applied sciences

MDPI

Article

Multi-Automated Guided Vehicles Conflict-Free Path Planning for Packaging Workshop Based on Grid Time Windows

Guopeng Yang, Meiyan Li * and Qin Gao

College of Energy and Mining Engineering, Shandong University of Science and Technology, Qingdao 266590, China; 202182010070@sdust.edu.cn (G.Y.); 202283010071@sdust.edu.cn (Q.G.)
* Correspondence: limeiyanqdu@163.com

Abstract: In order to solve the problem of multi-AGV path planning in a packaging workshop, this paper proposes a multi-AGV path-planning method based on time windows. A grid method is selected to build a map model. A penalty function is added to the A* algorithm to reduce the number of invalid turns made by AGVs during transportation. The AGV priorities are set according to the differences in AGV transport states. At the same time, the raster time window method is used to describe the five types of conflict that may occur in the process of multi-AGV transport. Combined with the AGV priorities, a multi-AGV anti-collision strategy is provided to realize the conflict-free path planning of multiple AGVs. The algorithm's effectiveness was verified by simulation, and a reasonable quantity of AGVs was proposed based on AGV sensitivity analysis. The improved A* algorithm combined with the time window method can realize multi-AGV collision-free transportation and improve the efficiency and reliability of multi-AGV transportation.

Keywords: automated guided vehicles (AGVs); multi-AGV path planning; anti-collision strategy; improved A* algorithm; time window method

check for updates

Citation: Yang, G.; Li, M.; Gao, Q. Multi-Automated Guided Vehicles Conflict-Free Path Planning for Packaging Workshop Based on Grid Time Windows. *Appl. Sci.* **2024**, *14*, 3341. https://doi.org/10.3390/app14083341

Academic Editor: Arkadiusz Gola

Received: 20 February 2024
Revised: 13 April 2024
Accepted: 14 April 2024
Published: 16 April 2024

1. Introduction

With the development of the new generation of information technology, automatic driving has been applied more and more in many river basin areas [1], such as ports and factories. In the production activities of manufacturing enterprises, the processing time of materials usually only accounts for 5~10%, and the remaining time (such as storage, waiting, handling, etc.) accounts for 90~95%. The achievement of efficient, rapid, and stable commercial transportation is an effective way of shortening the production cycle of enterprises. Automatic guided vehicles (AGVs), with their advantages of high efficiency, flexibility, and high automation levels, have been widely connected with and applied in industrial production and organization, playing a non-negligible role in reducing production costs and improving planning efficiency [2]. Planning AGV transportation routes, improving the efficiency of AGV use, and ensuring the safety of multi-AGV transportation have become hot research issues in recent years.

At present, there are different AGV path-planning algorithms for different situations. Common AGV path planning algorithms include the A* algorithm [3], Dijkstra algorithm [4], ant colony algorithm [5], genetic algorithm [6], reinforcement learning algorithm [7], artificial potential field method [8], etc. Among them, the A* algorithm is widely used in the field of AGV path planning due to its characteristics of efficient search and output of optimal paths [9,10]. However, there may be invalid node searches in the search process of the A* algorithm, and the planned path will have invalid turns. To solve this problem, Liu [11] adopted the spatio–temporal search method in the search process, optimized the focused search operator, and improved the search efficiency of the algorithm. Tao [12] et al. increased the obstacle coefficient in the process of the A* algorithm, extracted key nodes in the path process, improved the search efficiency, and combined the

A* algorithm with the DWA algorithm to reduce the AGV travel distance and improve transportation efficiency. Hole [13] used the weighted Manhattan distance as the heuristic function, reduced the number of nodes traversed by the A* algorithm, added the turning cost parameter, and improved the smoothness of the path.

In the actual transportation process of AGVs, in the face of complex transportation requirements, a single AGV cannot meet the transportation needs, and multiple AGVs are needed to meet transport requirements at the same time. Therefore, the problem of collision-free or collision avoidance path optimization for multi-AGV transportation has also been designated a research hotspot in recent years. Algorithms are used to realize conflict-free path planning [14] or multi-AGV path planning in specific environments, such as docks [15], logistics warehousing [16], etc. In the research on multi-AGV path planning for the A* algorithm, Yu et al. [17] proposed two A* algorithms to deal with multi-AGV path planning. In case of AGV conflict, AGVs that are automatically adjusted are set to point to the linked list according to the remaining AGV planning caused by AGV path replanning, and AGVs behind the linked list are replanned. The second method is a multi-step forward algorithm, which predicts the path that the AGV is about to pass to predict the conflict in advance and choose the path with the least possibility of conflict. Compared with passive AGV conflict processing, the proactive algorithm has better initiative and reduces the loss of path replanning after AGV conflict. However, the iterative method is used to gradually fit the path, greatly increasing the computation amount of the algorithm. Yuan et al. [18] added the degree of road congestion to the estimation function so that the AGV can avoid the congested path and actively choose the road with a clear path in the process of path specification. However, this method is suitable for one-way traffic paths. Deng et al. [19] proposed a hybrid genetic algorithm based on scheduling and the A* algorithm, used the A* algorithm to carry out path planning and scheduling to avoid AGV node conflicts and designed a task-based chromosome structure to solve AGV scheduling and path planning problems for small and medium-sized flexible jobs. Tian et al. [20] set the AGV priority according to the remaining current of the AGV battery, improved the A* algorithm, and realized multi-AGV dynamic collision-free path planning according to the priority strategy and collision avoidance rules.

Research Scene

The subject of this study is a rubber product packaging workshop equipped with two production lines and a single AGV to accomplish the transportation task of packaging materials. In response to the growing market demands, there is a plan to incorporate three additional production lines and increase the number of AGVs. Considering the potential collision risks among multiple AGVs, it becomes imperative to investigate the problem of conflict-free path planning for multi-AGV transportation in the packaging workshop.

The packaging workshop measures 60 m in length and 46 m in width, accommodating a total of five packaging production lines. Each production line is equipped with three stations to carry out packaging operations using various materials. All the packaging materials are stored temporarily on the left side of the workshop. When a specific station requires replenishment of certain packaging materials, the intelligent manufacturing system assigns transportation tasks to AGVs. Upon receiving the delivery task, the AGV will navigate from its current position to retrieve the designated tray containing the required packaging materials from the shelf and transport them to the corresponding temporary storage area for raw materials. Subsequently, the AGV will transfer empty trays used in that station's temporary raw material storage area to an empty tray storage area in an upper corner before returning to its parking spot in anticipation of future transportation tasks. The facility layout of the packaging workshop in Figure 1 and the AGV driving route can be observed in red lines in Figure 1.

