An Adaptive Ant Colony Algorithm for Autonomous Vehicles Global Path Planning

Yanqiang Li[1], Yu Ming[1], Zihui Zhang[1,*], Weiqi Yan[2], Kang Wang[1]
[1] Institute of Automation, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China
[2] Auckland University of Technology, Auckland 1010, New Zealand
liyq@sdas.org, 1181471757@qq.com, zihuizhang@163.com, weiqi.yan@aut.ac.nz, wangkang1551@163.com

***Abstract*—In order to improve the robustness of the autonomous vehicle path planning algorithm and reduce the number of turns in the planned path, this paper proposes an adaptive ant colony algorithm path planning method. The algorithm optimizes the initial pheromone matrix based on the environment map, reduces the blindness of the initial ant colony in pathfinding, and improves the convergence speed. Then an adaptive heuristic function is used, which adaptively adjusts according to the different proportions of the heuristic function in the algorithm process, so as to avoid the algorithm being trapped in local optimum. The pheromone is updated according to the corners of the planned route, reducing the acute angle of the route and unnecessary turns to further optimize the route. The simulation results show that the proposed algorithm achieves good results. The simulation results show that the improved adaptive ant colony algorithm has faster convergence speed, higher path planning quality, and improved stability of planned paths than classical ant colony algorithms and other adaptive ant colony algorithms.**

Keywords—path planning, ant colony algorithm, heuristic function, adaptive optimization

I. INTRODUCTION

With the development of artificial intelligence(AI), autonomous vehicles can provide safer, more comfortable and energy efficient ways of travel, which is an internationally recognized development direction and research focus [1]. Automatic driving technology is usually composed of three modules: sensing positioning, decision planning and control execution. Path planning is one of the main components of decision planning module. It is divided into global path planning and local path planning, which is also called static planning and dynamic planning [2]. At present, the path planning method of autonomous vehicles is mainly to plan the global path first, then plan the local path, and finally follow the path by the vehicle control module to reach the destination. This paper studies the method of global path planning. Global path planning is to plan a collision-free path from the start point to an end point in a known map by exploring feasible path, and its processing object is a known static obstacle. By using different path planning algorithms, such problems can be effectively solved.

Path planning algorithms can be divided into traditional algorithms, graph search algorithms, group intelligent optimization algorithms, and so on. Traditional path planning algorithms mainly include Rapidly-exploring Random Tree method (RRT) [3], Dynamic Window Approach (DWA) [4] and so on. Traditional algorithms are generally simple in structure and fast in calculation, but difficult to adapt to complex environment. Graph search algorithms include A* algorithm [5], D* algorithm [6], et al. The graph search algorithm has low probability bias and strong stability, but it cannot guarantee that the planned path is optimal. Group intelligence algorithms include ant colony algorithm [7], particle swarm algorithm [8] and so on. The group intelligent algorithm has strong adaptability and robustness, but it is easy to fall into local optimum. Recently, researchers have applied artificial neural network algorithms[9] and reinforcement learning methods[10] to path planning problems, and achieved good results. Ant Colony Algorithm (ACO), as a kind of swarm intelligent optimization algorithm, is characterized by simple integration of individual work for distributed solution, high quality path planning, but its disadvantage is that the convergence speed is slow and it is easy to fall into local optimization. Aiming at the shortcomings of ant colony algorithm, Liu Jianhua et al. introduced potential field forces into pheromone distribution strategy to improve the ant colony's local search ability and convergence speed[11]. Huang Chen et al. proposed using the A* algorithm's valuation function to set the initial pheromone, combining the multi-strategy evolution mechanism with the closed-loop control system to avoid local optimization and solve the stagnation problem of the ant colony algorithm[12]. Tan Huisheng et al. first used Dijkstra algorithm to find the rough path, and then used ant colony algorithm to get the shortest path near the rough solution, so as to quickly obtain the global optimal solution[13]. Gan Yi et al. quickly found the optimal path by choosing the way of releasing better pheromone and limiting pheromone concentration[14]. The current improved methods of ant colony algorithm mainly focus on the improvement of heuristic function, the optimization of pheromone update method and probability selection.

