An Stable Intelligence Medical Trolley Path Planning System Based on Improved Breadth First Search Algorithm

Xuesong Han

School of Mechatronic Engineering and Automation Shanghai University Shanghai, China

* Corresponding Author Email: hanxuesong@shu.edu.cn

Abstract. Intelligence medical trolley path planning system is a comprehensive intelligent system that integrates many functions such as environmental perception, decision planning and motion control. Due to the characteristics of low speed, simple operating conditions, and priority information, it is unmanned one of the first scenes that can be achieved, with high research value. Although Intelligence Medical Trolley System has been widely developed and developed, many problems still need to be resolved.For example, the search path is not the optimal path resulting in reduced efficiency, or the algorithm is not stable enough, causing the trolley to deviate from the running path.The Path Planning algorithm is the core content of the robot. This article focuses on the improvement of path planning algorithm and determine the feasibility of the algorithm through simulation experiments and optimize. The feasibility of the algorithm is confirmed and optimized by simulation experiments. Finally, through the infrared sensor and the physical sign combine the reality with the algorithm results, and verifies the path planning algorithm with the mobile robot entity. This design uses c++ as the development language, and then uses Arduino as the development platform, lets the mobile robot move according to the path which produced by the algorithm. The experimental results show that the system based on Breath First Search (BFS) algorithm is feasible and can be applied to the working environment of hospital.

Keywords: Path planning, trolley, intelligence, hospital.

1. Introduction

With the general trend of population aging and the emergence of the new crown epidemic, the hospital's demand for medical staff is getting higher and higher [1]. They are busy at work and often have physical overdrafts. If you introduce Intelligence Medical Trolley, you do not need medical staff to push a particularly heavy medical vehicle, or even medical staff to send drugs to the corresponding ward. Spray the disinfection solution to prevent the virus from spreading. The promotion of Intelligence Medical Trolley can reduce the burden on hospitals and medical staff, reduce the risk of infectious disease transmission, and improve the work efficiency of hospitals.

Medical path planning technology is one of the cores of the research of robotics. Path planning refers to in accordance with certain evaluation standards (such as selecting the shortest planning path, the shortest planning time, or the smallest work price, etc.), in the environmental map with obstacles, find a viable path from the starting point to the destination and avoid obstacle [2]. The collision path is a global static planning algorithm [3-55]. The classic path planning algorithm has BFS, DFS, Dijkstra algorithms, and A*algorithms [6-9]. This article proposes a BFS -based path planning solution based on the actual application scenarios of the hospital. Based on the Arduino development board design, the cost is low, effective and reliable [10].

The innovation point of this paper is to design a complete path planning system which can be used in hospital intelligent medical trolley based on BFS algorithm for practical application scenarios in hospitals, which is expected to reduce the burden of doctors and improve the work efficiency of hospitals.

2. Method

The process of robot path planning is divided into three main and key steps.

The first is to model the map data parameters related to the known map information, obstacle points, feasible points, weighted values and other related map data parameters. Secondly, choose the right algorithm for research, and optimize the actual optimization according to the tasks required by the robot. Essence finally, according to the improved path planning algorithm, find a non-collision shortest path from the stating location of the machine to the destination position for the robot. The robot seids the path according to the path information and the physical parameters configured.

In the Path Planning section, this work hope that the car can be go from one room to another to complete the function of sending medicines to the patient and transporting medical appliances to doctors.

Intelligence Medical Trolley path planning system use BFS algorithm to plan the path. BFS is a traversing algorithm where start traversing from a selected piont (starting piont), traverse the map layerwise thus exploring the neighbour piont. when the Simulation trolley find the destination node, it will search for the father node and the father' node, and step by step, until finding the starting point. This method will combine these nodes, and it is the shortest path. Figure 1 showed the principle of BFS.

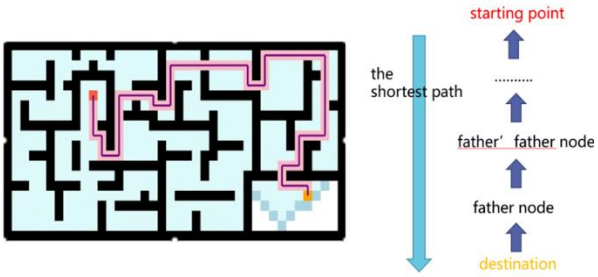


Fig. 1. The principle of Breath First Search (Picture credit: Original)

3. Based on the hospital environment improvement BFS

Use BFS to find a way should first enter the map and tell Trolley Road and the location of the room. In Figure 2, the map can be converted into a two-dimensional array, with non-negative numbers representing room numbers, -1 representing road, trolley can pass in these coordinates, and -2 representing obstacles such as walls, trolley cannot pass.



