



# An enhanced heuristic ant colony optimization for mobile robot path planning

Wenxiang Gao<sup>1</sup> · Qing Tang<sup>1</sup> · Beifa Ye<sup>1</sup> · Yaru Yang<sup>1</sup> · Jin Yao<sup>1</sup>

Published online: 10 February 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

To realize a fast and efficient path planning for mobile robot in complex environment, an enhanced heuristic ant colony optimization (EH-ACO) algorithm is proposed. Four strategies are introduced to accelerate the ACO algorithm and optimize the final path. Firstly, the heuristic distance in the local visibility formula is improved by considering the heuristic distance from ant's neighbor points to target. Secondly, a new pheromone diffusion gradient formula is designed, which emphasizes that pheromones left the path would spread into a region and the pheromone density would present a gradient distribution in the region. Thirdly, backtracking strategy is introduced to enable ants to find new path when their search is blocked. Finally, path merging strategy is designed to further obtain an optimal path. Simulations are carried out to verify each individual strategy, and comparisons are made with the state-of-the-art algorithms. The results show our proposed EH-ACO algorithm outperforms other algorithms in both optimality and efficiency, especially when the map is large and complex.

**Keywords** Ant colony optimization · Enhanced heuristic · Pheromone diffusion · Path planning

## 1 Introduction

Nowadays, mobile robots which could adapt to the harsh environment and reduce the labor intensity of people are widely used in manufacturing and service industries. Many topics about mobile robots have been studied, such as path planning, simultaneous localization and mapping (SLAM), navigation, trajectory tracking and control (Friudenberg and Koziol 2018; Li and Savkin 2018; Luo and Hsiao 2019; Miah et al. 2018; Patle et al. 2019). Among them, the automatic collision-free path planning algorithms have attracted much attention in recent years (Mac et al. 2016). As a swarm intelligence heuristic algorithm (Mavrovouniotis et al. 2017), ant colony optimization (ACO) has been successfully applied to solve the path planning problems in complex environment (Deneubourg et al. 1994; Dorigo et al. 1996; Dorigo and Di Caro 1999). The ACO algorithm is easy to design (Deng et al. 2015) and has good global optimization ability (Liu

et al. 2017) and robustness. Moreover, it could even be well integrated into other algorithms (Lee et al. 2008).

However, current ACO algorithms still have some drawbacks. Firstly, it is difficult to balance the efficiency and optimality. For example, some algorithm only focused on promoting the efficiency of the algorithm, but failed in guaranteeing the optimality of the ACO algorithm, and vice versa. Secondly, the efficiency of some existing ACO algorithm is still low when deployed in the complex environment. Unfortunately, environments in real applications are usually cluttered with dense obstacles, narrow passages, concavities, mazes and other features. The planning efficiency problem in these environments should not be neglected.

In this paper, an enhanced heuristic ACO algorithm (EH-ACO) is proposed. The improvement and innovation of the algorithm include the following four aspects, among which (1) promotes the efficiency of the EH-ACO algorithm, (2) and (4) are used to enhance the global optimization ability of the algorithm, and (3) is used to raise the planning success rate of the algorithm in complex environment.

- (1) Heuristic distance in local visibility formula is redefined. Not only the distance from current position of the ant to target point, but also the heuristic distance from ant's neighbor points to target point would be considered. This

---

Communicated by A. Di Nola.

✉ Qing Tang  
tangqing\_scu@163.com

<sup>1</sup> School of Mechanical Engineering, Sichuan University, Chengdu 610065, China

strategy could raise the probability of ants' searching for target and improve the efficiency of path planning.

- (2) A new pheromone diffusion gradient formula is designed, which emphasizes that pheromones left the path would spread into a region and the pheromone density would present a gradient distribution in the region. This strategy could improve global planning ability of the ACO algorithm by enhancing the role of pheromones in the environment.
- (3) Backtracking strategy is introduced to reuse the previous feasible path and avoid information losses. The ants are encourages to trace back to some previous location of the feasible path and start a new search, when they were blocked in some local environment. This strategy could improve the success rate of ant planning in complex environments, thus enhancing the adaptability of ants to the environment.
- (4) Path merging strategy is designed to concatenate different optimal path segments from different ants to obtain a global optimal path, since it is difficult to find the global optimal path using one single ant. This strategy greatly demonstrates the cooperative characteristics among ants.

The remainder of this paper is organized as follows: Sect. 2 reviews the state-of-the-art ACO algorithms on point-to-point (PTP) collision-free path planning. The EH-ACO algorithm is proposed in Sect. 3, including the above four improved strategies. Simulation is carried out to verify the proposed algorithm in Sect. 4. Finally, conclusions and further work are given in Sect. 5.

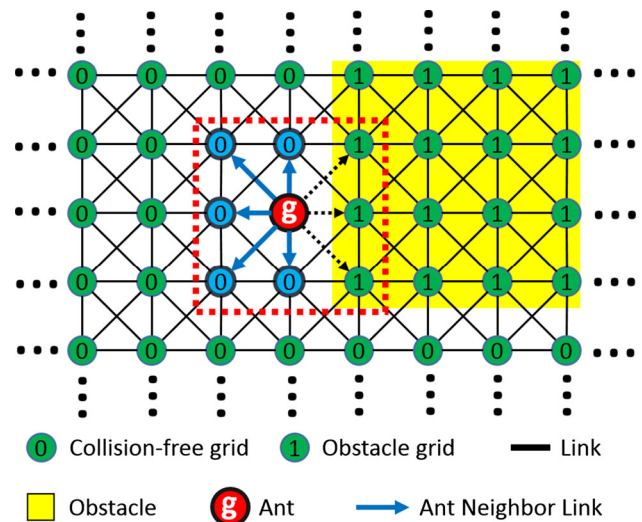
## 2 Related work on ACO solving PTP problem

Generally, the path planning problems are divided into two categories: point-to-point (PTP) path planning problems and multi-point tour problems (TSP). In this paper, we mainly focus on PTP collision-free path planning.

### 2.1 Traditional ACO algorithm

The ACO algorithm is used to solve the min-cost path planning problem by imitating the foraging behavior of real ant colony. When searching for food source, ants determine their movement direction by local visibility information and environmental pheromone intensity information. After finding food, ants would leave some pheromone on the route to attract and guide other ants. The pheromone strategy is a critical factor of the ACO, which is the medium of communication between different ants.

In each moving step, ant would find a feasible neighbor grid to which it could move. As shown in Fig. 1, the ant is located in grid  $g$ , and it has eight neighbor nodes, in which



**Fig. 1** The environment is divided into  $M \times N$  connected grids. The yellow region is obstacle space, and the nodes in this place are denoted by “1.” The other nodes are located in collision-free space and are denoted by “0.” The ant is located in grid  $g$ , and it has eight neighbor nodes, in which the five blue nodes are feasible options, and the other three grids are occupied by obstacles

the five blue nodes are feasible options. Then all the feasible grids would be evaluated by using local visibility and the pheromone intensity. Finally, the ant moves to one of these grids according to the probability related to the evaluations.