How to balance the relationship between exploration and development in the path planning algorithm is the main solution by the current path planning algorithm. The exploration process is the key to solving the local optimal problem. By expanding the search area, more feasible paths are explored, but the excessive proportion of exploration will result in slow convergence of the algorithm, increased invalid search, and waste of computing resources. The development process is more centralized pathfinding, the purpose is to find the path between the starting point and the target point more quickly, and the excessive proportion of the development part will produce local optimum and premature convergence. Through the appropriate probability selection method, the location of the waypoint can be updated according to different probability distributions and rules, and the exploration and development process can be reasonably allocated.

The adaptive ant colony algorithm (AACO) proposed in this paper optimizes the initial pheromone by preprocessing the environment map. Manhattan distance and adaptive factors are introduced into the heuristic function to improve the convergence speed of the algorithm. The optimized probability selection strategy balances the relationship between exploration and development, with the aim of avoiding the exploration getting trapped in local optimal values. Improve planning quality by reducing the number of turns in the path.

II. Description Of Classical Ant Colony Algorithm

Ant colony algorithm (ACO) is based on the principle that ants release pheromones on the way to forage. Other ants use pheromone concentration to determine the location of food. Ant colony algorithm consists of three parts: state transition probability, pheromone assignment and heuristic function. The probability of state transition $P_{ij}^k(\mathrm{t})$ represents the probability of ant k transferring from point i to point j at time t, as in (1).

$$P_{ij}^k(\mathrm{t}) = \begin{cases} allowed_k = \{C - Tube_k\} \\ \dfrac{[\tau_{ij}(\mathrm{t})]^\alpha [\eta_{ij}(\mathrm{t})]^\beta}{\sum_{s\in allowed_k} [\tau_{is}(\mathrm{t})]^\alpha [\eta_{is}(\mathrm{t})]^\beta}, j \in allowed_k \end{cases} \quad (1)$$

Where $\tau_{ij}(\mathrm{t})$ is the pheromone of ant k from point i to point j at time t, $\eta_{ij}(\mathrm{t})$ is the heuristic function of ant k from point i to point j at time t, α is the pheromone weight coefficient, and the proportion of determinant is determined, β is heuristic weight coefficient, which measures the proportion of the heuristic function, C represents the set of feasible points of the map, $Tube_k$ is the tabu table, and records that the ant k has passed the path.

Pheromone τ provides the reference direction of the path for other ants, and the pheromone concentration decreases with the increase of time, as in (2).

$$\begin{cases} \tau_{ij}(\mathrm{t+n}) = (1-\rho)\tau_{ij}(\mathrm{t}) + \Delta\tau_{ij}(\mathrm{t}) \\ \Delta\tau_{ij}(\mathrm{t}) = \sum_{k=1}^{m} \Delta\tau_{ij}^k(\mathrm{t}) \end{cases} \quad (2)$$

Where ρ is the pheromone volatility coefficient, which makes the pheromone function volatile. $\Delta\tau_{ij}(\mathrm{t})$ is the pheromone accumulation amount, and the pheromone left by each ant passing from point i to point j is recorded to update the pheromone concentration. According to different pheromone update strategies, there are three models for the calculation of $\Delta\tau_{ij}(\mathrm{t})$, which are Ant - Cycle model, Ant-Quantity model, and Ant-Density model, as shown in (3), (4), and(5) .