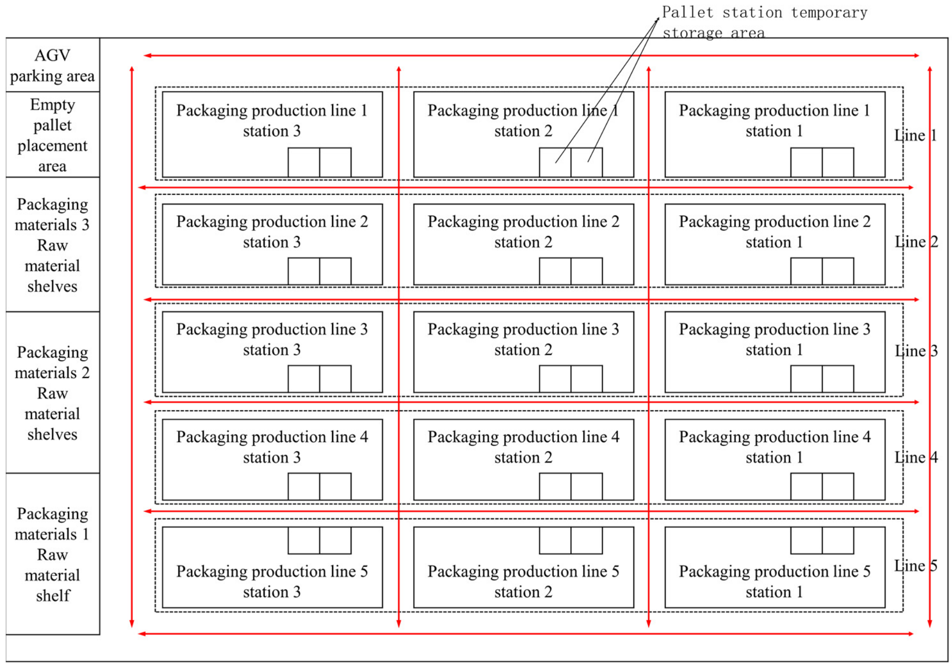


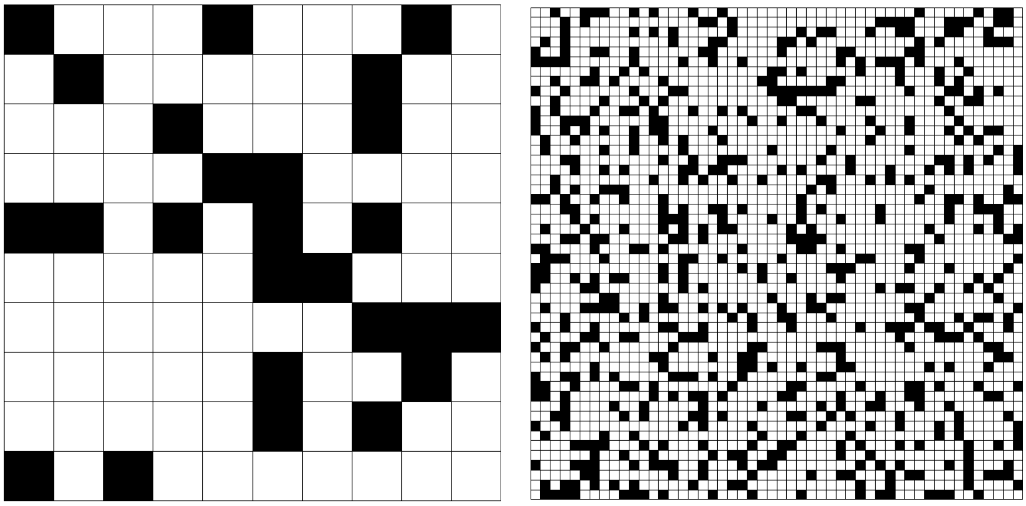
Figure 1. Packaging workshop diagram.

In this study, based on the actual situation of the packaging workshop, the passage area for the AGV was determined, and a grid map of the workshop was established. The single AGV path optimization algorithm was investigated, and a turning penalty factor was incorporated into the cost function of the traditional A* algorithm to minimize ineffective turns in the single AGV path. Five types of conflicts that may arise during multi-AGV transportation were identified, and conflicts on the grid were described using a time window method. An anti-collision strategy for each type of conflict was proposed by assigning priority to AGVs, enabling the identification of conflict-free paths with shorter transportation distances and fewer turns. Through the assignment of priority to AGVs, collision avoidance strategies were proposed for each type of conflict, facilitating the implementation of paths that were free from conflicts.

2. Single-AGV Path Planning Algorithm

2.1. Grid Map

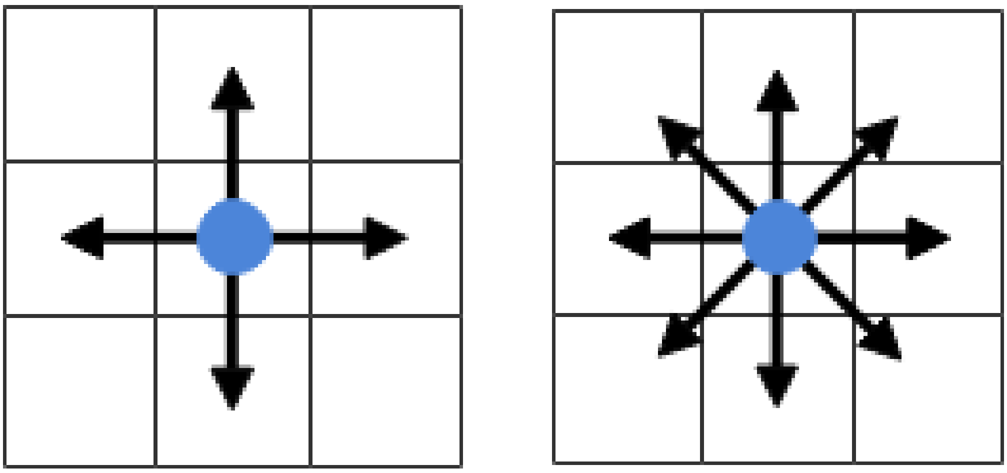
Common AGV path planning map models include the raster method [21], the topological diagram method [22], etc. Among them, the grid method is widely used because it can accurately depict the environmental information and display the current position of the AGV. It is also simple and easy to understand. The grid method refers to the map model obtained by processing the known information on the environment and completing the digitization of the working environment. The passable and impassable regions for AGVs are displayed, laying a foundation for the solution of subsequent path-planning problems [23]. Therefore, in this study, the grid method was used to build an AGV map model. The grid method transforms the two-dimensional environment in which the AGV operates into a grid map composed of uniformly sized black and white squares, with each grid cell's size equivalent to the occupied area of the AGV during transportation, and each cell's side length is 1. In the grid map, a white grid cell is a free cell, indicating that the AGV can pass; the black grid cell is a barrier cell, indicating that the AGV cannot pass. As shown in Figure 2, a is the 10×10 grid map, and b is the 50×50 grid map.



(**a**) 10 × 10 Grid map (**b**) 50 × 50 Grid map

Figure 2. Schematic diagram of raster map.

