Path Planning Based on Astar Algorithm in Mobile Robot

Zhenyang Peng *

School of Computer Science, BeiJing institute of technology, Beijing, China

* Corresponding Author Email: 20202330@stu.hebmu.edu.cn

Abstract. The food processing industry is one of the fundamental industries in our daily lives. In recent years, with the development of industrial robot technology, its application in food processing has become increasingly widespread, and the position of robots in industrial production has become higher and higher. At present, many factories choose to use robots to complete some assembly line work, while fully automated factories are widely used due to their high efficiency, reduced energy loss, and irreplaceable accuracy and sustained efficiency that cannot be replaced by manual labor. This trend is gradually increasing with the development of local path planning and other core robot technologies.In this context, we have decided to design a robotic arm mobile platform that can be applied to bakeries. For the fermentation process of bread, we use the mobile robot platform to complete the tasks of picking dough, placing it on the work loss line, and transferring it to the finished product placement point in a constant temperature environment. We even use the robotic arm to ensure that the bread is placed on the shelf from a high-temperature oven in a sterile environment before transportation, thereby saving labor costs, improving work efficiency, and ensuring the hygiene of the production environment. In addition, it is necessary to maintain the collection of surrounding obstacle information in complex indoor environments and improve the information processing ability of the obtained information, in order to plan the shortest path based on the layout of the bakery. In order to improve the efficiency of factory work, this work simulated the application scenario of a bakery and used Astar algorithm for path planning, aiming to ensure that the mobile platform of the robotic arm can complete its work while minimizing unnecessary consumption, collisions, and labor costs.

Keywords: Robot arm mobile platform, Astar algorithm, path planning.

1. Introduction

Most robotic arms can only perform one task, such as transportation, gripping, packaging, etc. However, it is evident that if fully automated mechanical factories are to be implemented, the demand for robotic arms will be high, which not only increases costs but may also lead to low work efficiency. Therefore, this robotic arm mobile platform can not only complete grasping and transportation, but also efficiently complete tasks such as packaging. Based on reducing energy consumption, path planning has become a key core technology for autonomous robot navigation. The purpose and significance of this study is to improve the work efficiency of factories and save labor costs. This type of car performs well in terms of effectiveness and long-term accuracy, especially in the widely used field of cargo transportation. For the application of automobiles, we simulated the application environment route of automobiles and modeled them. By using different algorithms to plan the path, we calculate the shortest driving route of the car to improve its efficiency. Obstacle avoidance planning is a very important part of local planning [1-3], which can greatly improve the intelligence level of autonomous driving. How to plan the shortest path for robots to avoid all obstacles remains a challenge in autonomous navigation research [4]. The current dynamic path planning algorithms can be divided into two categories: reaction-based obstacle avoidance algorithms and prediction-based obstacle avoidance algorithms [5, 6]. The reaction-based obstacle avoidance algorithm includes the artificial potential field method [7], which assumes that the robot is repelled by obstacles and attracted to the target point when moving in a virtual position, but may fall into local optima or oscillations in complex environments. In order to improve the efficiency of factory work, we simulated an application scenario of a bread factory and used the Astar algorithm for path planning, aiming to ensure that the mobile platform of the robotic arm can complete its work while minimizing unnecessary consumption and collisions [8]. The traditional A * algorithm combines the characteristics of breadth first search, namely the Dijkstra algorithm [9]. The experimental results show that under the path planning of the Astar algorithm, the robotic arm moving platform working in a bakery can achieve collision free motion from the starting point to the endpoint, and the path is the shortest, proving the stability and feasibility of the algorithm [10].

