The Impact of Path Planning Model Based on Improved Ant Colony Optimization Algorithm on Green Traffic Management

Huan Yu
School of Transportation Engineering, Chang'an University, Xi'an, 710064, China
School of Economics and Management, Shaanxi Xueqian Normal University, Xi'an, 710100, China

***Abstract*—In response to the demand for green city construction, low-carbon travel standards have been further implemented. This research focuses on intelligent transportation management and designs path planning algorithms. Firstly, the basic model of the proposed ant colony optimization algorithm was constructed. In response to the poor convergence of traditional algorithms, a rollback strategy was introduced to optimize the model taboo table. Subsequently, in response to the dynamic obstacle avoidance problem in practical applications, the optimized A* algorithm was studied and applied to global path planning. The improved ant colony algorithm was applied to local obstacle avoidance planning, further enhancing the accuracy and practicality of the algorithm. In simulation analysis, facing more complex simulation environments, this research method could better achieve obstacle avoidance path planning. The average number of search nodes decreased by 6, the average search time decreased by 4.11%, and the average path length decreased by 22.07%. In summary, the ant colony optimization algorithm designed through research is more suitable for path planning needs in different scenarios, with the best overall performance. It can plan the shortest driving path while ensuring precise obstacle avoidance, helping to achieve green traffic management.**

***Keywords*—Ant colony optimization; A*; path planning; obstacle avoidance; traffic control**

I. INTRODUCTION

With the continuous construction and development of green smart cities, traffic management has gradually become an important factor restricting urban development. Intelligent Transportation System (ITS) integrates advanced technologies such as information, data communication transmission, and electronic control, significantly improving the efficiency of traffic management. While ensuring traffic safety and improving traffic service levels, it also reduces the impact of vehicle driving on the environment. The Path Planning (PP) module is a core component of ITS, responsible for providing users with the optimal driving route based on real-time traffic environment data. PP technology can be divided into two categories: static PP and dynamic PP [1-2]. The former does not consider environmental changes and is simpler and more direct. The latter requires real-time updates of environmental information to achieve dynamic path adjustment, making it more suitable for complex actual traffic environments. Currently, with the iterative updates of sensor technology, cloud computing, and big data, dynamic PP has made significant progress. It can more accurately reflect real-time road conditions, and improve the efficiency and practicality of PP. Dynamic PP can effectively reduce traffic congestion, improve driving efficiency, and guide vehicles to drive reasonably. The reduction of driving route distance naturally helps to establish low-carbon and green cities. This is in line with the current severe environmental problems and energy crisis, and the needs and goals of various regions for green city construction. However, although dynamic PP has made certain progress in both theoretical and technical aspects, it still faces many challenges in practical applications. Firstly, existing PP algorithms have low computational efficiency when dealing with large-scale and highly complex road networks, making it difficult to meet real-time requirements. Secondly, the fusion and processing technology of multi-source heterogeneous traffic data is not yet mature, which affects the accuracy and reliability of dynamic PP [3-4]. Therefore, how to design a dynamic PP algorithm that is both efficient and accurate, while also taking into account multiple practical needs, is the focus of current research in intelligent transportation. A PP model based on Ant Colony Optimization (ACO) algorithm is proposed to address the aforementioned issues. Its purpose is to improve it through rollback strategies and introduce the A* algorithm to optimize obstacle avoidance accuracy. The contributions of the research are as follows: (1) the basic path planning model based on the improved ant colony optimization algorithm is constructed, and the table of the backward strategy optimization algorithm is introduced to improve the convergence performance of the algorithm. (2) The optimized A* algorithm is studied and applied to global path planning, and the improved ant colony algorithm is applied to local obstacle avoidance planning, which further improves the accuracy and practicability of the algorithm.

The study consists of five sections. Literature review given in Section II. Firstly, the research status of PP is introduced in Section III. Secondly, a dynamic obstacle avoidance model based on ACO is designed in Section IV. Then, actual experiments and simulation analyses are conducted on the performance of the design model. Finally, a summary of the experimental results is provided in Section V.

II. LITERATURE REVIEW

The PP algorithm, as a research hotspot in motion planning, has been widely applied in various fields such as robot design, traffic management, and tourism. Li X et al. proposed an improved compression factor particle swarm optimization method and applied it to the three-dimensional PP of Autonomous Underwater Vehicles (AUVs). In addition, they introduced three-dimensional seabed and Lamb vortex models to optimize navigation costs and ocean current constraints. Their model demonstrated better planning efficiency and path quality [5]. X Wang et al. designed an improved Q-learning algorithm and transformed its learning behavior into a discrete-time Markov chain model. By integrating strategies such as probability calculation tree logic, the effectiveness of PP and the reliability of control systems for mobile agents in uncertain environments were improved in this paper [6]. Y Huang et al. proposed a new underwater robot PP method, which transformed it into a deterministic optimization problem by using whale optimization algorithm and adaptive operator. The introduction of dynamic partitioning and other strategies for virtual individuals improved the search ability of the algorithm. Their method effectively solved the PP problem in complex terrain, improving the model's search ability and robustness [7]. S Zhang et al. proposed a PP model that combined timestamp collision detection and environment improved artificial potential field algorithm. Their model was applicable to the local PP technology of wave gliders, enhancing their obstacle avoidance ability and maneuverability during application [8].

Huo L proposed an improved path selection algorithm and applied it to wireless cloud computing environments to address the characteristics of frequent changes in urban traffic and rich driving paths. And the initial pheromones were non-uniformly dispersed, optimizing urban traffic management planning, improving path search efficiency and user satisfaction [9]. Yang X et al. chose the actor critic algorithm in reinforcement learning to design the PP model and introduced parameter updating and exploration strategies to further optimize it. Finally, it was applied to intelligent ship dynamic obstacle avoidance, improving the performance of the algorithm under complex meteorological conditions [10]. A Zou et al. proposed an innovative robot PP fusion algorithm by combining optimized mayfly algorithm and dynamic window method. The core of the former was the Q-learning algorithm, which could optimize convergence performance through adaptive parameter tuning. Their model reduced the average path length by 6.58% compared to traditional mayfly algorithms in complex environments [11]. Lyridis DV et al. proposed an improved fuzzy ACO for the PP of unmanned surface vehicles. Their method could effectively handle local obstacle avoidance problems and had better PP performance than other algorithms in complex environments [12].

