Improved Ant Colony Algorithm for AGV Path Planning

LI Jia-ning
School of Mechatronic Engineering and Automation, Shanghai University, China

Abstract: Given the shortcomings of the ant colony algorithm in the path planning process, such as low convergence speed and easiness of falling into local optimization, an improved ant colony algorithm (ACO) suitable for AGV path planning was proposed. The initial pheromone concentration was differentiated on the grid map according to the distance, which avoided the blind search in the early stage of the ant colony and sped up the convergence speed of the algorithm. The distance between the current grid and the grid to be selected and the distance between the grid to be selected, and the target grid were synthesized to improve the heuristic function to increase the direction of ant colony pathfinding. The dynamic heuristic factor was introduced to avoid the phenomenon of prematurity and falling into local optimization. It was proposed to label the direction of the adjacent grid of each grid, which increased the distance between the optimal path and obstacles, enhanced the security of the optimal path, avoided the occurrence of the dead corner phenomenon, and improved the robustness of the algorithm. The simulation results show that in the same environment, the improved algorithm's search efficiency and iterative stability are better than that of basic ACO algorithms in AGV path planning.

Keywords: AGV, Path Planning, Initial Pheromone, Dynamic Heuristic Factor, Direction Label

Introduction

With the increasing maturity of automation technology, an increasing number of automated guided vehicles (AGVs) have become the mainstream work method in the automatic handling process of factories. Its main function is to safely transport goods to the designated destination through the planned path in a complex working environment (De Ryck et al., 2020). AGVs can complete transportation tasks in harsh and complex environments compared to manual handling. The emergence of the AGV automatic navigation system has changed the logistics structure of traditional manufacturing workshops, greatly reduced the production cost and improved the production efficiency of the enterprise.

ICorrespondents Author:
LI Jia-ning, Shanghai University, China
Email : lijiangning01@gmail.com

With the continuous development of the AGV market and path-planning technology, many optimization algorithms are increasingly used to solve path-planning problems with different application backgrounds and task requirements. However, most of the previous works (Akka & Khaber, 2018) considered the performance improvement of a single algorithm in small-scale, low-complexity maps and often ignored the limitations of a single algorithm in large-scale, complex environments, and there is a lack of practical application verification. Several algorithms have long been applied to plan the optimal path for the AGV, including Dijkstra's, A∗ , and genetic algorithms (GA). Based on Dijkstra's algorithm, Kim and Tanchoco (Kim & Tanchoco, 1991) planned the AGV path using the TW graph's free time window (TW). Yu and Egbelu (Yu & Egbelu, 2001) classified the idle AGV and minimized the idle time of the AGV through genetic iteration. Mimicking the process of natural evolution, Umar et al. (Umar et al., 2012) applied GA to search for the optimal path for the AGV.

Although these algorithms can search the path distributed and have good global search ability, there are still some defects in some aspects, such as long operation time, easy to appear "premature" phenomenon, which leads to the search path effect is not good enough and the work efficiency is low. An optimal ant colony algorithm is proposed to address the above problems in this paper. By improving heuristic functions, state transition probabilities and pheromone update strategy, the algorithm can not only enhance the efficiency of intelligent guidance vehicle path planning but also help to improve the theoretical and technical system of the ACO and expand the application field of the ACO (Zhang, Luo, Yin, & Zou, 2023).

Research Method

Environment Modelling

Establishing the AGV's movement environment is an important step in the study of path planning for AGV. Many methods for modelling 2D planar maps include a topological map, visibility graph approach and free space methods (Li & Chen, 2022). This paper chooses the grid map as the 2D planar map model, considering that AGVs work mostly indoors and in an environment where obstacles are known. The grid map model is easy to create, simple to maintain and highly adaptable, and simple modifications can change the model environment. Its working principle is to assume that the path environment space of AGV is two-dimensional and divide the environment into square with the same size according to the specific requirements, which is called the grid map model. Grids can be divided into two categories: free grid and obstacle grid. The former is white, and the latter is black, which are represented by "0" and "1" respectively (Sun et al., 2021). The whole grid constitutes a complete grid environment model. A 5×5 grid environment is shown in Figure 1, with the lower left corner of the environment as the coordinates, upwards as the positive direction of the Y-axis and to the right as the positive direction of the X-axis, a plane right angle coordinate system established and a grid cell length of 1cm is selected (Zhao & Cheah, 2023).