2. Method And Material

2.1. Simulated Application Scenario Map

This simulation diagram includes a constant temperature box, luggage rack, packaging production line, processing line, and finished product collector. The main task of a car is to transfer the fermented dough from the constant temperature rack and luggage rack to the processing line and packaging production line, and then place the packaged finished product on the finished product collector. In this process, not only should the shortest path planning be considered, but also the obstacle collision and human factors of the robotic arm moving platform should be considered. Therefore, the installation planning of sensors and other components should also be considered, as shown in Figure 1. Considering the limited area of the factory, different small cars need to be arranged for installation

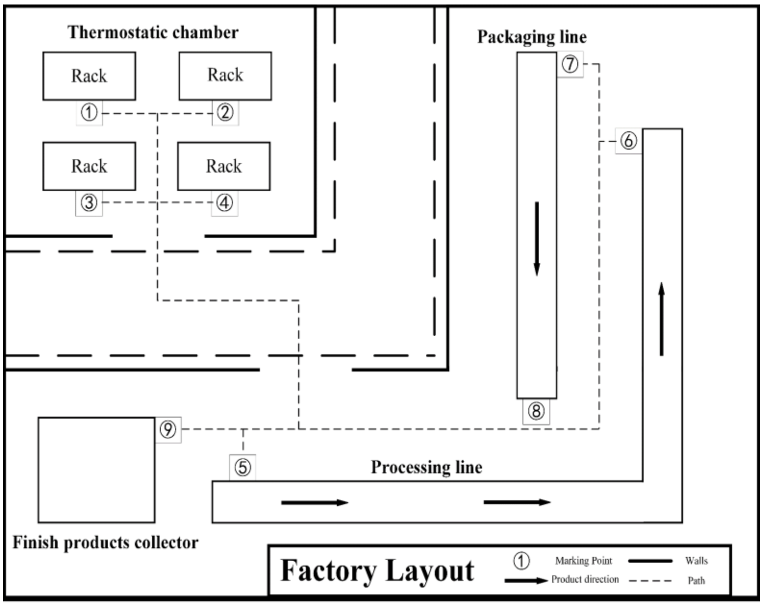

Fig 1. Planning of sensors and other components (Photo credited: Original).

For ease of testing, we have chosen to simplify it: From the four brackets in the upper left corner to the processing and assembly lines, only black squares are used to represent obstacles and destinations, providing a more intuitive view of the planned route for the car's movement (see Figure 2).

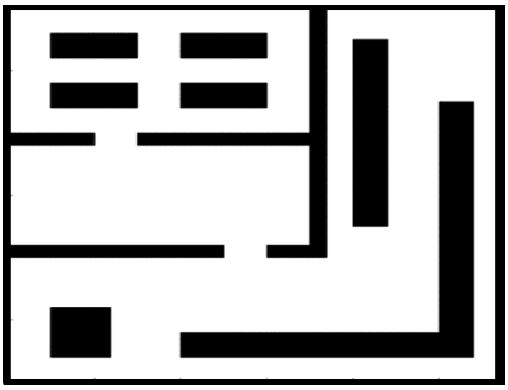

Fig 2. View of car movement (Photo credited: Original).

2.2. Calculation and Modeling of Robotic Arm Mobile Platform

We have designed and modeled the robotic arm mobile platform, as shown in Figure 3. This is a mobile robotic arm platform with five degrees of freedom, including the vehicle body, end effector, and other components.

