Journal of Intelligent and Connected Vehicles
2023, **6**(2): 102−112
https://doi.org/10.26599/JICV.2023.9210011

Research Article

Two-level vehicle path planning model for multi-warehouse robots with conflict solution strategies and improved ACO

Pan Wu[1], Lingshu Zhong[2,✉], Jingwen Xiong[3], Yuhao Zeng[3], Mingyang Pei[3]

[1]*College of Traffic & Transportation, Chongqing Jiaotong University, Chongqing 400074, China*
[2]*School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China*
[3]*Department of Civil and Transportation Engineering, South China University of Technology, Guangzhou 510641, China*

Received: April 12, 2023; Revised: May 13, 2023; Accepted: May 29, 2023

ABSTRACT: With the rapid development of warehouse robots in logistics and other industries, research on their path planning has become increasingly important. Based on the analysis of various conflicts that occur when the warehouse robot travels, this study proposes a two-level vehicle path planning model for multi-warehouse robots, which integrates static and dynamic planning to improve operational efficiency and reduce operating costs. In the static phase, the blockage factor is introduced to enhance the ant colony optimization (ACO) algorithm as a negative feedback mechanism to effectively avoid the blockage nodes during movement. In the dynamic stage, a dynamic priority mechanism is designed to adjust the routing strategy in real time and give the optimal path according to the real situation. To evaluate the model's effectiveness, simulations were performed under different operating environments and application strategies based on an actual grid environment map. The simulation results confirm that the proposed model outperforms other methods in terms of average running distance, number of blocked nodes, percentage of replanned paths, and average running time, showing great potential in optimizing warehouse operations.

KEYWORDS: warehouse robot, path planning, two-level model, conflict strategy, ant colony optimization

1 Introduction

With the rise of E-commerce industries, warehouse robots have gradually replaced manual labor as operational workers in the logistics environment to ensure timely responses to warehousing operations in massive cargo scenarios (Wang et al., 2022). Warehouse robots streamline operations by equipping their bodies with sensors and motion control systems to automate processes. It can significantly improve the efficiency of object transportation and sorting, reduce labor costs, and improve the space utilization of the warehouse environment. Leading companies in the E-commerce industry, such as Taobao and Jingdong, have independently developed a variety of autonomous guided vehicles (AGVs) and robotic arms to handle large numbers of orders efficiently (Huang et al., 2023; Lim et al., 2021; Yue et al., 2022; Zhang and Yang, 2021; Zhu et al., 2022). Amazon, known for its extensive use of automation in warehouses, has also developed a range of robots and drones to support its operations (Chen et al., 2021). Overall, the development of warehouse robots is part of a more significant trend toward automation in the logistics industry. As robots become increasingly autonomous and dependable, they possess the potential to revolutionize the storage, transportation, and delivery of products, resulting in faster and more efficient logistics operations.

Multi-warehouse robotic systems distribute control of individual warehouse robots to operate independently, reducing coupling within the system and avoiding overall paralysis (Li et al., 2023a, 2023b; Lian et al., 2022; Yin et al., 2023). This approach can increase operational flexibility and fault tolerance. However, it also creates the possibility of conflicts between robots as different robots have different goals and priorities. Therefore, path planning is a core problem for warehouse robot applications, which studies how to allow multiple robots to move from a standby position to a task point in an optimal path without collision for object pickup, delivery, and classification (Lv et al., 2022; Qin et al., 2023; Sun et al., 2023; Yan, 2022; Yuan et al., 2022). In practical problems, the path planning solution strategy first models the physical environment in which the problem is located and then searches for the optimal path based on different algorithms (Li et al., 2023a). As mobile robots are increasingly used in transportation, logistics, and manufacturing industries, there are many research results on algorithms for path planning (Li et al., 2022a). According to the planning operation area, the current path planning models for multi-warehouse robot systems can be divided into global path planning models (Daddi et al., 2022; Li et al., 2022b; Psotka et al., 2023) and local path planning models (Kobayashi and Motoi, 2022; Peralta et al., 2020). The global path planning models concern offline planning and are adequate for predictable tasks that do not change frequently. However, they are unsuitable for tasks requiring real-time adaptability and flexibility. On the other hand, local path planning models are essentially dynamic online planning, where the robot makes decisions in real time based on the surrounding environment and available information. This approach is suitable for more complex tasks that require real-time decisions but can be challenging to ensure that the robots do not collide with each other or with other obstacles.

According to the existing literature, algorithms used in robot

✉ Corresponding author.
E-mail: z.lingshu@gmail.com

path planning can be categorized into three main approaches: classical, metaheuristic, and hybrid (Gul et al., 2022). For comparison and analysis, these path planning algorithms are organized in Table 1. Classical methods include artificial potential field algorithms (Chen et al., 2022; Hao et al., 2022; Ma et al., 2022; Souza et al., 2022; Zhao et al., 2023), graph search algorithms (Jin et al., 2023; Li et al., 2022c), and sampling algorithms (Dian et al., 2022; Ding et al., 2023; Ma et al., 2023). With the development of computational techniques, metaheuristic optimization algorithms have emerged as an efficient path planning method for high-dimensional complex problems based on genetic evolution mechanisms and cluster foraging migration mechanisms theories. Common metaheuristic optimization algorithms include evolutionary algorithm (EA), particle swarm optimization (PSO), artificial bee colony algorithm (ABC), cuckoo search (CS), gray wolf optimization (GWO), etc. (Cui et al., 2023; Larsen and Kim, 2021; Rajamoorthy et al., 2022; Shao et al., 2020; Song et al., 2021; Tang et al., 2021; Xu et al., 2022). In addition to the models based on a single algorithm and a single objective function, studies on hybrid applications for multi-objective function utilities also exist (Sahu et al., 2023).

As a cluster-based algorithm, ant colony optimization (ACO) simulates the pheromone mechanism in the foraging process of ant colonies for path planning (Al-Amyal et al., 2023). Ants release a specific concentration of pheromones while crawling, and other ants search for optimal paths through indirect communication by sniffing the pheromones. Although traditional ACO has positive feedback and self-organization characteristics, they have limitations such as possible stagnation of convergence and easily falling into local optima (Dai et al., 2022; Qu et al., 2022; Tao et al., 2021; Wang and Wu, 2023; Yang et al., 2022). Therefore, various variants have been proposed to improve this algorithm's effectiveness and convergence speed. For instance, Wu et al. (2023) proposed a modified adaptive ant colony optimization algorithm (MAACO), which introduced a new heuristic mechanism with directional information and added directional guidance in the iterative process to improve the convergence speed of the algorithm further. To address the problem that the ACO does not fully utilize the historical paths explored by ants, Hou et al. (2022) suggested a new ant communication mechanism to accelerate the integration of historical paths and the exploration of better paths through direct communication between individual ants. Miao et al. (2021) also introduced the angle guidance and obstacle exclusion factors in the traditional algorithm. It also added an adaptive adjustment factor and volatile factor in the pheromone to balance the convergence and global search ability of ACO.

