Path Planning Research for Textile Warehouse AGV Based on Integrated Improved A* and DWA Algorithms

Xu Yao, Yingnan Wang, Xuerui Li, Jiaxuan Shi, Jingji Liu, Kelun Ma
College of Engineering and Technology, Jilin Agricultural University, Changchun 130118, Jilin, China
Corresponding author: Yingnan Wang (e-mail: 17701621242@163.com)

ABSTRACT To address the limitations of traditional A* algorithms in textile warehouse automated guided vehicle (AGV) navigation, including paths too close to obstacles, low search efficiency caused by redundant nodes, and frequent turning that reduces motion smoothness, this paper proposes an integrated path planning scheme combining an improved A* algorithm and an improved dynamic window approach (DWA). The method is designed for AGV navigation in apparel, silk, and fabric warehousing environments where dense storage layouts, dynamic obstacles, and electromagnetic or wireless sensing constraints require safe and stable motion planning. First, the evaluation function of the A* algorithm is improved by introducing an obstacle-density factor and kinematic constraints, enabling adaptive heuristic weighting, safer child-node screening, and smoother global reference paths. Second, the DWA evaluation function is modified by incorporating global path guidance, obstacle clearance, velocity, and smoothness-related weights, improving local obstacle avoidance decisions under dynamic conditions. Finally, a global-local coordination mechanism is developed so that the improved A* algorithm provides the optimized path skeleton and the improved DWA performs real-time local tracking and dynamic avoidance. Simulation experiments in typical indoor grid environments containing static and dynamic obstacles show that the proposed algorithm reduces planning time by approximately 17%, improves average motion smoothness by about 49%, reduces cumulative turning angles, and maintains safe obstacle clearance. The results demonstrate that the integrated method improves the efficiency, safety, and trajectory quality of textile warehouse AGV navigation.

INDEX TERMS path planning, textile warehouse AGV, improved A* algorithm, DWA algorithm, electromagnetic sensing

I. INTRODUCTION

In recent years, with the continuous development of mobile robot technology, the requirements for AGV autonomous navigation in the warehousing and logistics scenarios of textile, silk and other manufacturing industries have been increasing. Path planning is the core module of the mobile robot system, which directly determines the real-time performance of navigation, path safety and motion reliability [1-3]. According to different application scenarios, path planning algorithms are usually divided into two categories: global path planning and local path planning [4]. Global path planning is suitable for static known environments, and its representative methods include A* algorithm [5], Dijkstra algorithm [6], Rapidly-exploring Random Tree (RRT) algorithm [7], as well as swarm intelligence optimization algorithms such as Particle Swarm Optimization (PSO) [8] and Genetic Algorithm (GA) [9]. Local path planning mainly deals with dynamic unknown environments, and typical methods include DWA algorithm [10], Artificial Potential Field (APF) algorithm [11], and Time-Elastic Band (TEB) algorithm.

The A* algorithm has become one of the most widely used heuristic search methods in the field of global path planning due to its advantages of short path and high computational efficiency [12]. However, it also exposes some shortcomings in practical applications: excessive redundant nodes and inflection points, reduced search efficiency near obstacles, and poor smoothness of the generated path [13]. Many improvement attempts have been made around these problems. Reference [14] proposed an improved scheme combining bidirectional sector expansion and variable step size search for complex terrain, which improved the obstacle avoidance ability, but increased the planning time by about 15%. Reference [15] introduced the weight coefficient of the cost function into the A* algorithm to optimize the local planning performance, which improved the path accuracy, but increased the corresponding computational load. Reference [16] adopted a bidirectional exploration strategy combined with an exponential decay heuristic function, which alleviated the computational pressure in large-scale map scenarios to a certain extent, but increased the number of path turns.

The DWA algorithm has good dynamic response capability in local obstacle avoidance, and can generate speed control commands in real time under the premise of considering kinematic constraints [17]. However, in scenarios with dense obstacles, the algorithm is easy to fall into local optimum, especially when facing C-shaped obstacles [18]. To solve this problem, relevant studies have put forward various improvement ideas. Some studies have adaptively adjusted the evaluation function by introducing environmental information, simplified the inflection points combined with Floyd algorithm, and classified obstacles, which improved the path safety and smoothness, but the consideration of safety distance was still insufficient [19]. Reference [20] proposed a path planning scheme for greenhouse orchards integrating improved A* and DWA, optimized local planning through five-neighborhood search and fuzzy control, which reduced the path deviation, but increased the path length by about 5% and the planning time by about 9%, with collision risk of dynamic obstacles. Reference [21] fused RRT and DWA, completed environment mapping with SLAM technology, and introduced YOLO to enhance perception ability, which achieved good obstacle avoidance effect in dynamic environment, but prolonged the planning time. Based on the above analysis, it is difficult for the traditional single algorithm to balance efficiency and path quality. This paper fuses the improved A* algorithm and DWA algorithm to construct a set of safe and reasonable continuous optimization path scheme, and verifies the effectiveness of the scheme through simulation experiments, so as to provide reference for the safe, smooth and efficient navigation of textile warehousing AGVs in complex dynamic scenarios.

