improved, and the optimal path is shorter, but the search results are not stable. Wu et al. [13] adopted the method of integrating the current optimal track pheromone and the historical optimal track update, introduced the pheromone rollback clearing mechanism, and carried out offline calculation, which improved the search efficiency, but did not significantly reduce the number of iterations. Wang et al. [14] used the artificial potential field method to improve the heuristic function of the ant colony algorithm, allocated pheromones reasonably when the algorithm is not running, and improved the volatilization coefficient to find the optimal path of the algorithm.

It can be seen that many research studies are all based on the traditional ant colony algorithm and have achieved certain results. However, the traditional ant colony algorithm has the shortcomings of slow convergence speed and easy to fall into local optimum, which is not applicable to the real flight environment. Aiming at the defects of the basic ant colony algorithm, an improved ant colony algorithm is proposed in this study to make it suitable for path planning of multirotor UAV. First, a dynamic adjusting factor is added into the heuristic function to improve the directivity of path selection and search speed. Then, the state transition strategy is improved to solve the problem of slow convergence in the initial stage and easy to fall into local optimum in the later stage. At the same time, combined with the existing pheromone volatilization factor update strategy, the overall performance of the algorithm is improved. Finally, the path inflection point is smoothly optimized through the cubic B-spline curve, so that the path planning of multirotor UAV is more in line with the flight reality.

The study is organized as follows: Section 2 describes the environment in which the UAV performs its mission. Section 3 introduces the basic principle and deficiency of the ant colony algorithm. Section 4 improves the shortcomings of the ant colony algorithm and proposes the UAV track planning method and smoothing method. In Section 5, the proposed algorithm is simulated and analyzed, which verifies the effectiveness of the proposed algorithm. Finally, the full text work is summarized and the next research direction is pointed out.

2. Environmental Model

In order to accurately describe the path planning process of UAV, its two-dimensional environment map [15, 16] needs to be modeled. The conventional environment modeling methods include the grid method [17, 18], geometric information method, and view method. Considering that the grid method has the advantages of simple drawing, clear data structure, and easy implementation, we select the grid method to complete the UAV flight environment modeling. The grid method decomposes the plane map into a series of grids to facilitate the analysis of map information. The grid map is composed of 0 and 1 matrices with binary values. In the matrix, 0 stands for free grid, in which the UAV can enter and exit freely, represented by white grid, and 1 stands for obstacle grid, in which the UAV needs to move around, and represented by black grid. The grid environment and the

electronic map are shown in Figure 1. The map is numbered from left to right and top to bottom. The grid serial number corresponds to coordinates one by one, and the expression of the relationship between them is shown in equation (1). The result obtained is the center point of the grid.

$$\begin{aligned} x_i &= r \times \text{mod}(i, R) - 0.5, \\ y_i &= r \times (R + 0.5 - \text{ceil}(i, R)) - 0.5, \end{aligned} \quad (1)$$

where r is the grid size, i represents the number of the grid, R is the number of rows of the grid matrix, $\text{mod}()$ is the remainder operation function, and $\text{ceil}()$ is the rounding up operation function.

In order to prevent the collision between UAV and obstacles, when the obstacle is less than one grid, it is filled into a complete grid. At this time, the UAV is treated as a particle.

The flight rules of UAV are as follows: it can only appear in the white grid and cannot pass through the black grid, but it can pass through the four corners of the black grid; it is forbidden to repeat the same grid; and in unit time, it can only move between two adjacent grids. We use Euclidean distance to measure the length of the flight path. For example, if moving up, down, left, and right, the path length is 1 unit and if moving to the four corners of the grid, the path length is 1.41 units.

3. Basic Ant Colony Algorithm

Ant colony algorithms are inspired by the foraging behavior of ant populations, in which each ant in the colony leaves pheromones in its movement path to communicate information to the colony. The shorter the moving path is, the higher the pheromone concentration is, and the probability that the path is selected by ants, which forms the positive feedback mechanism for ants to find the shortest path. The basic principle of the ant colony algorithm is as follows.

Ants select paths according to pheromone concentration and heuristic function, and the probability of ant k moving from grid i to grid j can be expressed as

$$P_{ij}^k = \begin{cases} \left(\frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in A_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)} \right), & s \in A_k, \\ 0, & s \notin A_k, \end{cases} \quad (2)$$

where α is the pheromone heuristic factor and β is the distance expectation function factor, which affect the importance of pheromone and distance heuristic function, respectively, A_k indicates the next destinations ants can reach, $\tau_{ij}(t)$ is the pheromone concentration on the moving path at time t , and $\eta_{ij}(t)$ represents the distance heuristic function. Here,

$$\begin{aligned} \eta_{ij}(t) &= \frac{1}{d_{ij}}, \\ d_{ij} &= \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \end{aligned} \quad (3)$$