$$\Delta\tau_{ij}^k(\mathrm{t}) = \begin{cases} \dfrac{Q}{L_k}, Ant\ \mathrm{k}\ goes\ through\ \mathrm{(i,j)}\ in\ this\ cycle \\ 0, others \end{cases} \quad (3)$$

$$\Delta\tau_{ij}^k(\mathrm{t}) = \begin{cases} \dfrac{Q}{d_{ij}}, Ant\ \mathrm{k}\ passes\ through\ \mathrm{(i,j)}\ at\ time\ \mathrm{(t,t+1)} \\ 0, others \end{cases} \quad (4)$$

$$\Delta\tau_{ij}^k(\mathrm{t}) = \begin{cases} Q, Ant\ \mathrm{k}\ passes\ through\ \mathrm{(i,j)}\ at\ time\ \mathrm{(t,t+1)} \\ 0, \mathrm{others} \end{cases} \quad (5)$$

Where Q is the pheromone intensity coefficient. The Ant-Cycle model updated the pheromone on the overall path after the completion of the cycle, the pheromones of the Ant-Quantity model and the Ant-Density model is updated in real time based on the progress of each ant. We selected the Ant-Cycle model for more pheromones.

Heuristic function is used to guide ants to move towards the end point. The heuristic function of ant colony algorithm usually selects the reciprocal of the distance between two points. The probabilistic selection of the ant colony algorithm generally uses the method of roulette, and the probability selection is based on the individual fitness, which ensures the diversity of the algorithm.

III. An Adaptive Ant Colony Algorithm

An adaptive ant colony algorithm (AACO) is proposed for global path planning problems. The AACO algorithm optimizes the initial pheromone and pheromone accumulation function, introduces the Manhattan distance and adaptive factor into the heuristic function. Thus, the search efficiency and convergence speed are improved. Besides, the algorithm also adds a random probability selection strategy to prevent the algorithm from falling into a local optimum.

The flowchart of the adaptive ant colony algorithm is shown in Fig.1.

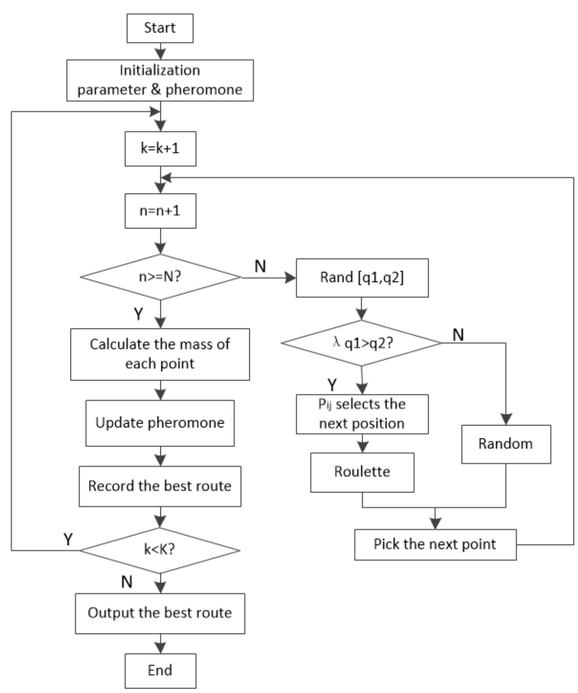
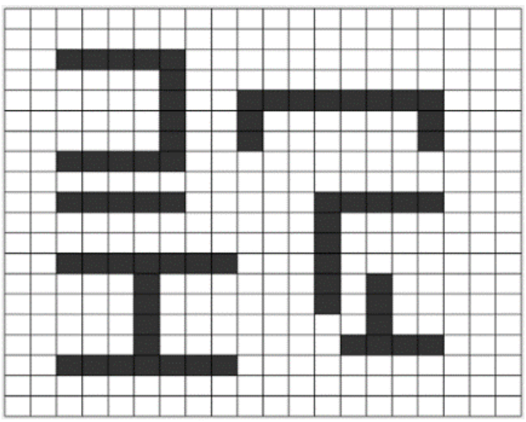