In summary, most PP methods are developed using global or local programming, and their performance needs to be improved. For example, the ability to handle dynamic environments is limited, especially in rapidly changing scenarios, which may cause path failure. The improvement of obstacle avoidance effect may lead to an increase in computational costs, and errors caused by environmental uncertainty may also lead to PP failure. This indicates that it needs to be optimized and improved in multiple aspects such as adaptability, generalization, and security. Therefore, this study achieves a combination of global and local obstacle avoidance through ACO and A* algorithms. This not only improves the obstacle avoidance accuracy of vehicles, but also reduces the computational burden of this model, making it more adaptable in complex operating scenarios.

III. An Intelligent Path Planning Model with Improved Ant Colony Optimization Algorithm and Dynamic Obstacle Avoidance Optimization

To achieve green intelligent traffic management, this study proposes applying ACO to the vehicle PP model. Firstly, the convergence performance of ACO is optimized by building environmental models and other methods. Secondly, to achieve dynamic obstacle avoidance in real-world application scenarios, the A* algorithm is introduced and a dynamic PP model is constructed.

A. *Design of Static Path Planning Model Based on Improved Ant Colony Optimization Algorithm*

The PP module is an important component of smart transportation systems, aimed at planning an optimal operating path under certain constraints. The study chooses classical heuristic ACO as the basis for the PP model. It assumes that the total number of ants is k, and their individuals represent different paths. Starting from the starting point, individuals continuously update the position of the next node as shown in Formula (1).

$$p_{ij}^{k} = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}{\sum_{s\in A}\left[\tau_{is}(t)\right]^{\alpha}\left[\eta_{is}(t)\right]^{\beta}} & if : j \in A \\ 0 & else \end{cases} \tag{1}$$

In Formula (1), p_{ij}^{k} represents the probability of ant k transitioning from position i to position j. t refers to a point in time. τ_{ij} means the concentration of pheromones between nodes at different locations. S is the current location node. A refers to optional location nodes for removing obstacles and other obstacles. α / β means the weights of pheromones and heuristic functions, respectively. η_{ij} represents a heuristic function. The heuristic function is the reciprocal of the distance between two position nodes, expressed by Formula (2) [13-14].

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \tag{2}$$

In Formula (2), d_{ij} represents the distance between two nodes. In addition, each location node will have corresponding pheromones. Whether a node is selected is related to the concentration of pheromones. The more times an individual passes through a node at that location, the higher the corresponding pheromone. However, traditional ACO has excessive computational pressure, poor convergence performance, and is prone to falling into local optima. Therefore, the study addresses the above issues and makes improvements to them. Firstly, the vehicles are simplified as particles, and a two-dimensional grid model is constructed based on the road environment in Fig. 1.

In the two-dimensional grid map of Fig. 1 (a), the black part refers to the obstacle area. Obstacle setting can effectively test the optimization ability of the model. The search path follows the rule of 8-neighborhood representation in Fig. 1 (b). The position of each search point is the center point of the grid. The same initial pheromones in traditional ACO can reduce the accuracy of path search and increase its computational time. Therefore, the study proposes using the starting and ending line as the criterion for pheromone allocation, and the smaller the distance from the line, the greater the pheromone value. This can further improve the efficiency of global search, represented by Formula (3).

$$\begin{cases} \tau = \tau_0 + \vartheta C \\ \vartheta = \mu\varepsilon \end{cases} \tag{3}$$

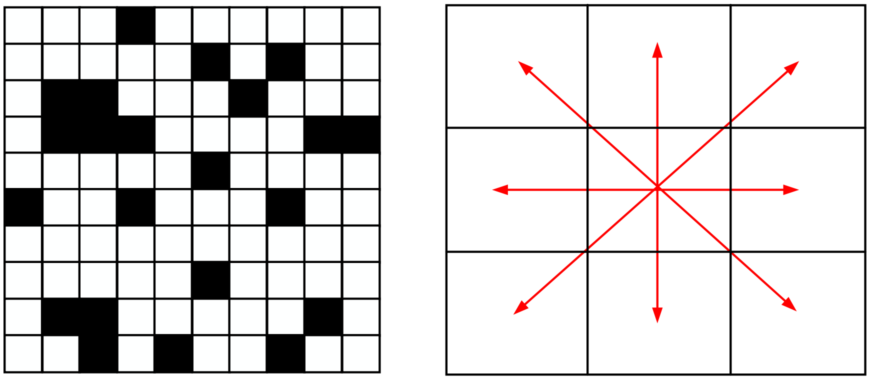
In Formula (3), τ, τ_0 represent the pheromone and basic pheromone of the algorithm, respectively. ϑ is an adaptive parameter. μ means the distance between nodes and connecting lines distributed within (0,1). ε refers to the proportion of obstacles distributed within (0,1). When there are no obstacles on the line, the depth of the grid color is proportional to the pheromone value. When there are obstacles on the connection, the overall pheromones of these other grids will decrease. This uneven initial pheromone concentration distribution is more conducive to the search for the optimal path. In addition, traditional heuristic function calculation methods suffer from low search efficiency and weak heuristic. Therefore, the study introduces a new Manhattan distance to calculate the heuristic function. Compared with Euclidean and diagonal equidistance, the calculation time of Manhattan distance is relatively shorter, expressed by Formula (4).

$$h(n) = D\left[abs\left(x_n - x_{end}\right) + abs\left(y_n - y_{end}\right)\right] \tag{4}$$

In Formula (4), x_n / y_n represents the two-dimensional coordinates of the starting node position. x_{end} / y_{end} is the two-dimensional coordinate of the termination node position. The heuristic function of Manhattan distance is introduced, represented by Formula (5) [15].