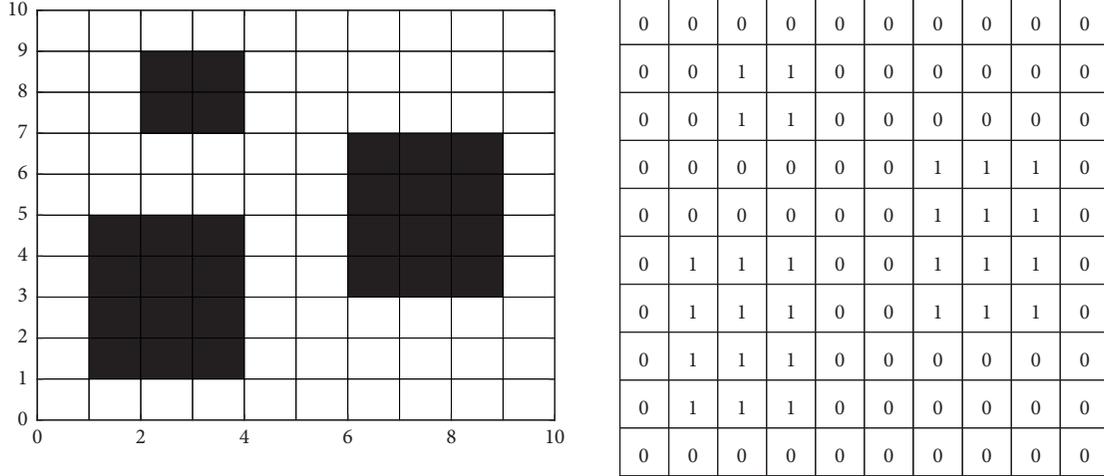


FIGURE 1: Grid map and electronic map.

Each ant will leave a certain amount of pheromone when moving. Therefore, when the algorithm iterates continuously, the pheromone content in the path gradually accumulates and volatilizes at the same time. When all ants in the population complete each round of iteration, the pheromone content on the path will be updated according to the following rules:

$$\begin{aligned}\tau_{ij}(t+1) &= (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t), \\ \tau_{ij}(t) &= \sum_{k=1}^m \Delta\tau_{ij}^k(t), \\ \Delta\tau_{ij}^k &= \begin{cases} (Q/L_k), \\ 0, \end{cases}\end{aligned}\quad (4)$$

where ρ is the pheromone volatilization coefficient, $\Delta\tau_{ij}$ represents the sum of pheromones released on the path between the two nodes, $\Delta\tau_{ij}^k$ denotes the pheromone increment, L_k is the path length of ant k , and Q is the pheromone enhancement coefficient.

4. The Improved Ant Colony Algorithm

4.1. The Improved Heuristic Function. When using the basic algorithm to plan the path, the ant colony has not left pheromone in the initial stage. At this time, the pheromone on the path is scarce. The ant cannot select the next grid according to the pheromone concentration. The search has no purpose and cannot quickly search the feasible path. On the basis of the heuristic function in [19], we add a dynamic regulation factor ξ . The new expressions of heuristic function and ξ are shown as

$$\begin{aligned}n_{ij} &= \frac{1}{(d_{ij} + d_{jE})^2} * \xi, \\ \xi &= \begin{cases} \exp(-5 * \text{iter}/\text{iter}_{\max}), & \text{iter} \neq \text{iter}_{\max}, \\ (1/\text{iter}_{\max}), & \text{iter} = \text{iter}_{\max}, \end{cases}\end{aligned}\quad (5)$$

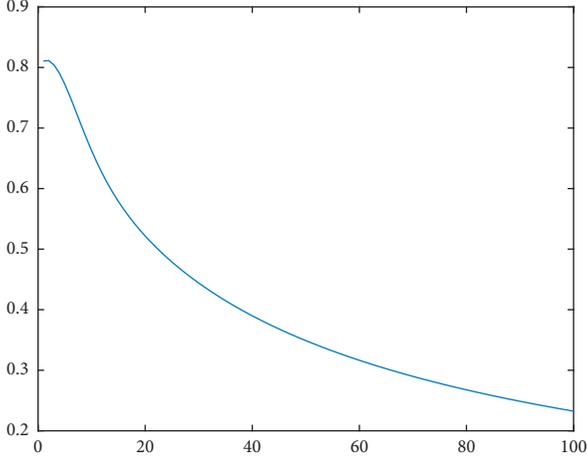
where d_{jE} is the Euclidean distance between j and the target point, iter is the current number of iterations, and iter_{\max} is maximum number of iterations.