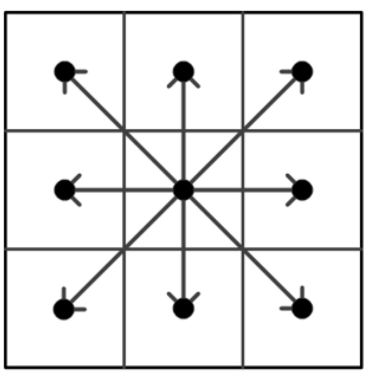
Fig. 1. Flowchart of adaptive ant colony algorithm

A. Build environment maps

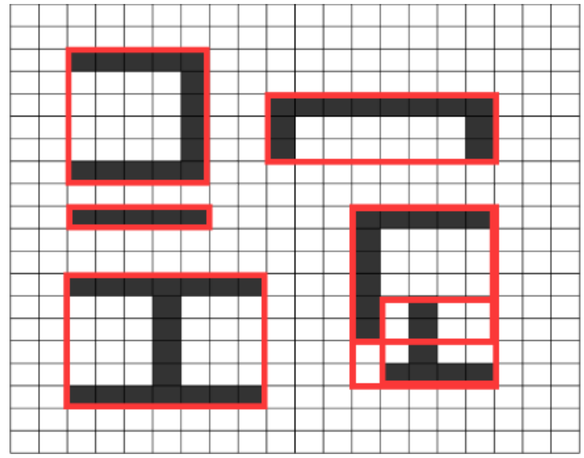
To simplify the problem, the autonomous vehicle is approximated as a particle. A two-dimensional grid map of the environment is known, the start and end point are determined in the map, and an optimal path from the start point to the end point is finally found. First, a two-dimensional grid map simulation environment is established. In the grid map, as long as there are obstacles in the unit grid, we will set the entire grid as an obstacle grid. The static environment is used as the premise, and assume that the obstacles grid has been inflated , as shown in Fig.2(a). The black grid represents obstacles and the white grid represents feasible grids. The grids are sorted in the order of top to bottom and left to right, the pathfinding rule is 8 neighborhood mode, and the pathfinding position is each grid center, as shown in Fig.2(b).



a. Two-dimensional grid map



b. The direction and position of the ant's movement



c. Obstacle regularization method

Fig. 2. Grid map and pathfinding rules

B. Pheromone Allocation Strategy

For the path planning problem, the AACO initializes the pheromone through the global grid map, and regularizes the concave or convex regions formed by the irregular obstacles, as shown in Fig.2 (c). Based on the processed raster map, the initial distribution of pheromone concentration was carried out to increase the concentration of free region and reduce the concentration of feasible region in the obstacle block. This method ensures that more ants will explore the effective area in the early stage of pathfinding, reduces the number of ants that fail in the early stage of pathfinding, and speeds up the convergence speed.

In the pheromone update phase, the ant records the pheromone increments of the adjacent two parts at each step. To avoid the acute angle appearing in the ant planning route, the angle factor γ is used to reduce the pheromone concentration of the planning path with the acute angle. At the end of the loop, the current optimal path is extracted, and obtain the corner point position and angle type on the path. According to (6), the angle factors γ of the corresponding obtuse angle, acute angle and straight line are substituted into $\Delta\tau_{ij}^{k}(\mathrm{t})$ to obtain (7). The acute angle corresponds to the smallest γ value, which reduces the probability that other ants choose the acute angle segment, and promotes the ant to find the path containing more straight lines and obtuse angles.

$$\begin{cases} \gamma_1, & \theta < 90 \\ \gamma_2, & \theta = 180 \\ \gamma_3, & \text{others} \end{cases} \tag{6}$$

$$\Delta\tau_{ij}^{k}(\mathrm{t}) = \begin{cases} \dfrac{Q}{L_k + \gamma} \\ 0 \end{cases} \tag{7}$$