Assuming that ant  $k$  has moved from grid  $g_i$  to  $g_j$  at time  $t$ , then the unvisited feasible neighbor grids are defined as:

$$N_k(g_j) = \left\{ g_j \mid \begin{array}{l} g_j \in W_{\text{free}}, g_j \notin \text{tabu}_k, \\ 1 \leq d(g_i, g_j) \leq \sqrt{2} \end{array} \right\}, \quad (1)$$

where  $W_{\text{free}}$  is the collision-free space of the environment,  $\text{tabu}_k$  includes the visited grids of ant  $k$ ,  $d(g_i, g_j)$  is the distance from grid  $g_i$  to  $g_j$ , and the distance could be defined as:

$$d(g, g_i) = \begin{cases} 1, & \text{in horizontal or vertical} \\ \sqrt{2}, & \text{otherwise.} \end{cases} \quad (2)$$

The state transition probability function  $p$  of the ant  $k$  moving from grid  $g_i$  to neighbor grid  $g_j$  is defined as:

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_j^\alpha(t) \eta_{ij}^\beta}{\sum_{s \in N_k(g_i)} \tau_s^\alpha(t) \eta_{is}^\beta}, & \text{if } g_j \in N_k(g_i) \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where  $\tau_j(t)$  is the pheromone intensity of grid  $g_j$  in time  $t$  and  $\eta_{ij}$  is the local visibility from grid  $g_i$  to grid  $g_j$ .  $\alpha$  is the coefficient of pheromone intensity, and  $\beta$  is the coefficient of the route cost.

At the end of each iteration, the pheromone intensity of each grid needs to be updated according to the ant's move-

ment. The updating formula is defined as follows:

$$\tau_i(t+1) = (1 - \rho) \tau_i(t) + \sum_{k=1}^m \Delta\tau_i^k, \quad (4)$$

where  $\rho$  is the pheromone evaporation rate and  $m$  is the number of ants.  $\Delta\tau_i^k$  is the pheromone intensity of the ant  $k$  leaves on grid  $g_i$ , which is typically given by:

$$\Delta\tau_i^k = \begin{cases} \frac{Q}{L_k}, & \text{if ant } k \text{ found food and passed grid } g_i \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where  $Q$  is a constant and  $L_k$  is the length of the route of ant  $k$ .

Finally, when a collision-free path is found, then the path could be represented by a series of adjacent collision-free grids:

$$\text{Path} = \{g_{\text{start}}, g_2, \dots, g_i, \dots, g_{\text{target}}\}, g_{\text{target}} \in W_{\text{free}}. \quad (6)$$

The length of the path  $\text{Len}(L)$  could be calculated using:

$$\text{Len}(L) = \sum_{i=1}^{n-1} d(g_i, g_{i+1}). \quad (7)$$

## 2.2 Review of existing improved ACO algorithm

Many improved ACO algorithms have been proposed to solve PTP path planning problems.

Pheromone is a bridge for ants to communicate and cooperate with each other. Some literature has studied pheromone update mechanism of ACO algorithm. Mou et al. (2008) introduced some specially designed variables to the pheromone update formula (formula (3)) to modify the original transfer probability. Besides, the pheromone evaporation rate  $\rho$  is replaced by introducing an operation function. The modified ACO algorithm could improve the efficiency of path selection and avoid the local optimum. Garcia et al. (2009) proposed an ACO meta-heuristic algorithm called SACOdm by introducing the information of Euclidian distance between the source and target nodes into the pheromone updating formula. Besides, the memory information was saved to avoid the algorithm degradation. Seçkiner et al. (2013) proposed a novel pheromone updating scheme for finding global minimum. The pheromone updates are performed based on the percentage of all ants searching for the optimal solution. Deng et al. (2015) proposed pheromone mark ACO (PM-ACO) and declared that the pheromones are left on node or path could influence its domain including  $k$ -nearest neighbor (KNN) nodes. “The  $r$ -best-node update rule” which introduces a variable  $r$  to control the node selection probability and “the relevant-node depositing rule”

which is a new pheromone adjusting strategy are introduced. Yu et al. (2017) added pheromone to optimal path when the global pheromone is updated at the end of each cycle. The algorithm reduces the search time by removing redundant path and has faster convergence speed. Yuan et al. (2019) enhanced pheromone of ants by introducing parameters into pheromone update formula, so as to speed up convergence. All these improved studies have achieved good results.

In addition to the research on the pheromone update mechanism, some other studies also consider combining the ACO algorithm with other algorithms to improve the planning efficiency of the ACO algorithm, such as combining ACO with artificial potential field (APF) (Liu et al. 2017; Chen and Liu 2019), introducing the mutation operator to ACO (Yang and Zhuang 2010), combining modified ACO (M-ACO) with Voronoi diagram (VD) (Habib et al. 2016) and combining ACO with particle swarm optimization (ACO-PSO) (Gigras et al. 2015).

In other aspects, Chen et al. (2013) used the “scent pervasion” rule to make a preprocess on the map and then used the “1 minus” search policy to finish the planning. This algorithm improves the efficiency of path planning, which can be adapted to most of the environments. However, the map preprocess would be time-consuming in the case of a large number of map grids existed, because each grid would be visited in the preprocessing. Cekmez et al. (2016) proposed multi-colony ACO to find a global optimal solution by considering the cooperation between ants. Zhu and Wang (2008) proposed a new ant algorithm based on scout ant cooperation (SAC). The algorithm makes each ant searches for some steps. Then all ants use the end of the path as a new starting point and search again. This process will be repeated until the goal is reached. The algorithm can adapt to complex environment, but the final path is not optimal. After that, Zhu et al. (2011) proposed a robot navigation ant (RNA) algorithm. The algorithm mapped the target position into the closest robot’s visual domain node and then made a local sub-goal path planning. Besides, Kumar et al. (2018) proposed an RA-ACO-based approach for navigation of a humanoid robot. They tested the RA-ACO approach for navigation of humanoid robots in both simulated and experimental environments and obtained a good agreement between them.

Although these existing studies have both improved the efficiency of the ACO algorithm and considered the global optimality, the efficiency and global optimality of the algorithm still need to be further improved. Firstly, some planners only improve one aspect of ACO algorithm, but pay little attention to the other evaluation indicators. When efficiency and optimality could not be achieved at the same time, they should be balanced in the algorithm. Secondly, although some modified strategies have been integrated into the algorithm to optimize the solution, reducing the possibility of falling into local optimum is still encouraged for the ACO

algorithm. Thirdly, the ACO algorithm is of low efficiency in complex environment, because most of the ants fail to find the food. To solve the above problems, this paper proposed an enhanced heuristic ACO algorithm with four strategies. The efficiency, optimality and adaptability of the ACO algorithm are improved.