4.2. The Improved State Transition Rule. In the basic ant colony algorithm, ants mainly rely on the probability function to select the next node. However, in the early stage of the algorithm, the difference between the pheromone concentrations of the path is very small, which cannot effectively guide the ant to choose the path. Based on the initial transition probability, the convergence speed of the algorithm is guaranteed by introducing pseudorandom rules. The ant will select the next node j according to the following formula.

$$P_{ij}^k(t) = \begin{cases} \max\{(\tau_{is})^\alpha * (\eta_{jE})^\beta\}, & q < q_0, \\ P_{ij}^k, & q \geq q_0, \end{cases}\quad (6)$$

where q represents a random number between (0, 1), and q_0 is a design parameter, which is usually determined by previous experience and repeated experiments. Its value determines the path selection mode. If the value of q_0 is too large, the path shift is more likely to be a deterministic pattern, and the convergence speed is faster, but the global search capability is reduced. On the contrary, if the value of q_0 is very small, the path shift is more inclined to the roulette mode, which not only increases the randomness of the global search but also reduces the convergence speed. Therefore, setting a reasonable value of q_0 has an important influence on the convergence process. In order to better improve the problem of slow convergence in the initial stage and easy to fall into local optimization in the later stage, we designed it as a dynamical parameter adjusted with the number of iterations. The details expression is as follows:

$$q_0 = 0.25 * \left(\exp\left(-\frac{1}{\text{iter}}\right)\right)^{-1}.\quad (7)$$

FIGURE 2: Changing curve of ρ .

4.3. Pheromone Volatilization Factor Update Strategy. Due to the particularity of path planning, different volatilization coefficients are required in different stages: if ρ is too large, ants cannot complete path search according to the pheromone information, resulting in slow convergence speed; if ρ is too small, the pheromone will accumulate excessively, easily make the path search fall into local optimization. Referring to the research ideas of [19], the adaptive updating strategies of pheromone volatile factors are designed as follows:

$$\rho(t+1) = \frac{\text{iter}_{\max}}{\text{iter}_{\max} + \text{iter}} * \frac{1}{\exp(1 - \rho(t))}. \quad (8)$$

In order to search as many paths as possible, the pheromone content should be controlled at a low level at the initial stage of algorithm optimization. Therefore, the value of the initial volatile factor ρ should not be too small. At this time, pheromone concentration has little interference in the process of path exploration, and ants can search more possible paths. With algorithm iteration step by step, the value of ρ gradually decreases, and the negative feedback effect is weakened. The pheromone content on the moving path is increased, and the guiding effect of pheromone concentration on ant colony is enhanced. When the number of iterations increases to a certain number, ant colony will converge to the path with a higher pheromone content. Therefore, the improved algorithm can expand the searching range of ant colony and improve the convergence speed. The changing curve of ρ is shown in Figure 2 (initial $\rho = 0.8$).

4.4. Path Smoothing Method. The optimal path generated by the improved ant colony algorithm is not smooth enough and has sharp inflection point. Therefore, cubic B -spline smoothing is introduced to optimize the path at the inflection point. The total equation of the B -spline curve is [20, 21]

$$P(t) = \sum_{i=0}^n P_i F_{i,k}(t). \quad (9)$$

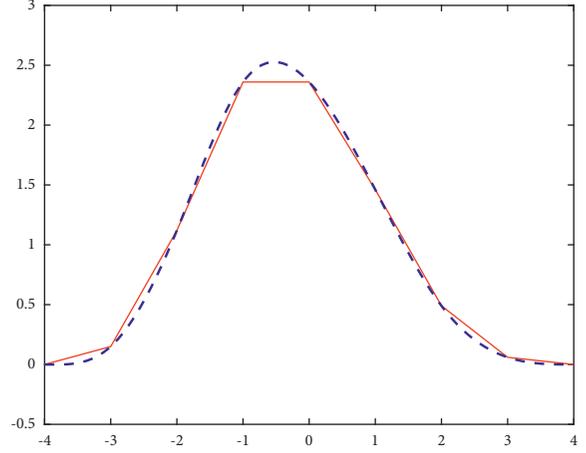


FIGURE 3: Simulation of local path smoothing results.

The basis function in cubic B -spline curve equation is

$$F_{i,k}(t) = \frac{1}{k!} \sum_{m=0}^{k-i} (-1)^m \binom{m}{k+1} (t+k-m-j)^k. \quad (10)$$

Furthermore, the mathematical expression of the basis function of cubic B -spline curve is

$$\left\{ \begin{array}{l} F_{0,3}(t) = \frac{1}{6}(1-t)^3, \\ F_{1,3}(t) = \frac{1}{6}(3t^3 - 6t^2 + 4), \\ F_{2,3}(t) = \frac{1}{6}(-3t^3 + 3t^2 - 3t + 1), \\ F_{3,3}(t) = \frac{1}{6}t^3. \end{array} \right. \quad (11)$$

The cubic B -spline curve equation can be converted by bringing the basis function into the general equation:

$$P(t) = P_0 * F_{0,3}(t) + P_1 * F_{1,3}(t) + P_2 * F_{2,3}(t) + P_3 * F_{3,3}(t). \quad (12)$$

Based on the above cubic B -spline curve equation, the flight path can be smoothed. Figure 3 shows the simulation of local path smoothing results. The red line is the path to be smoothed, and the blue dotted line is the smoothing result. The goal of this strategy is to replace the broken line with a curve near the inflection point, so that the obtained path is smoother and more in line with the reality of UAV flight.

The flow of the improved ant colony algorithm is shown in Figure 4.