The grid environment is labelled from left to right and from top to bottom. In an m x m grid environment, the serial number S corresponds to the coordinates (x, y) of the raster in which it is located, and the correspondence is:

$$\begin{cases} x = mod(S-1, m) + 0.5 \\ y = m + 0.5 - \text{ceil}\ (S/m) \end{cases} \tag{1}$$

Where mod is a remainder operation, ceil is an upward rounding operation. Thus, the travel path of an AGV can be represented as a series of sequences of numbers.

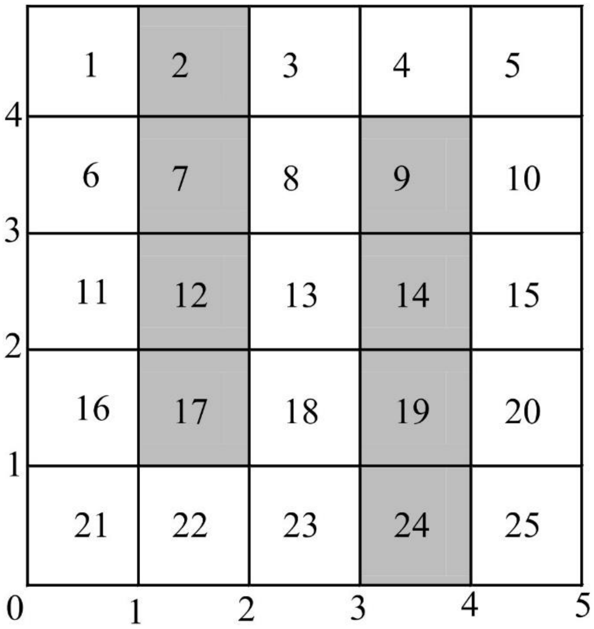

Figure 1 AGV Grid Environment

The directions of motion of the AGV are shown in Figure 2. Except for the edge grids, each grid generally has eight directions of motion: top left, top, top right, left, right, bottom left, bottom right and bottom right, which are numbered 1, 2, 3, 4, 5, 6, 7 and 8 numbers respectively (Zheng, Sayed, & Essa, 2019).

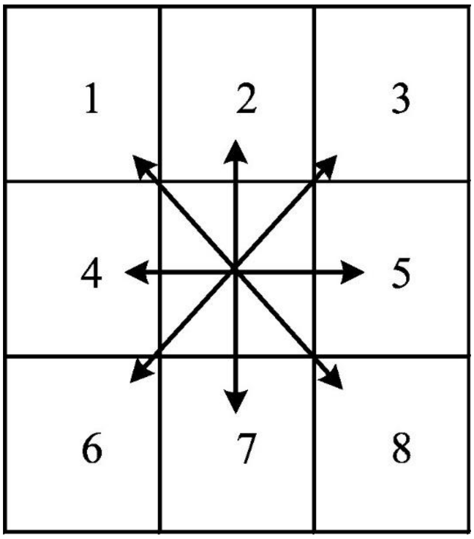

Figure 2 AGV Direction of Motion

Grid Environmental Processing

During indoor movement of the AGV, it is one of the considerations in AVG path planning to ensure that the optimal path avoids collisions with static obstacles and improves the safety of the path. If the optional grid contains an odd number of grids, it is determined whether two neighboring grids perpendicular to the direction of the path are obstacles. If one of the two neighboring grids is an obstacle, the grid is considered a "forbidden grid". This method reduces the likelihood of collision between the AGV and the obstacle. The correct path of travel is away from the edge of the obstacle as shown in Figure 3, reducing the possibility of collision between the AGV and static obstacles and increasing the safety of AGV indoor operations (Yu, Yuan, Li, Yuan, & Deng, 2023).