Previous research on the path planning problem of warehouse robots has limited exploration and analysis of the conflict types and corresponding strategies among robots. Moreover, the existing models for path planning typically rely on either a single global module or a single local module. In practical applications of a warehouse environment, the traditional ACO heuristic function only considers the path distance factor, which can result in blockage at the path intersection nodes, thereby diminishing the efficiency of operations. Additionally, ACO that employs only positive feedback information is prone to local minimum. Hence, this study incorporates the blockage factor as negative feedback information into the heuristic function to enhance the global performance of the algorithm. This work proposes a multi-warehouse robot two-level path planning model based on the analysis of various conflict accidents that occur when the warehouse robot travels. The model integrates static planning with dynamic planning and implements a dynamic prioritization mechanism to improve the operational efficiency of warehouse robots and reduce operating costs. The contributions of this study are as follows:

- This study proposes a two-level vehicle path planning model, including static and dynamic stages. In the static path planning stage, an improved ACO with negative feedback of blockage factor is introduced, realizing a parallel mechanism of positive and negative feedback. In the dynamic planning stage, the priority of the warehouse robot is adjusted in real time, and the conflict resolution strategy is selected dynamically.
- Based on a grid map of the real environment in the MATLAB platform, the model performance is simulated under various conditions, including different combination strategies, robot scales, and scenarios with and without conflicts. The superiority of the path planning performance of the proposed model is verified as compared to other existing models.

Table 1 Comparison of various methods

Ref.	Method	Advantages & limitations
Souza et al. (2022); Zhao et al. (2023)	Artificial potential field algorithms	Providing obstacle avoidance, as well as eliminating local minimum problems and oscillations in the influence threshold of repulsive fields
Jin et al. (2023); Li et al. (2022c)	Graph search algorithms	Improving the search efficiency, effectively reducing the turning angle
Dian et al. (2022); Ding et al. (2023)	Sampling algorithms	Being able to acquire the optimal path on the grid map; poor computational efficiency, falling into a local minimum
Larsen and Kim (2021)	Evolutionary algorithm	Have revolutionized the field of optimization algorithms, effective path planning approaches for high-dimensional complex conditions
Shao et al. (2020); Song et al. (2021)	Particle swarm optimization	Updating each particle's position and velocity; usually results in premature convergence
Cui et al. (2023)	Artificial bee colony algorithm	Outstanding exploration ability; poor exploitation capability, and low solution precision
Sahu et al. (2023)	Cuckoo search	Performing well for several problems such as nonlinear, multimodal, unconstrained, and constrained; suffering from premature convergence
Wang and Wu (2023); Yang et al. (2022)	ACO	Positive feedback and self-organization characteristics; possible convergence stagnation and tendency to fall into local optima
Wu et al. (2023)	Modified adaptive ACO	Improving the convergence speed of the algorithm
This study	Improved ACO	Enhancing the global performance of the algorithm, improving the operational efficiency of warehouse robots, and reducing operating costs

2 Conflict and resolution strategy

In the operation of multi-warehouse robots, conflicts between robots can occasionally arise, resulting in temporal or spatial conflicts that can have significant negative impacts on environmental resources, system stability, efficiency, and organization. Therefore, it is crucial to identify and implement effective measures to mitigate these conflicts and optimize operational efficiency. A prerequisite for achieving this goal is the classification and categorization of common types of disputes that may arise within the environment, which can then inform targeted conflict resolution strategies tailored to address specific conflicts. This approach ensures comprehensive and effective resolution of conflicts, as opposed to applying generic strategies to all conflicts. By employing targeted and differentiated conflict resolution strategies, greater precision and efficiency can be achieved in operating multi-warehouse robots.

2.1 Typical conflict types

During the driving process of a warehouse robot, the use of body-mounted detection devices enables it to continuously monitor the area around its body in real time for any potential collisions with other warehouse robots. Depending on the state of the conflict when it occurs, it can be roughly divided into the following types. A graphical representation of the various types of conflicts between the two robots, R_1 and R_2, and the corresponding time windows are shown in Fig. 1; t represents time, s.

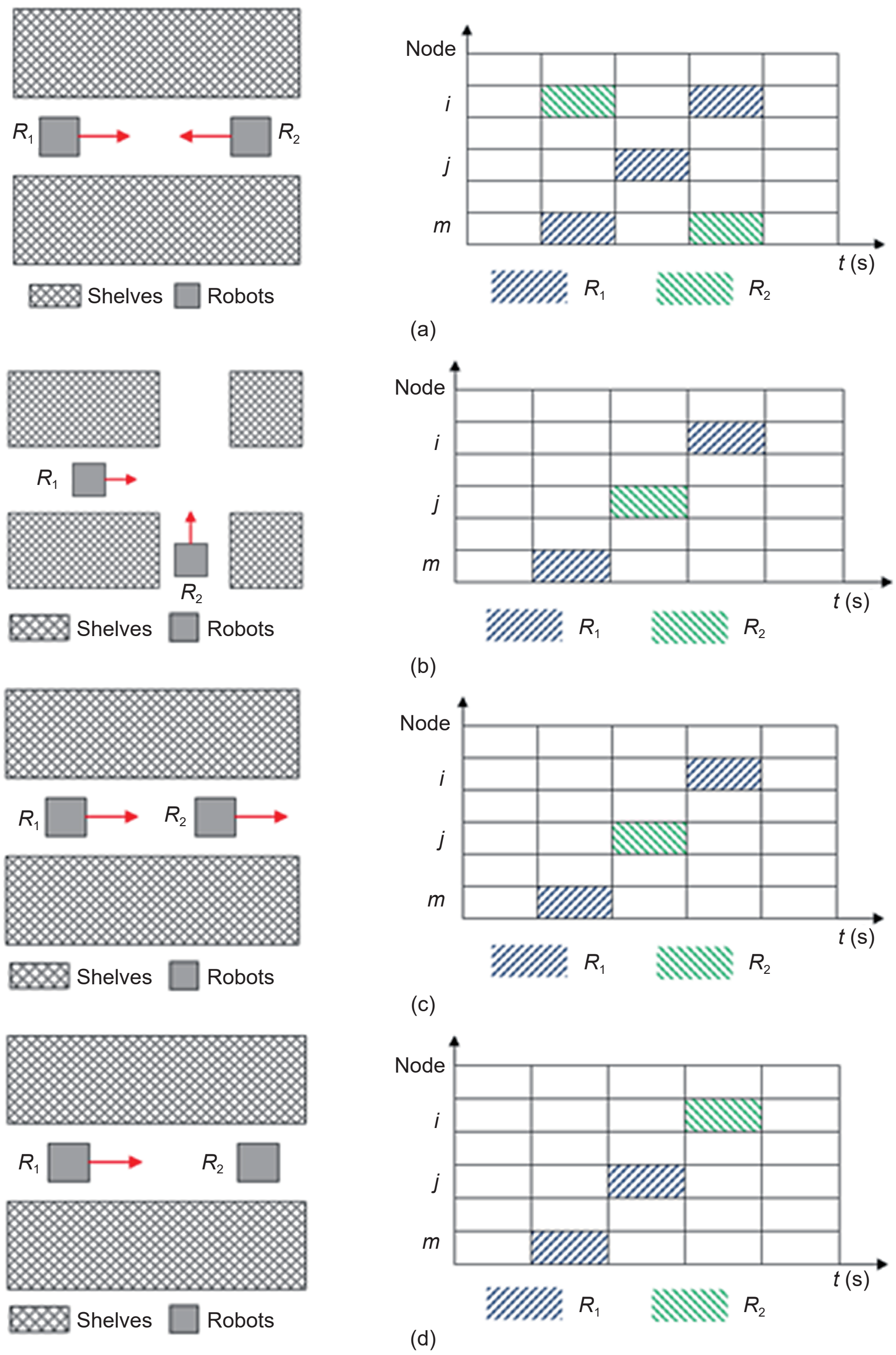

Fig. 1 Typical conflict types of the two robots R_1 and R_2, and the corresponding time windows. (a) Head-on conflict. (b) Crossing conflict. (c) Catch-up conflict. (d) Occupancy conflict. i, j, and m indicate the nodes.