$$\eta_{ij} = \frac{1}{d_{ij} + \left[abs\left(x_j - x_{end}\right) + abs\left(y_i - y_{end}\right)\right]} \tag{5}$$

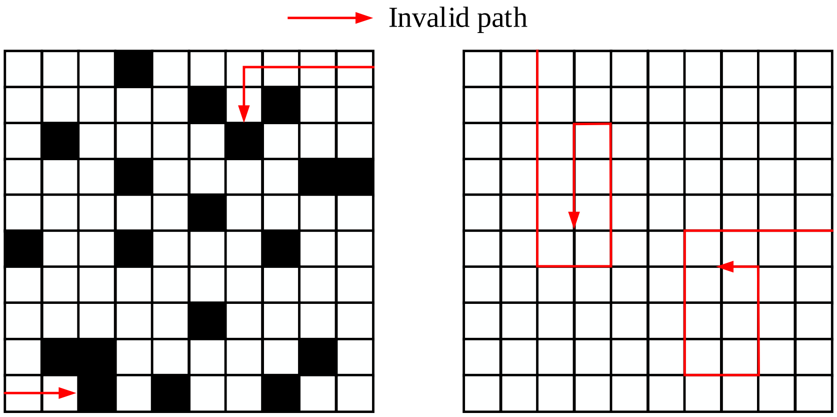
When an individual encounters obstacles in their search path, which cannot be avoided, or when there are taboo list restrictions, the path will be invalidated. Invalid paths include deadlocks and self-locking in Fig. 2.



(a) Two-dimensional raster map

(b) Rules of ant colony pathfinding

Fig. 1. Visual environment modeling.



(a) Deadlock condition

(b) Self-locking condition

Fig. 2. Diagram of invalid search path.

The study introduces a rollback strategy to optimize it. And the taboo table is divided into global and local categories, with the former recording deadlock routes and the latter recording self-locking and walking route nodes. Next, the study further introduces a pheromone update strategy to reduce data redundancy, represented by Formula (6).

$$\tau_{ij}\left(t+n\right)=\left(1-\rho\right)\tau_{ij}\left(t\right)+\Delta\tau_{ij}\left(t,t+n\right)+h\frac{L-L_n}{L_n} \tag{6}$$

In Formula (6), L, L_n represent the local and global optimal paths. h refers to the adjustable coefficient. ρ means the volatile factor of pheromones. The model only updates the shortest path pheromone. When $L > L_n$, it enhances the pheromone of the latest L. Otherwise, it decreases its pheromone. This method of updating pheromones only for the shortest path improves the convergence performance of the algorithm. In summary, the operational process of ACO has been improved in Fig. 3.

Firstly, it is necessary to build a virtual grid map through a real environment. Next, parameter initialization is carried out, which confirms the starting and ending points. After calculating the heuristic function at the starting node, path search can be performed. Node search needs to consider constraints such as taboo tables. This method iterates continuously until it reaches the endpoint.

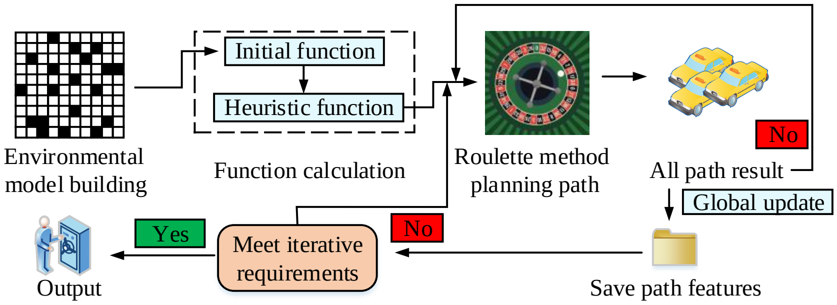


Fig. 3. Operation flow of improved ant colony optimization algorithm.

B. Dynamic Path Planning Model Integrating A*-ACO Optimization Algorithm

In practical intelligent traffic management applications, obstacles are often dynamically changing. Therefore, the study introduces the A* algorithm to improve ACO and builds a dynamic obstacle avoidance planning model. The classic A* algorithm updates the path by continuously updating the node cost and selecting the node with the lowest cost. Its performance is highly correlated with the heuristic function. If the heuristic function is too large, it will prioritize width. Otherwise, it is easier to complete the global optimal search. Common heuristic functions include Manhattan distance and Euclidean distance. However, both of these heuristic functions only have four search directions, which is not conducive to the global search of the algorithm and also increases the computational burden, resulting in a more tortuous path. However, to increase the operability of the search path, the path should be made as smooth as possible. Therefore, an improved heuristic function is proposed, which combines two heuristic functions to obtain twice the search direction. The estimated cost $h(n)$ is represented by Formula (7) [16-17].

$$h(n)=\max\left(abs\left(n_x-g_x\right),abs\left(n_y-g_y\right)\right) \tag{7}$$

In Formula (7), (n_x, n_y) represents the coordinates of the current node n. (g_x, g_y) refers to the coordinates of the target node G. abs means going to absolute values. In addition to smooth paths, this model also needs to implement dynamic obstacle avoidance. The motion model of obstacles is represented by Formula (8).

$$\begin{cases}\dot{x}=v\cos\left(\theta\right)\\ \dot{y}=v\sin\left(\theta\right)\\ \dot{\theta}=\kappa v\\ \omega=v\kappa\end{cases} \tag{8}$$

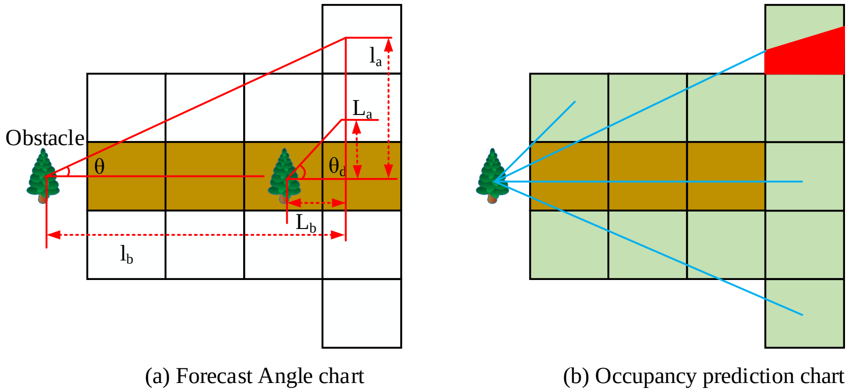
In Formula (8), v, ω represent velocity and angular velocity, respectively. θ is the safe steering prediction angle for speed. κ refers to curvature. Fig. 4 shows a dynamic obstacle model.

