Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 19(1): 225–252.
DOI: 10.3934/mbe.2022012
Received: 26 August 2021
Accepted: 21 October 2021
Published: 09 November 2021

Research article

LF-ACO: an effective formation path planning for multi-mobile robot

Liwei Yang*, Lixia Fu*, Ping Li, Jianlin Mao, Ning Guo and Linghao Du

School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650093, China

* **Correspondence:** Email: 18916336783ylw@gmail.com, 905771625@qq.com.

Abstract: Multi-robot path planning is a hot problem in the field of robotics. Compared with single-robot path planning, complex problems such as obstacle avoidance and mutual collaboration need to be considered. This paper proposes an efficient leader follower-ant colony optimization (LF-ACO) to solve the collaborative path planning problem. Firstly, a new Multi-factor heuristic functor is proposed, the distance factor heuristic function and the smoothing factor heuristic function. This improves the convergence speed of the algorithm and enhances the smoothness of the initial path. The leader-follower structure is reconstructed for the position constraint problem of multi-robots in a grid environment. Then, the pheromone of the leader ant and the follower ants are used in the pheromone update rule of the ACO to improve the search quality of the formation path. To improve the global search capability, a max-min ant strategy is used. Finally, the path is optimized by the turning point optimization algorithm and dynamic cut-point method to improve path quality further. The simulation and experimental results based on MATLAB and ROS show that the proposed method can successfully solve the path planning and formation problem.

Keywords: multi-robot; leader follower-ant colony algorithm (LF-ACO); formation path planning; dynamic tangent point method

1. Introduction

Cooperative group control of multiple mobile robots is an essential branch of robotic systems. In recent years, multi-robot systems that can perform complex tasks faster and better than individual robots，have been used in practical scenarios such as logistics and transportation [1], collision rescue [2], precise positioning [3], and mobile sensing networks [4]. In these applications, robots are typically required to satisfy formations or other constraints to accomplish complex tasks. Compared to a single robot, a multi-robot system provides adequate access to environmental information and improves the ability and role efficiency of the robot to complete the job. When multi-robot systems need to accomplish spatially distributed tasks [5], such as target search [6], their path planning [7] in complex environments is critical.

With increased environmental complexity and robot task difficulty, multi-robot formation collaboration has received extensive research and attention in recent years. Studying an effective formation path planning method can help solve the multi-robot obstacle avoidance and navigation problems. The research of multi-robot path planning based on formation control is a further extension of single-robot path planning. When the robots move in an environment full of obstacles, some specific evaluation metrics [8], such as path length, planning time, and energy consumption, must be considered and converged to a specific formation [9]. However, the problem of positional constraints between robots [10] and cooperative obstacle avoidance [11] exacerbates the difficulty of planning.

For the study of global path planning of robots, good results have been achieved by using intelligent optimization algorithms such as the sparrow search algorithm [12], the whale optimization algorithm [13], and the ant colony algorithm [14–20]. The ant colony algorithm was proposed by Italian scholar Dorigo [14]. It has strong robustness and better solving ability, but has disadvantages such as low search efficiency and ease of falling into local optimality [15]. Many scholars have conducted in-depth studies on the structural design, operation mechanism, and parameter optimization of the ant colony algorithm and proposed many improvement strategies in response to these problems. Liu et al. [16] combined the ant colony algorithm with geometric optimization to optimize the cross-path nodes generated during the pathfinding process, enabling subsequent pheromone updates to improve the algorithm's path quality and efficiency. You et al. [17] designed a new heuristic operator to improve the diversity and convergence of the population search. Dai et al. [18] improved the ant colony algorithm based on the A* algorithm and the maximum-minimum ant system to speed up global convergence speed and path smoothing, and introduced a retraction mechanism to solve the deadlock problem. Jiao et al. [19] used an adaptive state transfer strategy and an adaptive pheromone update strategy to ensure the relative importance of pheromone strength and heuristic information in the iterative process of the algorithm, which improved to some extent the adaptability of the algorithm for different environments and the ability to jump out of the optimal local solution. Khaled et al. [20] improved the state transfer formula to preferentially select the neighbor node with more exits as the next node, and by segmenting the multi heuristic function, and by rewarding and punishing the optimal worst path separately, the improved algorithm enhances the diversity of the search and attenuates the influence of invalid pheromones.

Theoretically, paths based on optimization algorithms can achieve global optimality, but most research has focused on individual robots. To improve the efficiency of robotic systems, collaborative path planning problems considering multiple factors have become a focus of researchers [21]. Dasa et al. [22] added the consideration of path deviation and energy consumption optimization by embedding the social and cognitive behavior of an improved particle swarm algorithm (IPSO) into the newtonian gravity of an improved gravity search algorithm (IGSA). They proposed IPSO-IGSA to implement path planning for multiple robots in dynamic environments and improve search capability by simultaneously updating particle positions using IPSO velocity and IGSA acceleration. In a subsequent research, the authors [22] also investigated the multi-robot collision-free planning problem by mixing improved particle swarm optimization (IPSO) and evolutionary operators (EOPs) [23]. Milad et al. [24] proposed a path planning algorithm for single and multiple robots that simultaneously optimizes path length, smoothness, and safety in continuous environments using a multi-objective enhanced genetic algorithm (EGA), which eliminates the risk of possible collisions between multiple robots by adding a collision elimination operator to the EGA. However, the research was only implemented in a static environment and lacked interference factors such as moving obstacles. In addition, some scholars [25] set different priorities for each robot by prioritization methods, thus reducing the possibility of robot collisions.

In recent years, scholars have combined intelligent optimization algorithms with formation control methods for multi-robot systems to further improve the effectiveness of multi-robot path planning. Zheng et al. [26] proposed a path planning method to maintain the team formation for agents, which used the A* algorithm to obtain the path of the leader, obtained the critical point table and team transformation table by optimizing the path of the leader, then get the path of followers in the team. The study by Zheng et al. [26] prioritized the leader path of A* realization and neglected the overall coordination of the formation. Jiang et al. [27] used an improved genetic algorithm to search for the shortest path from the starting point to the endpoint for the leader robot in a grid environment. Based on the path planning of the leader robot, a better path from the starting point to the endpoint is planned for the followers to achieve a multi-mobile robot population that achieves obstacle avoidance while maintaining a particular formation. The study of Jiang et al. [27], like Zheng et al. [26], ignored the problem of robot collaboration. The Rapidly-Exploring Random Tree algorithm (RRT) has been particularly favored by scholars in the study of cooperative path planning in formation. Similar to the implementation of Zheng et al. [26], Dong et al. [28] used the RRT algorithm to generate a leader UAV's trajectory, followed by generating the trajectory of the follower UAVs and verifying its effectiveness based on the ROS-Gazebo platform. Liu et al. [29] proposed a decentralized path planning method for multiple mobile robot formations on the basis of RRT, which plans a separate path for each robot and uses a dynamic prioritization strategy to avoid conflict in formation motion [30]. Wang et al. [31] proposed a multi-robot formation-keeping path planning method with an improved RRT algorithm, which dynamically adjusts the formation to change the formation orientation during the planning process by establishing a link between the search tree expansion and the formation direction. The study is feasible for robotic systems with a prime point model and an incomplete dynamics constraint model. The RRT method has the advantages of fast search speed and low difficulty in modeling the environment space in path planning, but the generated paths have many nodes and discontinuous corner curvature compared with the ant colony algorithm. If the robot travels along the path in practice, it will affect the stability of the trajectory tracking and even cause instability due to trajectory oscillation and increase the difficulty of multi-robot formation reconstruction. This problem is especially prominent in the complex map environment.

To further promote the effect of multi-robot path planning, this paper proposes a new formation path planning method (LF-ACO, leader follower-ant colony optimization) by considering the ant colony algorithm's excellent robustness and the advantages of the leader-following method on formation collaboration. The contributions can be summarized as follows:

- Firstly, the slow convergence of traditional ant colony algorithms and most improved ant colony algorithms [14–20] mostly ignore the initial path smoothness problem. Based on the idea of the greedy algorithm to improve the traditional distance heuristic function with Euclidean distance as the indicator to improve the convergence. Furthermore, the smoothing heuristic function proposed increases the chance of ants searching in the linear direction during local exploration to improve the smoothness of the initial path.

- The leader-follower formation structure is reconstructed for the multi-robot position constraint problem in a grid environment. Then, the path of the leader ant is generated using an improved ant colony algorithm, and the paths of the follower ants are efficiently generated based on the location information of the leader ant and the obstacle information.
- The pheromones of leader and follower ants are introduced in the pheromone update rule to improve the searchability of subsequent leader ants for the formation path. This solves the problem that other scholars [26–29,31] have not taken full advantage of the optimization algorithm in their studies of formation paths. Then the maximum-minimum ant system is used to improve the global searchability of the algorithm and avoid the problem of local optimum.
- The turning point optimization algorithm and dynamic tangent point method [32] optimize the planned path, remove the redundant turning points of the path, and perform corner arc optimization.
- Finally, the comparative simulation experiment is conducted for a single robot in the grid map environment to verify that the improved ant colony algorithm can converge to the shortest path faster and with better path smoothing. Then, we have performed formation path planning simulations in multiple environments to verify the effectiveness of our LF-ACO.

2. Environmental description and problem statement

2.1. Environment description

Using the grid method to construct the environment map can effectively represent the working environment of mobile robots, and the grid area can be divided into obstacle grid $\mathcal{T}^O$ and free grid $\mathcal{T}^f$ (where the robot can move). Most scholars [14–20] currently ignore the effect of map modeling accuracy on the pathfinding quality, and most of the obstacles are also treated with simple expansions, as shown in Figure 1(a) for the actual shape of obstacles and in Figure 1(b) for the shape of obstacles after the expansion process in the grid map. However, to some extent, the size of grid granularity and the determination of obstacle expansion determine the merit of the model building [33]. When the effect of the expansion rate is not considered, if the granularity of grid division is too small, it will intensify the difficulty of path search, and the search results are not guaranteed to be optimal. In contrast, the environment model will deviate from the natural environment, and the paths searched cannot guarantee authenticity. Moreover, when the grid granularity is determined, the selection of the set value of obstacle expansion rate affects the searchable space of the algorithm and the quality of the path search. For facilitating our subsequent experiments, the expansion of obstacles is defined as follows: when the percentage area of the free grid portion occupied by the edges of obstacles is greater than or equal to the set value, the expansion is performed; if it is less than the set value, the obstacles in the region are discarded.

The above work will build its workspace for the robot in a two-dimensional coordinate system of $M \times M$. As shown in Figure 2, the black grids represent the obstacles, and the white grids represent the free grids. Finally, for determining the path information of the robot, we use a sequence number method to mark each grid [18], and the relation between the coordinate centers (x, y) of the grid and the sequence number *N* is as follows:

$$\begin{cases} x = \begin{cases} \operatorname{mod}(N,M) - 0.5 & \mathit{if} \operatorname{mod}\ (N,M)! = 0 \\ M + \operatorname{mod}(N,M) - 0.5 & \mathit{otherwise} \end{cases} \\ y = M + 0.5 - ceil\left(N,M\right) \end{cases} \tag{1}$$

where mod is a remainder operation, $ceil$ is an upward rounding operation.