5. Simulation Analysis

The simulation hardware environment is the Huawei MateBook Pro laptop equipped with Intel Core I7-8565U processor. Software environment is the Win10 operating

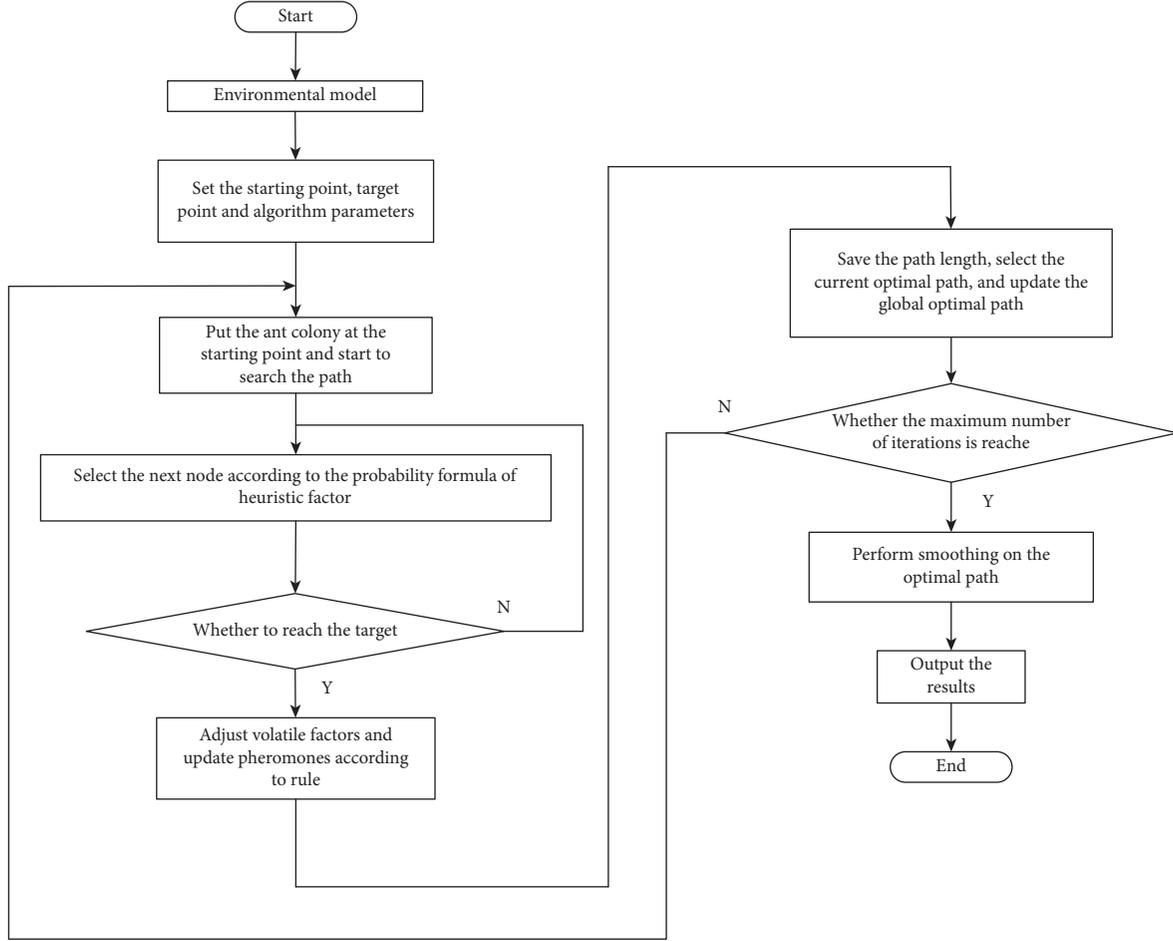


FIGURE 4: Improved ant colony algorithm flow.

TABLE 1: Basic parameter settings of BACA and IACA.

Parameter name	Number of ants (m)	Pheromone heuristic factor (α)	Expectation function factor (β)	Strength of the pheromone (Q)
Value	50	1	7	1

TABLE 2: Basic parameter settings of GA.

Parameter name	Number of swarms	Crossover probability	Variation probability	Number of terminative iterations
Value	50	0.6	0.001	100

system, installed with MATLAB 2017B simulation test environment. BACA, GA (genetic algorithm), and IACA algorithms are implemented in the simulation tests. The basic parameter settings of three algorithms are given in Tables 1 and 2.

5.1. Performance Test. In order to verify the optimization ability of the improved ant colony algorithm in this study, first, the typical single-mode test function Sphere and multimode test function Alpine are used for the optimization comparison test. The dimension value of the two functions is 30, the value range of Sphere is $[-100, 100]$, the value range of Alpine is $[-10, 10]$, and the theoretically

achievable minimum value of both functions is 0. The two function expressions are as follows, respectively:

$$\begin{aligned}
 f_1(x) &= \sum_{i=1}^n x_i^2, \\
 f_2(x) &= \sum_{i=1}^n |x_i \sin(x_i) + 0.1x_i|.
 \end{aligned} \tag{13}$$

The basic ant colony algorithm (BACA) and the improved ant colony algorithm (IACA) are independently run for 20 times on each test function, and the common parameters of the algorithm remained consistent. The results are given in Table 3.