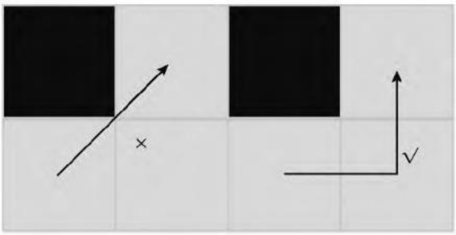

Figure 3 Collision Avoidance Effect

The "dead corner" phenomenon frequently occurs during the iterative process of the algorithm, affecting the efficiency of the ant colony search and the final result of the algorithm. This affects the search's efficiency and the algorithm's final result. As can be seen from Figure 4, both the basic ACO and the improved algorithm only consider the distance between the grid to be selected, and the target grid as the criterion for route selection, resulting in the grid labelled 3 still being included in the queue of selectable grids when the AGV travels to grid i. When the AGV travels to grid 3, it finds that it can only travel toward grid 2 or grid 4.

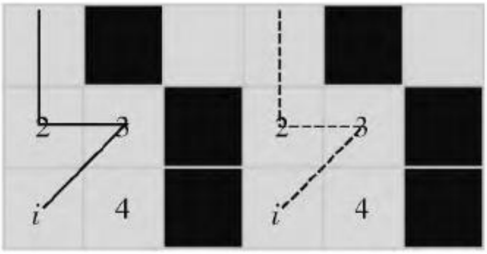

Figure 4 Comparison of Algorithm Results

To solve the problem of "dead corner" during the iteration of the algorithm, this paper extends a layer of the raster on top of the neighbouring raster for calculation. As shown in Figure 5, the neighbouring grid of grid i, labelled 3, is a feasible grid, and the two neighbouring grids perpendicular to it (grids 2,4) are both feasible grids, so grid 3 satisfies the path safety criterion. However, the neighbouring grids of grid 3, marked 2 and 4, are obstacles, so grid i moves to grid 3 and eventually moves to the left or down. Therefore, for grid i, grid 3 is called a "dead corner". In contrast to a colony that is trapped in a "dead-end" situation, this algorithm reduces the computational effort, the number of turns and the length of the optimal path by marking grid 3 as a "forbidden grid" in advance and moving directly from grid i to grid 2 or grid 4. This reduces the computational effort of the algorithm as well as the number of turns and the length of the optimal path, speeds up the convergence of the algorithm based on the search for the global optimal path and saves energy in the operation of the AGV.

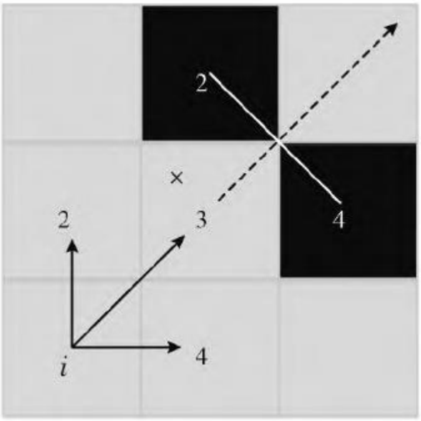

Figure 5 Anti-dead-end Principle

Ant Colony Algorithm Optimization

Traditional Ant Colony Algorithm

In the traditional ant colony algorithm mathematical model, the state transition probability and pheromone update strategy have a crucial role in the solution efficiency of the algorithm (Mullen et al., 2009)