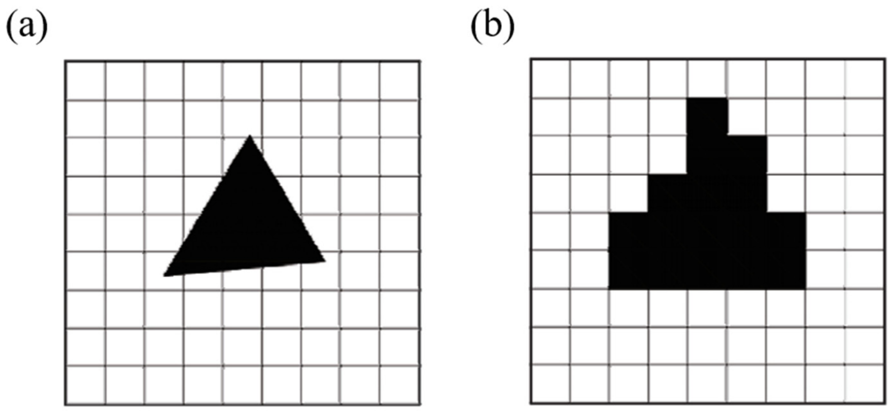

Figure 1. Grid environment for mobile robots. (a) The actual shape of the obstacle. (b) The result of the barrier dilatation treatment.

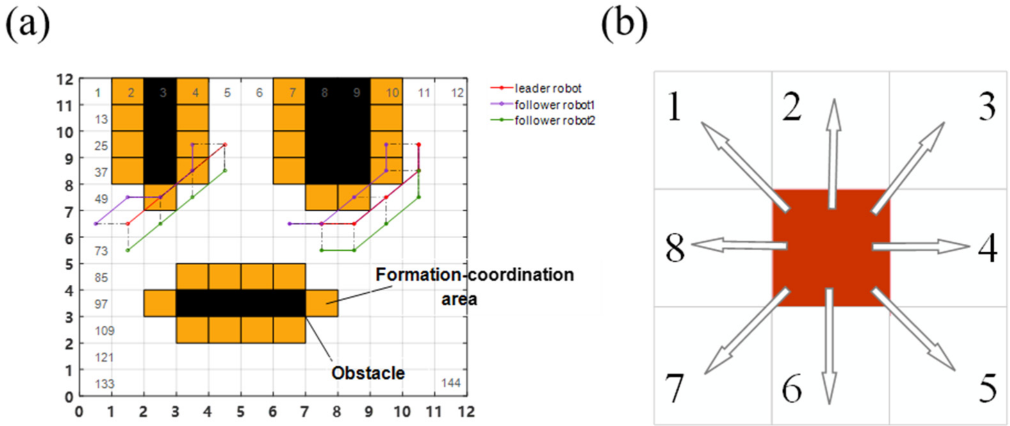

Figure 2. Illustration of formation-coordination area. (a) Multi-robot formations. (b) Search direction of leader ant.

Formation-coordination area: To ensure the integrity of the multi-robot formation, the initial path planned for the leader robot should be at a certain distance from the obstacle. This allows the follower robots to maintain a more stable formation when passing through narrow areas. As shown in Figure 2(a), given a$\mathcal{T}^{o}$, the $\mathcal{T}^{f}$ in four directions are defined as the formation-coordinate grids $\mathcal{T}^{h}$, and the leader robot should avoid passing through this area continuously. However, they should be allowed to pass if necessary (e.g., narrow passages). For this purpose, we improved the pathfinding mechanism of the ant colony algorithm. Figure 2(b) shows the path search direction of the leader ant. If the leader ant is located in the $\mathcal{T}^{h}$, the optional grid for the even transfer direction can be $\mathcal{T}^{h}$ and $\mathcal{T}^{f}$, but the odd transfer direction is that the grid can only be $\mathcal{T}^{f}$.

2.2. Problem statements

Here we present the problem statements considered in this work. The grid-based path planning problem is defined as follows.

Given a start grid S and a goal grid E, define a$\pi(t)$ to represent the path of the robot by finding a set of grids $\pi(t)$: $[0, t] \rightarrow \mathcal{T}^f$ such that $\pi(0) = S$ and $\pi(t) = E$. In addition, we define a formation structure of leader robot and follower robot, including the corresponding obstacle avoidance strategy. Then, $\pi(t)$ generates route nodes gradually from S to E and guarantees the path of the leader robot does not collide with any $\mathcal{T}^O$, which is necessary to improve the coordination of the whole multi-robot formation system. Next, the follower robot tracks the location information of the leading robot and executes the corresponding obstacle avoidance strategy. Finally, the path is optimized in terms of length and smoothness. As several criteria are optimized, the global path planning problem is also categorized as a multi-goal optimization problem [34].

3. Ant colony formation algorithm

For generating a better formation path in complex environments, we propose an ant colony formation algorithm (LF-ACO, leader follower-ant colony Optimization) by combining the ant colony algorithm and the leader-following method. We divide the ant population into the leader and follower layers based on the traditional ant colony algorithm, where: the initial path of the leader ant is generated by the improved ant colony algorithm, and the follower layer ant follows the path of the leader ant to reach the formation. This section focuses on the improved ant colony algorithm process, and Chapter 4 introduces the design ideas of the follower layer ant.

Algorithm 1 Ant Colony Formation Algorithm

```
procedure LF-ACO
  Building a robot working environment using the grid method
  initialize the number of ants $m$, maximum iteration number $Nmax$, weights $\alpha$, $\beta$, $\rho$, start $S$, goal $E$
  for $N$ = 1 to Nmax do
      Put all ants into $S$
      while ant $k$ is not in $E$ do
          $allowdi$ ← the set of reachable grids for k
          choose the next grid by (4)
      end while
      if all ants have arrived $E$ then
          $S_k$ ← comprehensive index of ant $k$'s path
          $S_{best}$,$b$ ← the best path in all ants
          $S_{worst}$,$w$ ← the worst path in all ants
      end if
      select the ants as the leader ants
      follower ant tracking the path of leader ants by formation control strategy ( 4.1. and 4.2. )
      $F_k$ ←comprehensive index of follower ants's path
          update the pheromone by (9)(17)
      end if
  end for
  output the best path
end procedure
```

To improve the ant colony algorithm, we propose an enhancement function based on the principle of optimal-worst ant system to make the ant search more focused around the best path found up to the current cycle and evolve toward the global optimum. The achieved path is poor because the traditional ant colony algorithm has only a single path length as the objective. For this reason, we improve the heuristic function and pheromone by weighing the path length and smoothness to improve the performance of ant pathfinding. In addition, we can adjust the goal weights of the algorithm to focus on a single goal. The pseudo-code of the ant colony formation algorithm is given in Algorithm 1.

3.1. Enhancement function

In foraging, once an ant finds the food source, it starts the return trip and deposits pheromones on the path that it has passed to guide the following ants. This creates a positive feedback loop that all ants in the colony eventually follow the optimum path to the food source. Since the paths established by ants in any loop affect the subsequent decisions of the colony, if the established paths are not optimal, they will mislead the subsequent ants.

In this paper, the optimal and worst solutions are found according to the principle of the best-worst ant system after one iteration, and the global pheromone update is effectively determined, enhancing the optimal solution and weakening the worst solution. Using the path enhancement factor, the ant's search can focus more on the neighborhood of the best path found in the current cycle and evolve towards the global optimum. The global pheromone enhancement factor is as follows:

$$q(t) = p\frac{S_w}{S_b} \tag{2}$$

$$p = \frac{b}{w} \tag{3}$$

where t is the current iteration number, $q(t)$ is a coefficient of enhancement factor, b and w are the best and worst ant count in this iteration, respectively, S_w and S_b are the comprehensive index for the best and worst routes in this iteration. The number of optimal ants is small, and the number of worst ants is significant in the early iterations of the algorithm, and the overall performance of the path is poor. As the number of iterations increases, the number of optimal ants increases, the number of worst ants decreases, and the overall optimization ability of ants become better, while the $q(t)$ gradually converges to p. That enhances the overall optimization capacity of the algorithm and accelerates the convergence speed of the algorithm.

3.2. Process of improved-ACO

Like most scholarly studies on the formation control of multi-agent systems, the leader ant plays a prominent role in this paper. To enhance the algorithm's efficiency, the follower ant only tracks the path of the optimal path leader ant for each iteration and feeds back its distance length and number of turns to affect the pheromone update of the ant colony algorithm path of the leader.

3.2.1. Initialization

Several parameters are initialized, and each of them is explained in Table 1. The value of these weights needs to be attributed to experiments. In addition, start S and goal E are decided, and all ants are situated on the starting grid.

Table 1. Parameters of the improved ant colony algorithm.

Variable	Description
m	Number of ants
N_{max}	Maximum number of iterations
α	Weight of pheromone
β	Weight of heuristic information
ρ	Pheromone evaporation ratio
Q	Pheromone intensity
τ_{ij}	Pheromone on the path between i and j
η_{ij}	Heuristic information on j
ξ	distance factor coefficient
ψ	distance correction parameter
U	the importance of ants going straight
δ	the parameter of the pheromone update

3.2.2. Selection

In the next step, each ant uses the roulette wheel to choose the grid to move forward. For ant k, the probability $p_{ij}^{k}(t)$ of moving from the current grid i to that grid j is expressed as:

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}^{k}(t)\right]^{a}\left[\eta_{ij}^{k}(t)\right]^{\beta}}{\sum_{j\subset\, allowdi}\left[\tau_{ij}^{k}(t)\right]^{a}\left[\eta_{ij}^{k}(t)\right]^{\beta}},j\in allowdi\\ 0,j\notin allowdi\end{cases} \tag{4}$$

where k is the number of ants, $allowdi$ denotes the set of transferable grids when the ant is in grid i.

3.2.3. Heuristic information

The traditional heuristic function is shown in Eq (5), which indicates the distance between grid j and target grid E. To deal with the complex and changing environment of the robot, we improve the heuristic information by using the smoothness and length of the path as optimization objectives, as in Eq (6).

$$\eta_{ij}^{k}(t)=\frac{1}{d(j,E)} \tag{5}$$

$$\eta_{ij}^{k}(t)=\varphi_{ij}^{k}(t)+r_{ij}^{k}(t) \tag{6}$$

In the early stage of the traditional ant colony algorithm, the difference in pheromone concentration on the path is slight, and the difference of distance between the next set of to-be-selected locations at the current position and the target point is also tiny, which causes the confusion of the ant colony pre-search. To solve this problem, the heuristic function is designed to increase the influence of the distance between the node to be selected and the endpoint grid, which makes the pathfinding strategy approximate to the greedy algorithm, and the improved distance heuristic function is as follows:

$$\varphi_{ij}^{k}(t)=\begin{cases}\left(\dfrac{|MAX-d(j,E)|}{MAX-MIN+Lgrid}\right)\xi+\psi \\ 0,\text{ others}\end{cases} \tag{7}$$

where $\varphi_{ij}^{k}(t)$ indicates the corrected distance between grid j and target grid $E, d(j,E)$ denotes the euclidean distance between two points, MAX and MIN denote the maximum and minimum distances to be transferred, respectively; $Lgrid$ is the size of the grid edge length.