● Head-on conflict

A head-on conflict is a frontal collision between two warehouse robots traveling in opposite directions on a single-lane road. The road in this study is set to be a single lane in both directions, so it is prone to such conflicts. Such conflicts can lead to road deadlock and require external intervention to resolve.

● Crossing conflict

This conflict occurs at the node where two lanes intersect. It happens when two warehouse robots travel from different directions and compete for space at an intersection, leading to side collisions between warehouse robots and blockage of the intersection.

● Catch-up conflict

Catch-up conflict arises when a following robot tries to overtake a slower front robot with a failure, deceleration, or load. Catch-up and crossing conflicts share similar time windows since both involve multiple robots occupying the same node.

● Occupancy conflict

An occupancy conflict is caused when a robot is blocked by another robot stopping at a node due to operational needs., blocking its normal travel route.

● Other conflicts

Others mainly include warehouse robot failures and human factors. Their failures include power shortages, communication failure, parts damage, etc. The human factor mainly refers to the situation in which people in the environment carrying out maintenance work that will affect the regular work of warehouse robots.

2.2 Conflict resolution strategies

In the previous section, several typical types of conflicts in the warehouse environment are introduced. Many conflict elimination strategies can be implemented to reduce the impact of conflicts on warehouse robots, and the main strategies adopted in this study include the waiting strategy, rerouting strategy, and dynamic prioritization strategy.

● Waiting strategy

The waiting strategy is a simple and easy-to-implement strategy. When a robot detects through its sensors that there may be a collision with another robot ahead, it stops and waits to avoid conflict until other robots pass. However, this strategy consumes more time resources, and waiting for too long may lead to occupancy conflict. Therefore, it is necessary to use this strategy according to the actual situation.

● Rerouting strategy

Rerouting strategy means replanning the travel route of a robot. After the robot detects a possible conflict, in addition to the wait strategy, it can first lock the conflict point and then replan a driving route to the target point. This strategy is also easy to implement, but it may still cause conflicts with other robots if there are many robots in the operating environment. Therefore, a conflict-free path needs to be planned in conjunction with the actual situation.

● Dynamic prioritization

When the warehouse robot in the system encounters a conflict in driving, the robot with lower priority should make avoidance. However, only a single waiting strategy may result in a lot of wasted time, thus creating occupancy conflicts. In addition, it is supposed to consider the distance driven by the robot during the operation in real time. Warehouse robots should be given a higher priority when they are far from the target point. Then the robot can be dispatched as soon as possible, shortening the waiting time, thus saving time cost. Therefore, the dynamic priority strategy in this study mainly considers the waiting time of the robot and the distance from the target point.

Since the planned routes and time windows of each warehouse robot in the system before the operation are known, the dynamic priority calculation expression for the warehouse robot a running to the node i is

$$T_{ai} = \frac{t_{ai}}{t_a} + \frac{s_{ai}}{s_a} \tag{1}$$

where T_{ai} is the priority level of the robot a at the node i; t_{ai} stands for the time used by the robot to drive to the node i and t_a is the total time taken by the robot to perform this task; s_{ai} represents the distance driven by the robot to node i; and s_a is the total distance traveled to perform this task.

Combining the conflict types and the three strategies mentioned above to eliminate conflicts, this study considers three cases: head-on conflict, crossing conflict, and occupancy conflict. By amalgamating the conflict types and the aforementioned strategies, the study seeks to eliminate conflicts that may arise from autonomous robot navigation. It is pertinent to note that this study assumes the uniform speed of robots and does not account for external factor interference. Consequently, conflicts such as catch-up conflicts and others are not within the purview of this research.

3 Methodology

The two-level planning model is grounded on a key principle, wherein the initial path for each warehouse robot is planned in the static planning phase, followed by the real-time adjustment of the path in the dynamic planning phase in response to the changes in the surrounding environment. This approach amalgamates the static and dynamic planning phases to enable effective and real-time robot path planning.

Based on the real grid environment map, the optimal path for each robot is developed in the static path planning stage using an improved ACO that considers lane blockage. However, the paths generated during this phase only consider the fixed information of the environment and individual robot. In the actual operating environment, conflicts may arise between multiple warehouse robots, and the paths planned during the global static planning phase may not remain optimal under the current circumstances. Hence, the paths planned during static planning must be dynamically adjusted based on operating conditions. The dynamic planning phase first ascertains if any conflict may arise between warehouse robots, following which appropriate response strategies are developed for different conflicts. In this study, conflicts between robots are resolved by adjusting the priority strategy of warehouse robots in real time during the dynamic planning phase. Subsequently, the next step for the warehouse robot is chosen based on whether to wait or replan the route, thereby ensuring conflict-free navigation.

3.1 Static path planning based on improved ACO

3.1.1 *Direction of algorithm improvement*

Considering the actual situation of the warehouse environment and multi-warehouse robots, we improve the ACO to make it better adapt to the actual situation of multi-warehouse robot research. The current path planning algorithm has been instrumental in increasing operational efficiency in intelligent warehouses. However, with the growing scale of warehouse robots, blockages at hotspots and nodes in the environment are more likely to occur. As the number of warehouse robots increases, the likelihood of blockage in the system rises, leading to the formation of more blocked sections and nodes. Fig. 2 illustrates the blockage diagram for 60 warehouse robots, with congestion deepening as the blue color intensifies. It can be found that the path intersection node is generally more susceptible to blockage. Failure to account for blockage factors in path planning can result in increasingly severe blockages, significantly reducing overall warehouse efficiency. Thus, it is essential to incorporate the blockage factor in the path planning of multi-warehouse robots to overcome potential blockages and enhance the operational efficiency.

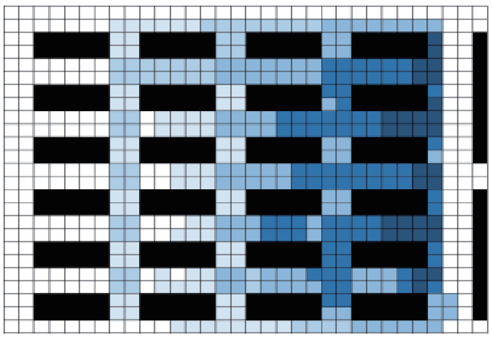

Fig. 2 Blockage diagram of 60 warehouse robots.