### 3 Enhanced heuristic ACO algorithm

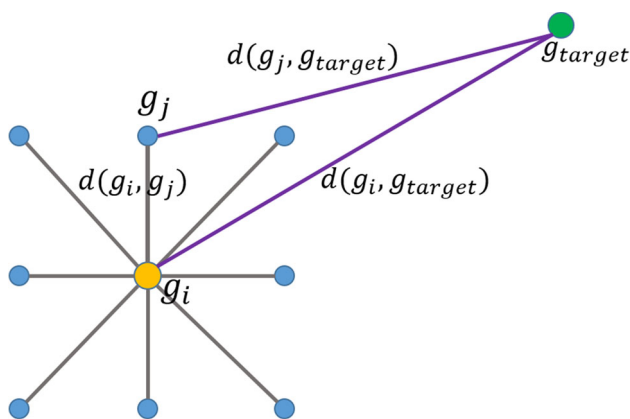
The enhanced heuristic ACO algorithm proposed in this paper has been improved in four aspects: pheromone update mechanism, pheromone diffusion model, backtracking mechanism and path merge mechanism. In the following sections, each improved strategies will be described, respectively.

#### 3.1 Local visibility enhancement

Local visibility has a great influence on the search capability of the ACO algorithm, especially at the beginning of iteration where the pheromone intensity of the environment is almost the same. The existing local visibility formula could be given as follows:

$$\eta_{ij} = \frac{1}{d(g_i, g_j) + d(g_j, g_{\text{target}})}, \quad (8)$$

where  $d(g_i, g_j)$  is the actual distance between current grid  $g_i$  and its neighbor grid  $g_j$ , and  $d(g_j, g_{\text{target}})$  is the heuristic distance from neighbor grid  $g_j$  to target grid  $g_{\text{target}}$ . As shown in Fig. 2, the local visibility model is provided. It could be found that the existing local visibility can be denoted by using the cost of  $d(g_i, g_j)$  and  $d(g_j, g_{\text{target}})$ .



**Fig. 2** Local visibility model.  $g_i$  is the current grid where ant located in, and  $g_j$  is the neighbor grid of  $g_i$ .  $d(g_i, g_j)$  is the actual distance between current grid  $g_i$  and its neighbor grid  $g_j$ .  $d(g_j, g_{\text{target}})$  is the heuristic distance from the neighbor grid  $g_j$  to the target grid  $g_{\text{target}}$ .  $d(g_i, g_{\text{target}})$  is the heuristic distance from current grid  $g_i$  to the target grid  $g_{\text{target}}$

Analyzing the existing local visibility formula, the cost of  $d(g_i, g_j)$  is far less than the cost of  $d(g_j, g_{\text{target}})$  at the beginning because the ant is far from target  $g_{\text{target}}$ . Therefore, according to formula (8), there is little difference between neighbor grids, and the heuristic information of the target position is weak, which is the main reason of low efficiency in ant's searching.

To enhance the heuristic information of the target, a new enhanced heuristic local visibility formula is proposed, as follows:

$$\eta_{ij} = \frac{1}{d(g_i, g_j) + d(g_j, g_{\text{target}}) - d(g_i, g_{\text{target}}) + \varepsilon}, \quad (9)$$

where  $\varepsilon$  is arbitrary constant,  $\varepsilon > 0$ . The parameter  $\varepsilon$  is used to ensure that the denominator is not equal to zero, and it also could be used to adjust the intensity of heuristic information. In the modified formula (9), the heuristic cost  $d(g_i, g_{\text{target}})$  between current grid  $g_i$  and target grid  $g_{\text{target}}$  is introduced to increase the distance difference of neighbor grids of grid  $g_i$ . After improvement, the grid which is closer to the target in orientation would have a higher probability to be selected. Therefore, the ant would search toward the target with high probability, and the efficiency would be improved.

#### 3.2 Pheromone diffusion

As a bridge of communication between ants, pheromone plays an important role in path planning. In most existing ACO algorithms, the pheromone only remains on the path that ants passed through. Therefore, when only the ants are close enough to the existing path, they could perceive the information. It limits the search ability of ants. As a result, the ant would be easy to fall into local optimum.

In order to improve this situation, some paper (Deng et al. 2015) has pointed out that pheromone left by ants on the path would spread into its neighbor grids. This new pheromone diffusion model is similar to the diffusion of pigment liquid drops on white paper. The pheromone diffusion model could be an arbitrary gradient function. In this paper, we present a new pheromone diffusion model. Figure 3a illustrates the model of unit pheromone diffusion. Assume that pheromone intensity is equal to one at the ant's current location. Then a half intensity of pheromone would spread into ant's neighbor grids. Therefore, the new pheromone diffusion gradient formula is designed as follows:

$$\nabla f(g, g_0) = \begin{cases} 1, & d(g, g_0) = 0 \\ 0.5, & 1 \leq d(g, g_0) \leq \sqrt{2}, \quad (g \in W_{\text{free}}) \\ 0, & d(g, g_0) > \sqrt{2}, \end{cases} \quad (10)$$

where  $g_0$  is the grid where ant is located. Then, the pheromone of ant  $k$  remain in the environment along a path  $P_k(g_1, g_2, \dots, g_n)$  is a convolution of pheromone diffusion model and the pheromone intensity. Therefore, the new pheromone intensity formula is designed by:

$$\Delta\tau^k(g) = \sum_{i=1}^n \frac{Q}{L_k} \nabla f(g, g_i), (g \in W_{\text{free}}). \tag{11}$$

In Fig. 3b, an example of pheromone diffusion is given when ant  $k$  passed through the path. The pheromone intensity of the grid is higher when many ants passed through. The improved pheromone diffusion mechanism not only enhances the role of pheromones, but also smooths the pheromone distribution. It is helpful to improve the global planning ability of the algorithm.

$$\begin{aligned} \text{Path}_1 &= \{\text{Sub}P_1(s_1, s_2), \text{Sub}P_1(s_2, s_3), \dots, \text{Sub}P_1(s_{n-1}, s_n)\} \\ \text{Path}_2 &= \{\text{Sub}P_2(s_1, s_2), \text{Sub}P_2(s_2, s_3), \dots, \text{Sub}P_2(s_{n-1}, s_n)\} \\ \text{Sub}P_{\text{new}}(s_i, s_{i+1}) &= \begin{cases} \text{Sub}P_1(s_i, s_{i+1}), & \text{if } \text{Len}(\text{Sub}P_1(s_i, s_{i+1})) \leq \text{Len}(\text{Sub}P_2(s_i, s_{i+1})) \\ \text{Sub}P_2(s_i, s_{i+1}), & \text{otherwise} \end{cases}, i = 1, 2, \dots, n - 1. \end{aligned} \tag{12}$$

### 3.3 Backtracking mechanism

The TABU table which is used to prevent the repeated searching is important in ACO algorithm. Some researchers make all ants share the same TABU table. However, this strategy brings a disadvantage of prematurely searching stop; we call it the ‘‘Lost Ant.’’ Many experiments have shown that ants may fall into the situation where there is no path to walk because its neighbor grids are occupied by obstacles or have been visited before. This is the reason that some ants fail to find food source. To overcome this shortcoming, backtracking strategy is introduced to avoid ant stagnation, so that the ‘‘Lost Ant’’ could move back to the previous grid that it has visited before when there is no path to go.

Figure 4 gives an ant’s search process from start grid ① to stop grid ⑨, and the path sequence: ① → ① → ... → ⑨. The ant would stop walking due to constraints of obstacles and TABU table. However, when backtracking strategy is introduced, the ant would take a step backward along the path until new grids are found, such as grid (a) and grid (b). Thus, a new path may be found ① → ① → ... → ⑦ → (a) or ① → ① → ... → ⑦ → (b), and the ant would start a new search by grid (a) and grid (b). It is useful for ants to search in complex environments.