The traditional ant colony algorithm tends to plan a path with more turns in the grid environment. If applied to the real-world scenario, the path obtained by the mobile robot has a shorter overall length, but there are individual unnecessary turn angles, and the robot needs to adjust its state to accommodate the change in angle when passing through the corners, leading to more complex driving. To solve these problems, the smoothing factor heuristic function is designed to reduce the turning points of the path by increasing the chance of the ants going straight while moving, as follows:

$$r_{ij}^{k}(t)=\begin{cases}U, t(J)=t(J-1) \\ \dfrac{U}{\sqrt{2}Lgrid}, otherwise\end{cases} \tag{8}$$

When the direction of the grid to be selected is the same as the direction of the previous grid, $r_{ij}^{k}(t)$ prompts the ants to continue in that direction.

3.2.4. Pheromone update

This step is to perform pheromone enhancement and decay after each round of iteration is completed, and the rules are as follows:

$$\tau_{ij}(t+1)=(1\text{-}\rho)\tau_{ij}(t)+\Delta\tau_{ij}(t)+\delta q(t) \tag{9}$$

and

$$\varDelta\tau_{ij}(t)=\sum_{m}^{k=1}\varDelta\tau_{ij}^{k}(t) \tag{10}$$

where $\varDelta\tau_{ij}^{k}(t)$ is the quantity of pheromone deposited by ant *k* on the path between *i* and *j* at time *t.*

The traditional pheromone update method is the ant-perimeter model, as shown in Eq (11). In practice, the optimal path not only needs to be shorter path but also smoother. Therefore, we make the following changes to the pheromone increment of the ant colony formation.

$$\varDelta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{Q}{L_k(t)}, j\in allowd_i \\ 0,\text{ others}\end{cases} \tag{11}$$

$$\varDelta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{Q}{S_k}+\dfrac{Q}{F_k}, j\in allowd_i \\ 0,\text{ others}\end{cases} \tag{12}$$

$$s_k = aL_k + bT_k \tag{13}$$

$$F_k = \sum_{z=1}^{n} \left[L_k^z + T_k^z \right] \tag{14}$$

where s_k is the integrated index of the path taken by the leader ant k, and the pheromone is assigned according to the integrated index, the smaller the index the better the path; F_k is the integrated index of the path taken by the follower ant; L_k is the path length of the leader; T_k is the number of path turns of the leader; a and b are moderators of the distance factor, which can be taken according to the path demand. L_k^z is the path length of the follower ants in this iteration; T_k^z is the number of path turns of the follower ants in this iteration; n is the number of follower ants. The expressions for the total length and turns of the path are as followers:

$$L_k, L_k^z = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{15}$$

$$T_k, T_k^z = card(S, \dots, i, j, \dots, E) \tag{16}$$

where x, y are the coordinate; T is the number of turning points.

In order to improve the global search ability of the algorithm and avoid the ants converging to the locally optimal path too quickly, the total number of pheromones on the path is controlled within a specific range $[\tau_{\min}, \tau_{\max}]$, and the pheromone update strategy is as follows:

$$\tau_{ij}(t) = \begin{cases} \tau_{\max}, \tau_{ij}(t) > \tau_{\max} \\ \tau_{ij}(t), \tau_{\min} \le \tau_{ij}(t) \le \tau_{\max} \\ \tau_{\min}, \tau_{ij}(t) < \tau_{\min} \end{cases} \tag{17}$$

4. Multi-robot formation control

In the previous section, we improved the ant colony algorithm for planning paths for the leader ant, and this section describes how the follower layer ant work. Assuming that the global information in the environment is known, the leader layer ants can achieve path search and obstacle avoidance based on the ant colony algorithm, and the follower layer ant needs to consider both formation control problems and obstacle avoidance. Based on this, the research in this section is as follows.

4.1. Determination of formation

The leader-follower method is a typical chain topology. The whole formation consists of a leader robot and follower robots, which only needs to make the follower obtain information about the leader's motion state and track the leader's motion trajectory in the implementation process to achieve the formation control.

Firstly, in this paper, the target formation is formed at the initial moment of planning, and the formation structure takes the form of the leader-follower method, and the formation is constructed by setting the desired angle φ_d and desired distance l_d of the leader and the followers.

Next, we present the path planning method with three robots to maintain a triangular formation in a common planar environment. In planning, the direction of the robot formation is dynamically adjusted, and the leader's direction from the current node to the next node is usually defined as the formation direction. Taking as an example Figure 3, when the leader moves from the path point $P_0(x_0, y_0)$ to the current position $P_1(x_1, y_1)$, l and φ denote the actual distance and the actual angle difference (relative direction) between the leader robotics and the follower robotics1, respectively, and θ is the angle between the robot's forward direction and the horizontal direction. Then the ideal team position $P_f(x_f, y_f)$ of the follower robotics1 at the current moment is:

$$\theta = \begin{cases} 90^{\circ}, & x_0 = x_1 \\ arctan\left(\frac{y_1 - y_0}{x_1 - x_0}\right), & x_0 \neq x_1 \end{cases} \tag{18}$$

$$\begin{bmatrix} x_f \\ y_f \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} - \begin{bmatrix} cos(\varphi_d + \theta) \\ sin(\varphi_d + \theta) \end{bmatrix} l_d \tag{19}$$

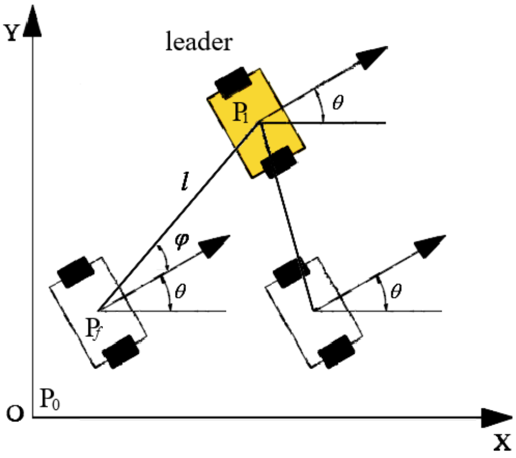

Figure 3. Schematic diagram of the relative positions of robot leader and follower.

The path planning research based on the ant colony algorithm generally uses a grid environment modeling, and for making the traditional formation control method better applicable to this environment, we make some adjustments: 1) the directional angle θ of the ant colony formation is not considered during the planning process; 2) we generate the formation at the initial moment of planning; 3) the desired angle φ_d and desired distance l_d of the initial leader and follower are the subsequent leader ants and following ants need to keep the formation. In the following parts, we demonstrate the effect of the improvement.

4.2. Avoiding obstacles

In the path planning process of the grid environment, path points that satisfy both leader and follower obstacle avoidance conditions should be searched. When an obstacle is encountered, the multi-robot planning process can be made to avoid the obstacle by team switching for a multi-robot system with a prime number model.

First, the leader ant uses an improved ant colony algorithm to plan its optimal path based on the map information.

Secondly, the leader ant runs along the optimal path toward the target point and uses the designed algorithm for obstacle avoidance. In contrast, the leader ant detects its position information in real-time and uses the l-φ control method to generate the trajectory of follower ants according to the set formation.

Finally, the leader ant in real-time and adjusts its speed and movement direction to follow the trajectory. Meanwhile, the follower ant detects the obstacle information around itself in real-time. When the follower ant encounters an obstacle, it waits in place for a certain time, and when the leader ant passes, the follower ants move to the position of the leader ant at the last moment, and as soon as passing the obstacle, the follower ants will return to the trajectory of the virtual ant and resume the set formation. At the end of each iteration, the follower ants feedback their path information for the leader ant's pheromone update.

Based on the above description of multi-robot formation keeping and formation direction, the purpose of this paper is to solve the path planning problem of multi-robot in formation operation and plan an obstacle avoidance path that satisfies the multi-robot formation constraints. Figure 4(a) shows the path planning of multi-robot formation control implemented by the ant colony algorithm combined with the unimproved leader-follower method, and Figure 4(b) shows the path planning implemented by our improved ant colony formation algorithm. The comparison shows that our improved algorithm is more consistent with the formation characteristics in the grid environment.

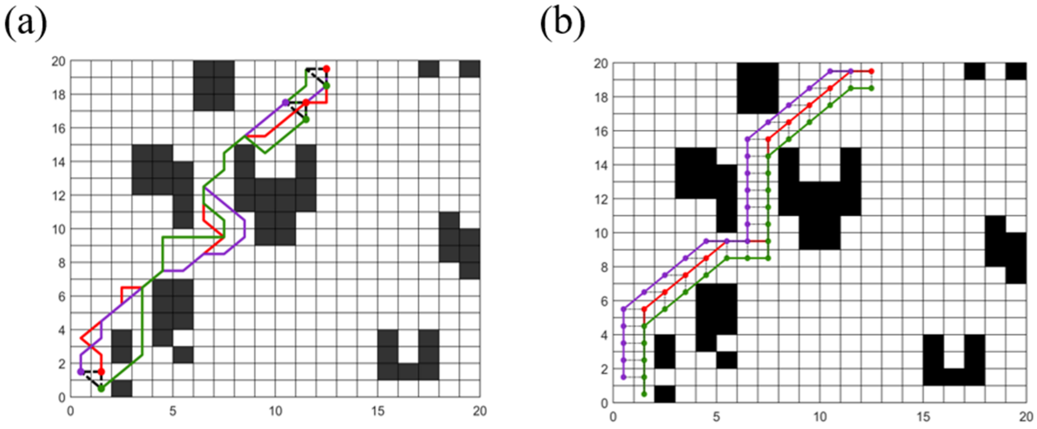

Figure 4. Multi-robot formation path planning under two strategies. (a) L-F combined with ant colony algorithm. (b) Improved ant colony formation algorithm.

5. Track optimization

5.1. Triangular pruning optimization algorithm (TPOA)

Most of the current path planning research are based on the ideal environment model established by the grid method, which does not consider the actual working conditions of mobile robots. Moreover, the grid modeling setting directly limits the accuracy of path planning, and the planned paths have redundant inflection points, which do not match the actual working characteristics of the robot. We smooth the optimal path planned by the improved ACO, which adjusts the route direction and improves the traditional fixed straight line direction or turn, making the mobile robot more flexible in the working environment, reducing the loss of running time and energy consumption of the mobile robot, and improving the working efficiency.