C. Heuristic Function

Heuristic functions in grid maps typically use Manhattan distances, Euclidean distances, diagonal distances, etc., where Manhattan distance calculations as faster. Therefore, AACO is optimized by introducing Manhattan distance into the heuristic function. The optimized heuristic function is shown in (8).

$$\eta_{ij} = \frac{1}{d_{ij} + [abs(x_j - x_{end}) + abs(y_j - y_{end})]} \tag{8}$$

In the initial stage of path planning, ants need to find the correct path direction and target point position as soon as possible. At this time, the heuristic function has a great influence on ants, and it is necessary to enhance the proportion of heuristic information. In the later stage of search, ants need to expand the search area to obtain the optimal solution, and need to weaken the heuristic function. Therefore, the adaptive coefficient ω is introduced into the heuristic function, as shown in (9). The adaptive coefficient is shown in (10), which can adaptively adjust the influence of the heuristic function during the path planning of the ant colony algorithm, improve the quality of the path planned by the algorithm, and ensure the convergence speed.

$$\eta_{ij} = \frac{1}{d_{ij} + \omega * [abs(x_j - x_{end}) + abs(y_j - y_{end})]} \tag{9}$$

$$\omega = 1 - \frac{\sqrt{(x_{start} - x_j)^2 + (y_{start} - y_j)^2}}{\sqrt{(x_{start} - x_{end})^2 + (y_{start} - y_{end})^2}} \tag{10}$$

Due to the guiding role of heuristic information, the optional probability is rapidly reduced, and the regional exploration ability is greatly reduced [12]. Therefore, the evolution factor λ and the strategy selection random number q_1 , q_2 are introduced, as shown in (11). When the grid with high probability of state transition is selected, the random selection probability is appropriately increased to avoid premature convergence of the ant colony algorithm and fall into local optimum.

$$\begin{cases} Roulette\ strategy, & \lambda q_1 > q_2 \\ Random\ strategy, & \lambda q_1 \leq q_2 \end{cases}, \lambda \in [1,10] \tag{11}$$

IV. SIMULATION

The algorithm is verified by 20×20 environment grid map. The starting point coordinate S=1 on the map and the ending point coordinate E=400. The traditional ant colony algorithm parameters are set to K=100, M=500, α =1, β =7, ρ =0.3, Q =1. In this paper, the basic parameters of the adaptive ant colony algorithm are set as above, and the rest of the parameters are set to γ_1 =4, γ_2 =6, γ_3 =9, λ =7. Parameter setting is based on multiple experimental results.

A. Heuristic function optimization experiment

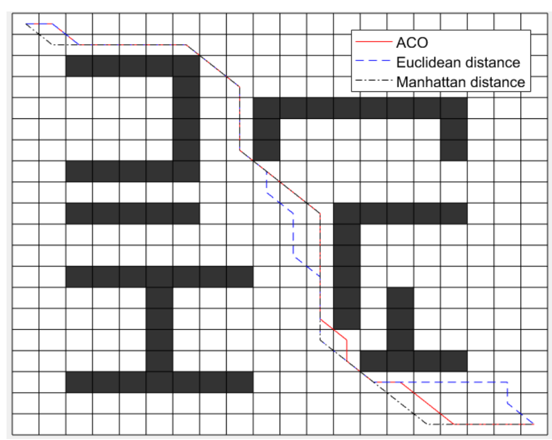
The selection of heuristic function directly affects the path planning quality and convergence of ant colony algorithm. Therefore, the classical ant colony algorithm and the ant colony algorithm with Euclidean distance heuristic function and Manhattan distance heuristic function are tested respectively, the experimental results of the three methods are analyzed and compared.

The ant colony algorithm of the three heuristic functions was tested for 10 times respectively, and the average value of each index was taken as shown in the TABLE I.