The traditional ACO is applied in the warehouse environment, and only the path distance factor is concerned in the heuristic function without considering other aspects. Hence it will generate blockage at the path intersection nodes during the path planning process of multi-warehouse robots, which dramatically reduces the efficiency of warehouse operations. Furthermore, the traditional ACO has positive feedback characteristics but lacks negative feedback information, which makes the algorithm optimization easily fall into local minimum. In this section, based on the actual warehouse environment, the blockage factor is added to the heuristic function as negative feedback information to improve the global nature of the algorithm. The specific improvements are as Eqs. (2)–(5):

$$p_{i,j}^{n}(t)=\begin{cases}\dfrac{[\tau_{i,j}(t)]^{\alpha}\left[\eta_{i,j}(t)\right]^{\beta}\left[\phi_{i,j}(t)\right]^{\gamma}}{\sum\limits_{s\subset \text{allowed}_i}[\tau_{i,s}(t)]^{\alpha}\left[\eta_{i,s}(t)\right]^{\beta}\left[\phi_{i,j}(t)\right]^{\gamma}}, & j\in \text{allowed}_i\\ 0, & j\notin \text{allowed}_i\end{cases} \tag{2}$$

$$\varphi_{i,j}(t)=1-\theta_j(t) \tag{3}$$

$$\theta_j(t)=\frac{r+r'}{r+r'+R} \tag{4}$$

$$\Delta\tau_{i,j}^{n}(t)=\begin{cases}Q/L_n+P\times\phi_{i,j},\ \{i,j\}\subset\text{visited}_{t,q}^{n}\\ 0, \qquad \text{others}\end{cases} \tag{5}$$

where $\varphi_{i,j}(t)$ represents the probability that the grid i to j will not produce blockage at the moment t; $\theta_j(t)$ denotes the blockage probability of grid j at the time t; r is the number of time node j that needs to be replanned; R is the number of time through grid j; r' stands for the number of times it is blocked again; and P represents the blockage factor.

3.1.2 Example analysis

In this study, the improved ACO is employed to incorporate the blockage factor as negative feedback information in the heuristic function of the original algorithm. To verify the performance of the improved ACO in avoiding the blockage area, simulation experiments are conducted based on the same grid map using the traditional ACO and the improved ACO of this study. The distribution of robot blockage in the hypothetical environment is shown in Fig. 3, where the green area represents the starting point with coordinates of (0.5, 5.5), the red area is the target point with coordinates of (11.5, 21.5), and the grid particle size is 1. The solid red line depicts the planning result of the traditional ACO, while the green dashed line is the planning result of the improved ACO.

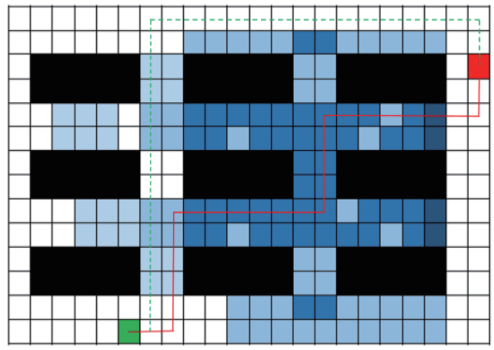

Fig. 3 Blockage verification simulation results.

According to Fig. 3, it can be seen that the improved ACO in this study is slightly longer than the traditional ACO in terms of path length. It is primarily due to the fact that the improved ACO sacrifices some path length to avoid the blockage area selection, but at the same time, greatly reduces the probability of blockage and prevents the robot from entering the highly congested area so as to improve the operation efficiency. The effectiveness of the improved ACO for avoiding blocked areas is verified.

3.2 Dynamic path planning based on real-time adjustment strategy

The time window model can accurately calculate the travel time of each warehouse robot in the system at road sections and nodes during the offline phase of the task, allowing for a reasonable allocation of resources. However, due to the constantly changing real operating environment, warehouse robots must receive new instructions during their journeys. As the scale of the operation increases, numerous uncertain situations may occur, including frontal and side collisions between robots in the lane, equipment failure, lane congestion, and other problems. In order to address these issues, it is essential to adjust warehouse robots in real time based on their current operation.

This study proposes a real-time adjustment strategy to enable online planning, which involves adjusting the priority of warehouse robots and implementing the conflict elimination strategies in real time. By employing this approach, the system can effectively address uncertainties and minimize delays, ensuring smooth and efficient operation.

3.2.1 Warehouse robot priority setting

Given that the requirements of each warehouse system can vary, the approach to priority setting may differ depending on the context. Common techniques for assigning priority include setting priorities based on the property of operation or the urgency level of the task at hand. In this study, the priority levels were determined based on the operational status of the warehouse robot and the timeframe for cargo transportation, tailored to the needs of warehousing and logistics enterprises. The priority levels are assigned a numeric value starting from 0, with higher priority levels corresponding to larger numerical values. A detailed description of the priority level settings is presented in Table 2.

Table 2 Initial priority setting

Operation status	Priority level
Failure/need for charging	Highest
Picking	High
Delivering	Low

3.2.2 Real-time adjustment strategy

The main conflict resolution strategies among robots in a warehouse system are waiting for strategies, rerouting strategies, dynamic prioritization, and rate adjustment. Traditional solutions generally use only one of these strategies, which may lead to undesirable situations such as long waiting time, lane congestion, and area deadlocks, resulting in reduced system operational efficiency. In this study, we integrate the dynamic priority, waiting for strategy, and rerouting strategy to plan the robot system. The approach comprises the following steps:

① Utilize the improved ACO algorithm to perform static planning and derive an optimal path for the current task. Compute the corresponding time windows for the paths and nodes to obtain the initial set of paths.

② Select the first path in the set of paths and calculate its corresponding time windows.

③ Check for the time window conflicts between the planned path and the selected path. If a conflict exists, choose the next path until a conflict-free path is identified. If no conflict-free path is available, determine the type of conflict.

④ According to the judged conflict type, choose different collision avoidance strategies from the waiting and rerouting strategies, combined with the real-time priority strategy simultaneously.

⑤ Based on the conflict types in the previous step, calculate and compare the time spent on different strategies. Choose the strategy that requires less time to resolve the conflict between robots, and finally furnish a conflict-free path.

4 Simulation results and discussions

4.1 Process of path planning algorithm

The process of the two-level planning model for the warehouse robot in this study is as follows, and the flow chart of the two-level planning model is shown in Fig. 4.

① Environment setting. According to the actual warehouse environment, establish the grid map, set task priorities in the system, and assign initial priority to the warehouse robot.

② Task delegating. Check whether there is a free warehouse robot in the environment, and if so, delegate the task with the highest priority to the warehouse robot closest to the starting point. Otherwise, wait for a free robot to appear in the environment.

③ Initial path planning. Utilize the improved ACO to plan the optimal driving route for the warehouse robots that have accepted the task to ensure the shortest total system running time.

④ Time window calculating and updating. The time windows on the planned route for the warehouse robot are calculated and updated.

⑤ Conflict detecting and resoluting. Determine whether there is a time window conflict for the warehouse robots, and if there is, determine the type of conflict and apply the conflict elimination strategy to the robots with low priority. Otherwise, go to step ⑦.

⑥ Real-time priority calculating. The real-time priority of the warehouse robot at the current node is calculated. Low-priority robots choose the least time-consuming conflict avoidance strategy between wait and rerouting strategies, and the system returns to step ④.

⑦ Task completion checking. The system checks if all tasks in the system are completed. If so, the job ends; otherwise, the system returns to step ②.