First, we connect two nodes in the path, not in the same line, and determine whether the current line passes through the obstacle area. If not, the current route is considered as a new path, and the intermediate nodes are removed. Otherwise, no path modification is performed. According to this method, all nodes of the path are traversed to generate a new path. The smoothed path is shown in Figure 5. The blue line is the smoothed path. Furthermore, the black dashed line is the preceding path. It is easy to see that removing these unnecessary transitions can effectively improve the length and smoothness of the robot.

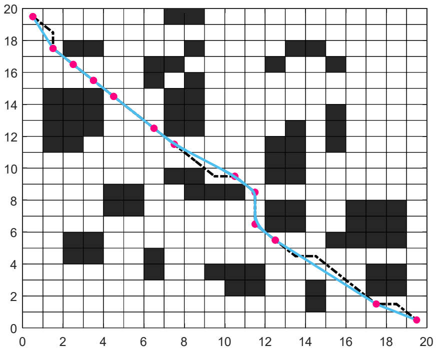

Figure 5. The original path and the path with unnecessary turn points removed.

When our TPOA smoothes the path, all the turning points need to be previously recorded to shorten the path length and reduce the number of turning points $T = \{S, T1, T2...Tn.E\}$.

Let us introduce the P function at the same time. Tx and Ty are two nodes on the path.

$$G(Tx, Ty) = \begin{cases} 1, & \text{if } Tx \text{ can go straight to } Ty \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

Algorithm 2 Triangular Pruning Optimization Algorithm

```
Procedure TPOA
  input the mobile robot path obtained by the ACO
  input the coordinates of nodes T
  n ← number of nodes on the path
while a < n – 1 do
    for b = n - a to 2 do
      calculate G(Ta, Tb) by (20)
      if G(Ta, Tb) == 0 then
        remove the middle nodes
      end if
    end for
  end while
  output the newRoute
end procedure
```

5.2. Dynamic cut point method

A smooth path is more conducive to the robot's work. If the smoothing optimization of the inflection point is performed with a fixed tangent angle or a fixed tangent point, the robot may fall into a dead zone with obstacles. To better optimize the path, this paper uses the dynamic tangent point adjustment method [32] to smooth the planned path, as shown in the smoothing schematic in Figure 6.

The robot starts from the initial position $A_1(x_1, y_1)$ and smoothes the corner sections sequentially along the direction of travel until the endpoint $A_n(x_n, y_n)$, as follows:

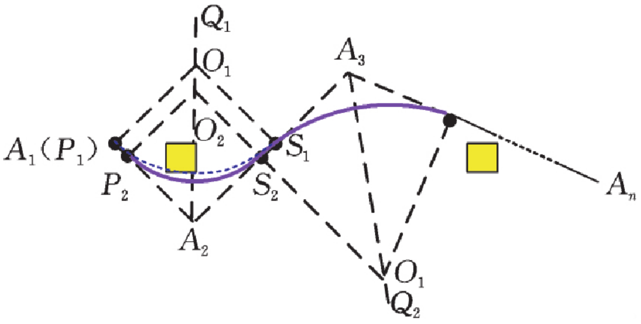

Figure 6. Path smoothing diagram.

Step 1: Choose the shorter sides of $A_{i-1}A_i$ and A_iA_{i+1}. Using the endpoint of the shorter side as the initial point $P(x_p, y_p)$, make a vertical line intersecting the angle bisector $A_iQ_{i-1}(i = 2, 3, \cdots,$ $n-1)$ of $\angle A_{i-1}A_iA_{i+1}$ at the point $O_{i-1}(x_0, y_0)$ $(i = 2, 3, \cdots, n-1)$, where:

$$x_0 = \frac{\left(x_p + k_{01}y_p + k_{01}(k_0x_2 - y_2)\right)}{(1 + k_0k_{01})} \tag{21}$$

$$y_0 = k_0(x_0 - x_2) + y_2 \tag{22}$$

The radius R of the tangent circle can be expressed as:

$$R = \sqrt{x_0^2 - 2x_0x_p + y_0^2 - 2y_0y_p + x_p^2 + y_p^2} \tag{23}$$

The equation of the tangent circle is:

$$(x - x_0)^2 + (y - y_0)^2 = R^2 \tag{24}$$

where k_{01} is the slope of the short side; k_0 the slope of the angle bisector.

Step 2: Determine whether there is an intersection S between the tangent circle and the long edge; if so, execute Step 3; otherwise, execute Step 4.

Step 3: Determine whether there is an obstacle on the arc PS; if there is, then execute Step 4; otherwise, use the arc PS instead of the corner and execute Step 5.

Step 4: The movement of tangent point $P(x_P, y_P)$ along the line segment where it is located to (x_{P2}, y_{P2}), x_{P2} can be expressed as:

$$x_{p2} = x_p + \lambda\left|x_2 - x_p\right| \quad \lambda \in (0,1) \tag{25}$$

where λ is set according to the actual situation. Also set P_2 as the initial cut point and return to Step 1.

Step 5: If all path nodes have been traversed, the algorithm is finished, otherwise return to Step 1 to continue the execution.

As can be seen from Figure 7, the blue line is a specific initial path, and the purple line is the path after smoothing and optimization by the algorithm in this paper. The optimized path reduces the number of turns and the accumulated turning angle, which improves the quality of the planned route.

6. Experiment

In this section, we conducted five experiments to illustrate the feasibility and superiority of the algorithm. In the first experiment, we performed path planning simulations for a single robot in different environments and compared them with other path planning algorithms. In the second experiment, we performed conducted path planning experiments on multi-robot formations. In the third experiment, we tested the role played by the smoothing parameter. In the fourth experiment, we conducted a raster map modeling study to extend the application scenario of LF-ACO. In the fifth experiment, We simulated a natural ROS-based environment and conducted path planning experiments for multi-robot formations. All simulations are performed on a computer with Intel Core i5 CPU @ 2.3 GHz and 8 GB of RAM under Windows 10.

Table 2. Main parameters of the simulation experiment.

Algorithm	m	N_{max}	α	β	ρ	Q	ξ	ψ	u	δ	a	b	λ
ACA	50	50	1	7	0.3	10	-	-	-	-	-	-	-
DLACA	50	50	1	3	0.3	100	-	1	-	-	-	-	-
RMACA	50	50	1	5	0.5	10	-	-	-	-	-	-	-
EACA	50	50	1	2	0.2	1	-	-	-	-	-	-	-
This paper	50	50	1	3	0.3	100	10	1	5	20	1	1	0.95

6.1. Single robot path planning experiment

To verify the effectiveness of our improved algorithm for single-robot path planning, we selected a 20 × 20 and 40×40 map environment from the literature [35] for comparison experiments. We chose four other algorithms for comparison: the traditional Ant Colony Algorithm (ACA), the Double Layer Ant Colony Algorithm (DLACA) [35], the Evolutionary Ant Colony Algorithm (EACA) [17], and the Retraction Mechanism Ant Colony Algorithm (RMACA) [18]. The convergence speed, shortest path and bending suppression effects of these algorithms are compared. Since the algorithms of literature [35], literature [17], and literature [18] differ from the algorithm of this paper in the values of various parameters, the simulated parameters from the original paper are used, as shown in Table 2.

1) Example 1. In this example, the robot's environment is modeled as a 20×20 grid and compared in a regular map environment. The coordinates of the robot's grid starting point and target are (0.5, 19.5) and (19.5, 0.5), respectively. (shown in Figures 7 and 9(a))

2) Example 2. In this example, the robot's environment is built in a 40×40 grid model and compared in a giant slot map environment. The coordinates of the robot's grid starting point and target are (5.5, 34.5) and (29.5, 5.5), respectively. (shown in Figures 8 and 9(b))

As shown in Figure 7, some ants die in the U-shaped deadlock region during the pathfinding process of traditional ACA, and the obtained paths fall into the optimum local situation, so it is the worst path. DLACA and RMACA solve the deadlock problem in the algorithm improvement, so the paths planned are better than EACA and traditional ACA. The strategy of RMACA will leave a large number of invalid pheromones in the deadlock edge region, which affects the path search of subsequent ants, and thus the algorithm converges poorly. Our improved algorithm mainly prevents ants from getting into the deadlock region through the smoothing heuristic function, and the global pheromone enhancement factor improves the algorithm's convergence. The original path length of our improved ACA is 28.6274, and the number of turns is 4. After optimization, it is 27.5622 and 3. Our algorithm is the most efficient among the five algorithms in path length and number of turns.

Figure 9(a) shows that our algorithm has the best iterative results among all the algorithms. Due to our late optimization, the algorithm runs longer but still outperforms ACA.

As shown in Figure 8, the worst path is still the traditional ACA trapped in a local optimum. Since experiment 2 is more complex than that of experiment 1, the EACA algorithm becomes worse in this environment and does not converge until the 44th iteration. The deadlock handling strategy of RMACA is somewhat adaptive to this environment and finds a path close to the optimal solution at some cost in the 31st generation. In terms of length and number of turns, RMACA and DLACA slightly outperformed the original paths of our algorithm. After post-optimization, our algorithm is the most effective among all algorithms, with 5.2% and 46.2% optimization in path length and turn count compared with RMACA, and 2.8 and 41.7% compared with DLACA. Figure 9(b) shows that our algorithm is also more effective among all algorithms in terms of iterations. From the experimental results, our algorithm effectively solves the path planning of a single human in a complex environment.

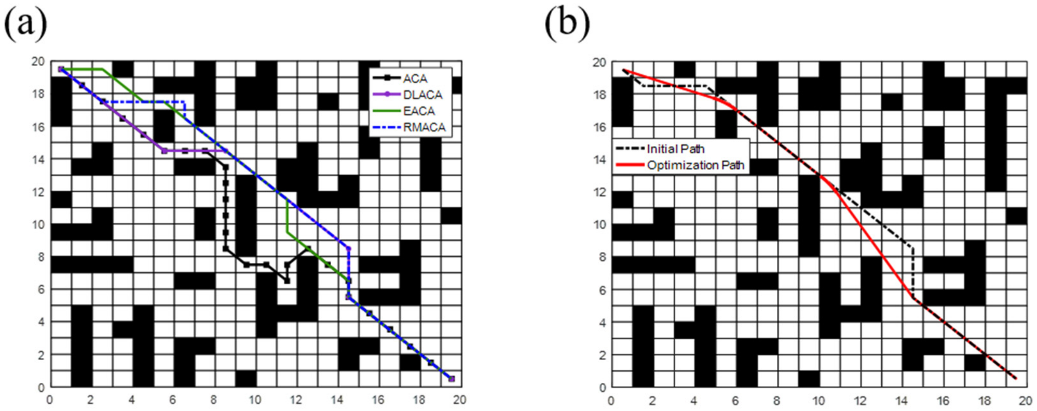

Figure 7. Comparison of path planning results under environment 1. (a) Path planning diagram of other algorithms. (b) Path planning diagram of the algorithm in this paper.

