Ant Colony Algorithm Based on Upper Bound of Nodes for Robot Path Planning Problems

Sizhi Qiu, Bo Hu, Yongbin Quan, Xiaoxin Jian, and Haibin Ouyang[(✉)]

School of Mechanical and Electric Engineering, Guangzhou University, Guangzhou 510006, China
oyhb1987@163.com

Abstract. In order to solve the problems of ant colony algorithm such as low search efficiency, slow convergence speed and easy to fall into local optimum, an improved ant colony algorithm is proposed in this paper. In the proposed algorithm, the upper bound of path nodes is adjusted and the grid map mode is integrated. The upper bound of nodes are adjusted for driving the optimization process of algorithm. Experiment simulation is carried out based on two specific robot path-planning examples. Results show that the proposed ant colony algorithm can guarantees the global optimization capability, and it also can improve the efficiency of path planning.

Keywords: Ant colony algorithm · Robot path planning · Grid map · Global optimization

1 Introduction

Robots can help humans do dangerous, repetitive and difficult work, and it also can liberate our human from high-risk, or physically demanding labor. One most important parts of robot is path planing, which it has become a hot issue. A good planning route not only can save time, but it also can reduce a great much of consume energy. Therefore, the research of path planning has an important theory and practical significance. Whatever the environment that the robot in, simple or complex, static or dynamic, known or unknown, the primary task for robot is to perceive the environment and avoid the barrier, which promote robot can finishes its work with minimum loss include time and space. This process can be called path planning. Nowadays based on the environment information, path planning can be divided into two categories: global planning (static planning) and local planning (dynamic planning). According to obstacle avoidance strategies of mobile robots, the path planning can be broadly divided into two categories [3]: The first kind of obstacle avoidance strategy is based on the completely known environment, for example, the obstructions' position and shape have been known or given. This kind of obstacle avoidance belongs to static planning issues. The main methods include the grid method, visibility graph method, and so on. The second category is the obstacle avoidance strategy based on sensor information, such as the algorithm with fuzzy logic, neural network and artificial potential field method.

Traditional path planning methods, such as artificial potential field method [4], simulated annealing algorithm [5], and fuzzy logic algorithm [6], which generally have such defects as slow convergence speed and poor global search ability. With the development of intelligent control technology, some new methods are presented and developed. For example, neural network algorithm, genetic algorithm, ant colony algorithm and other intelligent algorithms are gradually applied to solve the problem of path planning. Among them, neural network algorithm has the ability of self-adaptation and self-learning, but poor generalization ability is its fatal shortcoming. Genetic algorithm is suitable for global path planning, but the search space is large, the operation efficiency is not high, the running speed is slow. Ant colony optimization (ACO) is widely concerned by researchers because of its heuristic, parallelism and robustness. Ant colony optimization (ACO) is proposed by Marco Dorigo in 1992. It showed a probability model which is used to find optimal path algorithm [7], and its inspiration from the path in the process of ants searching for food, this algorithm has distributed computing, information positive feedback and heuristic search, the characteristics of the nature is a heuristic global optimization algorithm.

The establishment of environment model is the basic premise and key of mobile robot autonomous navigation and path planning. The establishment of environment model is not only a reflection of the robot's perception ability and intelligence level, but also enhances the robot's ability to complete intelligent tasks, and improves the flexibility, stability, robustness and efficiency of robot navigation. The most commonly used environmental modeling methods are grid, geometry and topology [8–10].

Based on the grid map modeling, an improved ant colony algorithm is proposed in this article through in the process of iteration the limitation on the path to the node number, effectively blocked obviously unreasonable ants search path, through the modeling and simulation, compared with other algorithms under the same environment, we obtained a better effect, and to a certain extent, it improved the ant search accuracy, the convergence speed of the algorithm and the efficiency of path planning.

2   Environment Overview and Modeling

The actual working environment of robot is a real physical space, while the space processed by path planning algorithm is an abstract space of the environment. Environment modeling is a mapping of physical space to abstract space. The purpose of environment modeling is to build an effective environment model, which it is convenient for using computer to solve path planning problem.