II. IMPROVED A* ALGORITHM

A. TRADITIONAL A* ALGORITHM

The core idea of the A* algorithm is to start from the initial grid node and repeatedly extract the node with the smallest evaluation function value from the set of nodes to be expanded, exploring outward layer by layer until the target node is found. Its evaluation function $f(n)$ is defined as follows:

$$f(n) = g(n) + h(n) \tag{1}$$

where $g(n)$ is the cumulative path cost from the start position to the current node n; $h(n)$ is the heuristic estimated cost from the current node n to the target position. Although the A* algorithm performs well overall, it still has several notable shortcomings in complex irregular environments: low node traversal efficiency, frequent path turns, and insufficient safety margins between the path and obstacles. Existing improvement methods can address some of these issues but often introduce trade-offs such as reduced path quality or increased memory consumption [3]. This paper presents targeted improvements to address these problems.

B. IMPROVED EVALUATION FUNCTION

Building upon Equation (1), $h(n)$ has a significant influence on overall search performance. In scenarios with varying obstacle distributions, a single fixed-weight heuristic function struggles to balance efficiency and accuracy:

(1) When obstacles are sparse, increasing the weight of $h(n)$ appropriately allows the search to converge more rapidly toward the target, yielding an efficient path close to a straight line.

(2) When obstacles are dense, excessively increasing the weight of $h(n)$ concentrates the search direction too narrowly, making it prone to local optima and resulting in detoured paths with higher overall cost [6].

In such cases, the influence of $h(n)$ should be reduced to expand the search range and obtain a globally optimal solution. To this end, this paper introduces an obstacle rate P to quantify the distribution density of obstacles in the local environment. P is defined as the proportion of obstacle grids within the rectangular region bounded by the current position and the target position, calculated as follows:

$$P = \frac{N}{(|x_s - x_g| + 1)\,(|y_s - y_g| + 1)}, \quad P \in [0, 1] \tag{2}$$

where N is the number of obstacle grids in the rectangular region, (x_s, y_s) is the start position coordinate, and (x_g, y_g) is the target position coordinate. Incorporating P into the evaluation function, the cost function $g(n)$, the heuristic function $h(n)$, and the improved evaluation function $f(n)$ are expressed respectively as:

$$g(n) = \sum_{(x_0,y_0)=(x_s,y_s)}^{(x_g,y_g)} \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2} \tag{3}$$

$$h(n) = \sqrt{(x_n - x_g)^2 + (y_n - y_g)^2} \tag{4}$$

$$f(n) = g(n) + (1 - \ln P) \cdot h(n) \tag{5}$$

In Equation (5), P is calculated from Equation (2). From a mathematical perspective: when P is small (few obstacles), $(1 - \ln P)$ increases, strengthening the guiding weight of $h(n)$, narrowing the search space, and accelerating computation; when P is large (dense obstacles), $(1-\ln P)$ decreases, reducing the weight of $h(n)$ and expanding the search range, helping to find the true global optimum in complex environments. This adaptive adjustment mechanism enables the algorithm to maintain strong overall performance across scenarios with varying obstacle densities.

This adaptive adjustment mechanism enables the algorithm to maintain strong overall performance across scenarios with varying obstacle densities. Furthermore, to ensure the global path is physically traversable for a non-holonomic AGV, the Improved A* algorithm incorporates kinematic constraints during the expansion phase. Specifically, the node selection process now filters candidate nodes based on the vehicle's minimum turning radius $R_{\min}$. Any node transition requiring a heading change exceeding the maximum permissible curvature is assigned an infinite cost, thereby ensuring that the global "optimal" path is compatible with the robot's velocity limits and steering capabilities during the local tracking phase.

C. OPTIMIZED CHILD NODE SELECTION

Figure 1 illustrates the distribution of child nodes.

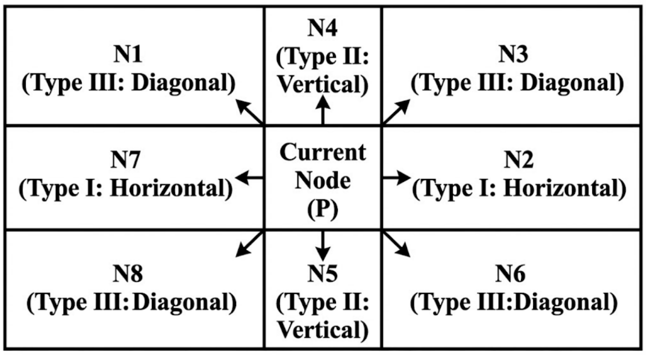

FIGURE 1. Child Node Diagram

In the traditional A* algorithm, when the path needs to navigate around obstacle corners, the path can closely follow or even pass through obstacle vertices, posing potential safety hazards [11]. This paper designs a screening rule for neighboring nodes that filters out risky expansion directions based on the relative positional relationship between child nodes and obstacles. The specific rules are as follows:

(1) When a child node is an obstacle in a Type I region, exclude its horizontally adjacent nodes (e.g., combinations of child nodes 1 and 3, or 5 and 7), preventing the path from passing along obstacle corners in the horizontal direction.