In Fig. 4, the angle of the dynamic obstacle also includes an emergency turn prediction angle θ_d, represented by Formula (9).

$$\begin{cases}\theta=\arctan\left(\dfrac{l_a}{l_b}\right)\\ \theta_d=\arctan\left(\dfrac{L_a}{L_b}\right)\end{cases} \tag{9}$$

In Formula (9), L_a, L_b represent the vertical and horizontal offset distances during emergency turns, respectively. l_a, l_b refer to the vertical and horizontal distances of slight path offset, respectively. To analyze whether the next node is affected by dynamic obstacles, the ratio of its grid area is calculated using Formula (10) [18].

$$f=\frac{\gamma}{L_i}+m \tag{10}$$



(a) Forecast Angle chart (b) Occupancy prediction chart

Fig. 4. Visual analysis of dynamic obstacles.

In Formula (10), $\gamma = \sin(\theta_i)$ represents the adaptive parameter. θ_i refers to the angle between the center of grid i and the direction of obstacle velocity, representing the proportion of the affected area of the grid to the total area. L_i means the vertical distance between the center of i and the direction of obstacle velocity. When f < 0.5, the corresponding grid is not affected by obstacles and is marked in green. On the contrary, the affected area is marked in red. The motion path of dynamic obstacles may move along the original direction or deviate to varying degrees toward the green area. The probability of deviation is positively correlated with the length of the obstacle's motion path, indicating a safe distance between the vehicle and the obstacle in Fig. 5.

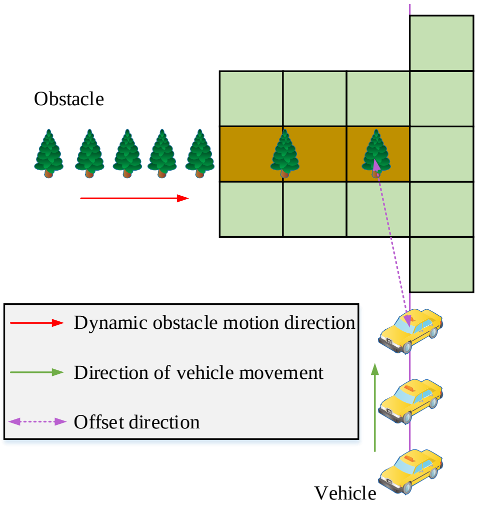

Obstacle
Dynamic obstacle motion direction
Direction of vehicle movement
Offset direction
Vehicle

Fig. 5. Location diagram of vehicle and obstacle.

In Fig. 5, the safety distance S is represented by Formula (11).

$$S = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \tag{11}$$

In Formula (12), $(x_a, y_a), (x_b, y_b)$ represent the vehicle position and obstacle position, respectively. The premise for predicting the distance of obstacle movement is that the distance between the vehicle and the obstacle is not greater than the safe distance. The cumulative trajectory length $F(s)$ of the obstacle is represented by Formula (12).

$$F(s) = \sum_{S<S_0} S_b \tag{12}$$

In Formula (12), S_b represents the trajectory length of the obstacle. S_0 refers to the safety distance threshold. Therefore, after detecting and predicting the movement direction of obstacles and potential occupied nodes, they should be added to the temporary taboo list. Subsequently, during ACO runtime, the past taboo nodes are deleted one by one until the temporary taboo table is cleared. When obstacles are detected, local obstacle avoidance is achieved through ACO, and the volume and position of obstacles are uncertain. The obstacle avoidance strategy includes two types. Firstly, the distinction is made based on the angle between the directions of two objects. If the angle is an obtuse angle, it is considered to be an encounter, and vice versa, it is considered a pursuit. Both need to call occupancy prediction after detection to realize obstacle avoidance. In addition, as the distance between the two gradually increases, it is necessary to make another occupancy prediction. To improve the operational efficiency of occupancy prediction, an information inheritance strategy is introduced, represented by Formula (13).

$$\begin{cases} Tua_d = Tua + Tua_s \\ TABU_d = TABU_s \end{cases} \tag{13}$$

In Formula (13), Tua represents the pheromone matrix of the original ant colony. Tua_d, Tua_s refers to the initial pheromone during occupancy prediction and the upper and lower bound optimization pheromone matrices after the call is completed, respectively. $TABU_d$ is the ACO global taboo table for obstacle avoidance. $TABU_s$ represents the ACO global taboo table after the end of the run. In summary, Fig. 6 shows a dynamic obstacle avoidance model that integrates A*-ACO.

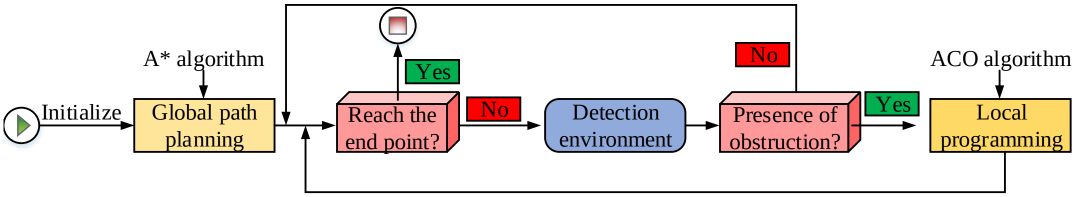


Fig. 6. Dynamic obstacle avoidance path planning process of A*-ACO algorithm.