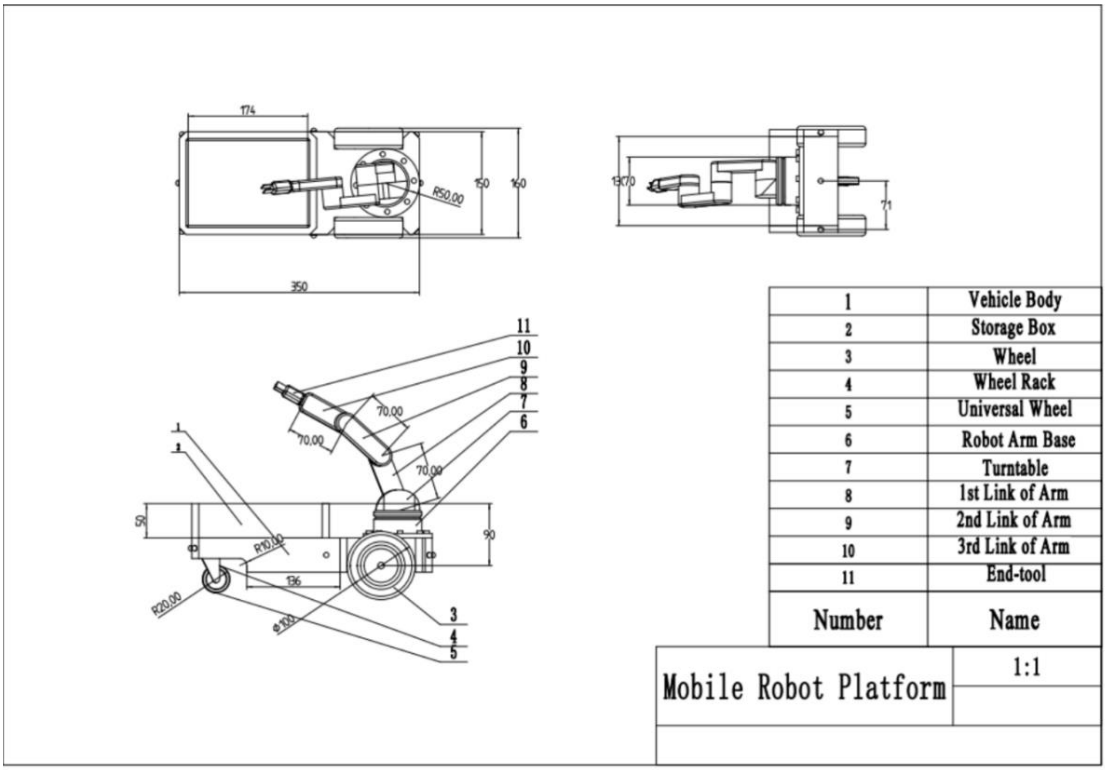

Fig 3. Robotic arm mobile platform (Photo credited: Original).

Under the simple modeling and calculation of the robotic arm moving platform, it can be seen that PID control consists of three parts: proportional term, integral term, and differential term.

$u(t)=K_pe(t)+K_i\int_0^t e(T)dT+K_d\, de(t)/dt$

In this project, as our car needs to be controlled at a certain point every moment, we can only use PD control, as shown in Figure 4.

$$u(t)=K_pe(t)+K_i+K_d de(t)/dt \tag{1}$$

Caculate the distance&angle:

$$d=\sqrt{(x-x_p)^2+(y-y_p)^2} \tag{2}$$

Caculate the distance&angle

$$\alpha=\arctan(y-y_p/x-x_p) \tag{3}$$

$$V_d=d/dt \qquad \theta_d=\alpha \tag{4}$$

Caculate the e (t) and de (t)

$$u = K_p * e(t) + K_d * de(t)/dt \tag{5}$$

$$v = v_{pre}+u*dt \qquad \omega=\omega_d \tag{6}$$

Control sample:
(One of our planed paths)
$K_p=1.2$, $K_d=0.01$

As shown in Figure 4, PID control has the characteristics of simple structure, good robustness, and high reliability. Compared with a path we prepared; it can be seen that this vehicle is being controlled to follow a point at every moment.