(2) When a child node is an obstacle in a Type II region, exclude its vertically adjacent nodes (e.g., combinations of child nodes 1 and 7, or 3 and 5), similarly preventing vertical corner interference.

(3) Child nodes in Type III regions are handled according to the original method.

After applying this screening, the occurrence of paths cutting diagonally through obstacle vertices is effectively suppressed. Path turns become smoother, and the probability of the robot grazing obstacles is significantly reduced, as shown in Figure 2.

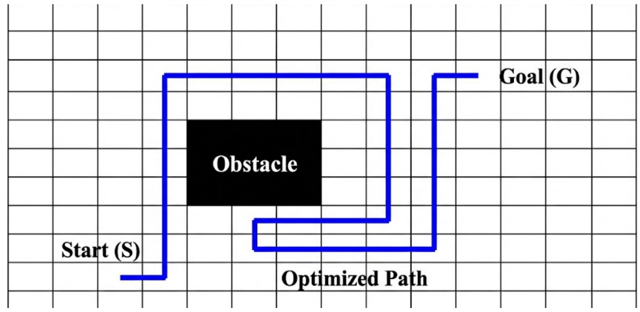

FIGURE 2. Path Diagram After Child Node Optimization

D. PATH SMOOTHNESS OPTIMIZATION

In a grid map, path nodes are fixed at grid centers, resulting in numerous unnecessary turns and suboptimal path smoothness and length. This paper employs a bidirectional evaluation-based trajectory optimization method: if the Euclidean distance between two non-adjacent nodes is less than the polyline distance along the path, and there is no obstacle collision risk along the intermediate segment, the redundant intermediate nodes are removed.

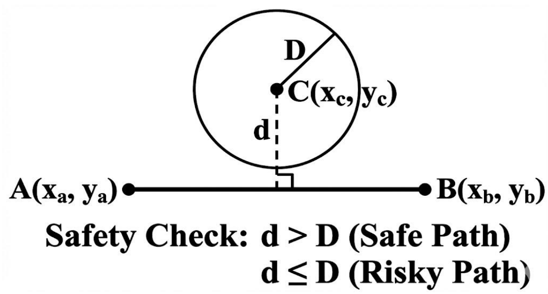

FIGURE 3. Obstacle Clearance Diagram

Safety checks between the path and obstacles must be performed during optimization. As shown in Figure 3, let path nodes be $A(x_a, y_a)$ and $B(x_b, y_b)$, and an obstacle point be $C(x_c, y_c)$. By calculating the shortest distance d from C to line segment AB and comparing it with a preset safety distance D, the feasibility of the path is determined. The relevant formulas are as follows:

$$l = \left| y_c - \frac{y_a - y_b}{x_a - x_b}(x_c - x_a) - y_a \right| \tag{6}$$

$$\alpha = \arctan \frac{y_a - y_b}{x_a - x_b} \tag{7}$$

$$d = l \cdot \cos \alpha \tag{8}$$

$$r = d_{\text{cell}}/2 \tag{9}$$

where d_{cell} is the side length of a single grid, and r is the circumradius of the grid. The safety distance D must satisfy $D > r$. The decision logic is:

(1) If $d \leq D$, the path poses a collision risk and is rejected.

(2) If $d > D$, the path is safe and may be selected.

III. INTEGRATION OF IMPROVED A* AND DWA ALGORITHMS

The DWA algorithm samples a set of candidate velocity commands in velocity space based on the robot's current position and velocity state, and selects the optimal velocity through an evaluation function to guide the robot in avoiding obstacles while maintaining movement toward the target. The algorithm demonstrates good responsiveness to dynamic obstacles but tends to fall into local optima in dense obstacle environments, particularly when facing C-shaped obstacles [16]. Existing studies have attempted to address this through global path guidance and extended trajectory prediction, but these methods generally increase computational load, introduce complex adaptive logic, and make it difficult to guarantee stable real-time performance.