TABLE 3: Comparison of test functions for optimization results.

Function name	Algorithm name	Average value	Worst value	Best value
Sphere	BACA	$2.43E-6$	$2.98E-5$	$7.92E-14$
	IACA	$7.64E-82$	$6.49E-69$	0
Alpine	BACA	$5.43E-2$	$1.21E-1$	$9.84E-3$
	IACA	$6.99E-79$	$2.14E-66$	0

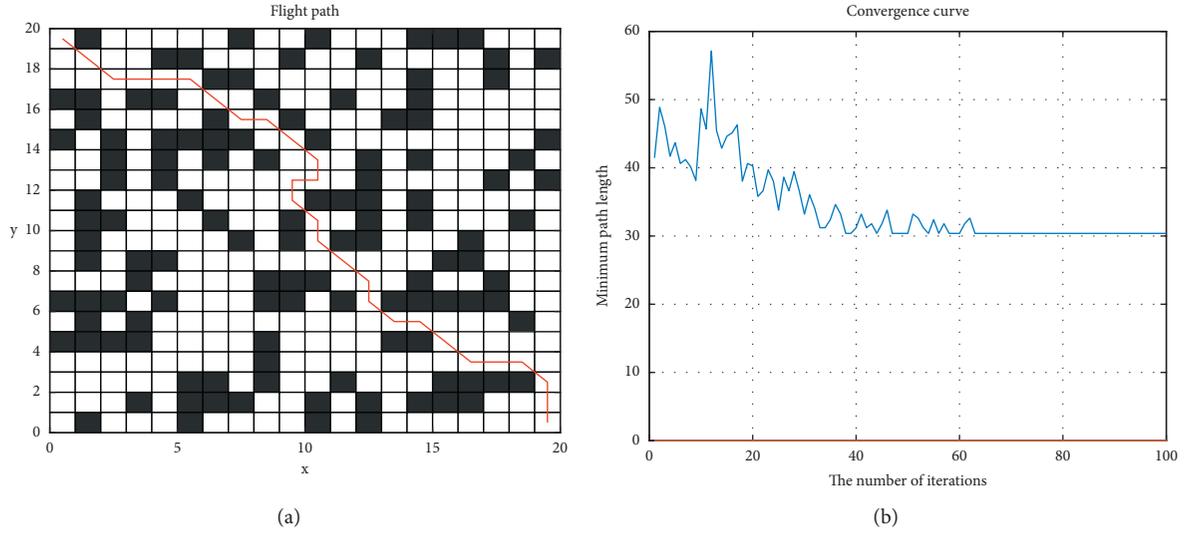


FIGURE 5: Path planning of BACA. (a) Optimal path. (b) Convergence curve.

TABLE 4: Comparison tests of 20×20 simulation environment.

Algorithm name	Length of the path	Time consuming	Number of path grid	Number of convergent iterations
BACA	36.04	4.62	32	62
GA	39.45	4.65	30	70
IACA	30.47	3.97	26	46

TABLE 5: Comparison tests of 30×30 simulation environment.

Algorithm name	Length of the path	Time consuming	Number of path grid	Number of convergent iterations
BACA	53.19	64.15	43	211
GA	51.94	62.31	46	161
IACA	45.62	58.59	38	170

According to the test results in the above table, the IACA algorithm proposed in this study has a good optimization effect. The theoretical global optimal value can be obtained by searching for Sphere function and Alpine function.

5.2. Comparison Test of Path Planning. In order to verify the effectiveness of IACA in UAV path planning, BACA and GA (genetic algorithm) are used as a comparison to conduct simulation comparison tests in 20×20 and 30×30 environment maps.

5.2.1. 20×20 Simulation Environment. The simulation test is first carried out in a 20×20 grid map, with the starting point at the upper left corner of the map and the target point at the lower right corner. BACA, GA, and IACA proposed in this

study are used to plan the flight path of UAV, respectively, with the maximum iteration times of 100. Path planning diagram and iterative convergence curve are shown in Figures 5–7. The experimental results are given in Table 4.

It can be seen from the above table that the shortest path length obtained by applying the IACA algorithm proposed in this study is 15.4% shorter than that obtained by applying the BACA algorithm, and the running time is reduced by 14.1%. Compared with GA, IACA is 22.8% shorter, and the running time is reduced by 14.6%.