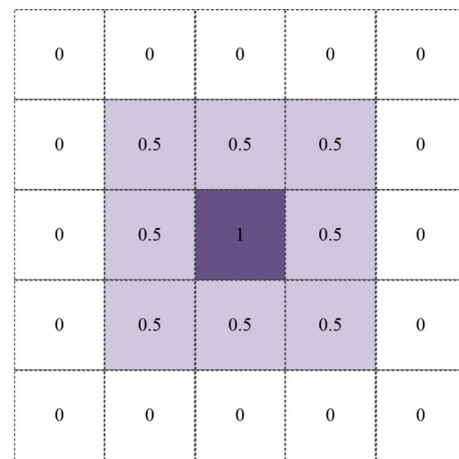
Different from other studies which solve the optimization problem to obtain a better optimization scheme (Benhala et al. 2015; Li and Tian 2006) or revisit the path node (Hee-

gaard and Wittner 2006), backtracking strategy proposed in this paper is specially designed for robot collision-free path planning to optimize the selection of extension nodes. With backtracking strategy, the ants would not easily fail searching in complex environment such as concave obstacle because of the influence of TABU table.

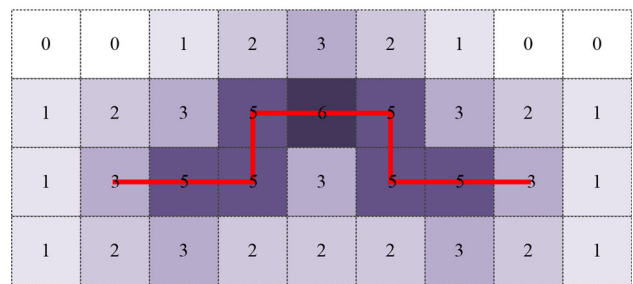
### 3.4 Path merging

Although each ant is hard to find the global optimal path, but one or more parts of the searched path could belong to the global optimal path. Therefore, the path merging strategy is put forward to further optimize the final path. This strategy could concatenate any two optimal sub-paths to form a better path.

Let  $\text{Path}_1$  and  $\text{Path}_2$  denote two different paths, and assume that they had passed a number of common grids called set

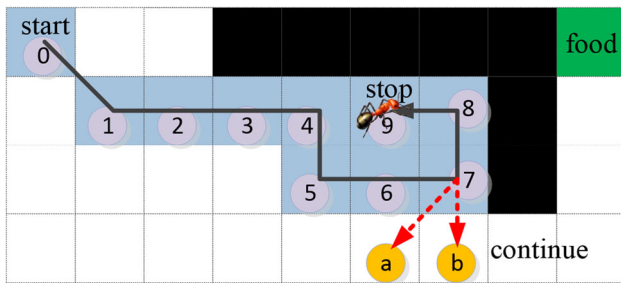


(a)

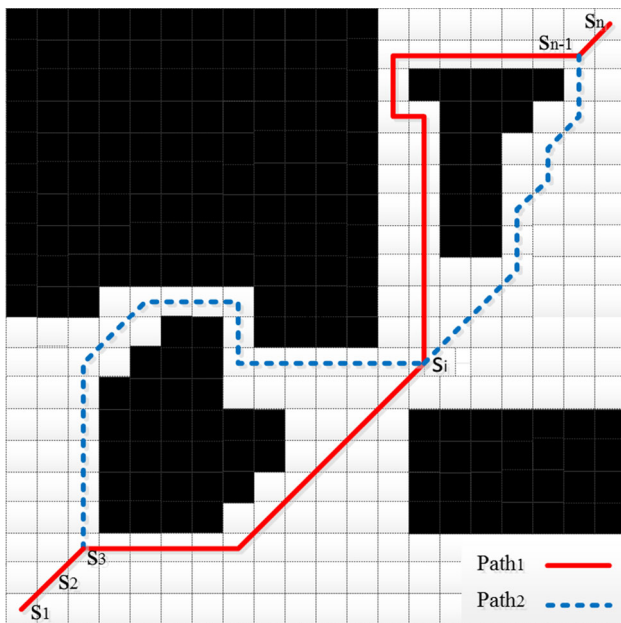


(b)

Fig. 3 a Diffusion model of unit pheromone intensity and b example of pheromone diffusion. The solid line is the ant path, and the values are the pheromone intensity



**Fig. 4** Backtracking model. The black grids are collision grids, and the light blue grids are the tube table. When the ant searched to the stop grid “(9),” it will backtrack and continue the new searching. The grid (a) and grid (b) are the new found



**Fig. 5** Path merging strategy, where  $s_i$  is the common grids of Path<sub>1</sub> and Path<sub>2</sub>. After path merging, a min distance path could be found

$S_{com} = \{s_1, s_2, \dots, s_n\}$ . Then these two paths could be divided into  $n - 1$  segments with the break point in these common grids, as shown in Fig. 5. Let  $SubP_k(g_i, g_j)$  denote the segment from grid  $g_i$  to grid  $g_j$  on the path Path<sub>k</sub>. Then, the  $n - 1$  segments of two path Path<sub>1</sub> and Path<sub>2</sub> could be denoted by formula (12).

The sub-segments between different paths with the same start and end point could be replaced with each other. Therefore, a new sub-segment could be generated by comparing the length of two paths, such as formula (13).

### 3.5 The overall EH-ACO algorithm

After introducing four improving strategies specifically, a complete EH-ACO algorithm is formed within the following steps:

- Step 1* Import world environment *WORLD*, start point  $g_{start}$  and goal point  $g_{goal}$ . Initialize pheromones of the environment, pheromone intensity coefficient  $\alpha$ , route cost coefficient  $\beta$ , number of ants  $m$ , maximum iteration times *IterMAX*. Set start time  $t = 0$ ,  $tabu_k = \emptyset$ ,  $path_k = \emptyset$ .
- Step 2* Let ant  $k = 1$  at start point  $g_{start}$ ,  $p_k = g_{start}$ , and add it into TABU table  $tabu_k$  and path  $path_k$ .
- Step 3* Find the neighbor grids  $N_k(p_k)$  of ant  $p_k$  according to formula (1). If no neighbor grids  $N_k(p_k)$  is found, then go to *Step 5*.
- Step 4* Ant  $k$  moves to one of its neighbor grids  $g_{next} \in N_k(p_k)$  according to transition probability formula (3). Then modify the local visibility according to formula (9). Set  $p_k = g_{next}$ , and add it into TABU table  $tabu_k = tabu_k \cup p_k$  and path  $path_k = path_k \cup p_k$ . If  $p_k \neq g_{goal}$ , then return to *Step 3*; else, then go to *Step 6*.
- Step 5* Ant  $k$  moves back to previous grids  $g_{pre}$ . Set  $p_k = g_{pre}$ , return to *Step 3*.
- Step 6* Optimize path  $path_k$  using geometric method and path merging strategy according to formula (13). Update optimal path  $path_{opt}$ . If  $k < m$ , then set  $k = k + 1$  and return to *Step 2*; else, set  $k = 1$ .
- Step 7* Update pheromone intensity by using formulas (4), (10) and (11).
- Step 8* Let  $t = t + 1$ , if  $t < IterMAX$ , return to *Step 2*; else, output solution and stop.