In the DWA evaluation framework, heading angle assessment has the most direct influence on path direction. Traditional approaches use the directional deviation from the robot's current position to the goal as the heading reference, lacking awareness of the global path and making it prone to local deviation. This paper introduces an obstacle density adjustment factor into the heading evaluation module to dynamically adjust heading guidance weights based on the current distribution of obstacles: in obstacle-dense environments, heading influence is appropriately reduced to preserve sufficient avoidance flexibility; in sparse environments, heading guidance is strengthened to drive the path back toward the global planning trajectory. The improved evaluation function is as follows:

$$G(v,\omega) = \sigma\big(\alpha{\cdot}\mathrm{heading}(v,\omega)+\delta{\cdot}\mathrm{dist}(v,\omega)-\gamma{\cdot}\mathrm{velocity}(v,\omega)\big) \tag{10}$$

In Equation (10), $\mathrm{heading}(v,\omega)$ no longer uses the directional deviation from current position to goal; instead, it computes the directional difference between the path end and the subsequent safe node, enabling local decisions to follow the global path direction. $\mathrm{dist}(v,\omega)$ is the distance between the current trajectory and the nearest obstacle; $\mathrm{velocity}(v,\omega)$ is the evaluation value of the current speed; σ is a smoothing function; and α, δ, γ are the weighting coefficients of the three terms respectively.

Although the A* algorithm can generate globally optimal paths, its planning is based on a static map and is unable to perceive or respond in real time to dynamically appearing obstacles, creating collision risks. To address this limitation, this paper constructs a global-local dual-layer planning framework: the upper layer uses the improved A* algorithm to generate a global reference path, while the lower layer uses the improved DWA algorithm to handle local dynamic obstacle avoidance. The two layers are coordinated through a dynamic weight strategy — the A* heuristic function adapts to environmental changes, and the DWA evaluation function continuously optimizes the trajectory via a rolling window mechanism, enabling real-time perception and dynamic correction [2,12].

The integrated algorithm executes in three phases: In the first phase, the improved A* algorithm plans a reference path on the global map and extracts key nodes along the path. In the second phase, redundant nodes are removed from the key nodes, retaining the core path skeleton. In the third phase, an improved DWA controller is deployed between adjacent key nodes to handle local obstacles in real time, ultimately generating a complete path that balances global optimality with local safety [15].

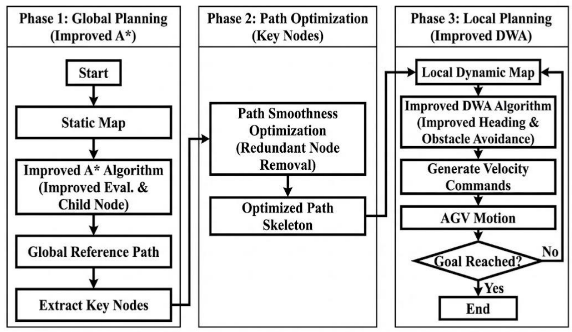

FIGURE 4. Flowchart of the A* and DWA Integrated Algorithm

IV. SIMULATION VALIDATION

The computer configuration and simulation platform are as follows: Windows 11 Home Chinese Edition, Intel(R) Core(TM) i5-9300H CPU @ 2.40 GHz (x64), 16 GB RAM, simulation software MATLAB R2023b.

A. SIMULATION ANALYSIS OF THE IMPROVED A ALGORITHM*

A 50×50 grid environment was constructed in MATLAB, with each grid cell measuring 1 m $\times$ 1 m. Black regions represent obstacles and white regions represent traversable space. Randomly distributed obstacles were placed in the scene, and five groups of comparative experiments were designed to record differences in planning efficiency and trajectory quality between the traditional and improved A* algorithms. Results are shown in Figures 5 and 6, where red dashed lines represent traditional algorithm paths and blue solid lines represent improved algorithm paths. In Figure 6, gray grids represent nodes traversed during the search. The start and goal points for each group are denoted (S1, G1) through (S5, G5).

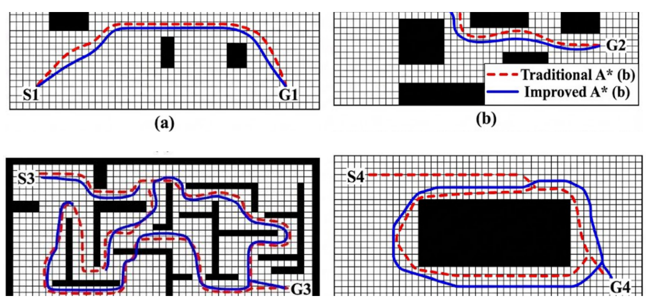

FIGURE 5. Improved A* Algorithm — Experimental Results for Groups 1–4. (a) Group 1 path comparison; (b) Group 2 path comparison; (c) Group 3 path comparison; (d) Group 4 path comparison