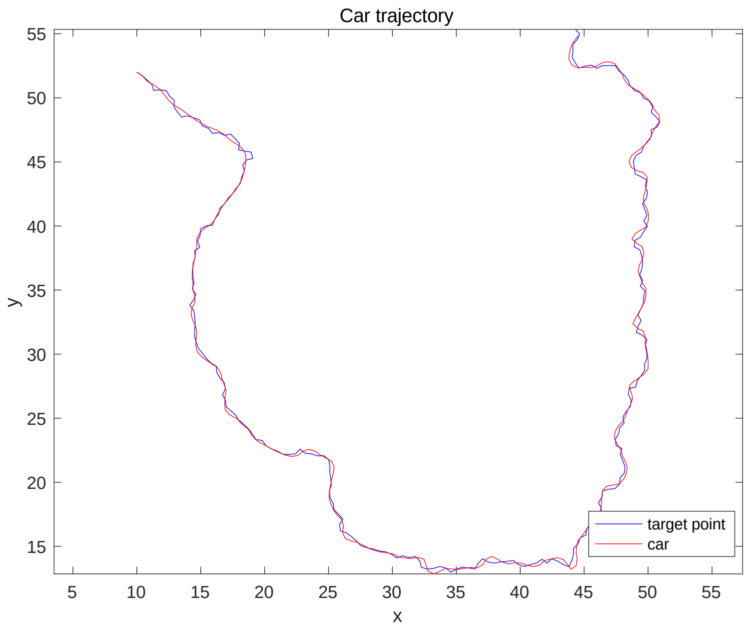

Fig 4. PID control results in this work (Photo credited: Original).

2.3. Astar Algorithm Method

Algorithm A draws inspiration from the Dijkstra algorithm and the best priority search algorithm. The core idea is to plan a minimum movement cost path from the starting position to the target position in a specified map. The estimation function f (n) for the movement cost from the starting position to the target position is f (n)=g (n)+h (n) (1), where n represents the current position node in the path search process; G (n) represents the true movement distance of the node from the starting position to the current position; H (n) represents a heuristic function from the current position node to the target position node, which estimates the movement cost [1]. The relationship between nodes and g (n) during the path search process is shown in Figure 5.

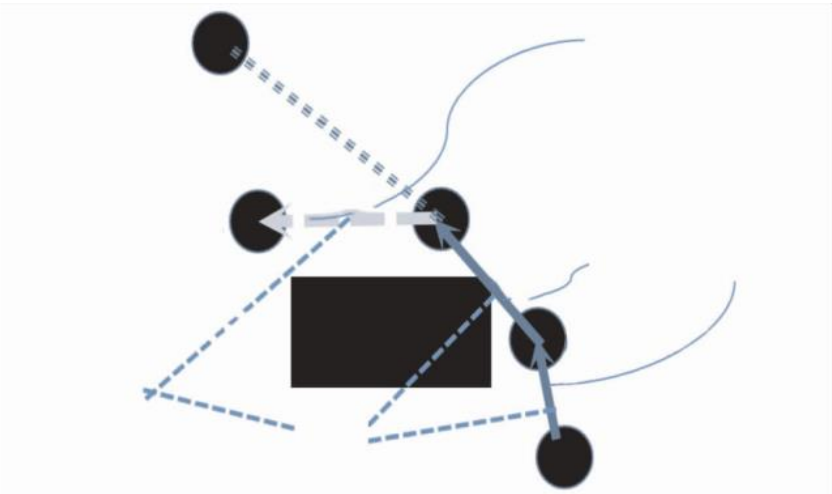

Fig 5. A Mobile robot planning based on improved A* algorithm [1].

Firstly, an initialization interface and button grid were created, and then a pop-up window with operation instructions was set to display. Next, create a class application, including methods for button operations, and then create a Tkinter application instance root and bind key events. Finally, start the main event loop.

For the output results, visualize them using the Tkinter library. First, determine the starting and ending points, using green and red to represent the starting and ending points respectively, and then using black to represent obstacles. I simulated obstacles on the map using a simple black grid, and then ran the code. The red route is the shortest path for the car from the starting point to the endpoint.

2.4. Program Analysis

This program defines a graphical interface application for path search based on the A * algorithm, built using the Tkinter library. It contains two main classes: 'Node' and 'AStarGUI'.

Node: Represents a cell (node) in the grid. Each node has its row and column positions, as well as some attributes, such as whether it is an obstacle (is_object), the cost from the starting point to the current node (g), the heuristic estimated cost from the current node to the target node (h), the total cost (f=g+h), and the parent node (parent).

Implemented the ‘lt’ method, which allows nodes to be compared based on the total cost ‘f’ and return the comparison result. This method prioritizes sorting in the ‘PriorityQueue’.