5.2.2. 30×30 Simulation Environment. When the flight environment is large and obstacles are densely distributed, the 100 iteration times of 20×20 map cannot be effectively converged to the shortest path. We increase the maximum

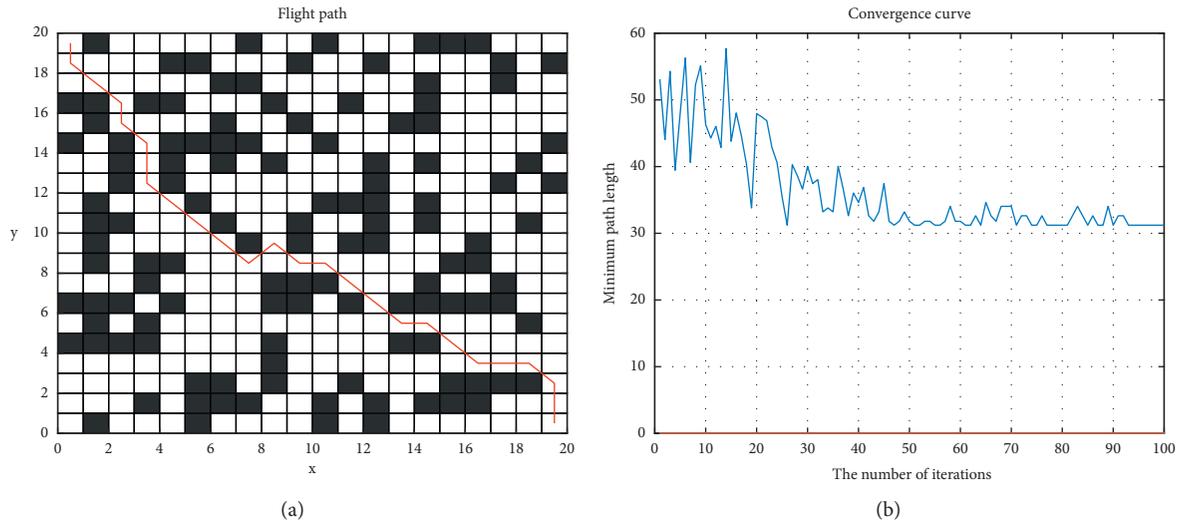


FIGURE 6: Path planning of GA. (a) Optimal path. (b) Convergence curve.

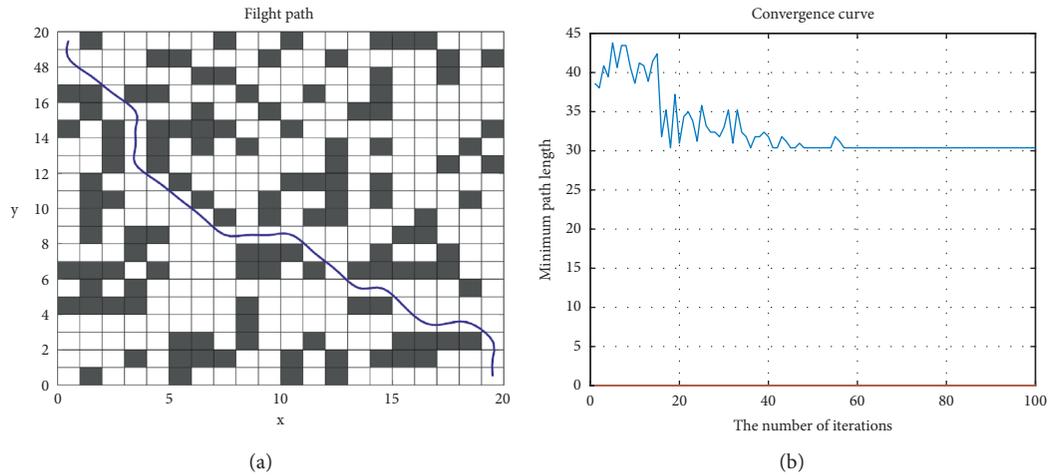


FIGURE 7: Path planning of IACA. (a) Optimal path. (b) Convergence curve.

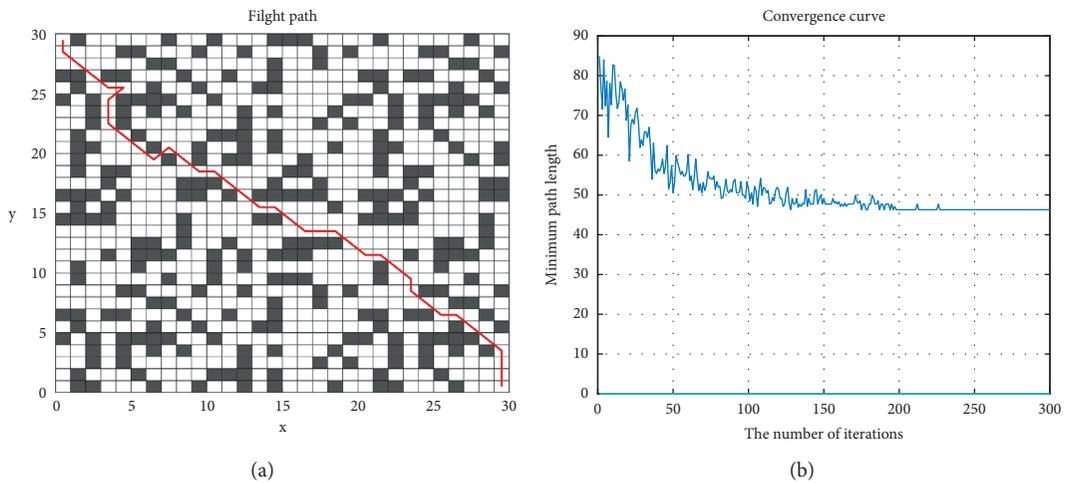


FIGURE 8: Path planning of BACA. (a) Optimal path. (b) Convergence curve.