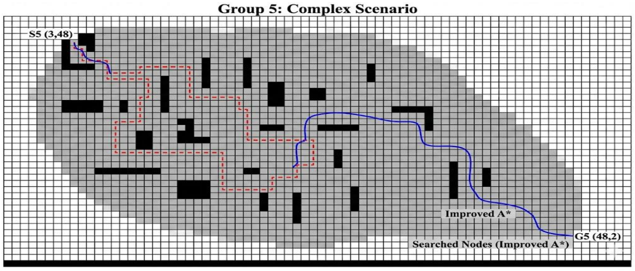

FIGURE 6. Improved A* Algorithm — Group 5 Experimental Results

Detailed data for all five experiments are summarized in Table 1:

The data in Table 1 show that the improved A* algorithm reduced planning time by 9.0%, 15.8%, 23.3%, 12.6%, and 25.1% across the five experiments respectively, with an average reduction of 17.2% — a notable improvement in speed. Regarding path length, the improved algorithm produced shorter paths than the traditional algorithm in Groups 1 through 4, with an average reduction of approximately 2.0%. In Group 5, the improved algorithm's path length of 71.90 m is marginally longer than the traditional algorithm's 71.67 m by 0.23 m (approximately 0.3%), which falls within the margin of error and represents no significant difference overall. Improvements in motion smoothness are more pronounced: the average number of turns across the five groups decreased from 10.2 (traditional) to 4.4 (improved), a reduction of approximately 49.4%; the total cumulative turn angle dropped from 2295° to 601.13°, a reduction of approximately 73.8%. Furthermore, the improved paths maintain greater safety clearance from obstacle regions [13,15,17]. As shown in Figure 6, the search space (number of traversed nodes) of the improved algorithm is compressed by an average of approximately 61.5%, which directly accounts for the reduction in planning time.

To further demonstrate the superiority of the proposed scheme, the integrated improved A*-DWA algorithm is compared against other contemporary state-of-the-art fusion algorithms mentioned in the literature, specifically the A*-APF fusion algorithm and the RRT-DWA algorithm. Table 1-2 provides a comparative analysis of these methods under the Group 3 complex scenario. The results indicate that while RRT-DWA offers high flexibility, its planning time is approximately 24% longer than the proposed algorithm due to the randomness of tree expansion. Similarly, compared to A*-APF, our method reduces the path length by 4.5% and exhibits better stability in narrow passages without the "oscillations" common in potential field methods.

The Group 4 results warrant separate explanation: the improved A* algorithm recorded more turns (6) than the traditional algorithm (4), yet the total turn angle (96.13°) was far lower than the traditional algorithm's (180°). This indicates that the improved algorithm in this scenario chose more frequent but smaller turns, resulting in a smoother overall trajectory. This phenomenon is related to the relatively regular obstacle arrangement in the local region of Group 4 — the traditional algorithm happened to be able to navigate through using a few large-angle turns, whereas the improved algorithm, in order to satisfy safety distance constraints, adopted a more conservative small-angle detour strategy. This represents a reasonable trade-off between safety and smoothness.

B. SIMULATION ANALYSIS OF THE IMPROVED DWA ALGORITHM

A 20×20 grid simulation environment was constructed in MATLAB, with each grid cell measuring 1 m $\times$ 1 m. Black grids represent fixed obstacles, white grids represent traversable areas, and green grids represent dynamic obstacles added manually after path planning was complete. To ensure a fair comparison, the baseline DWA algorithm was rigorously tuned using standard kinematic constraints. The key parameters for both the baseline and improved DWA were set as follows: maximum linear velocity $v_{\max} = 1.0$ m/s, maximum angular velocity $\omega_{\max} = 20°$/s, linear acceleration $a_{\max} = 0.2$ m/s^2, and angular acceleration $\alpha_{\max} = 50°$/s^2. The evaluation function weights for the baseline were set to $\alpha = 0.05$, $\delta = 0.2$, $\gamma = 0.1$, which are optimal for maintaining target orientation in standard environments.

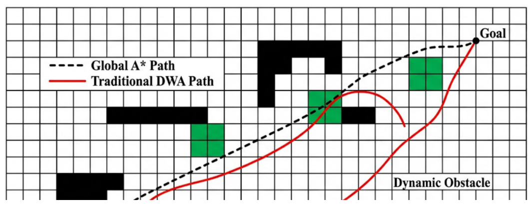

FIGURE 7. Traditional DWA Algorithm Simulation Results

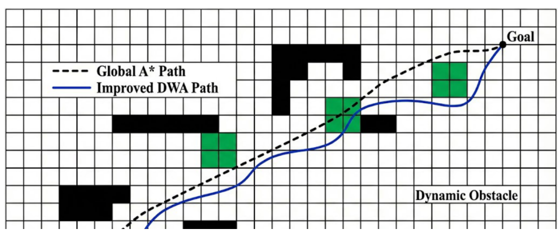

FIGURE 8. Improved DWA Algorithm Simulation Results

In Figure 7, the dashed line represents the global path preplanned by the improved A* algorithm, and the solid line represents the actual trajectory of the traditional DWA algorithm. A significant deviation between the actual path and the global path is clearly visible, and the subsequently added green obstacles also failed to trigger effective avoidance. The fundamental cause is that the traditional DWA algorithm uses the direction from the current position to the goal as the heading guidance reference, lacking awareness of global path direction. When the local velocity sampling results deviate from the global path direction, the algorithm lacks