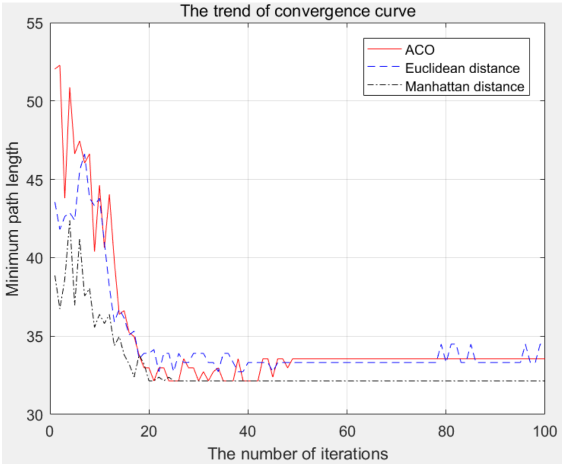
TABLE I. COMPARISON OF THREE HEURISTIC FUNCTIONS

Heuristic function mode	Number of turns	Path length	Longest and shortest paths difference	Running time
ACO	13	32.1	20.9	16.0
Euclidean distance	13	34.7	13.8	18.4
Manhattan distance	10	32.1	10.5	17.2

The experimental results are shown in figure 3. From the path planning effect and the convergence curve of the algorithm, it can be seen that the heuristic function of the traditional ant colony algorithm produces many path angles and the difference of the planned path is large. The heuristic function using Euclidean distance has poor convergence, and the quality of the planned path is not better. The heuristic function based on Manhattan distance has strong convergence and the planning path has high quality.



a. Planning Path Comparison



b. Convergence Curves Comparison

Fig. 3. Comparison of three heuristic function effects

B. *Comparison of algorithm simulation results*

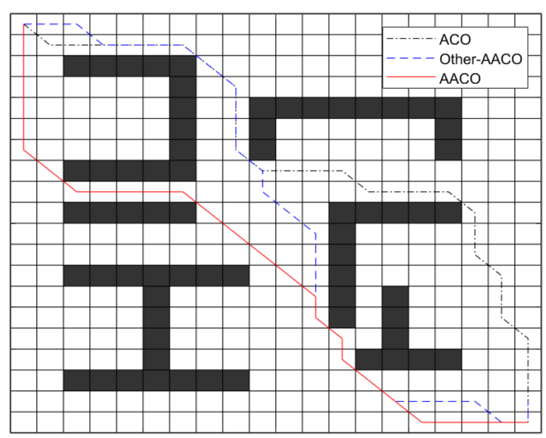
In order to verify the effectiveness of the adaptive ant colony algorithm proposed in this paper, two scenarios are used to compare the experimental results of classical ant colony algorithm (ACO), other adaptive ant colony algorithm (other-AACO) and the adaptive ant colony algorithm (AACO) proposed in this paper. The two scenarios are used to compare and verify the algorithm. The scenario A has a simple structure, and the obstacles are concentrated. It contains various obstacle information, such as narrow gaps, concave and convex obstacles, etc., which are used to test the pathfinding ability of the algorithm in multiple types of obstacles and large free space environment, as shown in Fig.4 (a). Scene B is a discrete obstacle map with many optional paths and less free space. It is used to test the path finding ability of the algorithm in a compact environment, as shown in Fig.5 (a).

In scenario A and scenario B, three algorithms were tested for 10 times respectively, and the average value of each index was shown in TABLE Ⅱ.

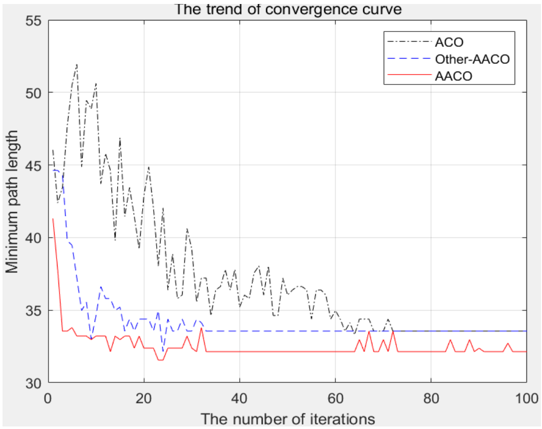
TABLE II. COMPARISON OF CLASSICAL ANT COLONY ALGORITHM AND ADAPTIVE ANT COLONY ALGORITHM