## 4 Experimental results

To verify the proposed EH-ACO algorithm, simulation experiments with four different strategies and comparisons with other ACO algorithms are carried out. All experiments are performed on a computer with an Intel Core i5, 2.3 GHz CPU and 8 GB RAM, and using a single-threaded MATLAB implementation.

### 4.1 Verification of four improved strategies

#### 4.1.1 Visibility enhancement experiment

In the first experiment, we would like to verify the effect of visibility enhancement strategy. The experiments are carried out on “50 × 50” obstacle map, and the main parameters of the experiments such as the pheromone intensity and the ant number are shown in Table 1. The visibility weight  $\beta$  which heavily influences the ant’s visibility is tested with different values. Parameter  $\varepsilon$  in formula (9) is set to 0.5 to adjust the intensity of heuristic information. The experiment would terminate as soon as the ant finds the food.

**Table 1** Experimental parameter setting

World size	50 × 50
Pheromone intensity $\alpha$	1
Testing times	100
Ant number	1
Start point	(1, 50)
Goal point	(50, 1)

**Table 2** Average test results of 100 experiments between the original visibility formula and the improved visibility formula

	Original	Improved	Reduced (%)
<b>(a)</b>			
Visited grids	266	147	44.74
Time	2.918	0.014	99.52
Distance cost	319.292	173.279	45.73
<b>(b)</b>			
Visited grids	226	82	63.72
Time	0.178	0.003	98.31
Distance cost	271.175	96.497	64.42

The related results “visited grids,” “time cost” and “distance cost” are recorded as shown in Table 2 after 100 experiments. It could be found that the “visited grids” is reduced by 44.74% in (a) and 63.72% in (b), “time cost” is reduced by 99.52% in (a) and 98.31% in (b), and the “distance cost” is reduced by 45.73% in (a) and 64.42% in (b) compared with no improvement strategy that is implemented. From the result, we could conclude that the visibility enhancement strategy provides heuristic to the ants and promotes their abilities of finding food. Figure 6 gives two examples of the path planning results from the experiment with  $\beta = 1$  and  $\beta = 5$ , respectively. It could be obviously noticed that ant search

tends to move toward the target consistently after the visibility enhancement strategy is applied under different visibility weights.

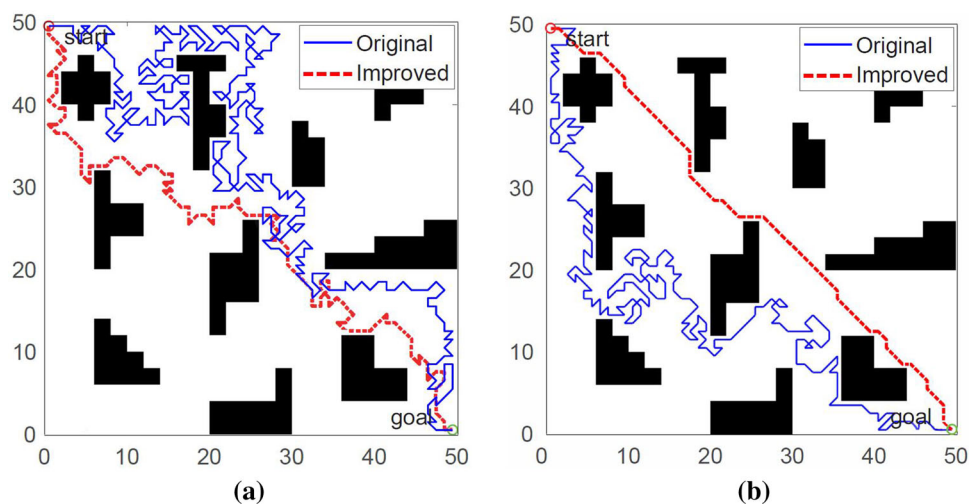
### 4.1.2 Pheromone diffusion strategy

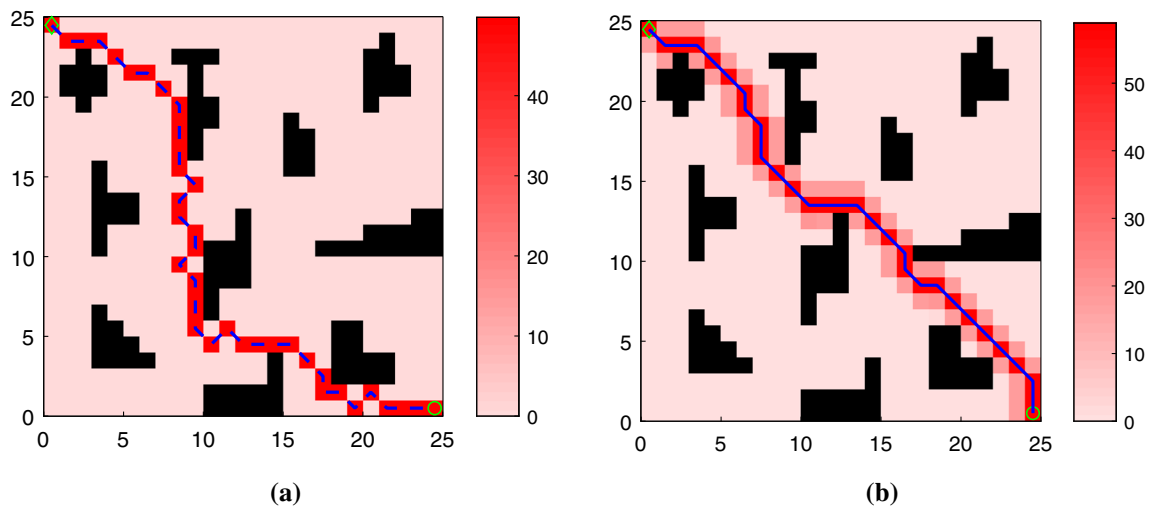
This experiment verifies the effectiveness of the pheromone diffusion strategy. The test is executed on “25 × 25” obstacle map. Parameters are set as follows: pheromone intensity  $\alpha = 10$ , visibility weight  $\beta = 5$ , ant number equals 10, iteration number is 100, start point is (1, 25) and goal point is (25, 1). The algorithm terminates after the predefined iterations. After planning, it could be found that the final path cost is 41.2132 without pheromone diffusion strategy and pheromones are concentrated on the final path, as shown in Fig. 7a. However, after applying the pheromone diffusion strategy, the final path cost is 37.4588, which is reduced by 9.12%. The pheromones show gradient distribution, and the final path follows the highest pheromone concentration path as shown in Fig. 7b. Therefore, we could conclude that the improved pheromone gradient diffusion model is effective in reducing the path cost.