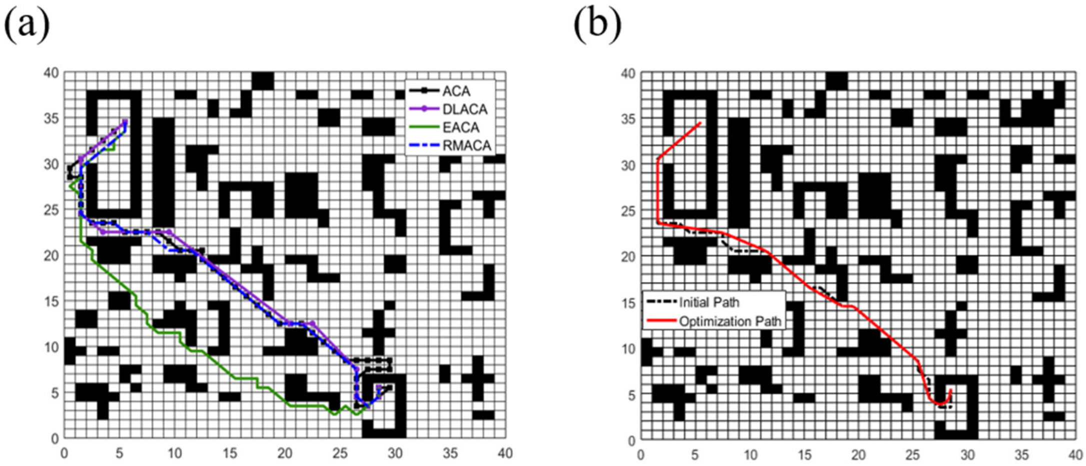

Figure 8. Comparison of path planning results under environment 2. (a) Path planning diagram of other algorithms. (b) Path planning diagram of the algorithm in this paper.

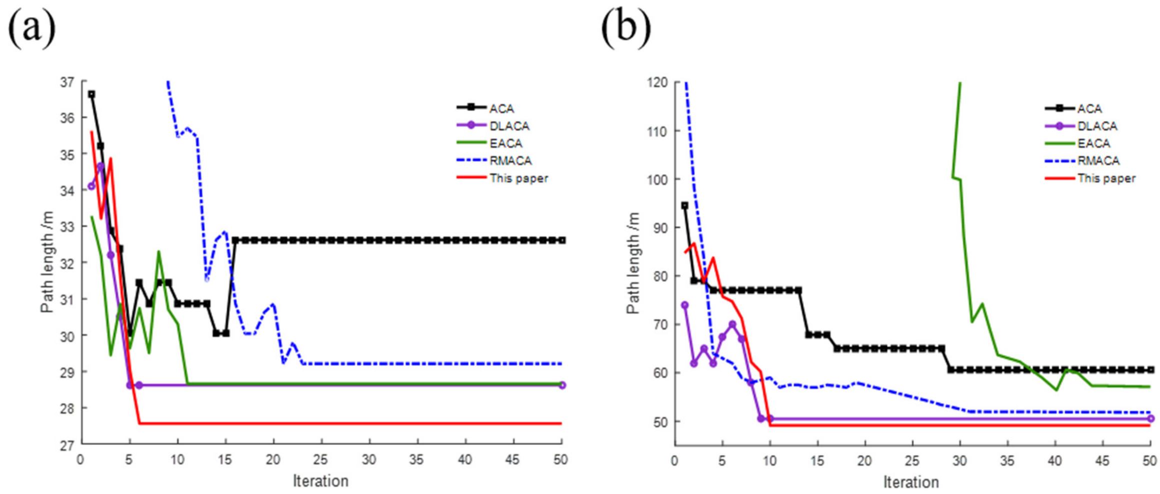

Figure 9. Iterative diagram of the optimal path in maps 1 and 2. (a) Comparison of algorithm convergence in map 1. (b) Comparison of algorithm convergence in map 2.

Table3. Algorithm comparison test results under different maps.

Map	Algorithm	Optimal solution of the algorithm	iteration times	Time consuming(sec)	Number of turns
1	ACA	32.6240	16	6.86	11
	DLACA	28.6274	5	1.89	4
	EACA	28.6557	11	1.5	7
	RMACA	29.2100	23	1.299	5
	This paper	27.5622	6	2.63	3
2	ACA	60.5269	29	75.64	21
	DLACA	50.5269	9	5.5	12
	EACA	57.1261	44	5.27	33
	RMACA	51.8471	31	58.83	13
	This paper	49.1306	10	4.89	7

6.2. Multi-robot formation path planning experiment

In previous experiments, we verified the application of the improved ant colony algorithm to single-robot path planning and validated the experimental property of the algorithm of complex environments. In this section, we conducted four formation path planning experiments on a triangular formation of three robots. In four experiments, we used a 20 × 20 grid environment as the working environment for multiple robots. The starting coordinates of the lead robot were set to (1.5, 1.5) and the coordinates of the target point were (19.5, 19.5), and the starting coordinates of the following two robots were (0.5, 1.5) and (1.5, 0.5), respectively, and the coordinates of the target point were (18.5, 19.5) and (19.5, 18.5).

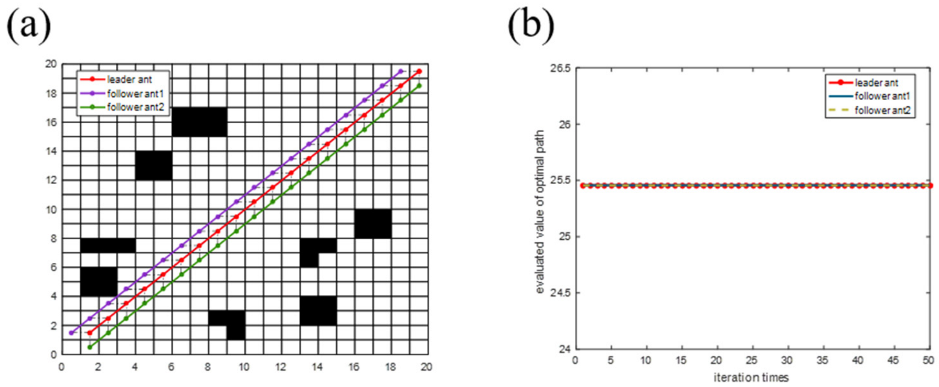

Figure 10. The optimal formation path diagram under environment 1. (a) Path planning figure in map 1. (b) Convergence curve of the optimal path in map 1.

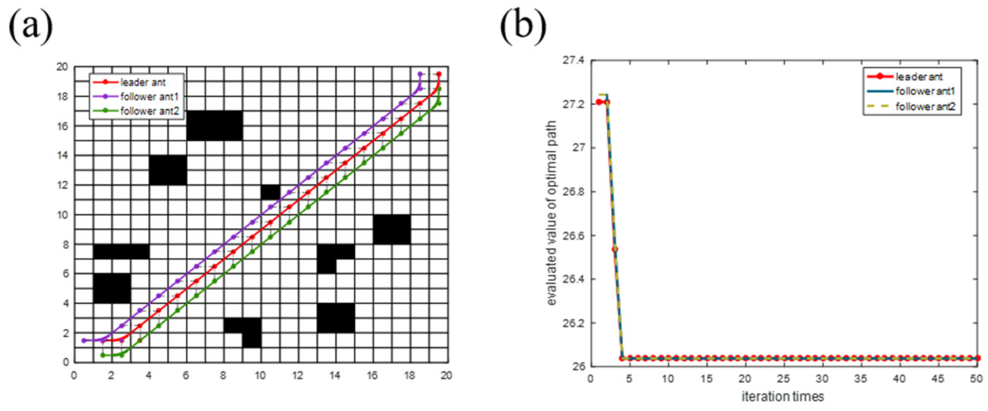

Figure 11. The optimal formation path diagram under environment 2. (a) Path planning figure in map 2. (b) Convergence curve of the optimal path in map 2.

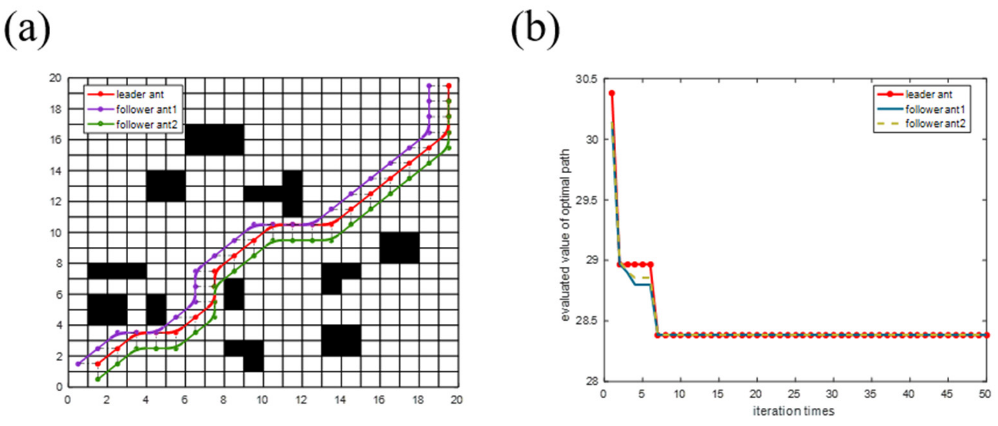

Figure 12. The optimal formation path diagram under environment 3. (a) Path planning figure in map 3. (b) Convergence curve of the optimal path in map 3.

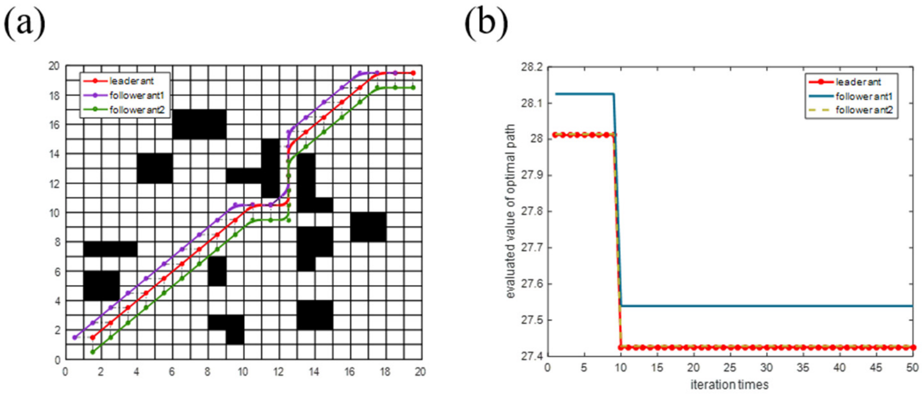

Figure 13. The optimal formation path diagram under environment 4. (a) Path planning figure in map 4. (b) Convergence curve of the optimal path in map 4.