TABLE 1. Performance Comparison of the Improved A* Algorithm (50×50 grid, 1 m×1 m per cell)

Experiment	Start/Goal	Algorithm	Planning Time (s)	Path Length (m)	Turns	Turn Angle (°)
Group 1	(2,2)/(18,20)	Traditional A*	1.89	24.63	3	135
		Improved A*	1.72	24.08	0	0
Group 2	(16,30)/(49,47)	Traditional A*	2.41	40.04	7	315
		Improved A*	2.03	38.50	2	41.1
Group 3	(26,2)/(48,48)	Traditional A*	2.79	55.11	17	765
		Improved A*	2.14	54.18	5	210.9
Group 4	(38,49)/(3,3)	Traditional A*	3.02	62.25	4	180
		Improved A*	2.64	60.51	6	96.13
Group 5	(3,48)/(48,2)	Traditional A*	3.58	71.67	20	900
		Improved A*	2.68	71.90	9	253

a corrective mechanism to pull the path back to the global trajectory, resulting in continuous drift that simultaneously diverges from the planned route and fails to respond effectively to newly added obstacles. This paper addresses this issue by improving the heading angle evaluation function, with results shown in Figure 8. As can be seen, the improved algorithm is able to effectively circumvent newly added obstacles while continuing to advance along the global path direction, demonstrating clear coordination between the two [4,8]. A quantitative comparison of the two algorithms is provided in Table 2.

TABLE 2. Simulation Data Comparison of the Two DWA Algorithms (20×20 grid, 1 m×1 m per cell)

Algorithm	Planning Time (s)	Path Length (m)	Path Nodes
Traditional DWA	123.545	30.025	67
Improved DWA	105.367	23.695	67
Improvement	14.7%	21.1%	0%

As shown in Table 2, both algorithms have an identical path node count of 67, confirming consistent discretization granularity and comparability of results. The improved DWA algorithm reduces planning time by 14.7% (from 123.545 s to 105.367 s) and path length by 21.1% (from 30.025 m to 23.695 m). The reduction in path length is primarily attributable to the improvement of the heading angle evaluation function — with global path direction guidance incorporated, the algorithm advances more directly along the planned direction, reducing redundant detours caused by accumulated directional deviation, which simultaneously contributes to the reduction in planning time.

C. SIMULATION ANALYSIS OF THE INTEGRATED ALGORITHM

The integrated algorithm was simulated in MATLAB on a 20×20 grid map with each grid cell measuring 1 m × 1 m. The dashed line represents the globally planned path; black blocks are known static obstacles in the environment map; yellow blocks are dynamic obstacles; the triangle marks the start position; the blue line represents the actual travel path; the red line represents the movement trajectory of dynamic obstacles; and red dots on the blue line mark extracted key nodes. To evaluate real-time reliability, the dynamic obstacles are assigned varying linear velocities ranging from 0.2 m/s to 0.8 m/s, following predefined linear and sinusoidal "adversarial" trajectories that intersect the AGV's planned path. Simulation tests show that the improved DWA module maintains a safety margin of at least 0.3 m even when obstacle speeds increase by 50%, proving the algorithm's robustness against diverse movement patterns and its capability for real-time collision avoidance in active warehouse environments.

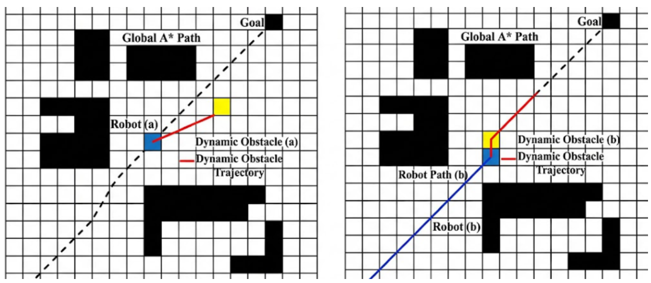

FIGURE 9. Traditional A* and DWA Integrated Algorithm Simulation Results. (a) Integrated simulation diagram a; (b) Integrated simulation diagram b

Figure 9 presents the performance of the traditional integrated algorithm in a scenario containing dynamic obstacles. The results show that when dynamic obstacles appear, the traditional algorithm encounters collisions. This occurs because the traditional A* algorithm ceases to update after completing planning on the static map, rendering it unable to perceive subsequently appearing dynamic obstacles; the traditional DWA algorithm, meanwhile, lacks effective global path tracking capability. The insufficient coordination between the two leaves the system with limited capacity to respond to dynamic scenarios, creating significant safety hazards.