### 4.1.3 Backtracking strategy

To verify the effectiveness of the backtracking strategy, the success rate which is calculated as Eq. (14) is adopted. The distribution of obstacles in this experiment is consistent with Fig. 7, and map has three types of size: “25 × 25,” “50 × 50” and “100 × 100.” Parameters are set as follows: pheromone intensity  $\alpha = 50$ , visibility weight  $\beta = 1$ , the ant number is 50 and the iteration number equals 1. The algorithm terminates after the predefined iterations. In the end, we counted the probability of 50 ants successfully search for the path, as

**Fig. 6** Result of an ant finding food for the first time. The solid line is the path that is not modified by the visibility formula, and the dotted line is the path that is after modified by the visibility formula: **a**  $\beta = 1$  and **b**  $\beta = 5$





**Fig. 7** Comparison of the final path. The intensity of the color in the region denotes the pheromone intensity, and the line is the final path: **a** not improved pheromone strategy, and the path distance cost is equal to 41.2132, and **b** improved pheromone diffusion strategy, and the path distance cost is equal to 37.4558

**Table 3** Success rate of ant finding food

Map size	No backtracking (%)	With backtracking (%)
25 × 25	98	100
50 × 50	62	100
100 × 100	18	100

shown in Table 3.

$$\text{Success rate} = \frac{\text{Number of finding path}}{\text{Ant number}} \times 100\%. \quad (14)$$

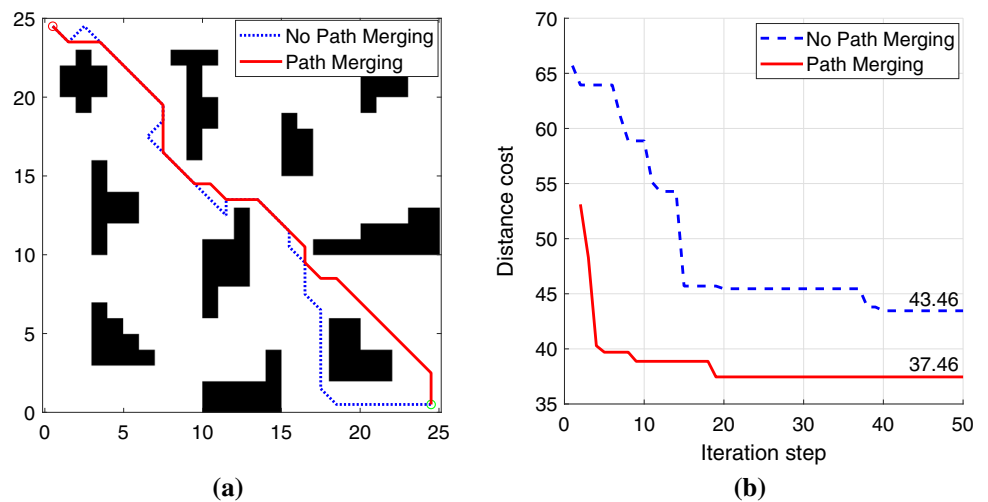
Table 3 shows that ant has a low probability of finding food when backtracking strategy is not considered, especially in the largest size map (100 × 100). However, all the ants could find the food when using the backtracking strategy, which

verifies that the backtracking strategy would help finding the final result.

#### 4.1.4 Path merging strategy

To verify the path merging strategy, experiments are carried out with the following parameters: pheromone intensity  $\alpha = 10$ , visibility weight  $\beta = 2$ , ant number is 10, iteration number equals 50, start point is (1,25), goal point is (25,1) and map size equals “25 × 25.” The algorithm terminates after the predefined iterations. Figure 8 gives results of final path and convergence property. Comparing the final path shown in Fig. 8a, we could found that the path merging strategy is able to merge several optimal sections together rather than just be additive to some good results. Thus, it could achieve a much shorter final result, which is shown intuitively in

**Fig. 8** Comparison of considering path merging strategy and not considering the path merging strategy: **a** the final path and **b** the convergence curve





**Table 4** Parameter setting of each algorithm

	Traditional ACO	ACO-PDG	Two-stage ACO	EH-ACO
$\alpha$	2.1	2.1	4	1
$\beta$	12	12	1.5	1
Ant number	10	10	10	10
Iteration times	50	50	50	50
Testing times	100	100	100	100
$\varepsilon$	–	–	–	0.5

**Fig. 9** Planning results of the proposed EH-ACO algorithm in different environments: **a** “Map (a)” general map, **b** “Map (b)” room map, **c** “Map (c)” complex environment map and **d** “Map (d)” large size “1000 × 1000” map