In this paper, the grid method can be equivalent to two-dimensional finite area AS through the fractal algorithm of text [1]. AS can be changed into any shape. Therefore, when building the robot environment, it is necessary to supplement the obstacle grid at the AS boundary to take the robot working environment as a square, each square is called a grid, and the size of the grid area is generally determined by the size of the robot car body. If the robot traverses the whole site, we can record the movement of the robot at the center of each grid, and we record the obstacle information during the robot movement. In this way, the movement and obstacle information of the robot are

recorded on a grid to define the grid properties. All of the grid properties constitute a map of the site, in which the grid occupied by obstacles is called the obstacle grid; and a grid without obstacles is called a free grid [11].

The rectangular coordinate system method was used to construct the grid plane, and the grid was divided into two color grids, black and white, as shown in Fig. 1. Black represents the obstacle grid, white represents the free grid, and the robot can move freely in the free grid in the plane. The obstacles are fixed, and the grid number is labeled 1, 2, 3, …, n. Each number represents a grid, and establish the relationship between the grid number and the coordinates in the plane of XOY.

$$\begin{cases} x = \text{floor}((C-1)/N) \\ y = N - \text{mod}((C, 20) + 0.5) \end{cases} \tag{1}$$

In the formula (1): N is the number of grids for each row and column in the grid environment; C is the grid number, floor is the downward integer, mod is the remainder.

Fig. 1. Simple grid diagram

3   Improved Ant Colony Optimization Algorithm

3.1   Basic Principle of Ant Colony Algorithm

Biologists have found that ant's foraging is a group activity, and it is not a solitary search for food source. When an ant searches for a food source, it releases a pheromone in its path, and it senses pheromones released by other ants. The pheromone concentration indicates the distance of the path. The higher the pheromone concentration, the shorter the corresponding path distance.

Generally, ants prefer the path with higher pheromone concentration with higher probability, and release a certain amount of pheromone to enhance the pheromone concentration in that path, thus these actions forming a positive feedback.

The basic optimization process of ant colony algorithm can be described as: the feasible solution of the problem to be optimized is represented by the ant's walking path, and all the paths of the whole ant colony constitute the solution space of the problem

to be optimized. Ants with shorter paths release more pheromones. With the increase of time, the pheromone concentration accumulated on the shorter paths gradually increases, and the number of ants choosing this path is increasing. In the end, the whole colony of ants will focus on the optimal path under the effect of positive feedback, and the corresponding path is the optimal path.

3.2 The Basic Model of Ant Colony Algorithm

Through pheromones, individuals of ant colony communicate with each other indirectly. A group composed of simple ants can accomplish some complex tasks that are difficult for individual ants to complete.

The mathematical model of basic ant colony algorithm is given by analyzing the classic TSP problem of n cities. The TSP problem is a combination optimization problem, in which the artificial ant has to choose the path to visit n cities, and the limit of the path is that each city can only visit once, and finally return to the original city. It is required to travel the nearest distance, visit each city only once, start from a certain city, and finally return to that city. In the optimization process, artificial ants calculate the state transition probability according to the information amount on different paths and heuristic information, as shown in Eq. (2):

$$p_{ij}^k (t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in allow_k} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & s \in allow_k \\ 0, & s \notin allow_k \end{cases} \quad (2)$$

p_{ij}^k (t) represents the transfer probability of ant k from element (city) I to element (city) j at time t; α and β represent the relative importance of pheromone and heuristic information in ant colony search path; $\tau_{ij}(t)$ is the information quantity on the path between city I and city j at time t; Tabu$_k$ (k = 1, 2, …, m) is used here to record the set of elements (cities) that the KTH ant has walked through. The explanatory formula (2) will not select the elements (cities) in Tabu$_k$ again, ensuring the legitimacy of solving the optimal path selection. In order to make the information contained in the pheromones after the ant has traversed all the elements (cities) effective, it is necessary to update the remaining pheromones. The change process of pheromone is shown in formula (3):

$$\begin{cases} \tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}, \\ \Delta\tau_{ij} = \sum_{k=1}^n \Delta\tau_{ij}^k, \end{cases} \quad 0 < \rho < 1 \quad (3)$$

$\Delta\tau_{ij}^k$ represents the pheromone concentration released by the kth ant on the connection path between the *i*th and the *j* grids; $\Delta\tau_{ij}$ represents the sum of pheromone concentration released by all ants in the connection path of grid I and grid *j*.

To solve the problem of pheromone release by ants, in order to make better use of the overall information, this paper adopts the ant cycle model proposed by Dorigo, $\Delta\tau_{ij}^k$ is calculated as a formula (4)

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, \text{ the } k^{th} \text{ ant } from \text{ grid } i \text{ to } j \\ 0, \text{ others} \end{cases} \quad (4)$$

Where Q is a constant, represents the amount of pheromone released by the ant in a cycle; L_k is the length of the path traveled by the kth ant.