In Fig. 6, the first step is to build a two-dimensional grid environment model, and then optimize the A* algorithm for static global PP. When conducting obstacle detection, if there are no obstacles, the global path is executed. If there are obstacles, occupancy prediction is made. Optimized ACO is used for local obstacle avoidance PP until reaching the endpoint.

IV. Results and Discussion

In the performance analysis and verification of the PP design algorithm, the study first analyzed the performance of the optimized ACO and the optimized A* algorithm to verify the effectiveness of their improvement strategies. Subsequently, the study applied it to practical simulations to compare the PP performance of various models under different vehicle driving conditions and environmental complexities.

A. Comparison and Analysis of A and ACO Performance Before and After Optimization*

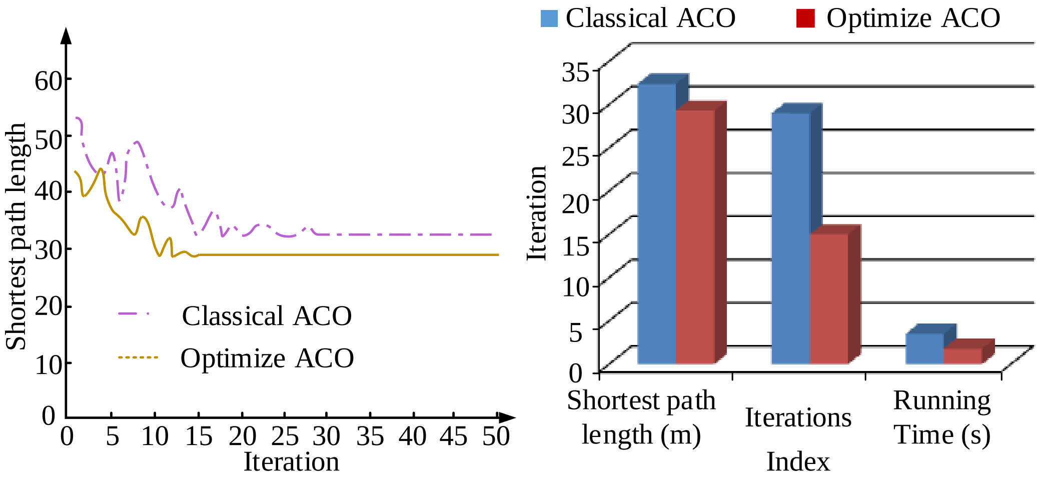
The study first focused on the optimized ACO and A* algorithms. Table I shows the experimental environment and parameter settings.

The study compared the convergence performance before and after ACO optimization in Fig. 7.

From Fig. 7, the optimized ACO showed a significant improvement in convergence performance, which was reflected in both convergence speed and final convergence value. The final convergence value of the optimized ACO, i.e. the output shortest path length, was 29.3 meters. The convergence value of the shortest path length in classical ACO was 32.4 meters, a relative increase of 9.57%. The convergence frequency of the optimized ACO was 15 times, which was a 48.28% decrease compared to the 29 times of the traditional ACO. The runtime of optimized ACO was only 1.8 seconds, while classical ACO took 3.5 seconds to complete the iteration. Therefore, in terms of runtime, this optimization algorithm had relatively decreased by 48.57%. In summary, the performance improvement of optimized ACO was mainly reflected in operational efficiency. In addition, there was also a certain improvement in the output path value. As the PP length increased, the gap between the two algorithms also became larger. The study then compared the performance of A* algorithm before and after optimization in Fig. 8.

TABLE I. Experimental Environment and Parameter Settings

Name	Settings
Operating system	ThinkPad E440 Ubuntu 16.04
GPU	GTX 2070 Super
Simulation platform	MATLAB
Search individual count	30.0
Pheromone heuristic factor α	1.0
Ideal heuristic factor β	7.0
Number of iterations threshold	50.0
Pheromone volatile factor ρ	0.7
Pheromone enhancement coefficient	1.0



(a) Model convergence graph

(b) Model convergence performance

Fig. 7. Comparison of ACO algorithm convergence performance before and after optimization.