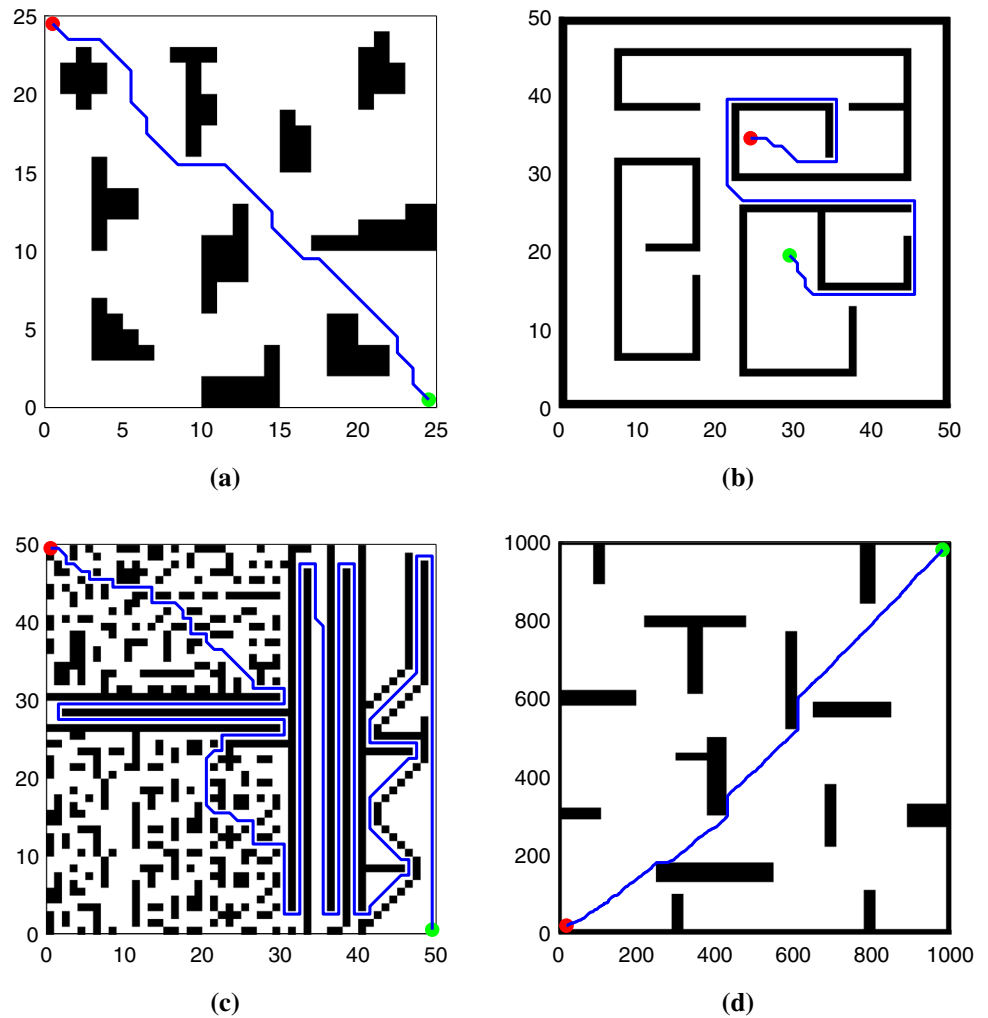
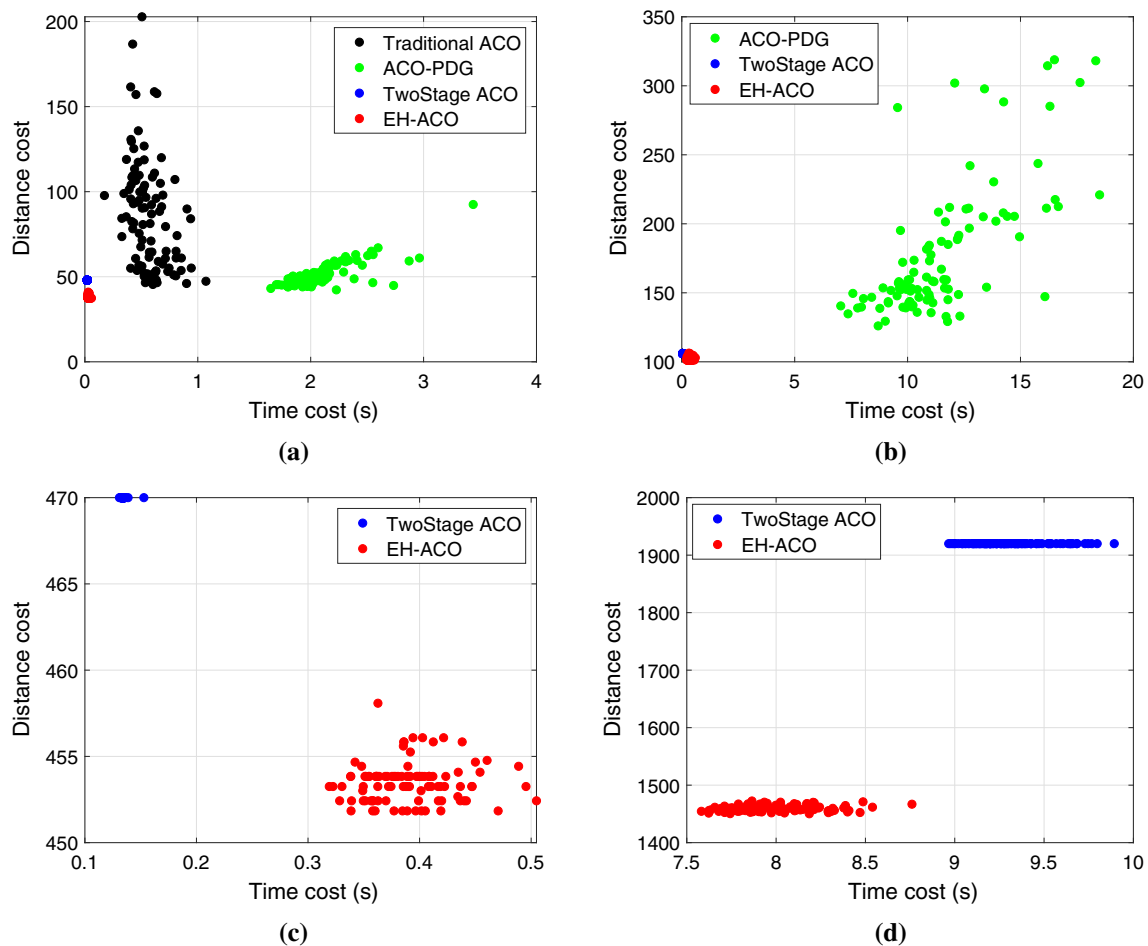


Fig. 8a. Figure 8b shows that the final distance cost eventually converges to 43.46 when path merging strategy is not considered. However, the path distance cost converges to a smaller value 37.46 by applying the path merging strategy. Compared with the former, the final path cost is reduced by 13.8%.

Figure 8b also shows that the original ACO algorithm might converge to a better solution as iteration increases, which is a common characteristic of ACO algorithms. However our proposed path merging strategy takes more advantages since it always selects the optimal segments from

all the paths that ants have searched. Thus, the results would not be better than the proposed strategy no matter how much iteration is used. Our proposed merging process is similar to the searching process of Dijkstra algorithm, which can ensure to find the optimal solution among all ant search route branches including the infeasible path. It is time-efficient as well as powerful in finding the global optimal solution in complex environment.



**Fig. 10** Statistical results of 100 tests on time cost and path cost of each algorithm in different maps: **a** “Map (a),” **b** “Map (b),” **c** “Map (c)” and **d** “Map (d)”

## 4.2 The overall comparison between EH-ACO and other ACO algorithms

In this section, we would like to compare our proposed EH-ACO algorithm with some other ACO algorithms in different maps. Traditional ACO, ACO-PDG (Liu et al. 2017) and two-stage ACO (Chen et al. 2013) are selected. The experimental setting is shown in Table 4. The optimal parameters of each algorithm are given, including pheromone intensity  $\alpha$ , visibility weight  $\beta$ , ant number, iteration times and constant  $\varepsilon$ . Four different maps are selected for experiments, such as “Map (a)” a general map, “Map (b)” a room map, “Map (c)” a complex environment map and “Map (d)” a large size “1000  $\times$  1000” map, as shown in Fig. 9. Black areas are occupied by obstacles. The red point and green points denote the start point and the goal, respectively. All these environments are common in mobile robot path planning.

All the experiments are repeated 100 times, and the final paths generated by our EH-ACO algorithm are drawn in blue lines in Fig. 9. Different algorithms are compared from both

the performance (distance cost of the optimal path) and the efficiency (time cost of finding the optimal path). Figure 10 shows the results comparing different algorithms in different maps. In each subgraph, the abscissa represents the time cost of the algorithm, and the ordinate represents the final distance cost of the algorithm.

In the first subgraph Fig. 10a, all four algorithms are compared. It could be easily noticed that both two-stage ACO and the EH-ACO algorithms have a better performance over the other two algorithms. Similar results could be obtained from the second subgraph Fig. 10b. The reasons are that the traditional ACO has difficulties in handling large complex environments and ACO-PDG only improves the path in local area for a global optimal, both of which are lack of heuristics so that they are not able to find the feasible result in time. Since the two-stage ACO and EH-ACO have relatively equal performance and efficiency in the above maps, more complex and large maps (Fig. 10c, d) are used to compare these two algorithms specifically. From the results, we could notice that in a complex environment map (Fig. 10c), the EH-ACO

algorithm has a better optimization performance, but the efficiency is a little lower compared with two-stage ACO. This is because EH-ACO focuses more on algorithm optimization performance than efficiency. Three out of four strategies proposed are improving the final result. On the other hand, the two-stage ACO preprocessed the map before planning. So it could have a good efficiency when the map is small. However, when the map is large the preprocess will take too much time. The efficiency will decrease as map increases. Figure 10d shows the result that our proposed EH-ACO has advantages in both performance and efficiency in large complex environment.