State transition probability

At time t, ant k is moved from node i to node j by calculating the state transition probability $P_{ij}^{k}(t)$; then, the next node is selected according to the roulette method, which is defined as:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}{\sum_{j\in \text{allowed}_k}\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}, j \in \text{ allowed}_k \\ 0, \qquad \text{otherswise} \end{cases} \tag{2}$$

Where $\eta_{ij} = \frac{1}{d_{ij}}, d_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2}, \eta_{ij}(t)$ is the heuristic function and represents the Euler distance from the current node to the candidate node, $\tau_{ij}(t)$ is the pheromone concentration between two nodes, allowed $_k$ is the set of next mobile nodes to be selected by the ants, and α and β are the pheromone importance factor and heuristic function importance factor, respectively.

Pheromone update strategy

The pheromone update strategy is a process in which the ant colony algorithm continuously realizes positive feedback. Through this method, the ant colony guides the descendants of the ants to continuously converge to obtain an optimal path (Xiao et al., 2022). After all contemporary ants have reached the target, each ant will update the pheromone concentration according to Equation 3, Equation 4, and Equation 5.

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{3}$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \tag{4}$$

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \dfrac{Q}{L_k}, & \text{if ant } k \text{ passes through path } (i,j) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Where $\rho \in (0,1)$ is the pheromone volatilization coefficient, $\Delta\tau_{ij}(t)$ is the sum of pheromone increments for all ants, $\Delta\tau_{ij}^{k}(t)$ is the pheromone increment of the k^{th} ant on the path (i, j), Q is the pheromone intensity, and L_k is the total path length of the k^{th} ant after one iteration.

Improvement Ant Colony Algorithm

In the early iterations of the ant colony algorithm, the lack of differentiated initial pheromone concentration in the map environment and the lack of guiding factors in the pathfinding process leads to blind searching by the ant colony in the early stages of the algorithm and slow convergence of the algorithm. Therefore, this paper designs a differential initial pheromone concentration based on the Euclidean distance, the known map environment model, and the initial pheromone's differential distribution around the starting point and the "sub-optimal path" produced by the target point (Sun et al., 2021).

$$\text{Tau}\ (i,j) = \begin{cases} \dfrac{\sigma^{*} d_{se}}{d_{si} + d_{ij} + d_{je}} & \text{no obstacle between node } (\text{s}, \text{e}) \text{ lines} \\ \dfrac{\sigma^{*}(d_{sn} + d_{mn} + d_{ne})}{d_{si} + d_{ij} + min\left(d_{jm}\right) + d_{me}}, & \text{otherwise} \end{cases} \tag{6}$$

Where Tau (i, j) is the pheromone concentration between the feasible grid i and the feasible grid j, d_{si} is the Euclidean distance between the departure grid S and the current grid i, d_{ij} is the Euclidean distance between the departure grid i and the current grid j, d_{je} is the Euclidean distance between the departure grid j and the current grid e, d_{se} is the Euclidean distance between the starting grid and the target grid, $\min\left(d_{jm}\right)$ is the minimum Euclidean distance of grid j from the "suboptimal path", $d_{sm}, d_{mn}, d_{ne}, d_{me}$ are the Euclidean distances between two grids on the "suboptimal path", σ is the discretization factor. A larger σ represents a larger discrepancy in the initial global pheromone concentration (Cenerini, Mehrez, Han, Jeon, & Melek, 2023).