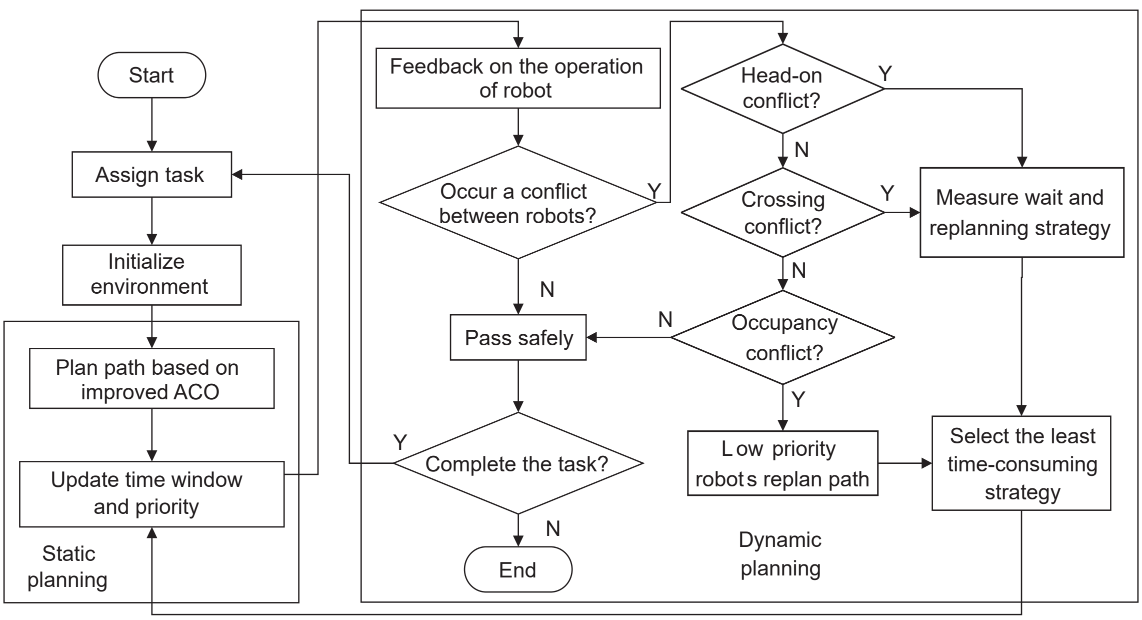

Fig. 4 Flow chart of the two-level planning model.

In the previous work, a mathematical path planning model was established for a multi-warehouse robot, and a two-level planning model for a multi-warehouse robot was designed. In this section, based on a real model of the warehouse operation environment, we compared the proposed path planning model and other models, and then experimented on the MATLAB simulation platform in terms of different task scales. At last, analyze the simulation results so as to test the effectiveness and feasibility of the proposed model.

4.2 Simulation conditions

MATLAB is used as a simulation platform for multi-warehouse robot path planning here. MATLAB software, as an advanced development software, is widely used in computational research, data analysis visualization, and statistical analysis. As a computer language, MATLAB can adapt well to the multi-warehouse robot simulation environment. It has powerful data processing capabilities and can visualize the calculation results. Other hardware conditions are Windows 10 64-bit operating system and 2.4 GHz AMD-A10 CPU processor with 8 GB of RAM. To save running time during the simulation, the model in this study omits the smoothing operation. Simulation-related parameters are set as in Tables 3 and 4.

4.3 Simulation example analysis

4.3.1 *Conflict-free scenario simulation analysis*

To verify the performance of the improved ACO, implement simulation with the different number of warehouse robots. The starting and ending points of the ten robots are shown in Table 5. There is no conflict between the paths, and the warehouse simulation environment is consistent with the previous section. The simulation results of the traditional and improved ACO at different scales are shown in Tables 6 and 7.

Table 3 Environment and robot parameters settings

Parameter	Indicator	Value
Goods shelve number	G_N	21
Charging pile number	P_N	8
Picking table number	PT_N	2
Node interval	N_T	5 m
Robot length	R_L	0.5 m
Safe distance	S_D	2.5 m
Uniform driving speed	U_S	1 m/s
Steering speed	E_T	0.5 m/s
Starting speed	S_S	0 m/s
Braking speed	B_S	0.7 m/s
Braking deceleration time	BD_T	0.5 m/s
Buffer time	B_T	0.4 s
Picking table dwell time	D_T	60 s

According to the results of Tables 6 and 7, the average running distance and average running time of a single warehouse robot applying the ACO before and after the improvement are plotted in Fig. 5. As shown in Table 7 and Fig. 5, the total running distance and total running time of warehouse robots in the system increase to different degrees as the scale of robot number increases. In a conflict-free scenario, compared with the traditional ACO, the average distance traveled by a single robot is reduced by 2.36 and 2.66 m for 5 and 10 robots, and the average running time is reduced by 9.06 and 10.66 s, respectively. Therefore, the improved ACO proposed in this study outperforms the traditional algorithm in terms of average running distance and movement time of a single storage robot, which reflects the effectiveness and advancement of the improved algorithm in a conflict-free environment.

Table 4 Algorithm parameter settings

Parameter	Indicator	Value	Parameter	Indicator	Value
Maximum number of robots	N_{cmax}	100	Correction coefficient	λ	5
Number of ants	K	50	A constant	u	20
Pheromone volatility coefficient	ρ	0.3	Heuristic function	η	0.5
Pheromone intensity	Q	100	Weighting of factors	x, y	1
Correction coefficient	μ	1	Weighting of factors	z	0
Correction coefficient	ω	10	Pheromone concentration constant	C	20
Height correction factor	λ, σ	0	Advantage ant adjustment coefficient	ε	20
Maximum pheromone concentration on the path	τ_{max}	40	Minimum value of total pheromones	τ_{min}	10
Pheromone expectation factor	α	1	Heuristic expectation factor	β	8
Clogging factor	γ	2.5			

Table 5 Warehouse robot task settings

Sequence	Priority	Starting point	Target point
1	1	(8.5, 0.5)	(11.5, 7.5)
2	2	(15.5, 0.5)	(16.5, 7.5)
3	3	(21.5, 0.5)	(24.5, 9.5)
4	4	(29.5, 0.5)	(30.5, 9.5)
5	5	(35.5, 7.5)	(34.5, 0.5)
6	6	(4.5, 39.5)	(5.5, 34.5)
7	7	(8.5, 39.5)	(10.5, 34.5)
8	8	(16.5, 32.5)	(19.5, 39.5)
9	9	(30.5, 39.5)	(20.5, 32.5)
10	10	(35.5, 34.5)	(36.5, 39.5)

Table 6 Simulation results of traditional ACO

Scale	Total running distance (m)	Total running time (s)	Average distance traveled by a single warehouse robot (m)	Average running time of a single warehouse robot (s)
5	493.21	543.45	98.64	108.69
10	1,012.92	1,126.41	101.29	112.64

Table 7 Simulation results of improved ACO

Scale	Total running distance (m)	Total running time (s)	Average distance traveled by a single warehouse robot (m)	Average running time of a single warehouse robot (s)
5	481.41	498.15	96.28	99.63
10	986.32	1,019.83	98.63	101.98

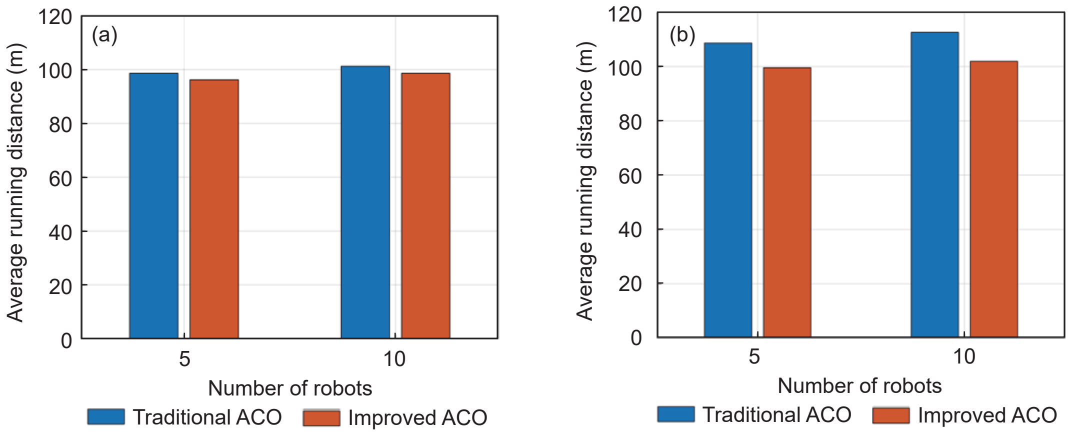

Fig. 5 Average running distance and average running time of a single warehouse robot applying traditional ACO and improved ACO. (a) Comparison of average running distance. (b) Comparison of average running time.