Fig. 2. Transform the map into a two-dimensional array (Picture credit: Original)

Then trolley uses BFS to calculate the optimal path based on the departure and destination. Figure 3 is the coordinate of path.

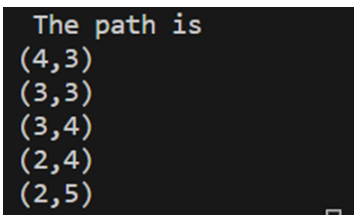


Fig. 3. The path from 0 to 9(Picture credit: Original)

Next, convert the coordinates of the path into a signal that the trolley can understand in the real world.

Stick the rail line on the ground, and the Trolley walks on the rail line. In the path planning system, trolley use TCRT5000 to detect Rail Line. TCRT5000 is an infrared launch and receiver, which continuously launch and receive infrared rays. Black does not reflect infrared rays, but white does it.So trolley can use TCRT5000 to distinguish the track. Figure 4 is the picture of the TCRT5000.

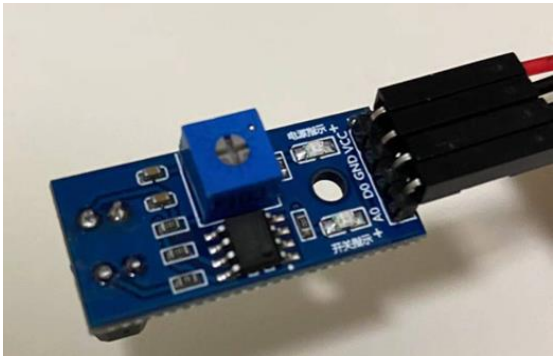


Fig. 4. Six choices of turning around(Picture credit: Original)

It can be seen in Figure 5. There are two sensors on both sides of the Trolley, then hospital ca n uses the black and white rail to guide the trolley. When the trolley needs to turn around, there are 3 situations. In these 3 situations, there are six choices. So it can be used six kinds of code for six cases to turn around.

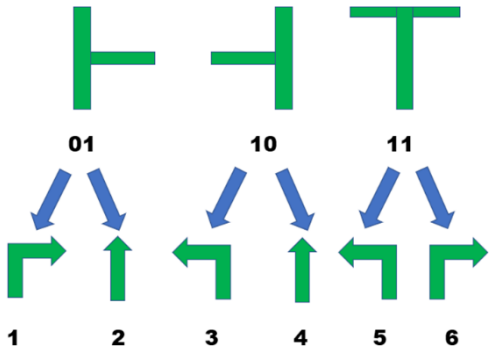


Fig. 5. Six choices of turning around (Picture credit: Original)

The trolley can set up the Queue of Selecting direction according to the coordinate position of the route. For example, in the route below, Trolley should turn right first, turn left, go straight, and then turn right. Figure 6 is the steps of prouding the queue of selection.

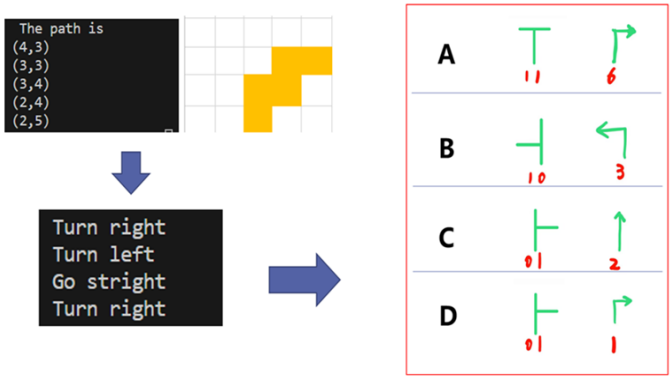


Fig. 6. Steps from path to queue of selection(Picture credit: Original)

Through the above steps, the trolley can realize the automatic way to path planning, complete the functions of helping doctors and patients transport medicine, medical equipment, etc.

4. Simulation Test

4.1. Simulation Results Test

First build different maps, 25*25, 50*50, 100*100, 200*200, 400*400 sets of data each. This map size is enough to describe the road environment on a certain floor of most hospitals. The map data, and the starting point are stored in the text document, the command program reads the data, operation, and the shortest path and steering strategy are finally output. Results showed that the BFS -based path planning algorithm can accurately plan the nearest route.

4.2. Time Test for Algorithm Running

Because the BFS algorithm will conduct a multi -layer traversal search of the map, the time which is produed by algorithm running is a problem that cannot be ignored. The following is the test of the running time of the algorithm in different situations. The SIZE was read and calculated by the SIZE to 25*25, 50*50, 100*100, 200*200, 400*400 sets of map data each, and finally output the results. Take the Time obtained 10 times per data, remove the maximum and minimum values, and take the average number as the final data source. The table below is the statistics of the experiment.