In the process of grid map simulation, the AGV adopts the four-direction or eight-direction search method. As shown in Figure 3, a is the four-direction, b is the eight-direction, blue dot in the figure represents the AGV position. Taking into account the packaging workshop, AGV diagonal driving is prone to collision so that the AGV can drive to the four feasible grid cells above and below the current cell.



(**a**) Four-direction of AGV (**b**) Eight-direction of AGV

Figure 3. Four-way search diagram.

According to Figure 1, the workshop floorplan was transformed into a raster map consisting of uniformly sized black and white squares. Each side of these squares has a length of 1 unit. In this grid map, white squares represent free cells, indicating passable routes for AGVs, while black squares represent obstacle cells, indicating impassable areas. To accurately describe the AGV's driving path, each free cell is sequentially numbered, starting from 1 in the upper left corner and continuing from top to bottom from left to right. The workshop map model after rasterization is shown in Figure 4.

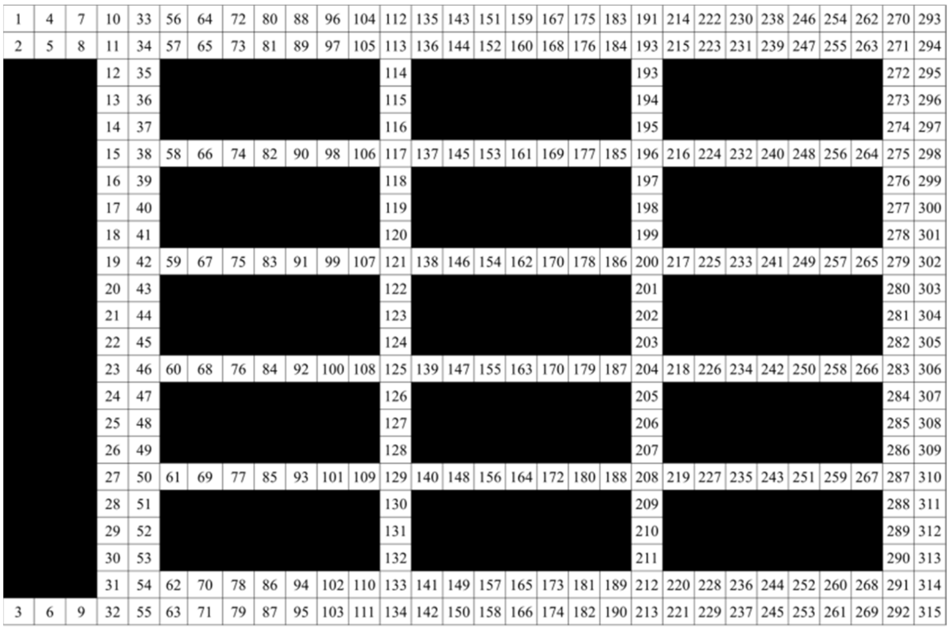


Figure 4. Grid diagram of workshop.

2.2. Improved A Algorithm*

2.2.1. Traditional A* Algorithm

The A* algorithm is derived from the Dijkstra algorithm. In comparison to the Dijkstra algorithm, the A* algorithm employs a greedy strategy that imparts a more pronounced sense of directionality during the search process. This characteristic enables it to narrow down the search space, thereby reducing path search complexity and enhancing overall efficiency [24]. Considering the constraint that the AGV in the packaging workshop is unable to move in the slope direction in transit, the A* algorithm employs a four-direction search mode in its search process. Starting from the current grid cell, only the unoccupied cells in the up, down, left, and right directions are utilized to search for the optimal path.

The shortest path obtained using the A* algorithm is found by calculating and comparing the cost functions of different paths, and the cost function formula is as follows [25]:

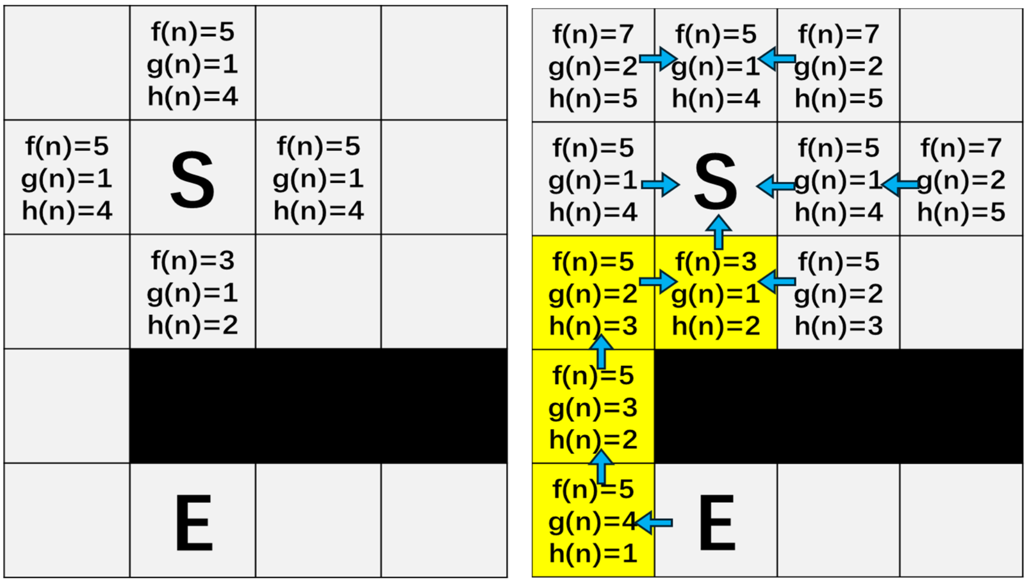
$$f(n) = g(n) + h(n) \tag{1}$$

Among them, f(n) is the total cost from the current grid cell n to the target grid cell, g(n) is the actual cost that has occurred from the starting grid cell n to the current grid cell, and h(n) is the estimated cost from the current grid cell to the target grid cell. Commonly, h(n) is used to represent the Manhattan distance, Euclidean distance, and Chebyshev distance. However, considering that there is no oblique path for AGV operation in the packaging workshop environment, the Manhattan distance was used for calculation:

$$h(n) = |X_n - X_z| + |Y_n - Y_z| \tag{2}$$

In Equation (2), X_n and Y_n are the X and Y coordinates of the current position of the AGV, and X_z and Y_z are the X and Y coordinates of the AGV transportation endpoint.

The A* algorithm uses three tables in the search process: open list, closed list, and traceability list. Among them, the open list is used to store grid information that has not yet been retrieved, the closed list records grid information that has already been retrieved, and the traceability list stores the relationships between all grid cells. Assuming that the AGV travels from the starting grid cell S to the target grid cell E, black represents the obstacle cells, and the cost of AGV movement between adjacent grid cells is 1. The A* algorithm searches for the shortest path, as shown in Figure 5:



(**a**) A* algorithm schematic (**b**) Track back to parent node

Figure 5. A* algorithm diagram.