3.3 Improved Ant Colony Optimization Algorithm

3.3.1 Improved Strategy

The concept of node upper limit is introduced and the coordinates traversed by ant are regarded as nodes, the sum of the distances between nodes is called the node upper limit. In the whole ant colony, when the number of nodes traversed by an individual is far more than that of the currently known optimal individual, the individual cannot plan towards the optimal solution, it results in the problem of long algorithm time and low efficiency. By setting the upper limit of nodes, this paper avoids the influence of too many individual nodes and effectively improves the execution efficiency of the algorithm.

3.3.2 Improved Method

Y is set as the upper limit of dynamic nodes. In the whole ant colony, when the number of path nodes of an ant in the traversal process is obviously larger than the number of the optimal path nodes at present, the ant is eliminated in advance and the next ant traverses. For example, the path of the first ant is

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9$$

The path traversed by the ant is a node. If 1 is the starting point of the ant and 9 is the target point of the ant, the distance of the first ant is 8. The path of the other ant is

$$1 \rightarrow 3 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 6 \rightarrow 8 \rightarrow 7$$

The second ant has not traversed all the nodes, and the ant's path distance is 14. The difference between the two ants is significant, and the upper limit of nodes is 1.5, so the ants that are within 12, and it can reach the target point are kept. Like the second ant, the distance is greater than 12, and it didn't go through all the nodes, so we eliminated it ahead of time and went through the next ant. In order to plan the path towards the optimal solution, the upper limit of dynamic nodes decreases with the increase of i of iteration number $iter_i$, so that the number of nodes of the planned path is closer and closer to the optimal solution. Take the upper bound multiple of the initial node $Yo = 2$ as an example, set the dynamic upper limit of the ith iteration node as Y_i, the maximum number of iterations is $iter_{max}$, the i times iteration is $iter_i$, Then the upper limit Y formula of dynamic node is calculated

$$Y_i = \frac{1}{3600}iter_i^2 - \frac{1}{30}iter_i + 2, \ iter \in [0, iter_{max}] \tag{5}$$

When the traditional ant colony algorithm is used to solve the robot path-planning problem, it is easy to fall into the local extreme value, and it is difficult to avoid the obstacles and find a reasonable path. After many experiments, it is found that in the middle and later stages of planning, because the path gradually converges, and in order to eliminate the errors caused by individual ants deviating from the optimal path greatly in the course of path transformation, we adjusted the upper limit of nodes dynamically. Through the dynamic adjustment in the evolutionary process of each generation, it not only guarantees the overall situation in the early stage, but also guarantees the operation efficiency in the later stage. The simulation experiment shows that this adjustment mode has obvious effect on the improvement of convergence speed.

3.3.3 Basic Steps of Solving Robot Path Planning

By combining the node upper limit ant colony algorithm with robot path planning, a reasonable and reliable path is finally found. The steps are as follows:

Step 1. Adopt the grip method to build the map, initialize the algorithm, empty the grid of pheromones, reset the number of iterations. Initialize the ant colony, put all the ants at the starting point S.

Step 2. Calculate the transition probability of the next grid by formula (2), the ant migrates between the grid points according to the probability and records the migration path. When the migration path is less than the upper limit of nodes and the ant does not reach the target point, repeat the migration path logging process; When the ant migration path is equal to or greater than the upper limit of nodes and does not reach the target point, reinitialize the ants, clear migration path and node access records, go back to our starting point S and the number of ants' re-initialization plus 1. If the re-initialization count value is less than the upper limit, the migration path recording process is performed; otherwise, the ant is discarded.

Step 3. For ants reaching the target point, calculate its path distance, and the upper limit of nodes is dynamically adjusted according to formula (5).

Step 4. Execute on all ants in the colony from **Step 2** to **Step 3**. Choose the smallest path of each ant.

Step 5. In the optimal path, the pheromone is updated by formula (3) and (4) according to the movement direction of the ant when it reaches each grid point. Number of iterations plus 1. When the number of algorithm iterations is less than the maximum iteration limit, go to **Step 2**; otherwise, the algorithm terminates.