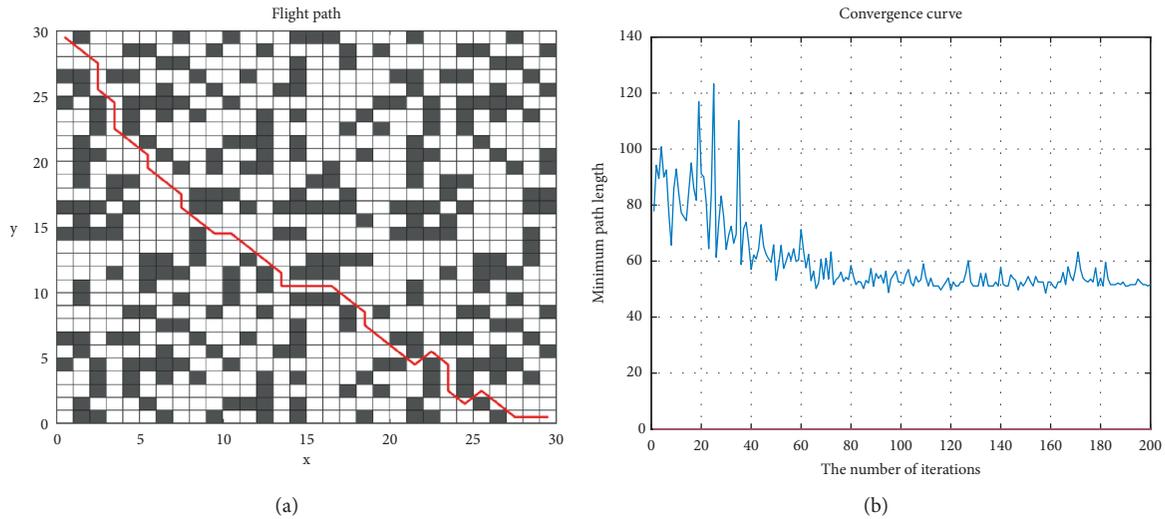


FIGURE 9: Path planning of GA. (a) Optimal path. (b) Convergence curve.

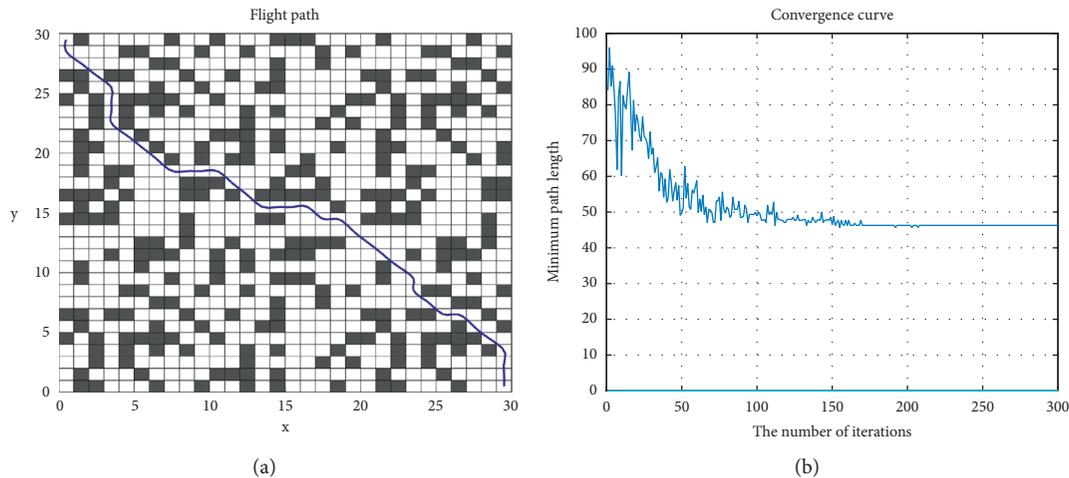


FIGURE 10: Path planning of IACA. (a) Optimal path. (b) Convergence curve.

iteration times to 300. Path planning diagram and iterative convergence curve are shown in Figures 8–10. As given in Table 5, the shortest path obtained by IACA is 14.2% shorter than that obtained by BACA, and the system running time is reduced by 8.7%, respectively. Compared with GA, though the system running time is added by 6.4%, the shortest path obtained by IACA is 13.8% shorter than that obtained by GA.

6. Conclusion

In this study, we proposed an improved ant colony algorithm for multirotor UAV path planning. By redesigning the heuristic function of the ant colony algorithm and improving the state transition matrix, the running time of the algorithm and the distance of the optimal moving path are effectively shortened. At last, we smoothed the final moving path to make it more in line with the flight reality of UAV.

Through the comparison of MATLAB simulation experiments, it can be seen that the improved algorithm has more advantages than the basic algorithm in the optimization effect and reflects the effectiveness of the algorithm in the operation time and the shortest path.