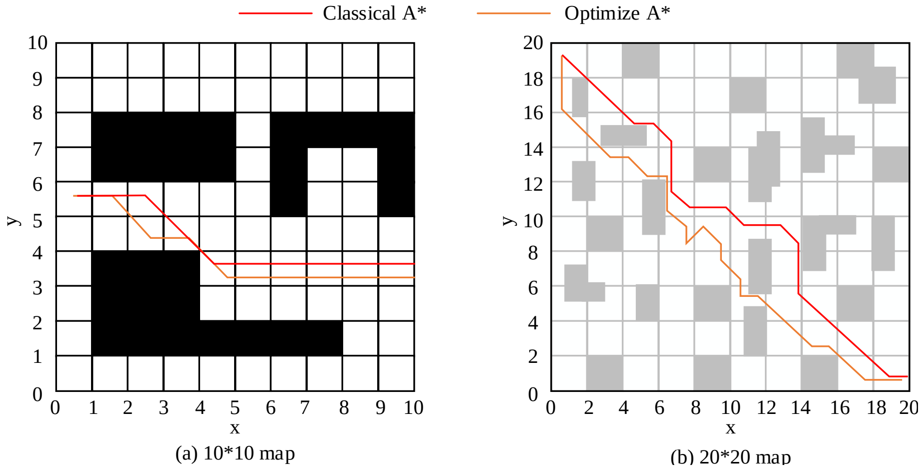


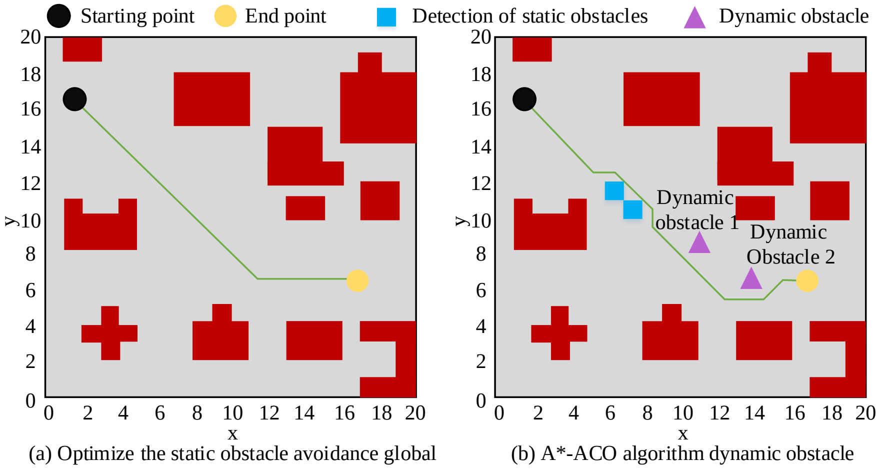
Fig. 8. Comparison of A* algorithm path planning before and after optimization.

The above obstacle avoidance PP experiments were all based on static obstacles. Fig. 8 (a) shows the PP results of the A* algorithm for each model in a 10*10 map before and after optimization. Due to the small size of the map and the concentrated distribution of obstacles, the PP results of each model were not significantly different. But overall, the optimized A* model had a smoother path with only two inflection points. The number of inflection points in traditional A* algorithms was twice that of optimization algorithms. This indicated that even in simple obstacle avoidance environments, the optimized A* algorithm exhibited better PP performance. In Fig. 8 (b), the map size had increased and the distribution of static obstacles was relatively scattered, resulting in smaller sizes. In complex obstacle avoidance scenarios, there was a more significant difference in the PP performance of the A* algorithm before and after optimization. The traditional A* algorithm had 17 inflection points in a 20*20 map, while the optimized A* algorithm had only 11 inflection points in a 20*20 map, a relative reduction of 35.29%. Therefore, the optimized A* algorithm produced smoother paths, shorter path distances, better adaptability in complex scenes, and could better achieve global PP.

B. Performance Comparison of Dynamic Obstacle Avoidance Path Planning Models Based on A-ACO*

A simulation obstacle avoidance environment was built on a 20*20 map and its PP process was simulated using A*-ACO in Fig. 9.

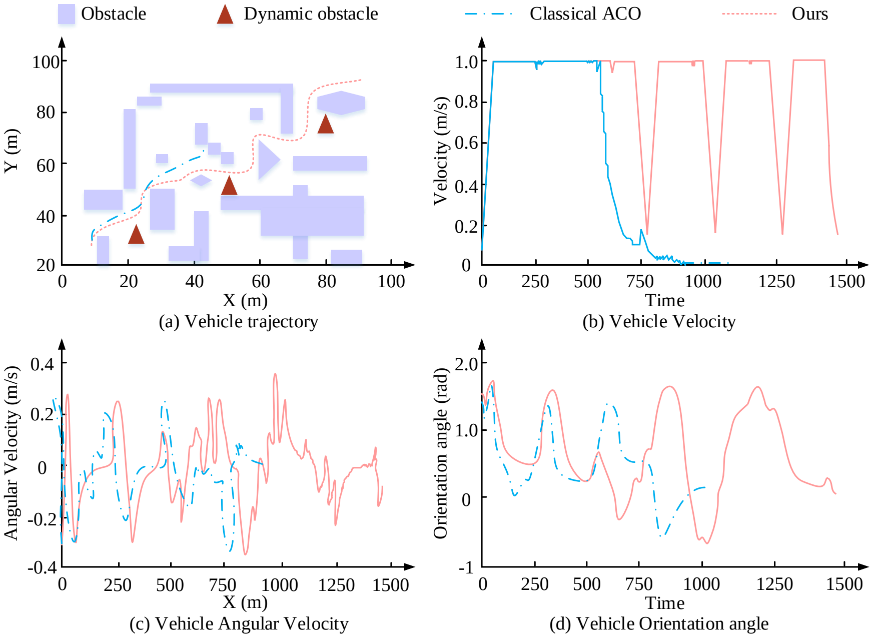
Fig. 9 (a) shows the initial global PP results of the improved A* algorithm. In this path, only static obstacles were considered. The optimized A* algorithm had a relatively smooth global planning path with fewer turning points, and overall smoothness, achieving good static obstacle avoidance PP. Then by detecting other obstacles and utilizing the improved ACO, dynamic obstacle avoidance local PP was achieved. Fig. 9 (b) shows the final dynamic obstacle avoidance PP result. It was completed even with the addition of static and dynamic obstacles. There were seven turning points. Compared to static global paths, the path length increased by 23.46%. In summary, the designed dynamic obstacle avoidance PP model based on A*-ACO could cope with the appearance of dynamic obstacles and output a relatively smooth driving planning path. The study continued to analyze the impact of optimization algorithms and classical ACO output paths on the operational performance of driving vehicles in Fig. 10.



(a) Optimize the static obstacle avoidance global planning path of A* algorithm

(b) A*-ACO algorithm dynamic obstacle avoidance path output

Fig. 9. Output result of A*-ACO algorithm dynamic obstacle avoidance process path.



(a) Vehicle trajectory

(b) Vehicle Velocity

(c) Vehicle Angular Velocity

(d) Vehicle Orientation angle

Fig. 10. Vehicle driving simulation results based on path planning model.