3.3.4 The Search Block Diagram

Dispatch 50 ants every time, the maximum number of iterations $iter_{max} = 60$, Pheromone importance factor $\alpha = 1$, Pheromone volatility $\beta = 0.5$.
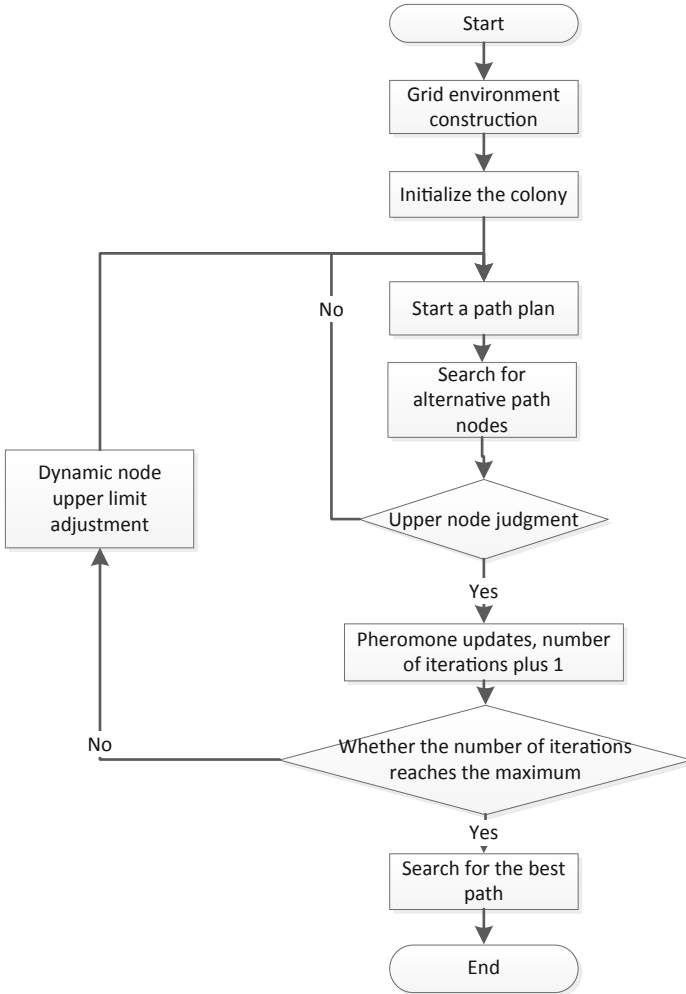
Fig. 2. MACO Algorithmic search framework

4   Discussion and Analysis of Experimental Simulation Results

Considering the comprehensiveness and fairness of the algorithm performance test, the general path problem in the literature is selected and it compared with the basic ACO algorithm and the improved ant colony algorithm. All algorithms are implemented by MATLAB 2016Ra programming, Intel (R) CORE(TM)2 Quad CPU Q9400 @ 2.66 GHz, 2.66 GHz, with the 3.50 GB of physical address expansion of memory, Microsoft Windows XP. It ran 30 times on the computer. In order to ensure the fairness of the comparison experiment, all the parameters in the comparison algorithm are set in strict accordance with the original literature. Parameter setting of the algorithm in this paper: the number of ants $N = 50$, iterations: 100, the relative importance of pheromones is 0.5; the pheromone volatile factor is 0.1.

4.1 Simulation and Results of Environment 1

According to the environment map given in literature [12], using the algorithm in this paper, A* algorithm, ACO and MACO apply to this path diagram, the simulation results are shown in Table 1. It can be seen from Table 1 that the algorithm in this paper is superior to A* in optimization efficiency, ACO and MACO, and the optimization time is the shortest. At the same time, the results of the optimization environment path of this algorithm in Fig. 1 are shown in Fig. 3.
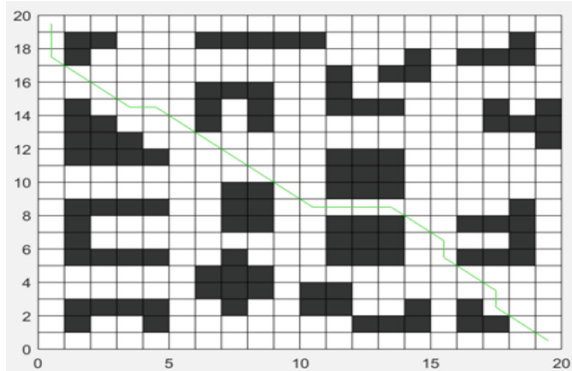