In practical applications, UAVs usually perform tasks in a swarm. Therefore, how to complete the coordinated trajectory planning of UAV swarms on the basis of this study will become the focus of our next research.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] T. Zhang, Q. Xiang, W. Zheng, Y. Sun, and X. Zhou, "Application of path planning based on improved A-star algorithm in war gaming of naval warfare," 2021, <https://kns.cnki.net/kcms/detail/11.2176.TJ.20211008.1542.002.html>.
- [2] B. Wang, D. Hu, and Y. Jiang, "Application of improved A* algorithm in path planning," *Computer Engineering and Applications*, vol. 57, no. 10, pp. 243–247, 2021.
- [3] Y. Wu, "A survey on population-based meta-heuristic algorithms for motion planning of aircraft," *Swarm and Evolutionary Computation*, vol. 62, no. 6, Article ID 100844, 2021.
- [4] S. Wang, J. Zhang, and J. Zhang, "Intelligent vehicles formation control based on artificial potential and virtual leader," *Journal of Shanghai Jiao Tong University*, vol. 54, no. 3, pp. 305–311, 2020.
- [5] W. Wu, L. Yang, W. Liu, W. Guo-Hong, and M. Wan-Jing, "Automatic parking path planning based on pseudo-spectral splicing method," *Acta Automatica Sinica*, vol. 46, no. 9, pp. 1971–1985, 2020.
- [6] G. Neng, Z. Kegang, P. Feng, G. Quancheng, and L. Sufen, "A novel autonomous vehicle trajectory planning and control model for connected-and-autonomous intersections," *Journal of Chongqing University of Technology (Natural Science)*, vol. 33, no. 4, pp. 26–31, 2019.
- [7] W. Tong and L. Chen, "Path planning for mobile robot based on improved genetic algorithm," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 46, no. 4, pp. 704–711, 2020.
- [8] X. Liang and J. Zhao, "A hybrid algorithm of artificial fish swarm and genetic algorithm and its application in collision avoidance of unmanned surface vessels," *Computer Engineering and Science*, vol. 41, no. 5, pp. 943–947, 2019.
- [9] M. Jiang, F. Wang, G. E. Yuan, and L. Sun, "Research on path planning of mobile robot based on improved ant colony algorithm," *Chinese Journal of Scientific Instrument*, vol. 40, no. 2, pp. 114–121, 2019.
- [10] D. Zhang, W. Chen, H. Zhang, and Y. Su, "Patrol path planning of unmanned surface vehicle based on A* algorithm and ant colony algorithm," *Journal of Huazhong University of Science and Technology*, vol. 48, no. 6, pp. 14–18, 2020.
- [11] Y. Zhang, Y. Wang, S. Li, and X. Wang, "Global path planning for AUV based on charts and the improved particle swarm optimization algorithm," *Robot*, vol. 42, no. 1, pp. 121–128, 2020.
- [12] Z. Li, Y. Huang, and Y. Xu, "Path planning of mobile robot based on improved variable step size ant colony algorithm," *Journal of Electronic Measurement and Instrumentation*, vol. 34, no. 8, pp. 16–21, 2020.
- [13] Y. Wu and C. Wang, "An improved three-dimensional path planning method of mobile robot," *Journal of South China University of Technology*, vol. 44, no. 9, pp. 54–60, 2016.
- [14] X. Wang, L. Yang, Y. Zhang, and S. Meng, "Robot path planning based on improved ant colony algorithm with potential field heuristic," *Control and Decision*, vol. 33, no. 10, pp. 1776–1781, 2018.
- [15] F. Causa, G. Fasano, and M. Grassi, "GNSS-aware path planning for UAV swarm in complex environments," in *Proceedings of the IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, IEEE, Piscataway, NJ, USA, June 2019.
- [16] W. H. Liu, X. Zheng, and Z. H. Deng, "Dynamic collision avoidance for cooperative fixed-wing UAV swarm based on normalized artificial potential field optimization," *Journal of Central South University*, vol. 28, no. 10, pp. 3159–3172, 2021.
- [17] Y. Wu and K. H. Low, "An adaptive path replanning method for coordinated operations of drone in dynamic urban environments," *IEEE Systems Journal*, vol. 15, no. 3, pp. 4600–4611, 2021.
- [18] Y. Wu, K. H. Low, B. Pang, and Q. Tan, "Swarm-based 4D path planning for drone operations in urban environments," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7464–7479, 2021.
- [19] J. Zhao, Y. Tang, G. Jiang, F. Xu, and J. Ding, "Mobile robot path planning based on improved ant colony algorithm," *Journal of Nanjing University of Posts and Telecommunications*, vol. 39, no. 6, pp. 73–78, 2019.
- [20] T. Zhang, B. Wu, and F. Zhou, "Research on improved ant colony algorithm for robot global path planning," *Computer Engineering and Applications*, vol. 58, 2021.
- [21] H. Chen, J. Fei, L. Yang, L. Hua, and X. Liu, "Smooth path planning method based on dynamic feedback A* ant colony algorithm," *Journal of Agricultural Machinery*, vol. 48, no. 4, pp. 34–102, 2016.