## 5 Conclusion

This paper introduces an enhanced heuristic ACO algorithm to solve the path planning problems for mobile robot. Four strategies are introduced to improve the performance and efficiency of the ACO algorithm. By enhancing the visibility of the ants, each ant is guided with heuristic, resulting in an algorithm acceleration. Pheromone diffusion strategy smooths the local path to obtain an optimal path. And the backtracking strategy raises the ant's ability of finding a feasible path, which would benefit the algorithms' performance. Path merging strategy which concatenates the optimal path segment would also accelerate the algorithm. Integrated with these four strategies, our proposed EH-ACO algorithm is able to find the optimal path in a large complex environment quickly, which could be implemented on the mobile robot for path planning. Simulation results show that the EH-ACO outperforms the other ACO algorithms in large complex environment in both optimality and efficiency.

For further work, we would like to modify the algorithm to adapt to the dynamic environment where real-time performance is seriously considered. The reuse and update of pheromones in ant colony algorithm need further improvement.

**Acknowledgements** This work was supported by the National Natural Science Foundation (NNSF) of China under 61503265 and the Sichuan Science and Technology Program under Grant 2017SZ0096.

## Compliance with ethical standards

**Conflict of interest** All authors declare that they have no conflict of interest.

**Human and animal rights** This article does not contain any studies with human participants or animals performed by any of the authors.

## References

- Benhala B, Kotti M, Ahaitouf A, Fakhfakh M (2015) Backtracking ACO for RF-circuit design optimization. In: Performance optimization techniques in analog, mixed-signal, and radio-frequency circuit design. IGI Global, pp 158–179
- Cekmez U, Ozsiginan M, Sahingoz OK (2016) Multi colony ant optimization for UAV path planning with obstacle avoidance. In: International conference on unmanned aircraft systems (ICUAS). IEEE, pp 47–52
- Chen G, Liu J (2019) Mobile robot path planning using ant colony algorithm and improved potential field method. *Comput Intell Neurosci*. <https://doi.org/10.1155/2019/1932812>
- Chen X, Kong Y, Fang X, Wu Q (2013) A fast two-stage ACO algorithm for robotic path planning. *Neural Comput Appl* 22(2):313–319
- Deneubourg JL, Clip PL, Camazine SS (1994) Ants, buses and robots-self-organization of transportation systems. In: Proceedings from perception to action conference. IEEE, pp 12–23
- Deng X, Zhang L, Lin H, Luo L (2015) Pheromone mark ant colony optimization with a hybrid node-based pheromone update strategy. *Neurocomputing* 148:46–53
- Dorigo M, Di Caro G (1999) Ant colony optimization: a new meta-heuristic. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), vol 2. IEEE, pp 1470–1477
- Dorigo M, Maniezzo V, Colnari A (1996) Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst Man Cybern Part B (Cybern)* 26(1):29–41
- Friudenberg P, Koziol S (2018) Mobile robot rendezvous using potential fields combined with parallel navigation. *IEEE Access* 6:16948–16957
- Garcia MP, Montiel O, Castillo O, Sepúlveda R, Melin P (2009) Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation. *Appl Soft Comput* 9(3):1102–1110
- Gigras Y, Choudhary K, Gupta K, et al (2015) A hybrid ACO-PSO technique for path planning. In: 2nd International conference on computing for sustainable global development (INDIACom). IEEE, pp 1616–1621
- Habib N, Purwanto D, Soeprijanto A (2016) Mobile robot motion planning by point to point based on modified ant colony optimization and voronoi diagram. In: International seminar on intelligent technology and its applications (ISITIA). IEEE, pp 613–618
- Heegaard PE, Wittner OJ (2006) Restoration performance vs. overhead in a swarm intelligence path management system. In: International workshop on ant colony optimization and swarm intelligence. Springer, pp 282–293
- Kumar PB, Sahu C, Parhi DR (2018) A hybridized regression-adaptive ant colony optimization approach for navigation of humanoids in a cluttered environment. *Appl Soft Comput* 68:565–585
- Lee ZJ, Su SF, Chuang CC, Liu KH (2008) Genetic algorithm with ant colony optimization (GA-ACO) for multiple sequence alignment. *Appl Soft Comput* 8(1):55–78
- Li H, Savkin AV (2018) An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments. *Robot Comput Integr Manuf* 54:65–82
- Li X, Tian P (2006) An ant colony system for the open vehicle routing problem. In: International workshop on ant colony optimization and swarm intelligence. Springer, pp 356–363
- Liu J, Yang J, Liu H, Tian X, Gao M (2017) An improved ant colony algorithm for robot path planning. *Soft Comput* 21(19):5829–5839
- Luo RC, Hsiao TJ (2019) Dynamic wireless indoor localization incorporating with an autonomous mobile robot based on an adaptive signal model fingerprinting approach. *IEEE Trans Ind Electron* 66(3):1940–1951

- Mac TT, Copot C, Tran DT, De Keyser R (2016) Heuristic approaches in robot path planning: a survey. *Robot Auton Syst* 86:13–28
- Mavrovouniotis M, Li C, Yang S (2017) A survey of swarm intelligence for dynamic optimization: algorithms and applications. *Swarm Evol Comput* 33:1–17
- Miah MS, Knoll J, Hevrdejs K (2018) Intelligent range-only mapping and navigation for mobile robots. *IEEE Trans Ind Inform* 14(3):1164–1174
- Mou C, Qing-xian W, Chang-sheng J (2008) A modified ant optimization algorithm for path planning of UCAV. *Appl Soft Comput* 8(4):1712–1718
- Patle B, Parhi D, Jagadeesh A, Kashyap SK (2019) Application of probability to enhance the performance of fuzzy based mobile robot navigation. *Appl Soft Comput* 75:265–283
- Seçkiner SU, Eroğlu Y, Emrullah M, Dereli T (2013) Ant colony optimization for continuous functions by using novel pheromone updating. *Appl Math Comput* 219(9):4163–4175
- Yang J, Zhuang Y (2010) An improved ant colony optimization algorithm for solving a complex combinatorial optimization problem. *Appl Soft Comput* 10(2):653–660
- Yu L, Wei Z, Wang H, Ding Y, Wang Z (2017) Path planning for mobile robot based on fast convergence ant colony algorithm. In: *IEEE international conference on mechatronics and automation (ICMA)*. IEEE, pp 1493–1497
- Yuan J, Wang H, Lin C, Liu D, Yu D (2019) A novel GRU-RNN network model for dynamic path planning of mobile robot. *IEEE Access* 7:15140–15151
- Zhu Q, Wang L (2008) A new algorithm for robot path planning based on scout ant cooperation. In: *Fourth international conference on natural computation, ICNC'08, vol 7*. IEEE, pp 444–449
- Zhu Q, Hu J, Cai W, Henschen L (2011) A new robot navigation algorithm for dynamic unknown environments based on dynamic path re-computation and an improved scout ant algorithm. *Appl Soft Comput* 11(8):4667–4676

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.