ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

Research on Trajectory Optimization of Upper Limb Rehabilitation Robot

Yuting Jin[1], Jiaxin Li[2]

[1]School of Mechano-Electronic Engineering, Xidian University, Xi'an, Shannxi, China
[2]CSUST International Institute of Technology, Changsha University of Science & Technology, Changsha, Hunan, China

Abstract: *For the current wide application of robots in the field of rehabilitation, in order to improve the auxiliary rehabilitation effect of upper limb movement disorder patients in the rehabilitation period and enhance the security of patients during use, this paper proposes a method for trajectory planning and optimization of upper limb rehabilitation robot, which can achieve good and safe auxiliary rehabilitation effect by modifying the tangent offset parameters. In order to solve the problem of redundant inflection points and poor smoothness of traditional A Star Algorithm, a hierarchical progressive algorithm optimization strategy is proposed, such as collision detection algorithm and dynamic tangent smoothing algorithm with variable tangent points. By using this strategy, the technical problems such as the judgment and deletion of redundant inflection points and the excessive corner of poor convergence of planned path are solved. The accuracy of optimal path judgment is significantly improved, and the safety performance and auxiliary rehabilitation effect of upper limb rehabilitation machine are significantly improved. In this paper, the kinematics modeling of upper limb rehabilitation robot is carried out by MATLAB, and the trajectory simulation of the robot is carried out, and a trajectory optimization scheme based on A Star Algorithm is proposed. The generated path conforms to the trajectory of the rehabilitation operation, and can effectively assist the rehabilitation movement, which provides a basis for kinematics.*

Keywords: *Rehabilitation robot; Trajectory planning; A Star Algorithm*

1. Introduction

In recent years, multi-technology integration has given robots a strong spillover effect, and robot-related technologies have developed rapidly and have received widespread attention [1]. The use of robots is no longer limited to the industrial field, but has begun to develop in the field of human-machine collaboration. The improvement of machine intelligence and the wide application of technologies such as voice interaction and image recognition have brought new development opportunities to human-machine interaction. The application field of robots has gradually expanded from manufacturing to medical and other advanced fields. With the continuous development of big data, artificial intelligence, new materials and other technologies as well as the close integration with the medical field, the industrial opportunities of medical robots will be greatly improved. The combination of artificial intelligence, automation and artificial intelligence has become a new trend in technological development [2]. In order to achieve better human-robot interaction and improve the flexibility and participation of human-robot collaboration, better motion performance and control of robots is required [3].

The traditional rehabilitation training method is mainly assisted by medical staff, and patients use simple instruments to drive the affected extremities for auxiliary training. Due to excessive physical consumption and the influence of subjective factors of the physiotherapists, it is difficult to ensure the intensity of rehabilitation training requirements, the persistence of endurance and the standardization of training effect. The emergence of rehabilitation robots has solved the shortcomings of traditional rehabilitation methods and improved the quality of rehabilitation [4]. The upper limb rehabilitation robots can be divided into two types: the end-guided type and the exoskeleton type. The end-guided type drives the hand movement through the end-effector to realize the rehabilitation training of each joint of the upper limb. The exoskeleton type can be worn to the upper limb of the human body to accurately control each joint [5]. The structure of the terminal traction rehabilitation robot is simple, and there are many research results at home and abroad, such as MIT-Manus upper limb rehabilitation robot developed by MIT and a 2-DOF upper limb rehabilitation robot system developed by Tsinghua University. Patients can

ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

perform rehabilitation training according to the preset trajectory under the traction of the robot [6-7]. LIU et al. proposed to use MATLAB to model the upper limb rehabilitation robot, import it into ADAMS for kinematic simulation and planning the end trajectory, and generate a smooth trajectory that can meet the needs of human training [8]. YANG et al. proposed to use five polynomial fitting to obtain the driving function of each joint of the rehabilitation robot and carry out kinematic simulation, so as to verify the correctness of the geometric model and mathematical model of the robot, which laid the foundation for the subsequent trajectory optimization and dynamic analysis [9]. XIA et al. proposed the establishment of human-computer interaction safety space, and based on the theory of human joint kinematics in rehabilitation medicine, the linear displacement of the end was achieved by using the pneumatic proportional servo control, which better met the requirements of occupational therapy in rehabilitation medicine [10].