TABLE I shows that as the map size increases, the algorithm running time will increase. When the size is 400*400, the algorithm runs the longest, reaching 1.537 seconds. In other SIZE, the operation time of the algorithm is shorter Essence For the working environment of the hospital, Trolley is mainly used for transportation items, and there is no particularly strict requirements. Compared with a few minutes of transportation time, it is acceptable to calculate the path with about 1.5 seconds.

Tab. 1 Time Test for Algorithm Running

Maps size	Time（s）
25*25	**0.0268**
50*50	**0.0662**
100*100	**0.2281**
200*200	**0.6350**
400*400	**1.4832**

5. Implementation of Intelligence Medical Trolley path planning system

5.1. Development Environment

Trolley uses Arduino Uno as the development motherboard. As shown in the figure, the CPU is Atmega328P, the operating system is Windows11, the development platform is Arduino IDE, and the development language is C., Trolley is equipped with two DC motors to control the movement of the wheels. Arduino UNO's output of serial ports, control the movement direction and advance speed of DC motors.

5.2. Implementation

First, convert the map to a two -dimensional array that can be processed by algorithms, and then the trolley will be controlled according to the results obtained by the algorithm. According to the path of the two -dimensional plane obtained by the algorithm, the direction is selected. Here is a example about the PSEUDO-CODE selected left-turning direction.

```
#define x = 1， y = 2；
//Define x and y coordinates to store path positions
for (int i = 0; i <= step - 2; i++)
{
    if ( ( path[i][y] + 1 == path[i+1][y] ) && ( path[i+1][x] - 1 == path[i+2][x] ) )
      {
```

```
            Turn left（）;
        }
        if ( ( path [i][y] - 1 == path [i+1][y] ) && ( path [i+1][x] + 1 == path [i+2][x] ) )
        {
            Turn left（）;
        }
        if ( ( path [i][x] - 1 == path [i+1][x] ) && ( path [i+1][y] - 1 == path [i+2][y] ) )
        {
            Turn left（）;
        }
        if ( ( path [i][x] + 1 == path [i+1][x] ) && ( path [i+1][y] + 1 == path [i+2][y] ) )
        {
            Turn left（）;
        }
}
```

Figure 7 shows there are four judgments correspond to the four cases of the figure, of which the number 1, 2, and 3 represent the order of the action of Trolly.

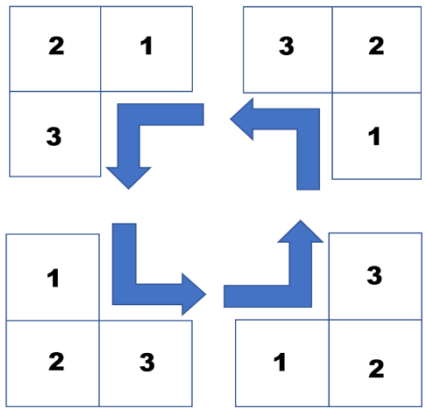


Fig. 7. Steps from path to queue of selection

By comparing the relative position of the path before and after, you can get the selection of other directions, and finally generate The Queue of Selecting Direction.

Through The Queue of Selecting Direction and the steering program written in advance, the robot is controlled, turning left, right, and stopping.

The following are PSEUDO-CODE which control the robot's forward, turn right, and turn left.

```
void go straight ()
{
   if (Sensor [left] == black && Sensor [right] == white )
     { Motor_Speed(100, 100);
     delay(1000);
     }//choice 2,go straight
if (Sensor [left] == white && Sensor [right] == black )
     { Motor_Speed(100, 100); //go straight
     delay(1000);
     }//choice 4 ,go straight
   else
      Motor_Speed(100, 100);
}
void turn right ()
{
```

```
            if (Sensor [left] == black && Sensor [right] == white )       { Motor_Speed(150, 0); //left
speed 150, right speed 0;
        delay(1000);
        }//choice 1, turn right
   if (Sensor [left] == black && Sensor [right] == black )         { Motor_Speed(150, 0); //left speed
150, right speed 0;
        delay(1000);
       }//choice 6, turn right
       else    //go straight
          Motor_Speed(100, 100);
   }
   void turn left ()
   {
      if (Sensor [left] == white && Sensor [right] == black)      { Motor_Speed(0, 150); //left speed
0, right speed 150;
        delay (1000);
        }//choice 3, turn left
   if (Sensor [left] == black && Sensor [right] == black)      { Motor_Speed(0, 150); //left speed 0,
right speed 150;
        delay(1000);
        }//choice 5, turn left
       else    //go straight
          Motor_Speed(100,100);
   }
```

It can set the corresponding forward speed and time according to the actual needs of the hospital and the ground.