The straight line between the two points is the shortest distance, but if only the line between the start and end grid is treated as a "sub-optimal path" and the static obstacle between the start and end grid is not considered, a direct pheromone concentration initialization of the line between the two grids will instead mislead the ant colony to search for the pheromone. As shown in Figure 6(a), the dashed line is the global optimal path and the solid line is the path finding result according to the initial pheromone. As the initialization of the pheromone does not consider the influence of static obstacles between the connecting lines, the ant colony needs to continue searching randomly along the obstacles to find the shortest path when it reaches the obstacles, which is inefficient and difficult to converge to the global optimal. The initial pheromone concentration between the grids is then determined based on the Euclidean distance, with the shorter the distance from the "next best path", the higher the initial pheromone concentration and the lower the opposite. The specific method for determining the 'second best path' is as follows: first determine the line between the starting grid and the ending grid, and when there is an obstacle in the line of grid (s, e), form a breakpoint at the obstacle. If the obstacle is located to the left or right of the breakpoint, a search will be carried out up and down the obstacle, recording the number of obstacles until a workable grid is found. Compare the number of obstacles in both directions and select the feasible grid in the direction with fewer obstacles as the next starting point. Compare the number of obstacles in both directions and select the feasible grid in the direction with fewer obstacles as the next starting point until you reach the end grid and finally find the 'next best route'. As shown in the solid line in Figure 6(a), there is an obstacle in the connection of grid (s, e), and the obstacle is located to the right of the breakpoint, so search up and down along the obstacle, and find that the number of obstacle grids downward is less, so the feasible grid below the obstacle is used as a new starting point to continue the search, which eventually forms the "second-best path" shown in the dashed line in the figure. The dashed line in Figure 6(b) shows that grid j is the grid to be selected from grid i. Grid j is the grid to be selected from grid i. The closest grid m to the "next best route" is selected as the origin. The initial pheromone concentration between grids (i, j) can be calculated using Equation 6.

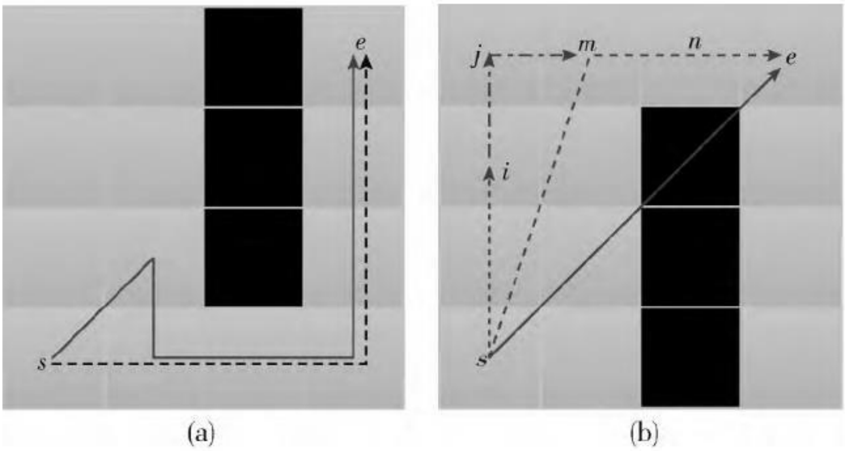

Figure 6 Principle of Pheromone Initialization

As shown in Figure 6, when grid i and grid j are closer to the "second-best path", the higher the concentration of initialized pheromone between grid i and grid j, the more likely the ant colony will search in the direction of that path in the pathfinding process, improving the efficiency of the ant colony pathfinding, speeding up the convergence of the algorithm and reducing the number of iterations of the algorithm.

Improvement Heuristic Function

In the basic ant colony algorithm, the heuristic function is determined only by the distance d_{ij} between the current grid i and the grid j to be selected. However, the difference in distance between the grids is small and it is difficult for the colony to select the best grids from the many available grids simply by the distance d between the grids. The best grid is selected by the ant colony only by the inter-grid distance d_{ij}. For example, when the unit grid width is 1 , the minimum distance between adjacent grids is 1 and the maximum distance is about 1.4; after the distance heuristic function of the basic ACO algorithm, the difference between the nearest adjacent grid and the farthest adjacent grid is only about 0.3 . The distance heuristic function has a weak pathfinding effect on the ant colony, which cannot combine the pheromone concentration between grids and the distance between grids. This reduces the search efficiency of the ant colony algorithm as well as the final pathfinding effect. Therefore, this paper improves a heuristic function whose value is determined by the distance d_{ij} between the current grid i and the grid j to be selected and the distance d_{je} between the grid j to be selected and the target grid e, as shown in Equation 7.