AStarGUI: Responsible for creating and managing graphical user interfaces, allowing users to set starting, ending, obstacles, and waypoints by clicking, and executing the A * algorithm to find the path.

In the initialization method, a Tkinter's' Canvas' object was created to draw the grid, and various buttons were used to trigger different setting modes (such as setting start, end, obstacles, etc.).

The 'handle_click' method updates the node's state or sets specific nodes (starting point, endpoint, etc.) based on the current state (set by the button).

The 'find_path' method is a key feature that is responsible for executing the A * search algorithm after the user has set all necessary points. This method sequentially searches from the starting point to the first midpoint, from the first midpoint to the second midpoint, and finally from the second midpoint to the endpoint.

The A_starSearch method implements the core of the A * search algorithm, using heuristic functions (Manhattan distance) and priority queues to find the shortest path from the starting point to the endpoint. The ‘draw_grid’ and 'draw_path' methods are used to draw grids, obstacles, special nodes (starting point, midpoint, endpoint), and final paths on the interface.

3. Results

Given the efficiency of this robotic arm mobile platform, we have decided to compare it with the output results of RRT *, as shown in Figure 6.

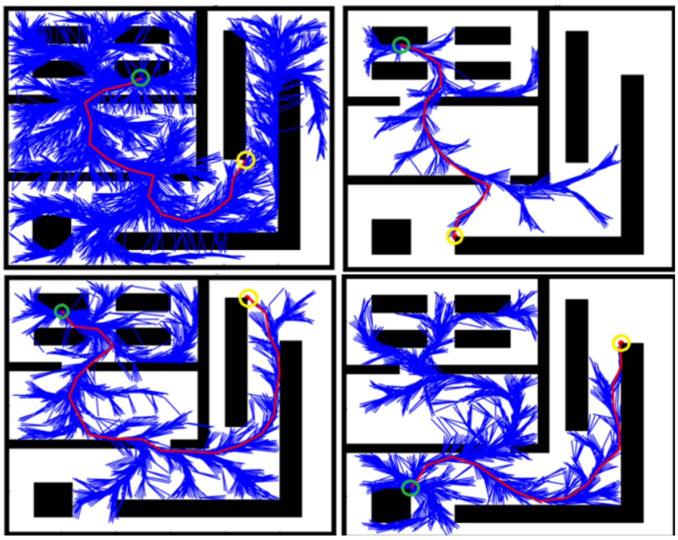

Fig 6. Output results of RRT * (Photo credited: Original).

It can be seen that the shortest route from the green starting point to the yellow ending point for this application environment has been calculated for different starting and ending points, simulating the process of mechanical mobile platforms picking up goods from shelves to the production line. It can be seen that the output results are significantly different from those of the Astar algorithm

On this basis, the Astar algorithm is used to plan the path from the starting point to the endpoint on a simulated map, and the pseudo code is shown in Figure 7.

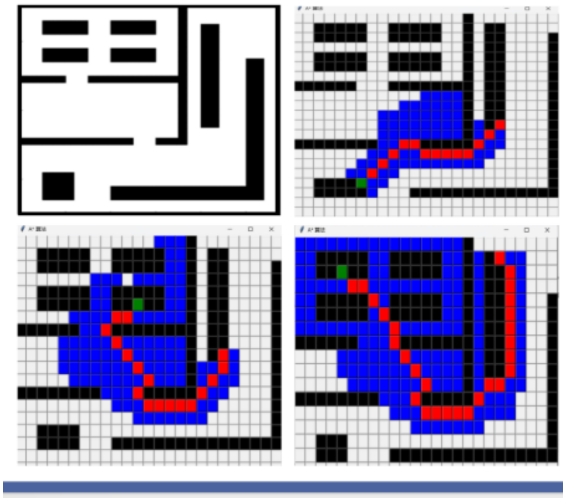

Fig 7. Output of Astar algorithm (Photo credited: Original).