4.3.2 Simulation analysis with conflict scenarios

For the different number of warehouse robots (the maximum scale is 80) under the same experimental environment, utilize the following three algorithms for 30 simulations each and take their average values for index comparison. The information related to each warehouse robot is set as shown in Table 8.

Table 8 Algorithm parameter settings

Sequence	Priority	Starting point	Target point
1	1	(8.5, 0.5)	(15.5, 12.5)
2	2	(15.5, 0.5)	(24.5, 7.5)
3	3	(21.5, 0.5)	(30.5, 9.5)
…	…	…	…
79	79	(29.5, 37.5)	(18.5, 29.5)
80	80	(35.5, 34.5)	(36.5, 19.5)

- Traditional ACO + fixed priority waiting strategy.
- Traditional ACO + fixed priority rerouting strategy.
- The algorithm of this study (improved ACO + real-time adjustment strategy).

1) Comparison of blocked nodes number

A comparison of blocked node number for each algorithm at different warehouse robot sizes is shown in Fig. 6. It can be seen that as the number of robots in the environment increases, the number of blocked nodes generated by each algorithm also increases. The red line in Fig. 6 represents the growing trend of the blockage nodes generated by the traditional ACO combined with the fixed priority waiting strategy, and it is evident that it generates more blockage nodes than the other two algorithms. The algorithm utilized in this study is more effective than the traditional ACO combined with the fixed-priority rerouting strategy in reducing blockage, and the advantage is more apparent when there are more robots. The main reason is that the improved ACO is used for static planning, which considers the blockage factor, thus significantly reducing the probability of blockage in the environment at the beginning. The traditional ACO combined with the fixed priority rerouting strategy is forced to choose to replan the path after encountering blockage, which only reduces the probability of secondary blockage for the warehouse robots.

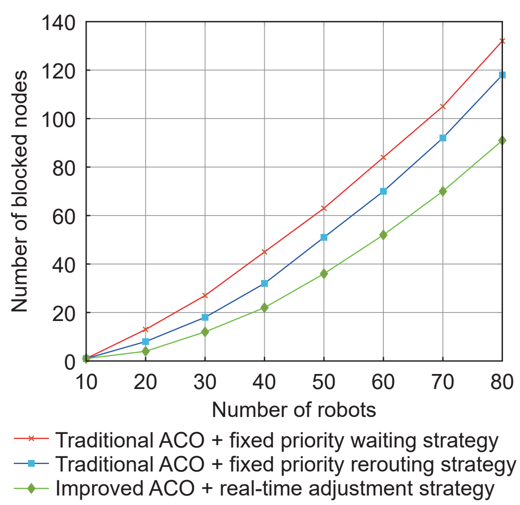

Fig. 6 Comparison of blocked node number.

2) Comparison of blocked nodes number

The comparison graph of the rerouting percentage is shown in Fig. 7. The traditional algorithm (traditional ACO + fixed priority rerouting strategy) and the algorithm in this study (improved ACO + real-time adjustment strategy) are simulated respectively.

Tsinghua University Press | IEEE

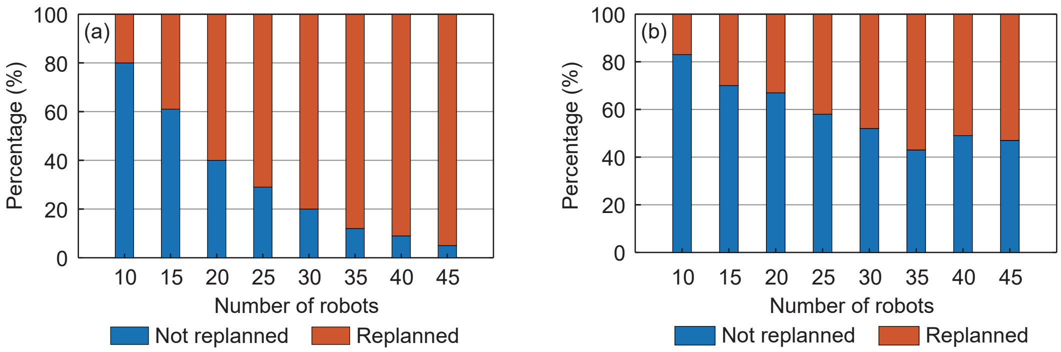

Fig. 7 Comparison graph of rerouting percentage. (a) Traditional algorithm rerouting percentage chart. (b) Improved algorithm rerouting percentage chart

The number of replanning of the original path was subsequently calculated. (after the first rerouting for the warehouse robot, the path still needs to be planned again or more due to conflicts, and multiple planning does not repeat the count). Fig. 7 indicates that an increase in the number of warehouse robots results in a corresponding gradual increase in the percentage of paths requiring replanning in both algorithms. Compared with the traditional ACO combined with the fixed priority rerouting strategy, the improved ACO combined with the real-time adjustment strategy is more effective in reducing the percentage of replanned paths. Moreover, the algorithm used in this study can maintain a high percentage of unplanned paths even with the expansion of the robot scale. For example, when the size of warehouse robots in the environment is 45, the algorithm in this study reduces the percentage of path replanning from over 90% to nearly 50%, highlighting the superiority of the proposed model. The advantage of this algorithm is that it can be obtained from the first static planning stage when there is no need for rerouting, which saves the resources of algorithmic operations and thus also relieves the pressure on the robot system. At the same time, the proposed algorithm draws dynamic priority into the dynamic planning stage. It combines two strategies of waiting and rerouting to solve the robot conflicts in the environment. and chooses the less time-consuming way between the two strategies. This kind of optimization effectively reduces the number of rerouting in the system, thus reducing the percentage of rerouting and making the performance of the two-level planning algorithm in this study fully developed.

(3) Comparison of average running time

Using the three different algorithms described above, we summarize the running time of warehouse robots at different scales, and the comparison results are shown in Fig. 8. As Fig. 8 shows, the average running time of each algorithm gets longer as the scale of the warehouse robot expands. In contrast, the two-level planning algorithm used in this study has a shorter average running time than the others. The fluctuation of indexes in terms of each algorithm is not significant when the robot size is 10–40 robots, but when 50 and more warehouse robots start working, the time of planning by the two traditional algorithms starts to increase significantly. The main reason is that either a single wait strategy or a single rerouting strategy after exceeding a certain size of robots in the environment will lead to more time consumed waiting or running, increasing the average running time. Meanwhile, the algorithm utilized in this study has a better planning effect because it considers the blockage factor and introduces a real-time adjustment strategy, which has more real-time characteristics. For example, when the robot size is 50 and 60 robots, the algorithm still maintains a high planning efficiency, outperforming the other two algorithms. As the number of robots

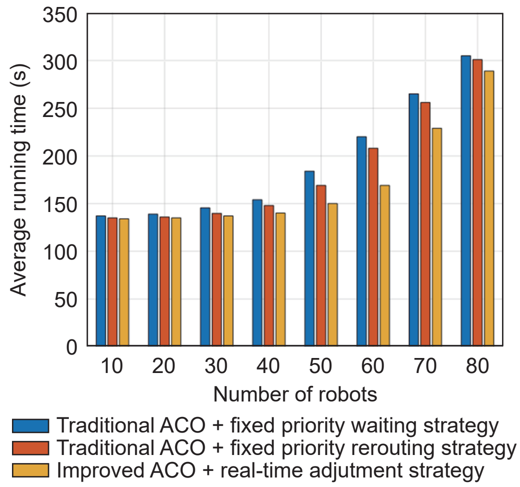

Fig. 8 Comparison chart of average running time.

in the system reaches 70 or more, the average running time of the proposed algorithm also starts to increase significantly. This is because most of the path resources in the environment are occupied, and there are more unstable factors, which ultimately leads to a decrease in the optimization effect.