$$\eta_{ij}(t) = \frac{\varphi_{ij}}{\omega^{*} d_{ij} + \lambda * d_{je}} \tag{7}$$

To avoid the problem of weak guidance of the heuristic function value on the ant colony pathfinding process due to the small distance difference between the grids, the method of amplifying the distance difference is used to enhance the "superiority and inferiority" between the grids to be selected.

$$\varphi_{ij} = \frac{\left(D_{max} - d_{ij}\right)}{(D_{max} - D_{\min} + 0.01)} * q \tag{8}$$

$$q = \frac{d_{se}}{d_{je}} \tag{9}$$

Where φ_{ij} is the amplification function of the distance between grid i and grid j, ω and λ are distance amplification factors, which can be set according to the specific environment, $D_{\max}, D_{\min}$ are the maximum and minimum values of the distance between the current grid and the optional grid, respectively, 0.01, to ensure that the denominator is not zero. The coefficient q is the amplification factor, and its value is related to the distance from the grid to be selected to the target grid.

https://doi.org/10.58291/ijec.v2i1.82

The final calculation of the heuristic function is shown in Equation 10:

$$\eta_{ij}(t) = \frac{d_{se} * \left(D_{max} - d_{ij}\right)}{d_{je} * (D_{max} - D_{min} + 0.01) * \left(\omega^{*} d_{ij} + \lambda * d_{je}\right)} \tag{10}$$

To facilitate the heuristic function calculation, the neighbouring grid of each grid is created distance matrix $D_{n^2 \times 8}$, as shown in Equation 11:

$$D(i,j) = \begin{cases} l, & G(i) = 0 \quad mod(j,2) = 0G(j) = 0 \\ \sqrt{2} * l, & G(i) = 0 \qquad mod(j,2) = 1 \\ & G(j) = 0 \qquad G(i') + G(i'') = 0 \\ & \qquad\qquad\quad G(i_1) + G(i_2) \neq 2 \\ \infty, & \text{otherwise} \end{cases} \tag{11}$$

Where l is the width per unit grid, and $G(i)$ is the state of grid i. A value of 0 indicates a feasible grid, and a value of 1 indicates a static obstruction. j is the direction of the grid to be selected with respect to grid i. i, i'', i_1, i_2 are the neighbouring grids perpendicular to the direction of the slash.

Pheromone Update Strategy

The pheromone factor determines how important it is for the ant to be influenced by the pheromone concentration as the ant travels to the next grid in the pathfinding process. To enhance the colony's global search ability and avoid the occurrence of local optimum or "early maturity," the pheromone factor was set smaller to reduce the pheromone guidance to the colony's pathfinding. As the number of iterations increases, the ant colony accumulates more pheromones on better paths and fewer pheromones on worse paths during pathfinding. By increasing the pheromone factor, the ant colony follows the paths with higher pheromone concentration, narrowing the search range and speeding up the algorithm's convergence. As shown in Equation 12:

$$\alpha = \begin{cases} \dfrac{NC_{max}}{NC_{max} - NC + 1}, & \alpha \leqslant \alpha_{max} \\ \alpha_{max}, & \text{otherwise} \end{cases} \tag{12}$$

Where $NC_{\max}$ is the maximum number of iterations, and NC is the current number of iterations. To avoid the algorithm falling into local optimum due to excessive pheromone importance in the later stages of the algorithm, $\alpha_{\max}$ is set as the upper bound threshold of pheromone importance.

Result and Discussion

Experimental Preparation

All algorithms in this paper have been implemented by MATLAB simulation. The map is modelled using the grid method with a map size of 20 x 20. To avoid errors from a single experiment, 30 simulations are carried out and the experimental data are averaged. To increase the realism of the map, static obstacles were randomly distributed in the map grid. The performance of the basic and improved ant colony algorithms was analyzed in a 20×20 raster size environment, and the relevant parameters are shown in Table 1.