The algorithm search steps are as follows:

Step 1: Add the starting point S as the parent node to the open list, and calculate the g(n) of m ($m \leq 4$) free grid cells above, below, left, and right at point S, as shown in Figure 5a. Add the calculated g(n) for the free grid cell to the open list, and calculate the h(n) and f(n) values of the newly added grid cell to the open list.

Step 2: Move the starting point S out of the open list and add it to the closed list. Select the grid cell with the smallest value in the open list as the new parent node.

Step 3: Repeat steps 1 and 2 continuously until the endpoint E is added to the open list. Obtain the planned path by tracing back to the parent node, as shown in Figure 5b.

1. The AGV loading and unloading time is set to 1 s;
2. AGV will not collide with obstacle grids.

2.2.2. Adding the A* Algorithm of the Turn Penalty Function

Since the traditional A* algorithm solely considers distance as the selection criterion for determining the optimal path, it fails to account for the impact of turning, which can significantly reduce the AGV's running speed [26]. Consequently, this leads to low actual transport efficiency, increased transport time, and a higher likelihood of failure [27]. To address this issue, an improvement can be made through the incorporation of a turning penalty factor into the cost function formula. This modification ensures that each turn increases the objective function value accordingly. The enhanced cost function formula is presented as follows:

$$f^*(n) = g(n) + h(n) + \sum ma \tag{3}$$

$$m = |X_{n+1} - X_{n-1}| \times |Y_{n+1} - Y_{n-1}| \tag{4}$$

$$m = \{0,1\} \tag{5}$$

Here, m is the turning judgment function, a is the turning penalty factor, and when the AGV turns, it will increase the size of f*(n) to a certain extent. Equation (4) is the formula for determining the value of m, and n is the current grid cell where the AGV is located, with coordinates (X_n,Y_n). Then, n − 1 is the previous grid cell on the AGV route, with coordinates (X_{n-1},Y_{n-1}). Cell n + 1 is the next grid cell on the AGV route, with coordinates (Xn+1,Yn+1) when the AGV turns at grid cell n, m = 1; otherwise, m = 0.

Through the use of the improved cost function and incorporating it into the algorithm flow, a single AGV transport path with fewer turns can be obtained.

3. Multi-AGV Path Planning Based on Time Window

3.1. The Time Window Method

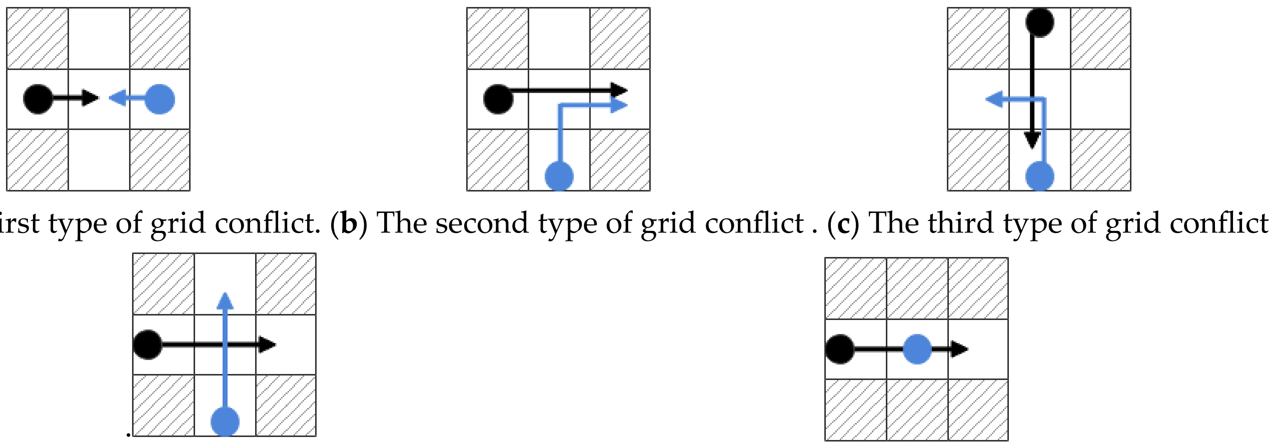
The time window refers to the time information of an AGV passing through a certain path [28], used to record the start time of an AGV entering a grid cell and the end time of an AGV leaving the cell. The time window of the grid cell is divided into an idle time window and an occupied time window. The idle time window is when there is no AGV driving in a certain grid cell, and any AGV can enter that grid cell at this time. The occupied time window refers to the presence of AGVs on the grid, during which other AGVs cannot enter the path of that segment. When the occupied time window of the grid corresponds to multiple AGVs simultaneously, it indicates that AGVs are in conflict here.

3.2. Types and Handling Strategies of Multiple AGV Path Conflicts

AGV path conflicts can be divided into three categories: directional conflicts, grid conflicts, and fixed-point conflicts [29], as shown in Figure 5. The shaded areas in the figure represent obstacle grid cells, different colored dots represent AGVs, and arrows represent the direction of AGV travel in that color:

(1) Grid conflict occurs when two or more AGVs meet at the same grid location at the same time. There are roughly four types of grid conflicts, as shown in Figure 6a–d. The first and second types of grid conflicts have overlapping AGV transportation paths, and it is necessary to redefine the AGV path to avoid the occurrence of conflicts. The third and fourth types of grid conflicts can be avoided by determining the priority order of traffic, as there are no overlapping transportation paths outside the conflicting grid cell;

(2) Fixed-point conflict occurs when the AGV is loading and unloading goods or experiencing sudden malfunctions, and the position of the AGV does not change in a short period of time. When other AGVs pass through the grid cell, fixed point conflict occurs, as shown in Figure 6e. At this point, it is necessary to set the fixed-point grid cell as an obstacle cell and replan the path of the AGVs to avoid conflicts.



(**a**) The first type of grid conflict. (**b**) The second type of grid conflict . (**c**) The third type of grid conflict

(**d**) The fourth type of grid conflict. (**e**) Fixed-point conflict.

Figure 6. Schematic diagram of AGV conflict.

In response to the above characteristics of conflict, the method of setting priorities for the AGV can be used to avoid collision of AGV paths in a predetermined grid cell:

(1) Replan the path scheme: When there is a conflict between the first and second types of grids, the AGV with higher priority maintains its original path scheme through the grid. The AGV with low priority sets the conflict grid cell as an obstacle cell in the grid graph, starts from the current position, replans the route, and drives according to the new path scheme.

When a fixed-point conflict occurs, the AGV that is malfunctioning or loading and unloading goods remains stationary. The AGV traveling in a normal manner designates the grid cell containing the stationary AGV as an obstacle cell and initiates path replanning from its current position to follow the newly planned path.

(2) AGV waiting scheme: When conflicts occur between the third and fourth types of grids, the AGV with higher priority maintains the original path scheme through the grid. The low-priority AGV waits in place, and after the high-priority AGV passes, it passes through the conflict grid cell according to the original path plan.