5.3. Physical Demonstration

This experiment prepared a map of plastic cloth material of 3 m × 3 m and painted the black solid trajectory to different places. Essence The map is unfolded on the open space. In the case of keeping flat, use Trolley to go from starting point to destination. Figure 8 is the model for physical verification.

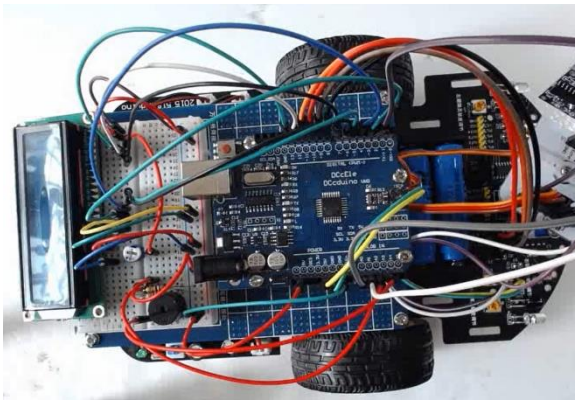


Fig. 8. Physical simulation of the Intelligence Medical Trolley(Picture credit: Original)

The body with an infrared sensor TCRT5000, enter the starting point, end point, the car can generate The Queue of Selecting Direction based on the improved BFS algorithm, and arrive at the end.

6. Conclusion

This article designs An Intelligence Medical Trolley Path Planning System Based on Improved Breadth First Search Algorithm.

Through experiments, the modified BFS algorithm system shows it can effectively and reliable planning paths. Based on this, this article generates the Queue of selection direct through the path, using the infrared sensors on Track and Trolley to achieve Trolley turning. Finally, testing in the physical environment proves the feasibility of this Intelligence Medical Trolley Path Planning System.

According to the path planning system, trolley can automatically transport medicines, medical equipment, etc. The system can be applied to reduce the work burden of doctors and nurses and provide more timely services to patients.

But the system designed in this paper, trolley running relies on the black track of the ground. If it is damaged by other factors, the running of trolley will also be affected.

In the future, more advanced positioning systems can be used to help trolley do path planning and improve the system's ability to resist environmental interference. In addition, the path planning algorithm can be further optimized. For example, in the face of multiple transportation tasks, how to calculate the optimal path and complete multiple tasks at one time. After the algorithm improvement, the working efficiency of trolley can be further improved.

References

[1] M. Nicole, M. Kathleen J, and P. David. “The Effect of Population Aging on Economic Growth, the Labor Force, and Productivity" American economic journal: macroeconomics, vol. 15, no. 2, pp. 306-32, April 2023

[2] X. Guo, M. Zhou, S. Liu and L. Qi, "Lexicographic multiobjective scatter search for the optimization of sequence-dependent selective disassembly subject to multiresource constraints", IEEE Trans. Cybern., vol. 50, pp. 3307-3317, Jul. 2020.

[3] J. Han and Y. Seo, "Mobile robot path planning with surrounding point set and path improvement", Appl. Soft Comput., vol. 57, pp. 35-47, Aug. 2017

[4] W. Wu, S. Bhattacharya and A. Prorok, "Multi-Robot Path Deconfliction through Prioritization by Path Prospects", IEEE International Conference on Robotics and Automation, 2020.

[5] A. Bircher, M. Kamel, K. Alexis and H. Oleynikova, "Receding horizon path planning for 3D exploration and surface inspection", Auton. Robots, vol. 42, no. 2, pp. 291-306, 2018.

[6] A. H. Qureshi and Y. Ayaz, "Potential functions-based sampling heuristic for optimal path planning", Auton. Robots, vol. 40, no. 6, pp. 1079-1093, Nov. 2015.

[7] Y. Qian, F. Wang, J. Li, L. Shi and K. Cai, "User association and path planning for UAV-aided mobile edge computing with energy restriction", IEEE Wireless Commun. Lett., vol. 8, no. 5, pp. 1312-1315, Oct. 2019

[8] C. Zhong, W. Chen. “ Research on Hybrid path Planning Algorithm for Mobile Robots. ” Automatic Instrumentation, vol. 42, no. 9, pp. 61-66, 2016.

[9] Q. Jia, X. Wang, "An improved potential field method for path planning", Proc. Chin. Control Decis. Conf., pp. 2265-2270, May 2010.

[10] M. Rickert, A. Sieverling and O. Brock, "Balancing exploration and exploitation in sampling-based motion planning", IEEE Trans. Robot., vol. 30, no. 6, pp. 1305-1317, Dec. 2014