Table 4. Formation path results for multiple robots in different maps.

Map	Multi-Robot	Initial path length	Optimized path length	Number of turns in formation	iteration times	Algorithm runtime
	leader ant	25.4558	25.4558			
1	follower ant1	25.4558	25.4558	0	1	11.6934
	follower ant2	25.4558	25.4558			
	leader ant	26.0416	25.9341			
2	follower ant1	26.0416	25.9341	2	4	15.6042
	follower ant2	26.0416	25.9341			
	leader ant	28.9690	28.4417			
3	follower ant1	28.9706	28.4443	7	6	18.032
	follower ant2	28.9706	28.4443			
	leader ant	27.7960	27.4256			
4	follower ant1	27.7990	27.5393	4	10	14.9257
	follower ant2	27.7990	27.4286			

As shown in Figure 10, neither the leader ant nor the follower ants encountered any obstacles, and the ants moved directly from the starting point to the endpoint along a straight line. The formation maintains a triangular formation from beginning to the end, and the length of all paths is 25.4558, and the number of turns is 0. The trajectory diagram is shown in Figure 9(a), and the iterations forming the ant colony algorithm are shown in Figure 10(b).

As shown in Figure 11, based on the first experiment, an obstacle is placed on the line between the starting and ending point of the leader ant. In the initial stage of the colony iteration, many leader ants move to the formation coordination grid around this obstacle before avoiding the obstacle, which generates more formation redundant nodes. To coordinate the formation, we consider both path length and the number of turns of the leader ant and the follower ants in the pheromone update of LF-ACO. Therefore, after several rounds of iterative optimization, to ensure the smoothness of the path, the leader ant finds a path that can maintain the formation as much as possible. So what we get is the formation path where the follower ant followed the leader ant to bypass the obstacle in advance. Due to the interference of the obstacles, the path length and the number of turns increased compared with the first experiment, and the optimal path was found only in the 4th iteration, as shown in Figure 11(b).

As shown in Figure 12, based on the second experiment, several obstacles were placed in the path of the follower robot. The formation changes briefly, and the follower ants will wait for a while in front of the obstacle to avoid collision problems. When the leader ant passes, the follower ants will move to the position of the leader ant at the last moment. After the followers bypassed the obstacle, the ants returned to their original formation and continued. Compared to the experiment 2, the environment complexity of this experiment increased, so the path length and number of turns for a single ant increased, and the algorithm converged only in the 7th iteration.

As shown in Figure 13, narrow terrain was designed based on the third experiment. When only one robot can pass through the narrow environment, two follower robots will wait in front of the obstacle. After the leader robot passes the narrow passage, the follower robots will move to the position of the leader robot and pass the narrow passage one by one in a "one" shape, and then return to the set formation after all robots pass the narrow area. The iteration diagram of the algorithm is shown in Figure 13(b).

The results of the ant colony formation algorithm in four scenarios are shown in Table 4. The experiments show that the leader-follower method designed in this paper enables all robots to maintain their formation and run safely and reach the target position without collision. When the robots encounter an obstacle, they can change their formation to avoid the obstacle and resume the formation after completing the obstacle avoidance task.

6.3. Multi-robot formation experiments in special map environments

The algorithm in this paper allows the formation path to perform exceptionally well in terms of smoothing performance, for example, by simply increasing the parameters associated with the number of turns or reducing weights of other factors, so that the number of route turns continues to decrease to achieve the desired effect. To verify the path advantage of LF-ACO in terms of formation smoothing characteristics, we conducted a particular 10 × 10 scale terrain for the experiment, and the results are shown in Figure 14 where Figure 14(a) is the formation path achieved based on the parameters in Table 2, and Figure 14(b) is the formation path obtained after adjusting the parameters u in the table to 15 and b to 5. The comparison results show that the former only pursues the shortest distance and chooses a highly tortuous path, while the latter avoids narrow and tortuous areas to achieve the best overall smoothness of the formation. However, the running time of the latter is three times longer than that of the former. The results of the ant colony formation algorithm in this scenario are shown in Table 5.

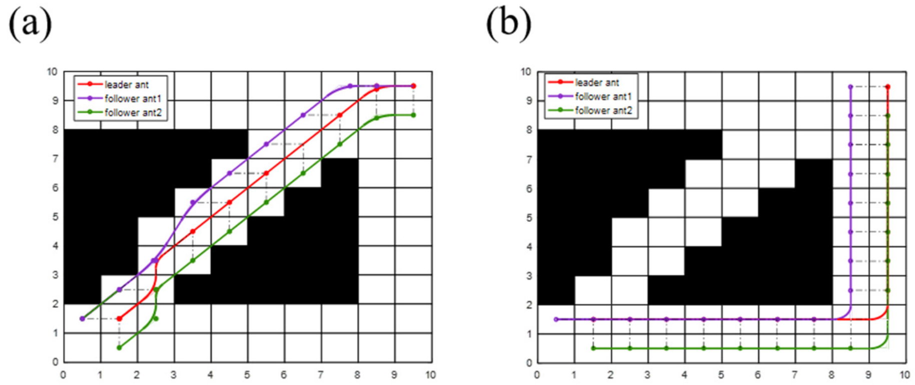

Figure 14. Diagram of experimental results of special barrier map. (a) Formation path diagram without changing parameters. (b) Formation path diagram after changing parameters.

Table 5. Characteristic formation path results for multiple robots.

Map	Multi-Robot	Initial path length	Optimized path length	Number of turns in formation	iteration times	Algorithm runtime
1	Leader ant	11.8980	11.7663	3	——	7.0497
	Follower ant1	11.7214	11.6450			
	Follower ant2	11.8995	11.7678			
2	Leader ant	16.0000	15.7961	1	——	24.5374
	Follower ant1	16.0000	15.7961			
	Follower ant2	16.0000	15.7961			

6.4. Multi-robot formation path planning experiments with real cases

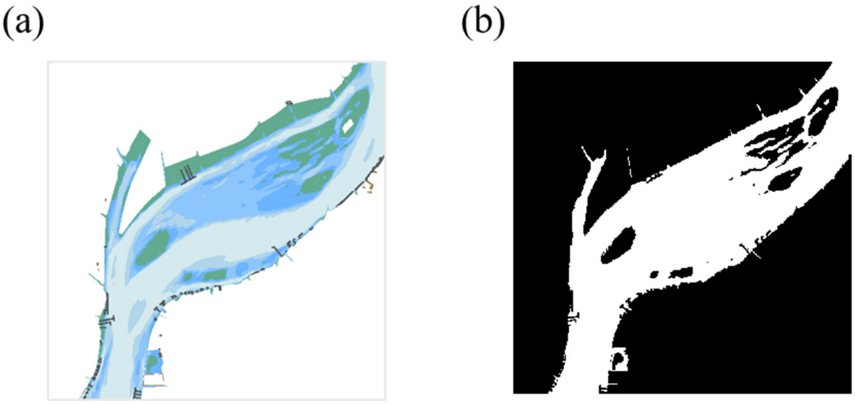

Figure 15. Navigational map of a watershed of the Yangtze River in China. (a) E-map. (b) Binarization diagram.

In the above experiments, we have only considered some limited map environments. To further extend the practical application value of our proposed LF-ACO, this experiment will investigate the path planning of USVs in conjunction with the Yangtze River basin of China in Figure 15(a). The green area in the figure represents land (impassable), and the blue and white areas represent feasible passable waters. Figure 15(a) is binarized to produce Figure 15(b) for the grid method modeling research that followed.

Most of the current scholarly studies on path planning [14–20,35] have ignored the impact of map modeling accuracy on path planning. In this experiment, a class of map modeling methods based on grid granularity and obstacle expansion rates, and the formation path study is conducted. The path results and modeling parameters are shown in Table 6 and Figure 16. Figure 16(a),(b) show the multi-robot path simulation maps with the leader's start point at (10.5, 3.5) and endpoint at (47.5, 46.5), with a grid granularity of 10 and an obstacle expansion rate of 20% and 5%, respectively. From the figure, we can see that the number of obstacles in Map 2 has become more than in Map 1, and the path space of LF-ACO search become smaller, making the formation path changed. Taking the leader as an example, the path length in the Map 2 scenarios increased by 4.73% compared with Map 1, but the number of turns decreased by 33.3%, and the algorithm search time was optimized. Figure 16(a),(b) show maps modeled with low accuracy, with significant differences from the original scene in Figure 15, keeping the expansion rate of obstacles constant and choosing a grid granularity of 5 and 3 to build Maps 3 and 4 of Figure 16(c),(d), respectively. The modeling accuracy of Map 3 is nearly twice that of Map 2, and the preset locations of the start and endpoint of leader in Map 3 are nearly twice that of Map 2. Figure 16(c) shows that the paths have deviated, but all data are almost twice as large as Map 2. The grid size of Map 4 is further reduced compared with Map 3, the modeling accuracy is further improved, and the difficulty of the path search process is further increased, and the path length and the number of turns of the leader are increased by 61.9% and reduced by 118.2%, respectively. Although the higher precision modeling increases the search difficulty of our algorithm, a formation path that matches the natural environment can be obtained.

Table 6. Formation path results for multiple robots with different accuracy maps.

Map	Multi-Robot	path length	Number of turns in formation	The starting and ending point of leader	Algorithm runtime	Grid edge length	Obstacle expansion rate
1	Leader ant Follower ant1 Follower ant2	63.8894 63.8595 63.8268	18	(10.5, 3.5) (47.5, 46.5)	6.1803	10	20%
2	Leader ant Follower ant1 Follower ant2	66.9142 66.8997 66.3180	12	(10.5, 3.5) (47.5, 46.5)	5.7537	10	5%
3	Leader ant Follower ant1 Follower ant2	136.3940 135.8450 135.4780	24	(20.5, 5.5) (95.5, 95.5)	14.7877	5	5%
4	Leader ant Follower ant1 Follower ant2	225.1826 226.9694 226.6619	19	(30.5, 6.5) (157.5, 159.5)	34.9412	3	5%

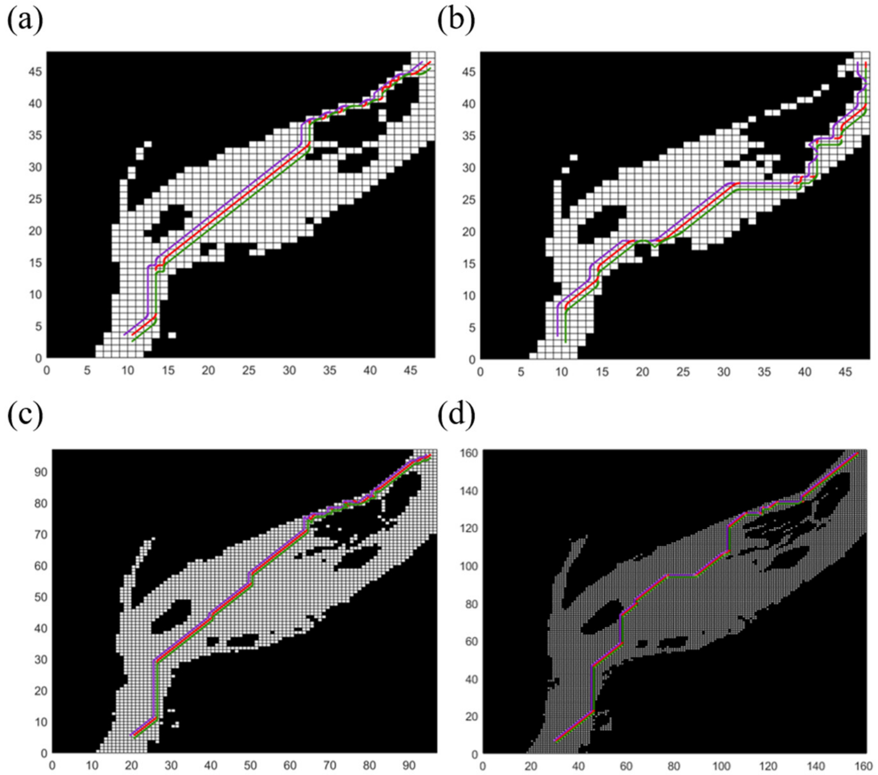

Figure 16. Multi-water robot surface navigation map. (a) 48 × 48 Scale Map. (b) 48 × 48 Scale Map. (c) 97 × 97 Scale Map. (d) 161 × 161 Scale Map.