The simulated environment for the above experiment was 100m*100m. In Fig. 10 (a), the traditional ACO did not complete the PP, although it successfully avoided the first dynamic obstacle, it collided with the static obstacle at coordinates around (45, 70). Therefore, traditional ACO needed further optimization. The designed hybrid optimization algorithm successfully avoided various static and dynamic obstacles and reached the endpoint. In addition, its output path was smooth, with fewer turning points except for necessary obstacle avoidance turning points. Fig. 10 (b) shows the speed curves of vehicle paths for each model. The linear velocity of the design model always maintained a relatively regular periodic variation, with only small fluctuations of less than 0.1%. After the traditional ACO failed to avoid obstacles, the linear speed dropped sharply and eventually returned to zero. In Fig. 10 (c), the angular velocity of the traditional ACO also returned to zero, and the steering angle in Fig. 10 (d) showed the same change. The linear velocity, angular velocity, and steering angle of the designed algorithm always maintained similar fluctuations without changing the kinematic characteristics. Therefore, this proposed algorithm could better achieve dynamic PP of vehicles. In addition, the study also introduced the Improved Compressed Factor Particle Swarm Optimal Algorithm (ICFPSO) proposed by Li X et al. and the Modified Q-Learning Algorithm (MQL) proposed by X Wang et al. for comparison. Table II shows the experimental results.

In Table II, this design algorithm had the best overall performance. In a 25*25 map, MQL performed the best, with an average decrease of 2.5 search nodes compared to other algorithms. The search time decreased by 10.51%. However, as the complexity of the map increased, this design algorithm gradually demonstrated better PP performance. In a 25*25 map, the average search node decreased by 3.85%, the average search time decreased by 3.62%, and the average path length decreased by 18.20%. In a 100*100 map, the differences between these models were even greater. The average number of search nodes for this design algorithm decreased by 6, the average search time decreased by 4.11%, and the average path length decreased by 22.07%. In summary, this design method was more suitable for PP needs in complex scenarios and had the best overall performance. In order to further confirm the superiority of the research method, the research was compared with the advanced algorithms in the current field, as shown in Table III.

TABLE II. COMPARISON OF PATH PLANNING PERFORMANCE OF DIFFERENT MODELS IN DIFFERENT SCENARIOS

Index	**Environmental dimension (m)**	**Model**		
		Ours	***ICFPSO***	***MQL***
Search node mean	25*25	83	82	80
	50*50	234	242	235
	100*100	694	699	701
Search time mean (ms)	25*25	11.21	11.23	10.66
	50*50	88.24	90.37	92.50
	100*100	287.35	288.91	291.47
Mean path length (m)	25*25	28.03	28.55	27.64
	50*50	65.91	65.88	68.96
	100*100	120.74	130.64	134.83

TABLE III. COMPARISON BETWEEN RESEARCH METHODS AND ADVANCED ALGORITHMS

Method	**Research method**			**Li X et al. [5]**			**X Wang et al. [6]**		
Map scale	25*25	50*50	100*100	25*25	50*50	100*100	25*25	50*50	100*100
Average number of search nodes	83	234	694	103	277	793	108	286	834
Average search time (ms)	11.21	88.24	287.35	14.22	107.69	392.48	13.71	121.34	402.95
Average path length (m)	28.03	65.91	120.74	31.02	77.68	143.23	29.97	79.31	142.39

As can be seen from Table III, the research method has certain advantages in terms of the average number of search nodes, average search time and average path length. In order to ensure the accuracy and reliability of the results, the study conducted comprehensive verification. The results show that the convergence frequency of the proposed method is 48.28% higher than that of the traditional algorithm. The generated path has fewer turning points and higher smoothness, and the number of turning points is reduced by 35.29% compared to the traditional algorithm. By analyzing the data of 100 independent simulation runs, the research method is 30% lower in the standard deviation of path length and 25% lower in the standard deviation of search time than the traditional algorithm. The results verify the accuracy and importance of the research method, and clarify its research status and potential application value in the field of intelligent traffic management.

C. Discussion

The proposed path planning model based on improved ant colony optimization algorithm shows good obstacle avoidance path planning ability in simulation analysis. Specifically, compared with the traditional algorithm, the number of search nodes in this algorithm is reduced by 6, the average search time is reduced by 4.11%, and the average path length is reduced by 22.07%. These improvements are mainly due to the following aspects: (1) The convergence performance of the algorithm is effectively improved by introducing a backtracking strategy to optimize the tabu table of the algorithm. (2) Manhattan distance is used instead of the traditional Euclidean distance, which simplifies the calculation of the heuristic function and improves the search efficiency. (3) Combined with the A* algorithm, it achieves effective avoidance of dynamic obstacles and improves the adaptability and practicability of the algorithm in the actual traffic environment. X Wang et al. [6] used an improved Q learning method to translate the learning behavior into a discrete-time Markov chain model. Through the improvement of ant colony algorithm, this study strengthens the heuristic information utilization in path planning, improves the search efficiency and the smoothness of the path. The routing algorithm proposed by Huo L [9] takes into account the characteristics of frequent changes in urban traffic and rich driving paths. This study further improves the adaptability and obstacle avoidance effect in complex dynamic environment through the combination of dynamic obstacle model and A*-ACO algorithm. The method proposed by Lyridis DV et al. [12] performs well in dealing with the local obstacle avoidance problem of unmanned surface vessels. The method in this study also focuses on local obstacle avoidance, but through improved ant colony algorithm and dynamic obstacle avoidance strategy, higher obstacle avoidance accuracy and practicability are achieved. Compared with the improved compression factor particle swarm optimization method proposed by Li X et al. [5], the research method achieves a better balance among multiple objectives. The good performance of the research method in route planning is further explained, which can provide more technical support for traffic development in the future.