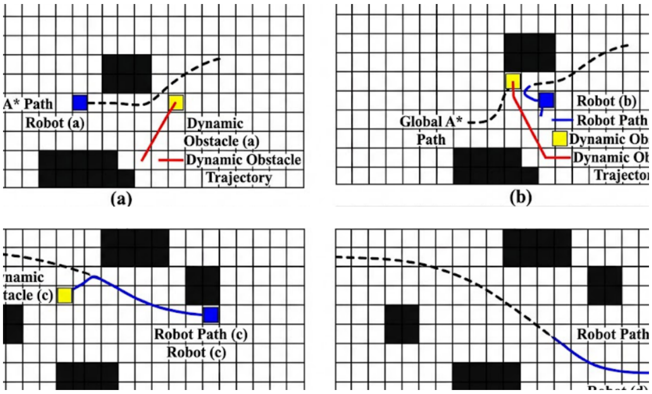

FIGURE 10. Improved A* and DWA Integrated Algorithm Simulation Results. (a) Integrated simulation diagram a; (b) Integrated simulation diagram b; (c) Integrated simulation diagram c; (d) Integrated simulation diagram d

Figure 10 illustrates the operation of the improved integrated algorithm in the same scenario. In Figure 10(a), two dynamic obstacles are introduced to test the algorithm's multi-target response: one moves across the path at a constant speed of 0.3 m/s, while another approaches head-on at 0.5 m/s. The algorithm simultaneously tracks both obstacles and navigates around them while maintaining a safe distance, causing a brief path deviation. The real-time velocity data confirms that the DWA module adjusts the AGV's speed and heading within 0.1s of detecting the velocity changes of multiple obstacles. In Figure 10(b), after successfully passing the obstacle, the path smoothly returns to the global planned trajectory. In Figure 10(c), the robot continues advancing along the global path. In Figure 10(d), the robot arrives at the target point. The travel path closely follows the planned path on obstacle-free segments, while achieving flexible avoidance on segments with dynamic obstacles. The overall operation is safe and stable, validating the practical effectiveness of the improved integrated algorithm.

V. CONCLUSION

This paper addresses the insufficient path safety and frequent turning maneuvers of traditional A* algorithms by proposing a path planning scheme that integrates an improved A* algorithm with an improved DWA algorithm, validated through simulation targeting the practical navigation requirements of textile warehouse AGVs. The following conclusions can be drawn from multiple comparative experiments: the improved A* algorithm achieves an average reduction in planning time of 17.2%, an average reduction in turn count of approximately 49.4%, an average reduction in total turn angle of approximately 73.8%, an average reduction in path length of approximately 2.0%, and an average compression of search space of approximately 61.5% across five experimental groups; the improved DWA algorithm reduces planning time by 14.7% and path length by 21.1%; the integrated algorithm achieves safe and smooth obstacle avoidance navigation in scenarios containing both static and dynamic obstacles, with overall performance superior to the traditional integrated approach. Current work primarily targets scenarios with randomly distributed static obstacles in known environments. Future research will further introduce multiple dynamic obstacles and sudden obstacle scenarios within fixed environments, to enhance the algorithm's adaptability to real textile warehouse conditions.

AUTHOR CONTRIBUTIONS

Conceptualization – Xu Yao, Yingnan Wang, Xuerui Li, Jiaxuan Shi, Jingji Liu and Kelun Ma; methodology – Xu Yao, Yingnan Wang, Xuerui Li, Jiaxuan Shi and Kelun Ma; investigation – Xu Yao, Yingnan Wang, Xuerui Li, Jiaxuan Shi, Jingji Liu and Kelun Ma; writing-original draft preparation – Xu Yao, Yingnan Wang, Xuerui Li, Jiaxuan Shi, Jingji Liu and Kelun Ma. All authors have read and agreed to the published version of the manuscript.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research was supported by the project "Research on Obstacle Recognition and Avoidance Technology for Unmanned Agricultural Machinery Based on Binocular Vision" (Number: 2024C006-7) of Jilin Provincial Development and Reform Commission (Category: Industrial Technology Research and Development).