3.3. Multi-AGV Conflict Determination Based on Time Window

Through the above analysis, it was found that in an environment with multiple AGVs running at the same time, the improved A* algorithm can first be used to determine the best path for each AGV and then to compare it with the path of other AGVs. If two or more AGVs pass through a certain grid cell at a given time, it can be judged that AGV conflict will occur in the grid. By combining the position of each AGV before and after passing through the grid cell, it is possible to determine the type of AGV conflict.

The time window method is used to determine whether a grid cell has an AGV passing through it. The time window refers to the time information of an AGV passing through a certain path [30], which is used to record the beginning time of an AGV entering the grid cell and the end time of an AGV leaving the grid cell. The time window of the grid cell is divided into idle time windows and occupied time windows. An idle time window is for a grid cell without an AGV driving in it; at this time, any AGV can enter the cell. The occupation time window means that an AGV is occupying the grid cell; therefore, other AGVs cannot enter the path. When the occupancy time window of the grid cell corresponds to multiple AGVs at the same time, there is a conflict between the AGVs here. By setting a time window for each grid cell, the specific time window can determine the order in which an AGV passes through each grid cell. If AGV conflicts occur, the time windows of AGVs passing through the grid will coincide. In this case, different AGV priority rules are set according to different conflict types so that AGVs with lower priorities wait or replan paths to delay or rearrange the time windows. The AGV with a higher priority is driven according to the planned path so that no time window conflict occurs in the driving process of multiple AGVs; finally, AGV conflict-free path planning is realized.

3.4. Multi-AGV Priority Design

In the process of multi-AGV conflict handling, priority design is a necessary prerequisite to determine the AGV traffic order. This study adopts the method of combining task priority and number priority to set AGV priority. First, all AGVs are numbered in integer order starting from 1. When AGV conflicts occur, AGVs are prioritized according to the following principles:

Principle 1: AGVs that are loading and unloading and are idle have the highest priority;

Principle 2: AGVs that are performing transportation tasks have a high priority;

Principle 3: When multiple AGVs are performing transportation tasks, the AGV whose transportation task started earlier has a higher priority;

Principle 4: When the task start time is the same, the AGV with a smaller AGV number has a higher priority.

Thus, in the process of multi-AGV operations, the improved A* algorithm can be used to obtain the optimal path of each AGV. Then, the time window method is used to determine whether these optimal paths will conflict on a certain grid cell at a certain time. At the same time, according to the AGV before and after the grid position to determine the conflict type, combined with the AGV priority setting rules, the multi-AGV conflict-free path planning can be realized.

3.5. Multi-AGV Path Planning Algorithm Based on Time Window

Based on the above analysis, the running steps of the time window-based multi-AGV conflict-free path optimization algorithm are as follows:

Step 1: Convert the AGV work environment into computer-recognizable information based on raster maps and set the AGV operating parameters;

Step 2: Given a task list, read the task number, position information of starting point S, and position information of ending point E from the AGVs' task allocation table;

Step 3: Assign tasks based on the task list, then search for available AGVs. If there are available AGVs, assign tasks to nearby available AGVs and continue to the next step; if there is no idle AGV, search again after one second;

Step 4: Use the improved A* algorithm to plan the shortest transportation path for the AGV and obtain the time window information of the AGV passing through each grid cell on the shortest path. Continue to the next step;

Step 5: Determine whether the shortest path passes through the grid cell where the idle AGV is located. If so, proceed to step 10; if not, proceed to the next step;

Step 6: Determine if there is a time window conflict with the AGV carrying out the transportation task. If one exists, proceed to step 7. If it does not exist, proceed to step 11;

Step 7: Determine the type of AGV conflict. If there is a first or second type of grid conflict with the AGV in transportation status, proceed to step 8. If there is a third or fourth type of grid conflict with the AGV in loading and unloading status, proceed to step 9. If there is a fixed-point conflict with the AGV in loading and unloading status, proceed to step 10;

Step 8: Determine the priority of conflicting AGVs. The AGVs with a higher priority should maintain normal operation, while the AGVs with a lower priority should set the conflict grid cell as an obstacle cell in the grid graph. Go to step 4 and replan the path;

Step 9: Determine the priority of conflicting AGVs. The higher priority AGVs pass through the planned path, while the lower priority AGVs wait for the higher priority AGVs to pass through the conflict grid cell, update the time window information, and proceed to step 5;

Step 10: Set the grid cell where the AGV is idle or in loading and unloading status as an obstacle cell, and proceed to step 4 to replan the path;

Step 11: The AGV performs transportation tasks. When the AGV reaches the endpoint, set the AGV status to idle and proceed to the next step;

Step 12: Retrieve the task list to see if there are any unfinished transportation tasks. If so, proceed to step 3. If all tasks have been completed, proceed to the next step;

Step 13: The algorithm ends.

The flowchart is shown in Figure 7.

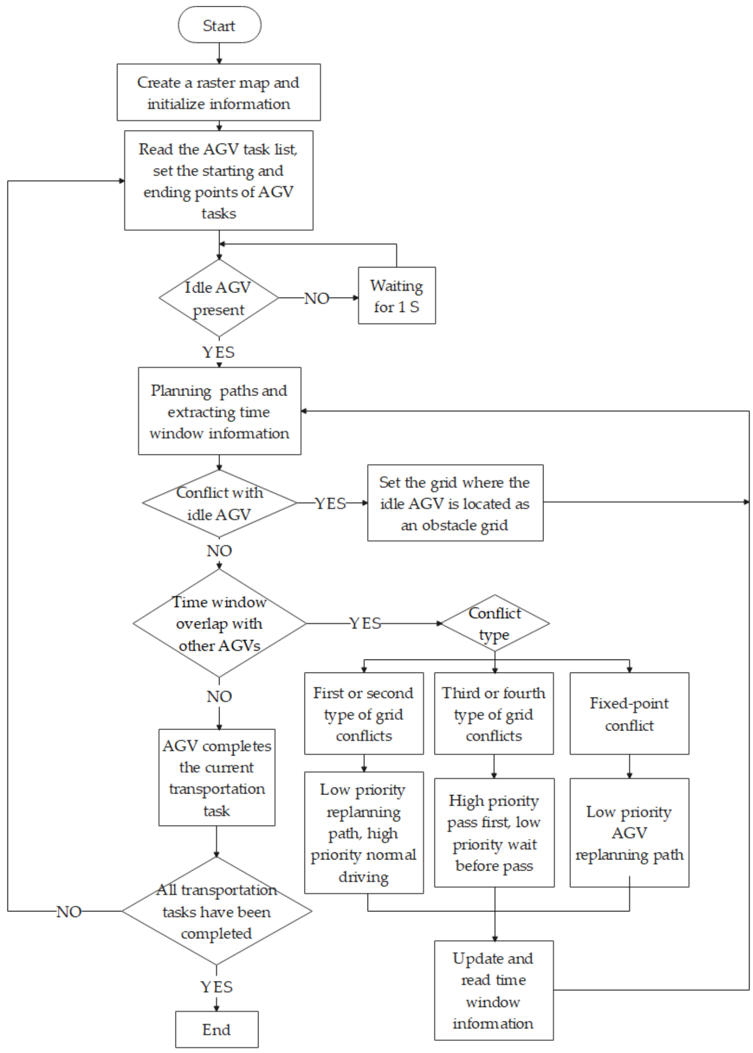


Figure 7. Multi-AGV path planning flowchart.