Table 1 Parameter values of the improved ACO

Parameter	**Value**
$NC_{\max}$	50
Q	100
$\mathrm{Tau}_{\min}$	5
λ	27
ω	27
$\alpha_{\max}$	2
M	50
$\mathrm{Tau}_{\max}$	30
σ	20
l	21
β	3

Comparison of Algorithm Performance

The basic ACO algorithm was compared with the improved ACO algorithm on a map size of 20×20 grid. The simulation results are shown in Figure 7 and Table 2.

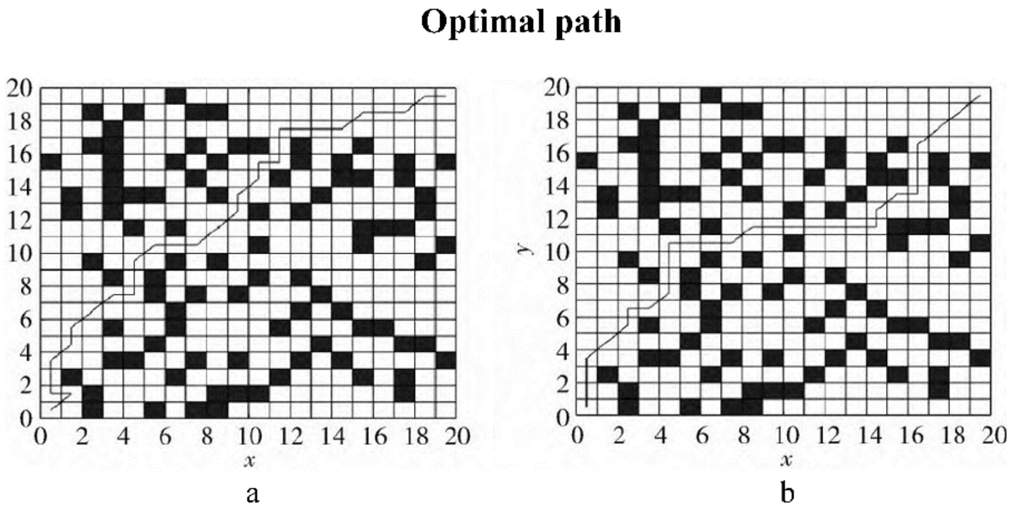

Figure 7 Simulation Results

Table 2 Simulation Results

Optimal Path Indicators	The Basic ACO	The Improved ACO
Path length	35.1	33.6
Turning times	19	14
Number of iterations stable	16	9
Iteration stabilization time / s	0.259	0.175
Program run time / s	1.130	1.268

Figure 7 (a) and Figure 7 (b) show the optimal planning paths for the basic and improved ant colony algorithms for a grid map size of 20 × 20, respectively. It can be seen from the diagram that the improved ACO algorithm avoids "dead ends" and reduces the length of the path and the number of turns. And as can be seen from Table 2, the improved ACO reduces the path planning by 4.3%, the search efficiency by 43.8%, and the iteration stabilization time by 32.4% compared to the basic ACO.

Conclusions

Improvements to the ACO's initialization of pheromone concentrations and heuristic functions in the construction of 2D planar maps have improved the overall performance of the ACO, for example, by increasing the speed of convergence, finding the global optimal path and increasing the stability of the algorithm. By introducing a "directional numbering" for each raster to avoid "dead ends" and a "cornering" mechanism, the distance between the optimal path and the obstacle is reasonably increased. This improves the safety and reliability of the AGV path. By introducing a dynamic pheromone factor, the algorithm avoids the phenomenon of "premature" in the early stage and partial optimization in the later stage. Simulation experiments and results show that the algorithm can achieve the optimal ratio between path length reduction and the number of turns and plan a safe and reliable optimal path for the AGV. Compared with other algorithms, the algorithm in this paper outperforms in terms of convergence, stability and finding the shortest path. The algorithm performs better than others regarding convergence, stability and finding the shortest path. The algorithm plans the global optimal path based on the AGV path's safety, which improves convergence speed and enhances the algorithm's stability. In future research, the particle swarm algorithm can be combined with parameter optimization to plan more intelligent paths. The sliding window method can be introduced to enable the AGV to achieve dynamic obstacle avoidance in dynamic environments.