In the experiments, we found that with the improvement of modeling accuracy, the optimization search of the ant colony under large-scale maps became difficult. In Maps 1–3, the algorithm can find the optimal solution quickly, and the paths have good metrics. However, a poor path is often obtained when run in Map 4 at 161 × 161 scale. The formation path in Figure 16(d) is the optimal path we obtained after conducting more than ten experiments. Since we did not improve the ACO for large-scale maps, we did not obtain a 483×483 scale map path. In future research, we should further study intelligent optimization algorithms, such as the ant colony algorithm [20] and particle swarm algorithm [22], combined with accurate map modeling to expand the application of mobile robots under large-scale maps.

6.5. Multi-robot formation path planning experiments in ROS environment

We proposed an improved ant colony fusion Leader-Follower formation path planning algorithm. In this section, we constructed a ROS experimental platform to study the cooperative control strategies for multiple mobile robots, where the blue and red lines representing the x and y axes of the ROS world coordinate system, the green box is used to fix the ROS world coordinate system and ensure that the motion state of the robot is continuous. The platform consisted of three four-wheeled robot carts and an indoor environment, each equipped with sensors for path tracking, and a hybrid wireless LAN communication structure was used to achieve information interaction between multiple robots. In order to verify the effectiveness and rationality of the algorithm, a more realistic working environment was constructed, and the three mobile robots were subjected to experiment to verify the path tracking, formation keeping, and switching capabilities of the robots. The starting point of the leader robot is (-2m, -1.5m) and the ending point is (3m, 4m), and the φ_d and l_d of the following robot 1 and the leader robot are 5/4 Π and 1, respectively, and the φ_d and l_d of the following robot 2 and the leader robot are -1/4 Π and 1, respectively. The parameters of the ant colony algorithm are shown in Table 2. The robot's maximum speed is 1.2m/s, the maximum angular speed is 1.5rad/s, and the sampling period is 0.01s.

The experiment starts with a global path planning of multiple robots in advance in this environment, followed by multiple robots running along the planned path from the starting position. When the leading robot encounters an obstacle in the tracking path, the leading robot turns at a more uniform speed. The following robots avoid the obstacle by switching their formation by tracking the position and angle of the leader and resume the original formation after the obstacle avoidance is completed. In this obstacle avoidance approach, we can see that the formation of robots changes, and each robot's linear velocity and angle change significantly. The motion state information of the robots at different moments is shown in Figures 17–19.

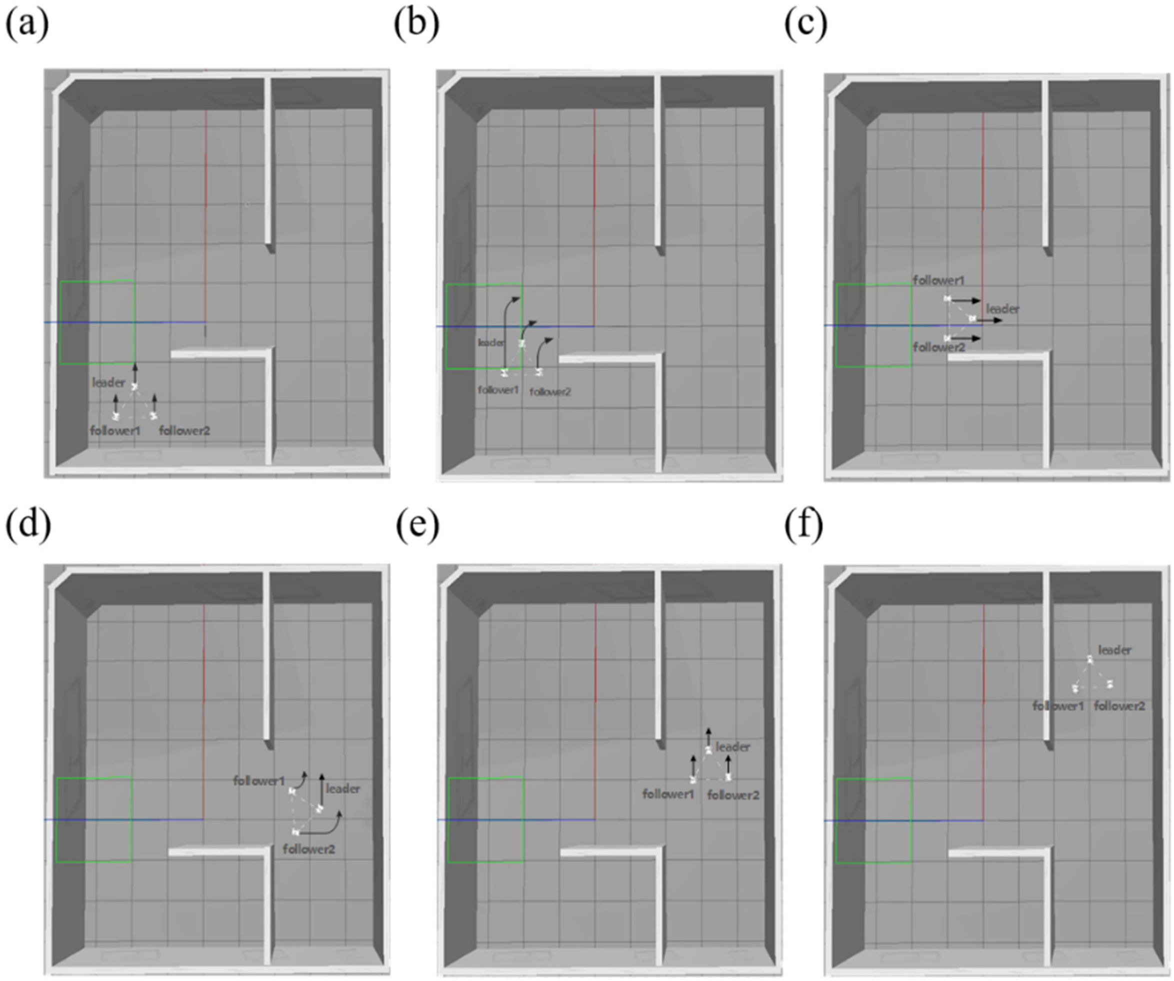

Figure 17. Graph of ROS experimental results in complex environment. (a) t = 0s initial position. (b) t = 11s Obstacle avoidance process 1 (c) t = 20s Obstacle avoidance process 2. (d) t = 40s Obstacle avoidance process 3. (e) t = 48s Obstacle avoidance process 4. (f) t = 64s Reach the target location.

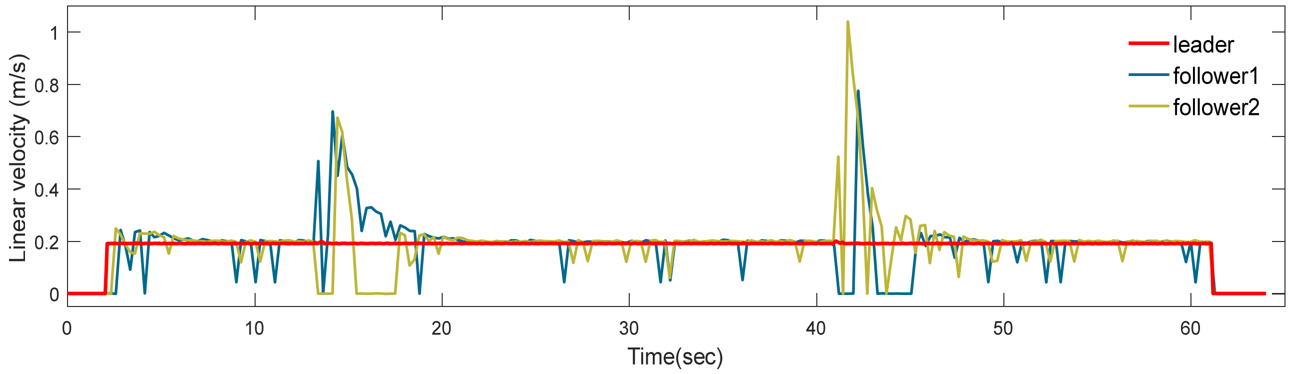

Figure 18. Linear velocity diagram.

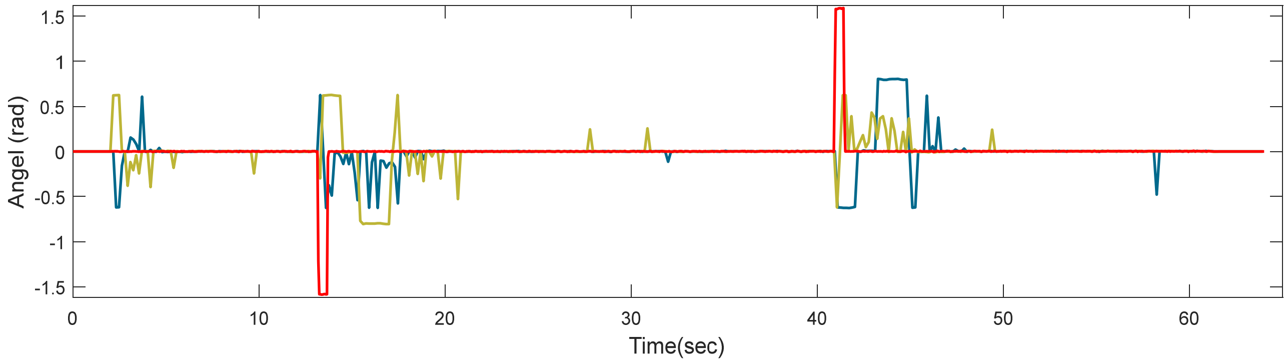

Figure 19. Angel diagram.