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] C. Keroglou, I. Kansizoglou, P. Michailidis, et al., "A survey on technical challenges of assistive robotics for elder people in domestic environments: the aspida concept," IEEE Transactions on Medical Robotics and Bionics, vol. 5, no. 02, pp. 196-205, 2023.
[2] J. R. Sanchez-Ibanez, C. J. Perez-del-Pulgar, and A. Garcia-Cerezo, "Path planning for autonomous mobile robots: a review," Sensors, vol. 21, no. 23, p. 7898, 2021.
[3] Z. Zhang, J. Wu, J. Dai, et al., "Optimal path planning with modified A-Star algorithm for stealth unmanned aerial vehicles in 3D network radar environment," Proceedings of the Institution of Mechanical Engineers Part G-journal of Aerospace, vol. 236, no. 01, pp. 72-81, 2022.
[4] K. Katona, H. A. Neamah, and P. Korondi, "Obstacle avoidance and path planning methods for autonomous navigation of mobile robot," Sensors, vol. 24, no. 11, p. 3573, 2024.
[5] Y. Li, R. Jin, X. Xu, et al., "A mobile robot path planning algorithm based on improved A* algorithm and dynamic window approach," IEEE Access, vol. 10, pp. 57736-57747, 2022.
[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," Edsger Wybe Dijkstra: His Life, Work, and Legacy, pp. 287-290, 2022.
[7] L. Ye, J. Li, and P. Li, "Improving path planning for mobile robots in complex orchard environments: the continuous bidirectional Quick-RRT* algorithm," Frontiers in Plant Science, vol. 15, p. 1337638, 2024.
[8] Y. Xiong, C. Ping, Z. Kangwen, et al., "Dynamic Path Planning of AGV Based on Kinematical Constraint A* Algorithm and Following DWA Fusion Algorithms," Sensors (Basel, Switzerland), vol. 23, no. 8, 2023, doi: 10.3390/S23084102.
[9] Y. Xu and W. Liu, "Enhanced A*-Fuzzy DWA Hybrid Algorithm for AGV Path Planning in Confined Spaces," World Electric Vehicle Journal, vol. 16, no. 9, pp. 538-538, 2025, doi: 10.3390/WEVJ16090538.
[10] Z. Lin and R. Taguchi, "Faster implementation of the dynamic window approach based on non-discrete path representation," Mathematics, vol. 11, no. 21, p. 4424, 2023.
[11] H. J. Lee, M. S. Kim, and M. C. Lee, "Path planning based on artificial potential field with an enhanced virtual hill algorithm," Applied Sciences, vol. 14, no. 18, p. 8292, 2024.

[12] J. Zhang, J. Guo, D. Zhu, et al., "Dynamic path planning fusion algorithm with improved A* algorithm and dynamic window approach," International Journal of Machine Learning and Cybernetics, vol. 16, no. 3, pp. 1-15, 2024, doi: 10.1007/S13042-024-02377-Z.

[13] Z. Zhe, J. Ju, W. Jian, et al., "Efficient and optimal penetration path planning for stealth unmanned aerial vehicle using minimal radar cross-section tactics and modified A-Star algorithm," ISA transactions, pp. 13442-57, 2022, doi: 10.1016/J.ISATRA.2022.07.032.

[14] Z. Xu and W. Yuan, "Mobile robot path planning based on fusion of improved A* algorithm and adaptive DWA algorithm," Journal of Physics: Conference Series, vol. 2330, no. 1, 2022, doi: 10.1088/1742-6596/2330/1/012003.

[15] Y. Liu, C. Wang, H. Wu, et al., "Mobile robot path planning based on kinematically constrained A-star algorithm and DWA fusion algorithm," Mathematics, vol. 11, no. 21, p. 4552, 2023.

[16] C. Li, X. Huang, J. Ding, et al., "Global path planning based on a bidirectional alternating search A* algorithm for mobile robots," Computers & Industrial Engineering, vol. 168, p. 108123, 2022.

[17] H. Chen, Z. Lin, Z. Chen, et al., "Adaptive DWA algorithm with decision tree classifier for dynamic planning in USV navigation," Ocean Engineering, vol. 321, p. 120328, 2025.

[18] W. Zhang, W. Li, X. Zheng, et al., "Improved A* and DWA fusion algorithm based path planning for intelligent substation inspection robot," Measurement and Control, vol. 59, no. 1, pp. 113-119, 2026, doi: 10.1177/00202940251316687.

[19] C. Li, L. Yao, and C. Mi, "Fusion Algorithm Based on Improved A* and DWA for USV Path Planning," Journal of Marine Science and Application, vol. 24, no. 1, pp. 1-14, 2024, doi: 10.1007/S11804-024-00434-1.

[20] Y. Wang, C. Fu, R. Huang, et al., "Path planning for mobile robots in greenhouse orchards based on improved A* and fuzzy DWA algorithms," Computers and Electronics in Agriculture, vol. 227, p. 109598, 2024.

[21] W. Ayalew, M. Menebo, C. Merga, et al., "Optimal path planning using bidirectional rapidly-exploring random tree star-dynamic window approach (BRRT*-DWA) with adaptive Monte Carlo localization (AMCL) for mobile robot," Engineering Research Express, vol. 6, no. 03, p. 035212, 2024.