4. Market Research

After studying and understanding the current applications of robotic arms entering the market, we found that the industrial robotic arm market size will exceed 14.5 billion US dollars in 2022, and with the development of industrial manufacturing and artificial intelligence technology, it is expected that the compound annual growth rate will exceed 12% by 2032. Most robotic arms can only perform a single task, such as transportation, grabbing, packaging, etc. However, it is evident that if a fully automated mechanical factory is to be implemented, the demand for robotic arms will be high, which not only increases costs but also may result in low work efficiency. Therefore, this robotic arm mobile platform can not only complete grabbing and transportation, but also efficiently complete packaging and other tasks.

According to data previously reported by The Wall Street Journal, 500000 industrial robots were installed globally in 2021 alone, which directly broke the record for 2018. And as of 2021, the total number of industrial robots worldwide has reached 3.5 million.

5. Conclusion

In order to improve the efficiency of factory work, we simulated the application scenario of a bakery and used Astar algorithm for path planning, aiming to ensure that the mobile platform of the robotic arm can complete its work while minimizing unnecessary consumption, collisions, and labor costs. In summary, the model of the bakery robotic arm mobile platform is mainly introduced. The path planning of the Astar algorithm is of great significance in the application field of industrial robots, enabling the robotic arm mobile platform to maintain efficiency in the working environment of the bakery, reduce energy loss and labor costs, and ensure the hygiene conditions for producing bread. But overall, there is a lack of more algorithm comparisons and improvements, as well as factors related to search sensors. This study even believe that deep learning and other technologies can be used to improve the mobile platform of the robotic arm so that it can independently cope with most of the possible situations that may occur in the bakery. However, overall, this algorithm has proven its feasibility and superiority in the field of robotics. Next, other algorithms can be combined to simulate the operation of robots quickly planning the shortest path in different dynamic environments.

References

[1] Shen Keyu, You Zhiyu, Liu Yongxin, Huang Tao. Mobile robot path planning based on improved A* algorithm. Southwest University for Nationalities, School of Electrical Engineering, Key Laboratory of State Ethnic Affairs Commission. Computer Application Research. 2023. 40(1):75-79.

[2] Liu Zhijun, Su Liang, Wu Shaoxiong. Development of Local Planning Method for Obstacle Avoidance Paths Based on A* Algorithm. Bus Technology and Research. 2023. 3:6-9.

[3] Wang Haiyao, a Tianyang. Robot Path Planning Based on Improved A* Algorithm and DWA Algorithm. Electronic production. 2023, 21:48-51.

[4] Yu Tengwei, Liu Changli, Research on Obstacle Avoidance Strategies for Mobile Robots in Dynamic Environments [J] Journal of Chongqing Jiaotong University (Natural Science Journal) 2021, 9:131-136.

[5] Steven M. LaValle. Rapidly-Exploring Random Trees: A New Tool for Path Planning. Iowa State University, Computer Science Department. 1998.

[6] Chen Qiulian, Jiang Huanyu, Zheng Yijun. A Review of Rapid Extended Random Tree Algorithms for Robot Path Planning. Computer Engineering and Applications. 2019, 16:10-17

[7] James J. Kuffner, Steven M. LaValle. RRT-Connect: An Efficient Approach to Single-Query Path Planning. 2000 IEEE International Conference on Robotics and Automation, ICRA 2000.

[8] Wang Bin, Nie Jianjun, Li Haiyang, Jie Xiaolin, Yan Hongzhen. Optimizing A* and Dynamic Window Method for Mobile Robot Path Planning. Computer Integrated Manufacturing Systems. 2022, 4 4:1743-1753

[9] Xianmengyu Design of a logistics distribution path optimization algorithm based on Dijkstra [J]. Electronic Design Engineering, 2023, 31(2):20-24.

[10] Design of Agricultural Robot Path Planning System Based on Improved A * Algorithm by Cheng Yonghong and Wang Meng in ournal of agricultural mechanization research. 2023, (3): 83-86.