7. Conclusions

This study aims to find a new formation path planning method. A multi-robot collaborative path strategy based on the improved ant colony algorithm fused with the leader-follower method is proposed. We improve the distance factor heuristic function by amplifying the attraction of the target point to the ants when they transfer based on the greedy algorithm idea. We propose the smoothing factor heuristic function to solve the difficulty that most optimization algorithms need to remove the redundant turning points later to smooth the path. In addition, we establish a formation model for leader ants and follower ants to achieve cooperative obstacle avoidance based on the leader-follower formation control method and update the pheromone by the overall characteristics of the formation. To prevent the algorithm from falling into local optimal solutions, the maximum-minimum ant strategy is used to enhance the global search capability of the algorithm. Moreover, we performed a secondary smoothing optimization to improve the path quality, used a turning point optimization algorithm to remove the redundant turning points of the path, and performed circular smoothing by a dynamic tangent point algorithm. Finally, simulation experiments based on Matlab and ROS verify the effectiveness and practicality of the algorithm in this paper.

In the subsequent research, we can also consider other optimization objective problems (e.g., energy consumption, time) for the ant colony algorithm to further investigate the multi-objective formation problem for multi-robot systems. In addition, we can use neural network algorithms to calculate the parameters and weights of each objective to make the planning more intelligent.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61163051) and the Yunnan Provincial Key R & D Program Project “Research on key technologies of industrial robots and its application demonstration in intelligent manufacturing” (202002AC080001).

Conflict of interest

The authors declare no conflict of interest.

References

1. S. K. Tse, Y. B. Wong, J. Tang, P. Duan, S. W. W. Leung, L. Shi, Relative state formation-based warehouse multi-robot collaborative parcel moving, in *2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, IEEE, (2020), 375–380. doi: 10.1109/ICPS49255.2021.9468127.
2. J. P. Queralta, J. Taipalmaa, B. C. Pullinen, V. K. Sarker, T. N. Gia, H. Tenhunen, et al., Collaborative multi-robot search and rescue: planning, coordination, perception, and active vision, *IEEE Access*, **8** (2020), 191617–191643. doi: 10.1109/ACCESS.2020.3030190.
3. Z. Li, Y. Yuan, F. Ke, W. He, C. Y. Su, Robust vision-based tube model predictive control of multiple mobile robots for leader-follower formation, *IEEE Trans. Ind. Electro.*, **67** (2019), 3096–3106. doi: 10.1109/TIE.2019.2913813.
4. C. Y. Kim, D. Song, Y. Xu, J. Yi, X. Wu, Cooperative search of multiple unknown transient radio sources using multiple paired mobile robots, *IEEE Trans. Rob.*, **30** (2014), 1161–1173. doi: 10.1109/TRO.2014.2333097.
5. L. Z. Du, S. Ke, Z. Wang, J. Tao, L. Yu, H. Li, Research on multi-load AGV path planning of weaving workshop based on time priority, *Math. Biosci. Eng*, **16** (2019), 2277–2292. doi: 10.3934/mbe.2019113.
6. J. Han, Y. Chen, Multiple UAV formations for cooperative source seeking and contour mapping of a radiative signal field, *J. Intell. Rob. Syst.*, **74** (2014), 323–332. doi: 10.1007/s10846-013-9897-4.
7. X. Zhang, J. Wang, Y. Fang, J. Yuan, Multilevel humanlike motion planning for mobile robots in complex indoor environments, *IEEE Trans. Autom. Sci. Eng.*, **16** (2018), 1244–1258. doi: 10.1109/TASE.2018.2880245.
8. B. Patle, A. Pandey, D. Parhi, A. Jagadeesh, A review: on path planning strategies for navigation of mobile robot, *Def. Technol.*, **15** (2019), 582–606. doi: 10.1016/j.dt.2019.04.011.
9. X. Wang, X. Li, Z. Zheng, Survey of developments on multi-agent formation control related problems, *Control Decis. (Chin.)*, **28** (2013), 1601–1613. doi: 10.13195/j.kzyjc.2013.11.026.
10. Y. Gu, B. Seanor, G. Campa, M. R. Napolitano, L. Rowe, S. Gururajan, et al., Design and flight testing evaluation of formation control laws, *IEEE Trans. Control Syst. Technol.*, **14** (2016), 1105–1112. doi: 10.1109/TCST.2006.880203.
11. Y. He, M. Wu, S. Liu, An optimisation-based distributed cooperative control for multi-robot manipulation with obstacle avoidance, *IFAC Pap. Online*, **53** (2020), 9859–9864. doi: 10.1016/j.ifacol.2020.12.2691.
12. G. Liu, C. Shu, Z. Liang, B. Peng, L. Cheng, A modified sparrow search algorithm with application in 3d route planning for UAV, *Sensors*, **21** (2021), 1224. doi: 10.3390/s21041224.
13. A. Chhillar, A. Choudhary, *Mobile Robot Path Planning Based Upon Updated Whale Optimization Algorithm*, in *2020 10th International Conference on Cloud Computing*, *Data Science & Engineering (Confluence)*, IEEE, (2020), 684–691. doi: 110.1109/Confluence47617. 2020.9058323.
14. M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, *IEEE Trans. Cybern.*, **26** (1996), 29–41. doi: 10.1109/3477.484436.
15. Q. Luo, H. Wang, Y. Zheng, J. He, Research on path planning of mobile robot based on improved ant colony algorithm, *Neural Comput. Appl.*, **32** (2020), 1555–1566. doi: 10.1007/s00521-019-04172-2.

16. J. Liu, J. Yang, H. Liu, X. Tian, M. Gao, An improved ant colony algorithm for robot path planning, *Soft Comput.*, **21** (2017), 5829–5839. doi: 10.1007/s00500-016-2161-7.
17. X. You, S. Liu, C. Zhang, An improved ant colony system algorithm for robot path planning and performance analysis, *Int. J. Rob. Autom.*, **33** (2018), 527–533. doi: 10.2316/Journal.206.2018.5.206-0071.
18. X. Dai, S. Long, Z. Zhang, D. Gong, Mobile robot path planning based on ant colony algorithm with A* heuristic method, *Front. Neurorobotics*, **13** (2019), 15. doi: 10.3389/fnbot.2019.00015.
19. Z. Jiao, K. Ma, Y. Rong, P. Wang, H. Zhang, S. Wang, A path planning method using adaptive polymorphic ant colony algorithm for smart wheelchairs, *J. Comput. Sci.*, **25** (2018), 50–57. doi: 10.1016/j.jocs.2018.02.004.
20. K. Akka, F. Khaber, Mobile robot path planning using an improved ant colony optimization, *Int. J. Adv. Rob. Syst.*, **15** (2018). doi: 10.1177/1729881418774673.
21. J. Xin, M. Chuang, S. Frederik, P. Jinzhu, L. Yanhong, R. R. Negenborn, A time-space network model for collision-free routing of planar motions in a multirobot station, *IEEE Trans. Ind. Inf.*, **16** (2020), 6413–6422. doi: 10.1109/TII.2020.2968099.
22. P. K. Das, S. B. Himansu, K. P. Bijaya, A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning, *Swarm Evol. Comput.*, **28** (2016), 14–28. doi: 10.1016/j.swevo.2015.10.011.
23. P. K. Das, K. J. Prabir, Multi-robot path planning using improved particle swarm optimization algorithm through novel evolutionary operators, *Appl. Soft Comput.*, **92** (2020), 106312. doi: 10.1016/j.asoc.2020.106312.
24. M. Nazarahari, E. Khanmirza, S. Doostie, Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm, *Expert Syst. Appl.*, **115** (2018), 106–120. doi: 10.1016/j.eswa.2018.08.008.
25. R. K. Dewangan, A. Shukla, W. W. Godfrey, A solution for priority-based multi-robot path planning problem with obstacles using ant lion optimization, *Mod. Phys. Lett. B*, **34** (2020), 2050137. doi: 10.1142/S0217984920501377.
26. Y. Zheng, X. Li, D. Duan, A path planning method to maintain the team formation of Agent, *Comput. Technol. Dev. (Chin.)*, **19** (2009), 159–162. doi: 10.3969/j.issn.1673-629X.2009.07.046.
27. T. Jiang, Z. Zhang, Z. Cheng, J. Li, J. Lu, Improved robot formation method with genetic algorithm and pilot-following method, *Comput. Eng. Appl. (Chin.)*, **56** (2020), 240–245. doi: 10.3778/j.issn.1002-8331.1906-0376.
28. Y. Dong, C. Fu, E. Kayacan, RRT-based 3D path planning for formation landing of quadrotor UAVs, in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, IEEE, (2016), 1–6. doi: 10.1109/ICARCV.2016.7838567.
29. S. Liu, D. Sun, C. Zhu, A dynamic priority based path planning for cooperation of multiple mobile robots in formation forming, *Rob. Comput. Integr. Manuf.*, **30** (2014), 589–596. doi: 10.1016/j.rcim.2014.04.002.
30. D. Sun, C. Wang, W. Shang, G. Feng, A synchronization approach to multiple mobile robots in switching between formations, *IEEE Trans. Rob.*, **25** (2009), 1074–1086. doi: 10.1109/TRO.2009.2027384.
31. L. Wang, Z. Sui, Z. PU, Z. Liu, J. Yi, An improved RRT algorithm for multi-robot formation path planning, *Acta Electonica Sin. (Chin.)*, **48** (2020), 2138. doi: 10.3969/j.issn.0372-2112.2020.11.007.

32. H. Wang, C. Hao, P. Zhang, M. Zhang, P. Yin, Y. Zhang, Path planning of mobile robots based on A* algorithm and artificial potential field algorithm, *China Mech. Eng. (Chin.)*, **30** (2019), 2489. doi: 10.3969/j.issn.1004-132X.2019.20.012.
33. W. Zhang, Y. Ma, H. D. Zhao, L. Zhang, Y. Li, X. D. Li, Obstacle avoidance path planning of intelligent mobile based on improved fireworks-ant colony hybrid algorithm, *Control Decis.*, **34** (2019), 335–343. doi: 10.13195/j.kzyjc.2017.0870.
34. H. Yang, J. Qi, Y. Miao, H. Sun, J. Li, A new robot navigation algorithm based on a double-layer ant algorithm and trajectory optimization, *IEEE Trans. Ind. Electron.*, **66** (2018), 8557–8566. doi: 10.1109/TIE.2018.2886798.
35. H. Zhang, L. He, L. Yuan, T. Ran, Path planning for mobile robots based on improved two-layer ant colony algorithm, *Control Decis. Making (Chin.)*, (2021), 1–10. doi: 10.13195/j.kzyjc.2020.0610.