4. Algorithm Test and AGV Quantity Analysis

In this study, MATLAB 2021a was used to verify and analyze the algorithm.

4.1. Single AGV Path Planning Simulation

In order to verify the effectiveness of the improved A* algorithm, the algorithm was used to randomly produce raster maps and simulate the start and endpoints. The size of the grid was 30 × 30; the proportion of barrier cells was 20% and 30%. The grid side length was 1, m = 10. The traditional A* algorithm and the improved A* algorithm were each used to conduct 50 simulations. After removing the abnormal data, the AGV driving distance and number of turns were recorded. The results are shown in Table 1.

Table 1. Comparison table of single-AGV path planning.

	Total Path Length with 20%	Total Turns Times with 20%	Total Path Length with 30%	Total Turns Times with 30%
A* algorithm	1412	449	1458	574
Improved A* algorithm	1412	285	1458	396

According to Table 1, total turn times with 20% were down 36.6%, and total turn times with 30% were down 27.7%. In the process of single AGV path planning, the improved A* algorithm can reduce the number of AGV turns and improve the AGV transportation efficiency.

4.2. Planning Paths for Multi-AGV

(1) Replanning the path scheme in a conflict where two AGVs arrive at the same grid cell at the same time. The route of AGV1 is 18-24, as shown in the red route in Figure 8; the route of AGV2 is 24-18, as shown in the blue route in Figure 7. The arrows in Figure 8 indicate the direction of each AGV. Both AGVs departed at 0 s, and both AGVs simultaneously reached the cell numbered 21 at 3 s and collided.

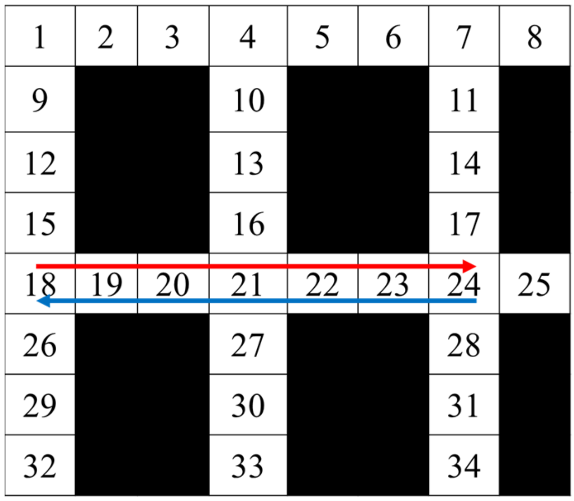


Figure 8. AGV Replanning roadmap.

If a conflict occurs, AGV2 needs to plan paths again. The AGV travel time window after relief is shown in Figure 9.

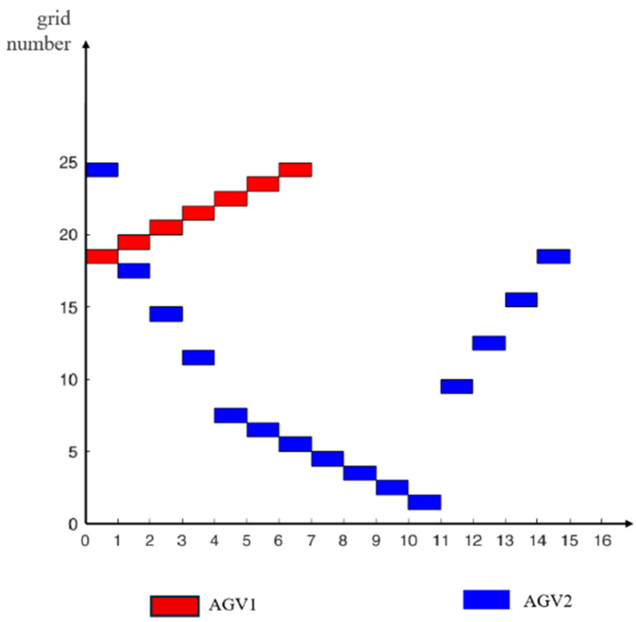
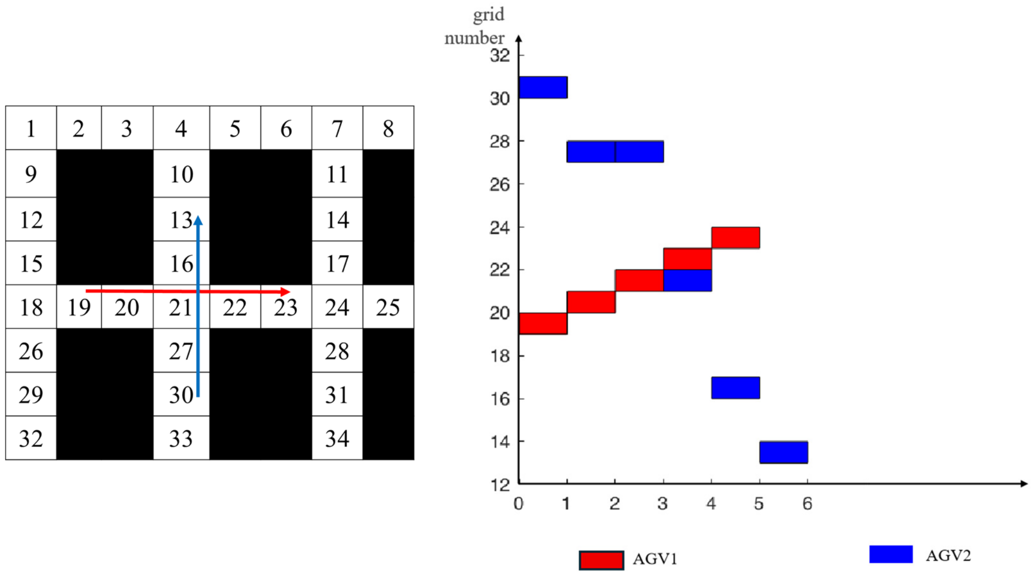


Figure 9. AGV Replanning travel time window.

(2) For the AGV executing the waiting scheme, as shown in Figure 10a, the route of AGV1 is 19-23, as shown in the red route, and the route of AGV2 is 30-13, as shown in the blue route. The arrow indicates the direction of each AGV. When a conflict occurs, AGV2 waits for AGV1 to pass through cell 21 before following its route. The AGV travel time window after mitigation is shown in Figure 10b.



(**a**) AGV driving route (**b**) AGV time window

Figure 10. Wait for conflict resolution schematics.