References

Akka, K., & Khaber, F. (2018). Mobile robot path planning using an improved ant colony optimization. *International Journal of Advanced Robotic Systems*, *15*(3), 1729881418774673.

De Ryck, M., Versteyhe, M., & Debrouwere, F. (2020). Automated guided vehicle systems, state-of-the-art control algorithms and techniques. *Journal of Manufacturing Systems*, *54*, 152–173.

Kim, C. W., & Tanchoco, J. M. A. (1991). Conflict-free shortest-time bidirectional AGV routeing. *The International Journal of Production Research*, *29*(12), 2377–2391.

Li, W., & Chen, X. (2022). UAV path planning based on improved ant colony algorithm with multiple heuristic factors. *International Conference on Advanced Manufacturing Technology and Manufacturing Systems (ICAMTMS 2022)*, *12309*, 702–709.

Mullen, R. J., Monekosso, D., Barman, S., & Remagnino, P. (2009). A review of ant algorithms. *Expert Systems with Applications*, *36*(6), 9608–9617.

Sun, J., Xin, L., & others. (2021). Research on Path Planning of AGV Based on Improved Ant Colony Optimization Algorithm. *2021 33rd Chinese Control and Decision Conference (CCDC)*, 7567–7572.

Umar, U. A., Ariffin, M. K. A., Ismail, N., & Tang, S. H. (2012). Priority-based genetic algorithm for conflict-free automated guided vehicle routing. *Procedia Engineering*, *50*(9), 732–739.

Xiao, J., Yu, X., Sun, K., Zhou, Z., & Zhou, G. (2022). Multiobjective path optimization of an indoor AGV based on an improved ACO-DWA. *Mathematical Biosciences and Engineering*, *19*(12), 12532–12557.

Yu, W., & Egbelu, P. J. (2001). Design of a variable path tandem layout for automated guided vehicle systems. *Journal of Manufacturing Systems*, *20*(5), 305–319.

Cenerini, J., Mehrez, M. W., Han, J.-w., Jeon, S., & Melek, W. (2023). Model Predictive Path Following Control without terminal constraints for holonomic mobile robots. Control Engineering Practice, 132, 105406. doi:https://doi.org/10.1016/j.conengprac.2022.105406

Yu, Z., Yuan, J., Li, Y., Yuan, C., & Deng, S. (2023). A path planning algorithm for mobile robot based on water flow potential field method and beetle antennae search algorithm. Computers and Electrical Engineering, 109, 108730. doi:https://doi.org/10.1016/j.compeleceng.2023.108730

Zhang, D., Luo, R., Yin, Y.-b., & Zou, S.-l. (2023). Multi-objective path planning for mobile robot in nuclear accident environment based on improved ant colony optimization with modified A∗. Nuclear Engineering and Technology, 55(5), 1838-1854. doi:https://doi.org/10.1016/j.net.2023.02.005

Zhao, X., & Cheah, C. C. (2023). BIM-based indoor mobile robot initialization for construction automation using object detection. Automation in Construction, 146, 104647. doi:https://doi.org/10.1016/j.autcon.2022.104647

Zheng, L., Sayed, T., & Essa, M. (2019). Validating the bivariate extreme value modeling approach for road safety estimation with different traffic conflict indicators. Accident Analysis & Prevention, 123, 314-323. doi:https://doi.org/10.1016/j.aap.2018.12.007