In summary, the algorithm proposed in this study outperforms other algorithms in conflicting scenarios. The main features are: First, the algorithm proposed in this study has a better effect on reducing blockage compared with other algorithms, and the advantage is more obvious when there are more robots. Second, compared with other algorithms, the improved ACO combined with the real-time adjustment strategy is more effective in reducing the proportion of replanned paths, and the algorithm can still maintain a high proportion of unplanned paths as the robot number increases. In addition, the algorithm can choose the least time-consuming way and can effectively reduce the number of replanning in the system, thus reducing the proportion of replanning. Finally, the improved algorithm has a shorter average running time than others. At the same time, the algorithm proposed in this study has stronger real-time performance and better planning effect due to the consideration of the blockage factor and the introduction of real-time adjustment strategy. Therefore, the improved ACO proposed in this study outperforms other algorithms in terms of the number of blocked nodes, the proportion of replanned paths, and the average movement time.

5 Conclusions and implications

In this study, we propose a vehicle path planning model for multi-warehouse robots that applies a two-level planning approach. To begin with, our analysis of common types of conflicts in warehouse environments informs the development of targeted dissipation strategies to decrease the probability of conflicts occurring. The proposed model integrates both static and dynamic planning phases to address various conflicts that may arise in these complex environments. In the static phase, we enhance the ACO algorithm by introducing a blockage factor to generate negative feedback mechanisms and enable robots to avoid congested areas during travel. In the dynamic planning phase, a real-time adjustment strategy is used to design a dynamic prioritization mechanism, which is combined with other strategies to help each robot in the system to find the optimal path.

We verify our proposed vehicle path planning model for multi-warehouse robots on the MATLAB platform. Our simulation results demonstrate the effectiveness of the conflict resolution strategy employed in our model. Specifically, our model outperforms the other two traditional models in terms of average running distance, blocked node number, rerouting percentage, and average running time for different sizes of conflict-free or conflicted scenarios. The promising results suggest that with further development and refinement, our proposed model can revolutionize the way warehouse operations are conducted. Furthermore, the model could be integrated with other technologies, such as autonomous vehicles or drones.

As the study still has some limitations: the proposed model has been tested only on the MATLAB platform, and the model only addresses conflicts between robots. In our future work, we plan to focus on several potential extensions for further research. Firstly, we will consider the integration of human operators and other equipment in the warehouse system. Secondly, we aim to further develop and test the proposed path planning model in real-world scenarios to validate its practicality and extend its application beyond multi-warehouse robots. These extensions will not only enhance the effectiveness of the warehouse system but also contribute to the development of more advanced robotics and automation technologies.

Replication and data sharing

The data and code are withheld due to confidentiality requirements and can be obtained by contacting the corresponding author upon request.

Acknowledgements

This work is funded by the Basic and Applied Research Foundation of Guangdong Province (2020A1515111024).

Declaration of competing interest

The authors declare that they have no competing financial interests or personal relationships that might influence the work published in this paper.

References

Al-Amyal, F., Számel, L., Hamouda, M., 2023. An enhanced direct instantaneous torque control of switched reluctance motor drives using ant colony optimization. Ain Shams Eng J, 14, 101967.

Chen, C., Demir, E., Huang, Y., Qiu, R., 2021. The adoption of self-driving delivery robots in last Mile logistics. Transp Res E Logist Transp Rev, 146, 102214.

Chen, X., Gao, M., Kang, Z., Zhou, J., Chen, S., Liao, Z., *et al*., 2022. Formation of MASS collision avoidance and path following based on artificial potential field in constrained environment. J Mar Sci Eng, 10, 1791.

Cui, Y., Hu, W., Rahmani, A., 2023. Fractional-order artificial bee colony algorithm with application in robot path planning. Eur J Oper Res, 306, 47−64.

Daddi, G., Notaristefano, N., Stesina, F., Corpino, S., 2022. Assessing an image-to-image approach to global path planning for a planetary exploration. Aerospace, 9, 721.

Dai, Y., Xiang, C. F., Liu, Z. X., Li, Z. L., Qu, W. Y., Zhang, Q. H., 2022. Modular robotic design and reconfiguring path planning. Appl Sci, 12, 723.

Dian, S., Zhong, J., Guo, B., Liu, J., Guo, R., 2022. A smooth path planning method for mobile robot using a BES-incorporated modified QPSO algorithm. Expert Syst Appl, 208, 118256.

Ding, J., Zhou, Y., Huang, X., Song, K., Lu, S., Wang, L., 2023. An improved RRT* algorithm for robot path planning based on path expansion heuristic sampling. J Comput Sci, 67, 101937.

Gul, F., Mir, I., Alarabiat, D., Alabool, H. M., Abualigah, L., Mir, S., 2022. Implementation of bio-inspired hybrid algorithm with mutation operator for robotic path planning. J Parallel Distrib Comput, 169, 171−184.

Hao, L., Liu, D., Du, S., Wang, Y., Wu, B., Wang, Q., *et al*., 2022. An improved path planning algorithm based on artificial potential field and primal-dual neural network for surgical robot. Comput Meth Programs Biomed, 227, 107202.

Hou, W., Xiong, Z., Wang, C., Chen, H., 2022. Enhanced ant colony algorithm with communication mechanism for mobile robot path planning. Robot Auton Syst, 148, 103949.

Huang, X., Lin, P., Pei, M., Ran, B., Tan, M., 2023. Reservation-based cooperative ecodriving model for mixed autonomous and manual vehicles at intersections. IEEE Trans Intell Transp Syst, 24: 9501−9517.

Jin, J., Zhang, Y., Zhou, Z., Jin, M., Yang, X., Hu, F., 2023. Conflict-based search with D* lite algorithm for robot path planning in unknown dynamic environments. Comput Electr Eng, 105, 108473.

Kobayashi, M., Motoi, N., 2022. Local path planning: Dynamic window approach with virtual manipulators considering dynamic obstacles. IEEE Access, 10, 17018−17029.

Larsen, L., Kim, J., 2021. Path planning of cooperating industrial robots using evolutionary algorithms. Robot Comput Integr Manuf, 67, 102053.

Li, C., Huang, X., Ding, J., Song, K., Lu, S., 2022a. Global path planning based on a bidirectional alternating search A* algorithm for mobile robots. Comput Ind Eng, 168, 108123.

Li, Q., Soleimaniamiri, S., Li, X., 2022b. Optimal mass evacuation planning for electric vehicles before natural disasters. Transp Res D, 107, 103292.