The algorithm can alleviate the multi-AGV conflict according to the set link strategy.

4.3. AGV Quantity Analysis of Packaging Workshop

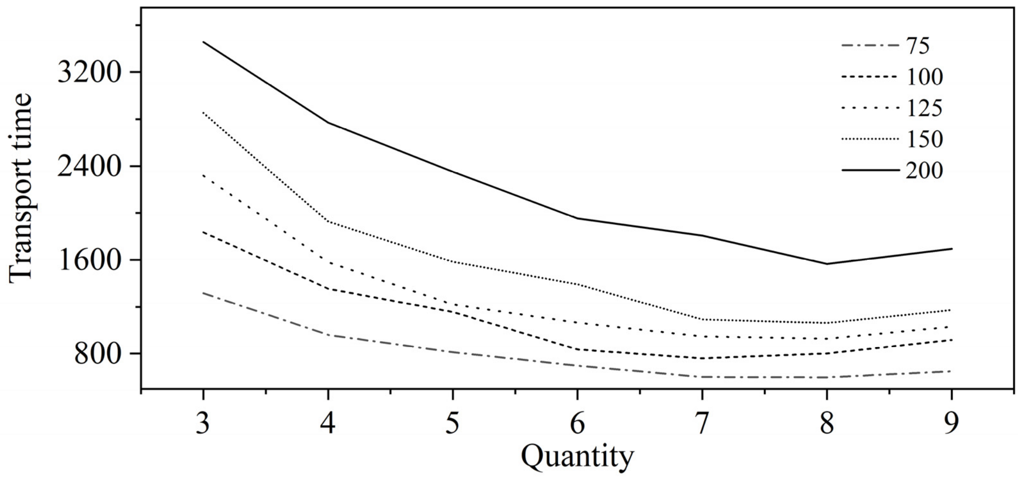
In this study, the multi-AGV conflict-free path planning algorithm based on time windows is used to analyze the sensitivity of the numbers in conflict-free AGV traffic. The number of missions increased from 70 to 200, and the number of AGVs increased from three to nine. The conflict-free transport time and the number of avoided collisions of AGVs under different transport tasks are simulated. The simulation results are shown in Table 2 and Figure 11. The top digit in the table is the transit time, and the bottom digit is the number of collisions avoided.

Table 2. AGV transit time and collision number table.

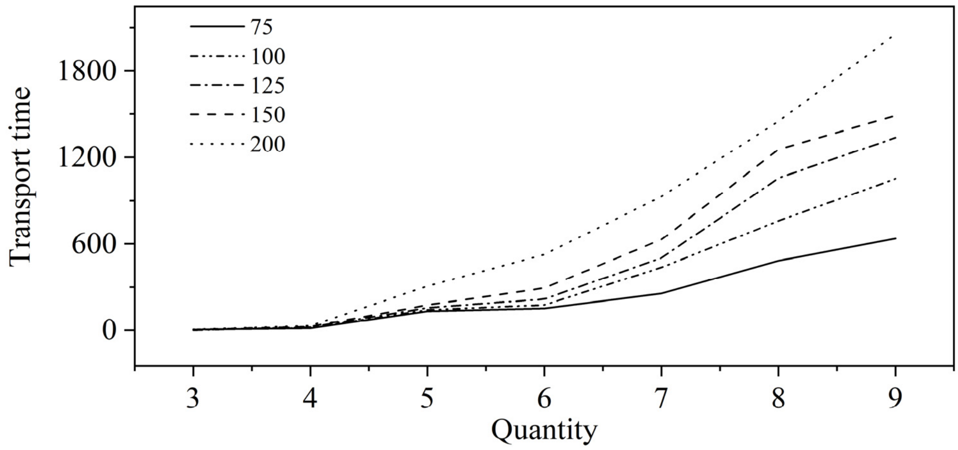
Transportation Time	**AGV Quantity**						
	3	**4**	**5**	**6**	**7**	**8**	**9**
75	1318 3	960 14	812 133	697 152	601 255	598 483	650 637
100	1834 2	1355 16	1157 142	835 176	759 437	801 761	917 1053
125	2319 0	1581 21	1221 156	1065 217	947 503	927 1057	1031 1336
150	2853 3	1925 25	1583 175	1393 293	1092 631	1062 1255	1173 1489
200	3455 5	2771 31	2352 307	1951 527	1807 930	1565 1449	1695 2057

AGV transport time and collision times are shown in Figure 11a,b.

The completion times of the AGV transportation tasks increase in line with the number of transportation tasks, as illustrated in Figure 11a. This phenomenon can be attributed to several factors: Firstly, when a fixed number of AGVs are assigned to perform multiple tasks continuously without interruption, they may encounter path conflicts and deviate from the shortest distance route. Consequently, the efficiency of AGV transportation fluctuates to some extent but has minimal impact on the overall completion time.



(**a**) Transport time of different AGVs



(**b**) Collision time of different AGVs

Figure 11. AGV quantity configuration figure.

When the number of transportation tasks is fixed at the initial moment, the task completion time remains relatively constant as the number of AGVs increases. However, when the number of AGVs becomes excessively large, there is a subsequent increase in the task completion time. At this juncture, augmenting the number of AGVs does not effectively mitigate the total task completion time. The underlying reasons for this phenomenon are analyzed. As depicted in Figure 11b, an escalation in the quantity of AGVs results in a corresponding rise in collisions between them. With a substantial number of AGVs present, collision occurrences become more frequent and urgent, ultimately impeding the overall transportation efficiency and prolonging the total AGV task completion time.

Considering that the daily transportation task of packaging is undertaken about 180 times, and there is a certain upward trend, it is recommended that the appropriate number of AGVs in the packaging workshop is four. In this case, each AGV can complete its transportation task quickly, and the possibility of AGV conflict in the transportation process is low.

5. Conclusions

In this study, the multi-AGV transportation path planning problem of a packaging workshop was examined. Firstly, the path planning problem of a single AGV was studied, and an improved A* algorithm with a penalty factor was proposed to solve the problem of invalid turns in AGV transportation. Then, for the multi-AGV path planning problem, the time window method was combined with the improved A* algorithm to judge multi-

AGV path conflicts. The collision phenomena occurring during AGV transportation were classified, and corresponding collision avoidance strategies were established. Finally, AGV conflict-free path planning was realized; the algorithm simulation verified that the improved A* algorithm can effectively reduce the number of AGV turns. The multi-AGV path planning algorithm based on time windows can avoid AGV conflict and complete the established transportation task. In this study, combined with the actual transportation situation of enterprises, the AGV transportation situation of the workshop was simulated, and the appropriate number of AGVs for this packaging workshop was proposed. This study has certain reference significance for the actual transportation issues of enterprises.