Scenario	Algorithm	Number of turns	Path length	Longest and shortest paths difference	Running time
Scenario A	ACO	11	33.5	20.6	10.5
	Other-AACO	12	32.6	18.2	10.9
	AACO	10	31.7	11.7	11.7
Scenario B	ACO	13	29.8	16	10.7
	Other-AACO	14	29.8	13.2	9.9
	AACO	12	29.8	6.0	10.1

By comparing the experimental results of the three algorithms, the adaptive ant colony algorithm(AACO) performs well in path quality, path length and running time. Especially in the planned path quality, the number of turns in the path is significantly reduced compared with the ACO and Other-AACO. According to Fig.4, the free area in the scene A is large, and the classical ant colony algorithm has a short running time due to the single probability selection strategy and the pheromone allocation strategy, but the planned path has a large number of turns and poor convergence. Although other adaptive ant colony algorithms shorten the planned path length, the number of turns is large. The adaptive ant colony algorithm proposed in this paper uses random selection strategy and angle judgment factors to the pheromone allocation strategy, which significantly reduces the number of path turns, shortens the path length, and improves the convergence speed of the algorithm. But it also shows that the running time has increased. In scenario B, the obstacles are compact and have few free regions, as shown in Fig.5. The simple selection strategy in the classical algorithm is no longer dominant. There are a large number of turns in other adaptive ant colony algorithm planning paths, which affects the quality of planned paths. The path quality of the adaptive ant colony algorithm is higher, and the search path length is stable. Although the exploration range is large, the running time is shortened and the convergence effect is better.

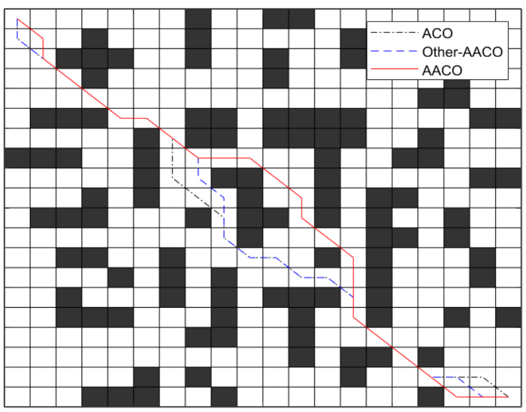


a. Algorithm Planning Path Comparison

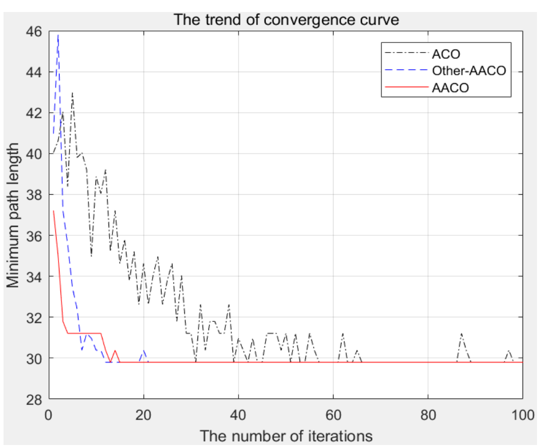


b. Algorithm Convergence Curves Comparison

Fig. 4. Comparison of Calculation Results in Scenario A



a. Algorithm Planning Path Comparison



b. Algorithm Convergence Curves Comparison

Fig. 5. Comparison of Calculation Results in Scenario B

V. Conclusion

In this paper, the ant colony algorithm is optimized, and the initial pheromone based on the environment map is optimized to improve the purpose of ant colony in the early stage of pathfinding. The Manhattan distance is introduced into the heuristic function and the adaptive coefficient of the heuristic function is added to accelerate the pathfinding speed of the ant colony and increase the stability of the algorithm. Optimize the transition probability to avoid premature and fall into local optimum. The final path planning was optimized by considering the route turning angle. The future work will be based on vehicle dynamics, and the dynamic path planning of ant colony algorithm by predicting the motion state of dynamic obstacles.