Fig. 3. Optimization of environment path example 1

Table 1. Comparison results of environment path example 1

Algorithm	Average shortest path length	Average running time
A*	31.30	40.15
ACO	30.10	39.06
Reference [12]	29.21	38.20
MACO	29.44	30.84

4.2 Simulation and Results of Environment 2

According to the environment map given in reference [11], the algorithm in this paper, ACO and MACO are applied to the path planning diagram the simulation results are shown in Table 2. Meanwhile, the results of the optimized environment path of the algorithm in Fig. 2 are shown in Fig. 4.
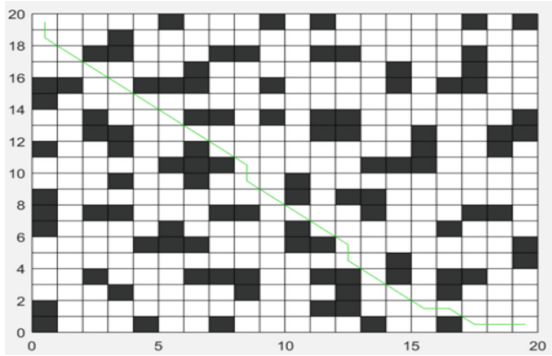
Fig. 4. Optimization of environment path example 2

It can be seen from Table 2 that the optimization efficiency of the algorithm in this paper is better than ACO and MACO, and the optimization time is the shortest, indicating that the algorithm in this paper has a good optimization performance.

Table 2. Comparison results of environment path example 2

Algorithm	Average shortest path length	Average running time
ACO	36.1	46.3
Reference [11]	35.8	32.4
MACO	29.7	27.1

5   Conclusion

In this paper, an improved ant colony algorithm (MACO) is proposed based on grid map modeling. By setting the upper limit of nodes, the number of path nodes is compared with the number of existing path nodes as the number of iterations increases, if the limit is exceeded, the ants are weeded out. The results show that the proposed ant colony algorithm can not only ensure the global optimization ability of the ant colony algorithm, but also improve the efficiency of path planning.

References

1. Ge, S., Cui, J.: New potential functions for mobile robot path planning. IEEE Trans. Robot. Autom. **16**(5), 615–620 (2000)
2. Mo, H., Xu, L.: Research of biogeography particle swarm optimization for robot path planning. Neurocomputing **148**, 91–99 (2015)
3. Mac, T., Copot, C., Tran, T., et al.: Heuristic approaches in robot path planning: a survey. Robot. Auton. Syst. **86**, 13–28 (2016)

4. Zhang, J., Zhao, Z., Liu, D.: A path planning method for mobile robot based on artificial potential field. J. Harbin Inst. Technol. **38**(8), 1306–1309 (2006)
5. van Laarhoven, P.J.M., Aarts, E.H.L.: Simulated annealing. In: van Laarhoven, P.J.M., Aarts, E.H.L. (eds.) Simulated Annealing: Theory and Applications. MAIA, vol. 37, pp. 7–15. Springer, Dordrecht (1987). https://doi.org/10.1007/978-94-015-7744-1_2
6. Qing, L., Chao, Z., Caiwei, H., et al.: Path planning based on fuzzy logic algorithm for mobile robots in dynamic environments. J. Central South Univ. (Sci. Technol.) **44**(s2), 104–108 (2013)
7. Dorigo, M., Gambardella, M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evol. Comput. **1**(1), 53–66 (1997)
8. Delmerico, J., Mueggler, E., Nitsch, J., et al.: Active autonomous aerial exploration for ground robot path planning. IEEE Robot. Autom. Lett. **2**(2), 664–671 (2017)
9. Fu, B., Chen, L., Zhou, Y., et al.: An improved A* algorithm for the industrial robot path planning with high success rate and short length. Robot. Auton. Syst. **106**, 26–37 (2018)
10. Yu, J., Rus, D.: An effective algorithmic framework for near optimal multi-robot path planning. In: Bicchi, A., Burgard, W. (eds.) Robotics Research. SPAR, vol. 2, pp. 495–511. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-51532-8_30
11. Deng, G., Zhang, X., Liu, Y.: Ant colony optimization and particle swarm optimization for robot-path planning in obstacle environment. Control Theory Appl. **26**(08), 879–883 (2009)
12. Xinlei, H., Lianzhi, Y.: Mobile robot path planning based on improved ant colony algorithm. Softw. Guide **12**, 162–164 (2017)