At present, in view of the poor smoothness and large corner of the end trajectory of the upper limb rehabilitation robot, in order to develop a reasonable rehabilitation plan, this paper improves the traditional A Star Algorithm, and combines the collision detection algorithm and the dynamic tangent point smoothing algorithm to plan the end trajectory of the robot. The end trajectory of upper limb rehabilitation robot is simulated and analyzed by Matlab, and the trajectory path with good cohesion and curvature continuity is obtained, which increases the flexibility and safety of rehabilitation training to better meet the needs of human motion training.

2. Robot trajectory planning

2.1. Robot kinematics modeling

The kinematics of the robot is to study the relationship between the angle change of each joint and the pose of the robot end, including forward kinematics and inverse kinematics. The forward kinematics is a process of expressing the pose equation of the robot end by changing the angles of each joint through multiple homogeneous transformations. Inverse kinematics is the process of obtaining the angle change of each joint by multiplying the inverse matrix of the adjacent two-link transformation matrix of the equation continuously left by the positive kinematics. Whether the robot connecting rod structure is complex or not, using D-H modeling method is a more general and suitable choice. The D-H representation method is used to describe the relationship between translation and rotation between adjacent bars. It establishes a 4×4 homogeneous transformation matrix for the bar coordinate system at each joint, indicating its relationship with the previous bar coordinate system. In this way, the base coordinate can be used to represent the pose of the robot end. In this paper, a six-degree-of-freedom upper limb rehabilitation robot is taken as an example. The standard D-H method is used to describe the pose transformation between connecting rods with homogeneous transformation matrix, and the kinematics equation is established. The Schematic diagram of robot D-H modeling method is shown in Figure 1 below.

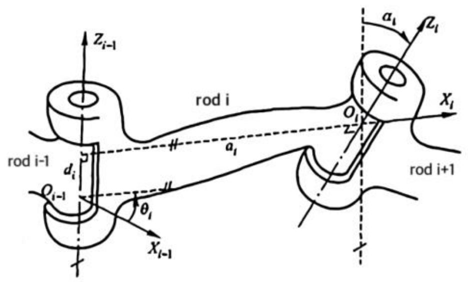

Figure 1: Schematic diagram of robot D-H modeling method

There are four parameters of D-H method: a is the distance from Zi-1 to Zi, that is, the distance between the common normal of the two rotating axes; α is the angle from Zi-1 to Zi, namely the angle perpendicular to the two rotation axes in the plane where a is located; d is the distance from Xi-1 to Xi, namely the distance between the two connecting rods; θ is the angle from Xi-1 to Xi, that is, the angle between two connecting rods. The homogeneous coordinate transformation matrix of adjacent connecting rod i-1 and connecting rod i is shown in formula (1).

$$^{i-1}T_i = Trans_z(d_i)Rot_z(\theta_i)Trans_x(a_i)Rots_x(\alpha_i)$$
$$= \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

2.2. Kinematics position solving

The kinematic analysis of the upper limb rehabilitation robot is to express the spatial displacement of the robot analytically as a function of time, that is, the relationship between the joint variable space of the rehabilitation robot and the position and attitude of the robot end point. Robot kinematics analysis is the premise of robot control, including forward kinematics analysis and inverse kinematics analysis. The forward kinematics analysis is to solve the pose of the end-link coordinate system relative to the base coordinate system by giving the joint angle variable of the robot; the inverse kinematics analysis is to solve the angle variable of each joint for a given robot end pose. For a six degree of freedom manipulator, if the axes of three adjacent joints intersect at one point or are parallel to each other, the closed solution can be obtained. The axes of the first three joints of the 6 - DOF upper limb rehabilitation robot studied in this paper intersect at a point, and the closed solution can be obtained by using the geometric method and the separation variable method, and then the inverse kinematics solution can be solved. The geometry of inverse kinematics is shown in Figure 2 below.

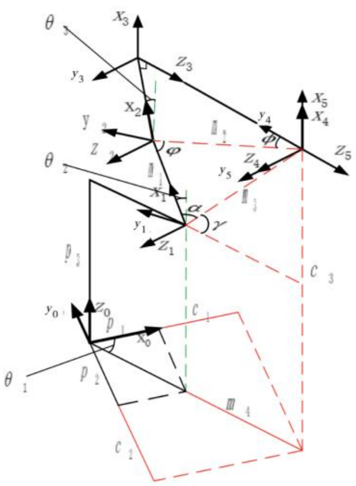

Figure 2: Geometric diagram of inverse solution of robot kinematics

2.3. Trajectory Planning Method

A Star Algorithm is a heuristic-based path planning algorithm, which has faster computing speed than other traditional algorithms. A Star uses g(n) to represent the distance cost from the starting node to any node n, and h(n) to represent the heuristic evaluation distance cost from node n to the target node. The quantitative trade-off between the advantages and disadvantages of nodes in the path is realized by constructing the evaluation function