V. CONCLUSION

A PP method based on ACO was proposed to address the demand for dynamic obstacle avoidance in ITS. The improved ACO algorithm improves the convergence speed and stability by introducing backtracking strategy and dynamic obstacle avoidance optimization. Combined with A* algorithm, the proposed model can effectively deal with dynamic obstacles in actual traffic and realize real-time path adjustment. The algorithm achieves a good balance among multiple objectives such as search efficiency, path length and smoothness. These results confirmed that the optimized ACO reduced the shortest path length by 9.57% compared to traditional algorithms. The convergence frequency was 15 times, a decrease of 48.28% compared to before. In a 10*10 map, the number of inflection points in the traditional A* algorithm was twice that of the optimization algorithm. In a 20*20 map, the optimized A* algorithm had only 11 inflection points, a relative reduction of 35.29%. Subsequently, simulations were conducted on the practical application of A*-ACO. In a 20*20 map, the optimized A* algorithm's global planning path was relatively smooth and smooth, achieving good static obstacle avoidance PP. On the basis of adding static and dynamic obstacles, the path length increased by 23.46%. Then, on a map of 100m*100m, it was compared with traditional algorithms. These results confirmed that traditional algorithms had failed to complete the path, while the various indicators of this optimized algorithm always maintained similar fluctuation amplitudes and had not changed the kinematic characteristics. In comparison with other models, although the performance of this design algorithm was slightly lower in simple environments, in a 100*100 map, the average search time decreased by 4.11% and the average path length decreased by 22.07%. In summary, these design methods are more suitable for PP requirements in complex scenarios. In future studies, collaborative path planning in multi-vehicle environments will be studied, considering the interaction and collaboration between vehicles to improve the efficiency of the overall traffic flow. And establish a comprehensive evaluation framework to evaluate the performance of intelligent transportation systems in different scenarios, including environmental impact, economic benefits and social benefits.

ACKNOWLEDGMENT

The research is supported by: Scientific Research Program Funded by Shaanxi Provincial Education Department

(Provincial Department of Education Program No.22JK0047, Provincial Federation of Social Sciences Program No.2022HZ1131); Scientific Research Program Funded by the Xi'an Social Science Planning Fund (Program No. 24GL20).

REFERENCES

[1] RJ Godwin, DR White, ET Dickin, M Kaczorowska-Dolowy, WAJ Millington, EK Pope, et al.The effects of traffic management systems on the yield and economics of crops grown in deep, shallow and zero tilled sandy loam soil over eight years. Soil & Tillage Research, 2022, 223(1): 1-15. DOI:10.1016/j.still.2022.105465.

[2] Kallinen V, Mcfadyen A. Collision Risk Modeling and Analysis for Lateral Separation to Support Unmanned Traffic Management. Risk Analysis, 2021, 42(4): 854-881. DOI:10.1111/risa.13809.

[3] H Xu, CR Wang, A Berres, T Laclair, J Sanyal. Interactive Web Application for Traffic Simulation Data Management and Visualization: Transportation Research Record, 2022, 2676(1):274-292. DOI:10.1177/03611981211035760.

[4] H Xu, M Guo, N Nedjah, J Zhang, P Li.Vehicle and Pedestrian Detection Algorithm Based on Lightweight YOLOv3-Promote and Semi-Precision Acceleration. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(10): 19760-19771. DOI:10.1109/TITS.2021.3137253.

[5] Li X, Yu S. Three-dimensional path planning for AUVs in ocean currents environment based on an improved compression factor particle swarm optimization algorithm. Ocean Engineering, 2023, 280(7): 114610-114621. DOI:10.1016/j.oceaneng.2023.114610.

[6] X Wang, J Liu, C Nugent, I Cleland, Y Xu. Mobile agent path planning under uncertain environment using reinforcement learning and probabilistic model checking. Knowledge-based systems, 2023, 264(3): 110355-110365. DOI:10.1016/j.knosys.2023.110355.

[7] Y Huang, Y Li, Z Zhang, Q Sun. A novel path planning approach for AUV based on improved whale optimization algorithm using segment learning and adaptive operator selection. Ocean Engineering, 2023, 280(7): 114591-114605. DOI:10.1016/j.oceaneng.2023.114591.

[8] S Zhang, H Sang, F Liu, X Sun, Y Zhou, P Yu. A real-time local path planning algorithm for the wave glider based on time-stamped collision detection and improved artificial potential field. Ocean Engineering, 2023, 283(9): 115139-115162. DOI:10.1016/j.oceaneng.2023.115139.

[9] Huo L. The global path planning for vehicular communication using ant colony algorithm in emerging wireless cloud computing. Wireless Networks, 2023, 29(2): 833-842. DOI:10.1007/s11276-022-03152-0.

[10] Yang X, Han Q. Improved reinforcement learning for collision-free local path planning of dynamic obstacle. Ocean Engineering, 2023, 283(9): 115040-115053. DOI:10.1016/j.oceaneng.2023.115040.

[11] A Zou, L Wang, W Li, J Cai, H Wang, T Tan. Mobile robot path planning using improved mayfly optimization algorithm and dynamic window approach. Journal of supercomputing, 2023, 79(8): 8340-8367. DOI:10.1007/s11227-022-04998-z.

[12] Lyridis DV. An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. Ocean engineering, 2021, 241(12): 109890-109897. DOI:10.1016/j.oceaneng.2021.109890.

[13] W He, S Meng, J Wang, L Wang, R Pan, W Gao. Weaving scheduling based on an improved ant colony algorithm: Textile Research Journal, 2021, 91(5):543-554. DOI:10.1177/0040517520948896.

[14] Ntakolia C, Lyridis DV. A comparative study on Ant Colony Optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles. Ocean engineering, 2022, 255(7): 111418-111424. DOI:10.1016/j. oceaneng.2022.111418.

[15] Liu D, Hu X, Jiang Q. Design and optimization of logistics distribution route based on improved ant colony algorithm. Optik, 2023, 273(1):170405-170410. DOI:10.1016/j.ijleo.2022.170405.

[16] Li CY, Zhang TC, Bao HG, Xiao ZL, Zhou YH, Sun Z, et al. Accelerated multi-physics analysis of electromagnetic energy selective surfaces with a space mapping algorithm. Engineering analysis with boundary elements, 2023:155(1): 140-147. DOI:10.1016/j.enganabound. 2023.05.044.

[17] WJ Liu, HF Ding, MF Ge, XY Yao. Cooperative control for platoon generation of vehicle-to-vehicle networks: a hierarchical nonlinear MPC algorithm. Nonlinear dynamics, 2022, 108(4): 3561-3578. DOI:10.1007/s11071-022-07400-y.

[18] Williams A. Human-Centric Functional Computing as an Approach to Human-Like Computation. Artificial Intelligence and Applications. 2023, 1(2): 118-137. DOI:10.47852/bonviewaia2202331.