Li, Q., Li, X., 2022c. Trajectory planning for autonomous modular vehicle docking and autonomous vehicle platooning operations. Transp Res E, 166, 102886.

Li, L., Shi, D., Jin, S., Yang, S., Zhou, C., Lian, Y., *et al*., 2023a. Exact and heuristic multi-robot dubins coverage path planning for known environments. Sensors, 23, 2560.

Li, Y., Zhao, J., Chen, Z., Xiong, G., Liu, S., 2023b. A robot path planning method based on improved genetic algorithm and improved dynamic window approach. Sustainability, 15, 4656.

Lian, Y., Yang, Q., Liu, Y., Xie, W., 2022. A spatio-temporal constrained hierarchical scheduling strategy for multiple warehouse mobile robots under industrial cyber-physical system. Adv Eng Inform, 52, 101572.

Lim, K. L., Whitehead, J., Jia, D., Zheng, Z., 2021. State of data platforms for connected vehicles and infrastructures. Commun Transp Res, 1, 100013.

Ma, G., Duan, Y., Li, M., Xie, Z., Zhu, J., 2023. A probability smoothing Bi-RRT path planning algorithm for indoor robot. Future Gener Comput Syst, 143, 349−360.

Lv, T., Zhang, J., Zhang, J., Chen, Y., 2022. A path planning algorithm for mobile robot based on edge-cloud collaborative computing. Int J Syst Assur Eng Manag, 13, 594−604.

Ma, W., Zhu, Y., Wu, Z., 2022. Vehicle motion prediction algorithm based on artificial potential field correction and fuzzy C-mean driving intention classification. Electronics, 11, 3857.

Miao, C., Chen, G., Yan, C., Wu, Y., 2021. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. Comput Ind Eng, 156, 107230.

Peralta, F., Arzamendia, M., Gregor, D., Reina, D. G., Toral, S., 2020. A comparison of local path planning techniques of autonomous surface vehicles for monitoring applications: The ypacarai lake case-study. Sensors, 20, 1488.

Psotka, M., Duchoň, F., Roman, M., Michal, T., Michal, D., 2023. Global path planning method based on a modification of the wavefront algorithm for ground mobile robots. Robotics, 12, 25.

Qin, H., Shao, S., Wang, T., Yu, X., Jiang, Y., Cao, Z., 2023. Review of autonomous path planning algorithms for mobile robots. Drones, 7, 211.

Qu, F., Yu, W., Xiao, K., Liu, C., Liu, W., 2022. Trajectory generation and optimization using the mutual learning and adaptive ant colony algorithm in uneven environments. Appl Sci, 12, 4629.

Rajamoorthy, R., Arunachalam, G., Kasinathan, P., Devendiran, R., Ahmadi, P., Pandiyan, S., *et al*., 2022. A novel intelligent transport system charging scheduling for electric vehicles using Grey Wolf Optimizer and Sail Fish Optimization algorithms. Energy Sources A, 44, 3555–3575.

Sahu, B., Das, P. K., Kumar, R., 2023. A modified cuckoo search algorithm implemented with SCA and PSO for multi-robot cooperation and path planning. Cogn Syst Res, 79, 24–42.

Shao, S., Peng, Y., He, C., Du, Y., 2020. Efficient path planning for UAV formation via comprehensively improved particle swarm optimization. ISA Trans, 97, 415–430.

Song, B., Wang, Z., Zou, L., 2021. An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve. Appl Soft Comput, 100, 106960.

Souza, R. M. J. A., Lima, G. V., Morais, A. S., Oliveira-Lopes, L. C., Ramos, D. C., Tofoli, F. L., 2022. Modified artificial potential field for the path planning of aircraft swarms in three-dimensional environments. Sensors, 22, 1558.

Sun, J., Zhao, J., Hu, X., Gao, H., Yu, J., 2023. Autonomous navigation system of indoor mobile robots using 2D lidar. Mathematics, 11, 1455.

Tang, J., Liu, G., Pan, Q., 2021. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. IEEE/CAA J Autom Sin, 8, 1627–1643.

Tao, Y., Gao, H., Ren, F., Chen, C., Wang, T., Xiong, H., *et al*., 2021. A mobile service robot global path planning method based on ant colony optimization and fuzzy control. Appl Sci, 11, 3605.

Wang, Z., Wu, Y., 2023. An ant colony optimization-simulated annealing algorithm for solving a multiload AGVs workshop scheduling problem with limited buffer capacity. Processes, 11, 861.

Wu, L., Huang, X., Cui, J., Liu, C., Xiao, W., 2023. Modified adaptive ant colony optimization algorithm and its application for solving path planning of mobile robot. Expert Syst Appl, 215, 119410.

Wang, L., Wu, J., Li, X., Wu, Z., Zhu, L., 2022. Longitudinal control for person-following robots. J Intell Connect Veh, 5, 88–98.

Xu, L., Cao, M., Song, B., 2022. A new approach to smooth path planning of mobile robot based on quartic Bezier transition curve and improved PSO algorithm. Neurocomputing, 473, 98–106.

Yan, P., 2022. Research on the selection of path planning algorithm: A case study in Leeds. J Phys: Conf Ser, 2179, 012039.

Yang, L., Fu, L., Li, P., Mao, J., Guo, N., 2022. An effective dynamic path planning approach for mobile robots based on ant colony fusion dynamic windows. Machines, 10, 50.

Yin, Y., Chen, Z., Liu, G., Guo, J., 2023. A mapless local path planning approach using deep reinforcement learning framework. Sensors, 23, 2036.

Yuan, Q., Sun, R., Du, X., 2022. Path planning of mobile robots based on an improved particle swarm optimization algorithm. Processes, 11, 26.

Yue, L., Abdel-Aty, M., Wang, Z., 2022. Effects of connected and autonomous vehicle merging behavior on mainline human-driven vehicle. J Intell Connect Veh, 5, 36–45.

Zhang, Z., Yang, X. T., 2021. Analysis of highway performance under mixed connected and regular vehicle environment. J Intell Connect Veh, 4, 68–79.

Zhao, Y., Hao, L. Y., Wu, Z. J., 2023. Obstacle avoidance control of unmanned aerial vehicle with motor loss-of-effectiveness fault based on improved artificial potential field. Sustainability, 15, 2368.

Zhu, J., Easa, S., Gao, K., 2022. Merging control strategies of connected and autonomous vehicles at freeway on-ramps: A comprehensive review. J Intell Connect Veh, 5, 99–111.

Pan Wu received her B.S. degree in turbine engineering in 2015 from Chongqing Jiaotong University. Then, she received her M.S. degree in marine and offshore engineering in 2018 and her Ph.D. degree in traffic information engineering and control in 2022 from the South China University of Technology. She is currently working at Chongqing Jiaotong University as a lecturer. Her research interests in big data mining and analysis for transportation.

Lingshu Zhong received his B.Eng. and M.Sc. degrees from the South China University of Technology, Guangzhou, China, in 2013 and 2016, respectively. He received his Ph.D. degree at the same university. He was also a visiting Ph.D. student at the University of Saskatchewan, Saskatoon, Canada, from 2018 to 2019. After Ph.D. graduation, he worked as a postdoctoral researcher at the Department of Architecture and Civil Engineering, Chalmers University of Technology from 2020 to 2022. His research interests include transportation electrification and decarbonization, joint optimization of power and transportation networks, and big data analysis of power/transportation systems.