Author Contributions: Methodology, G.Y.; writing—original draft preparation, G.Y.; validation, Q.G.; data curation and supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used and analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hu, B.; Seragiotto, C.; Prandtstetter, M.; Binder, B.; Bader, M. Guiding Autonomous Vehicles Past Obstacles–Theory and Practice. In Proceedings of the Computer Aided Systems Theory–EUROCAST 2019: 17th International Conference, Las Palmas de Gran Canaria, Spain, 17–22 February 2019; Revised Selected Papers, Part I 17. Springer International Publishing: Cham, Switzerland, 2020; pp. 271–278.
2. Li, Z.; Liu, Y. Job shop scheduling considering multiple AGVs with charging. *Comput. Integr. Manuf. Syst.* **2021**, *27*, 2872–2879. [CrossRef]
3. Fu, B.; Chen, L.; Zhou, Y.; Zheng, D.; Wei, Z.; Dai, J.; Pan, H. An improved A* algorithm for the industrial robot path planning with high success rate and short length. *Robot. Auton. Syst.* **2018**, *106*, 26–37. [CrossRef]
4. He, J.; Li, P.; Shen, Y. Research on obstacle avoidance trajectory planning of AGV with moving obstacle. *J. Chongqing Univ. Technol.* **2022**, *36*, 135–143.
5. Gbadamosi, O.A.; Aremu, D.R. Design of a Modified Dijkstra's Algorithm for finding alternate routes for shortest-path problems with huge costs. In Proceedings of the International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), Lagos, Nigeria, 18–21 March 2020.
6. Meng, G.; Chen, X.; Tao, X.; Zhang, W. AGV Path Planning Based on Hybrid Ant Colony Algorithm. *Modul. Mach. Tool Autom. Manuf. Tech.* **2021**, *1*, 70–73. [CrossRef]
7. Xu, L.; Liu, Y.; Wang, Q. Application of adaptive *genetic* algorithm in robot path planning. *Comput. Eng. Appl.* **2020**, *56*, 36–41.
8. Wang, H.; Qin, G.; Yang, C. AGV Path Planning of Customized Furniture Plate Transportation. *Packag. Eng.* **2021**, *42*, 203–209. [CrossRef]
9. Niu, Q.Y.; Li, M.F.; Zhao, Y. Research on improved artificial potential field method for AGV path planning. *Mach. Tool Hydraul.* **2022**, *50*, 19–24.
10. Chen, D.; Liu, X.; Liu, S. Improved A*algorithm based on two-way search for path planning of automated guided vehicle. *J. Comput. Appl.* **2021**, *41*, 309–313.
11. Deng, X.; Hu, X.; Jiang, D.; Peng, Z. A hybrid GA approach to the scheduling of machines and automated guided vehicles in flexible job shops. *J. Sichuan Univ. (Nat. Sci. Ed.)* **2021**, *58*, 73–82. [CrossRef]
12. Liu, C.; Peng, T. Improved time window path planning for large-scale AGVs. *Appl. Res. Comput.* **2019**, *40*, 52–56. [CrossRef]
13. Guo, T.; Sun, Y.; Liu, Y.; Liu, L.; Lu, J. An Automated Guided Vehicle Path Planning Algorithm Based on Improved A* and Dynamic Window Approach Fusion. *Appl. Sci.* **2023**, *13*, 10326. [CrossRef]
14. Kong, H.; Sheng, Y. Research on AGV path planning based on improved A∗algorithm. *Mod. Eng.* **2021**, *10*, 60–64. [CrossRef]
15. Han, Z.; Wang, D.; Liu, F.; Zhao, Z. Multi-AGV path planning with double-path constraints by using an improved genetic algorithm. *PLoS ONE* **2017**, *12*, e0181747. [CrossRef] [PubMed]
16. Fan, H.; Mu, S.; Yue, L. Collaborative optimization of automated guided vehicle scheduling and path planning considering conflict and congestion. *J. Comput. Appl.* **2022**, *42*, 2281–2291.
17. Yu, H.; Bai, H.; Li, C. Research and simulation of path planning for warehouse multi-AGV system. *Comput. Eng. Appl.* **2020**, *56*, 233–241.
18. Yuan, Y.; Ye, F.; Lai, Y.Z.; Zhao, Y. Multi-agv path planning combined with load balancing and A* algorithm. *Comput. Eng. Appl.* **2020**, *56*, 251–256.
19. Deng, X.; Hu, X.B.; Jiang, D.Y. Flexible job shop machine and AGV planning based on hybrid genetic algorithm. *J. Sichuan Univ. (NSE)* **2021**, *58*, 73–82. [CrossRef]

20. Tian, S.; Fan, L. Multi-agv path planning for express distribution center under dynamic priority strategy. *Comput. Eng. Appl.* **2019**, *59*, 275–284.
21. Yang, Y.; Zhong, M.; Dessouky, Y.; Postolache, O. An integrated scheduling method for AGV routing in automated container terminals. *Comput. Ind. Eng.* **2018**, *126*, 482–493. [CrossRef]
22. Prandtstetter, M.; Seragiotto, C. Towards system-aware routes. In Proceedings of the International Conference on Computer Aided Systems Theory, Las Palmas de Gran Canaria, Spain, 19–24 February 2017; Springer International Publishing: Cham, Switzerland, 2017; pp. 291–298.
23. Yuan, Z.; Yang, Z.; Lv, L.; Shi, Y. A Bi-Level Path Planning Algorithm for Multi-AGV Routing Problem. *Electronics* **2020**, *9*, 1351. [CrossRef]
24. Zhao, J.; Meng, C.; Wang, X.; Hao, C.; Li, R.; Liu, H.; Wang, Z. Modeling and Analysis of AGV Grid Method Based on Feature Points Extraction. *Comput. Eng. Appl.* **2022**, *58*, 156–167.
25. Wang, Y.; Cao, X.; Guo, X. Warehouse AGV path planning method based on improved A* algorithm and system short-term. *Comput. Integr. Manuf. Syst.* **2023**, *29*, 3897–3908. [CrossRef]
26. Zhang, Y. AGV Path Planning Based on Improved Astar Algorithm. *Inf. Comput.* **2022**, *34*, 67–70.
27. Yang, J.; Tian, Z.; Liu, G. Collision-Free Path Planning for Multiple AGVs in Logistics Center. *Mach. Des. Manuf.* **2022**, *08*, 271–277, 281. [CrossRef]
28. Xu, L.; Huang, B.; Zhong, H. Time Window Model and Algorithm witt AGV System Path Planning. *J. Guangxi Norm. Univ. (Nat. Sci. Ed.)* **2019**, *37*, 1–8. [CrossRef]
29. Li, K.; Liu, T.; He, B.; Xu, D. A Study on Routing and Scheduling of Automated Guided Vehicle in "Cargo-to-Picker" System. *Chin. J. Manag. Sci.* **2022**, *30*, 240–251. [CrossRef]
30. Jiang, C.K.; Li, Z.; Pan, S.B.; Wang, Y. Collision-free path planning for AGVs based on improved Dijkstra algorithm. *Comput. Sci.* **2020**, *47*, 272–277.