Acknowledgments

This paper was supported by the International cooperation project of Shandong academy of sciences 2019GHZD08 and funded by the Key Research and Development Program of Shandong with grant number 2017GGX50107.It is also grateful for Specially Recruited Experts of Taishan Scholar.

References

[1] K.Q.Li, Y.Dai, S.Li and M.Bian, "State-of-the-art and technical trends of intelligent and connected vehicles," J. Auto. safety& energy. 8(1) pp. 1-14.

[2] S.Li, J.Tang, S.Wu and L.Li,"Driverless cars: the future is beginning," in The first book on driverless technology. Beijing: Publishing House of Electronics Industry, 2017, pp. 2-4.

[3] Z. Tahir, A. H. Qureshi, Y. Ayaz and R. Nawaz, "Potentially Guided Bidirectionalized RRT* for Fast Optimal Path Planning in Cluttered Environments," Robotics and Autonomous Systems. 108(2018) pp. 13-27.

[4] J. Ballesteros, C. Urdiales, A.B.M. Velasco and G. Ramos-Jiménez, "A Biomimetical Dynamic Window Approach to Navigation for Collaborative Control," IEEE T HUM-MACH SYST. 47(6) pp. 1123-1133.

[5] W. Wei, P. Dong and F. Zhang, "The shortest pathplanning for mobile robots using improved A* algorithm," Journal of Computer Applications. 38(5) pp. 1523-1526.

[6] X. Zhang and B. Xiao, "Improved D* algorithm for mobile robot path planning," Transducer and Microsystem Technologies. 37(12) pp. 57-59,63.

[7] L. M. Gambardella and M. Dorigo, "Ant-Q: A Reinforcement Learning approach to the traveling salesman problem," Machine Learning Proceedings. 170(3) pp. 252-260.

[8] P.B.Fernandes, R.C.L.De Oliveira and J.V.F.Neto, "A Modified QPSO for Robotic Vehicle Path Planning," in 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, 2018, pp. 1-7.

[9] X. L. Geng and C. J. Li, "Application of Parallel Reinforcement Learning Based on Artificial Neural Network to Adaptive Path Planning," Science Technonlogy and Engineering. 11(4) pp. 756-759.

[10] A. Sharma, K. Gupta, A. Kumar, A. Sharma and R. Kumar, "Model based path planning using Q-Learning," in 2017 IEEE International Conference on Industrial Technology (ICIT), Canada 2017, pp. 837-842.

[11] J.H. Liu, J.G. Yang, H.P. Liu, P. Geng and M. Gao, "Robot Global Path Planning Based on Ant Colony Optimization with Artificial Potential Field," Transactions of The Chinese Society of Agricultural Machinery. 46(09) pp. 18-27.

[12] C. Huang, J. Fei, Y. Liu, H. Li and X.D. Lui, "Smooth Path Planning Method Based on Dynamic Feedback A* Ant Colony Algorithm," Trans. of Agricultural Machinery 48(04) pp. 34-40+102.

[13] H.S. Tan, W. Liao and X.Y. He, "A fast path planning algorithm for mobile robot based on improved ant colony algorithm," Journal of Dynamics and Control. 17(2) pp. 104-111.

[14] Y. Gan, F.T. Qu, F.J. Sun, W.M. He and P.J. Zhang, "Research on path planning for mobile robot based on ACO," in 2017 29th Chinese Control And Decision Conference (CCDC), 2017, pp. 6738-6743.