$$f(n) = g(n) + h(n) \tag{2}$$

Through the optimal selection of extended nodes, this algorithm makes the number of extended nodes less obvious, reduces the path search range, and improves the retrieval efficiency of the algorithm. In practical application, due to the heuristic function of A Star Algorithm, the search of A Star Algorithm is directional. In some specific cases, this directionality will lead to the generation of redundant inflection angles, so that the generated optimal path has a poor accuracy. In the case of small grid map, the A Star Algorithm may have too large corners and poor smoothness of the generated path due to the construction principle. In view of the above problems, this paper puts forward the search path optimization strategy of A Star Algorithm. Through the collision detection algorithm and the dynamic tangent point smoothing algorithm, the deletion of redundant inflection points and the improvement of path smoothness are

ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

realized, so that the operation efficiency of the improved A Star Algorithm and the accuracy of the shortest path decision are significantly improved.

Figure 3 is the schematic diagram of A star algorithm trajectory planning. In this figure, the dark gray grid represents obstacles, the red grid is the target node, the yellow grid represents the optimal path searched, and the purple grid represents the node in Close List except the optimal path searched by A star algorithm.

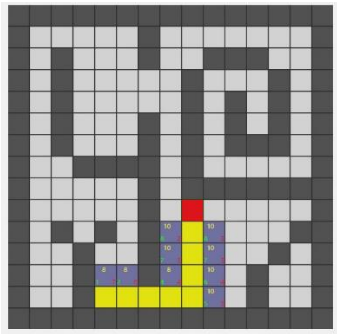

Figure 3: Schematic diagram of algorithm trajectory planning

2.4. Elimination of redundant inflection points

In practical application, due to the heuristic function of A Star Algorithm, the search of A Star Algorithm is directional. In some specific cases, this directionality will lead to the generation of redundant turning angles, so that the accuracy of the generated optimal path is not high. In order to solve the above problems, this paper uses the collision detection algorithm to identify the collision points in the path, and realizes the elimination of redundant inflection points.

The path of the traditional A-star algorithm is extracted, and the number of all nodes in the path is n. The starting point is denoted as the No.0 node of the path planning. The i node and the i+2 node of the initial planning are taken out to form a line segment a (i= 1, 2, ..., n-2). All obstacles in the map are extracted to obtain the four vertex coordinates of each obstacle, and the four vertices are connected to form four-line segments to determine whether the line segment a intersects the four-line segments of any obstacle. If no collision is proved, delete the i+1 node in the path; if intersection is proved to have collisions, retain the i+1 node in the path. Repeat the above steps to determine whether the path formed by the n-2 node and the n node of the initial planning collides with the obstacle. Connect the remaining nodes in turn to form a path to eliminate redundant nodes, the flow chart is shown in Figure 4.

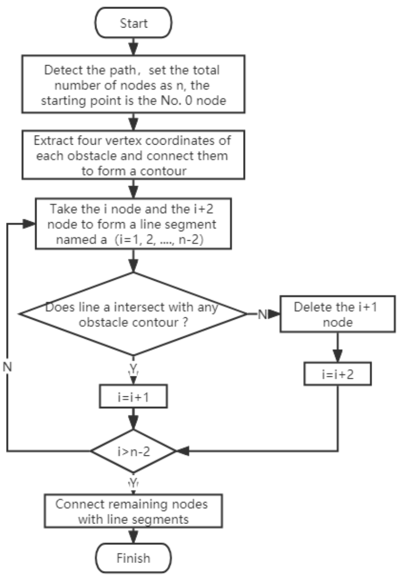

Figure 4: Flow chart of eliminating redundant inflection points

ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

2.5. Dynamic tangent smoothing algorithm to improve path smoothness

When the raster map is small, the A Star Algorithm may have the problems of too large corners due to the construction principle, which leads to poor smoothness and poor cohesion of the generated path. In this paper, the dynamic tangent smoothing algorithm is used to remove the concave and convex points of the planned path. By using this method, the path with good smoothness and curvature continuity can be obtained. The improved A Star Algorithm significantly improves the running efficiency and the accuracy of the shortest path determination.

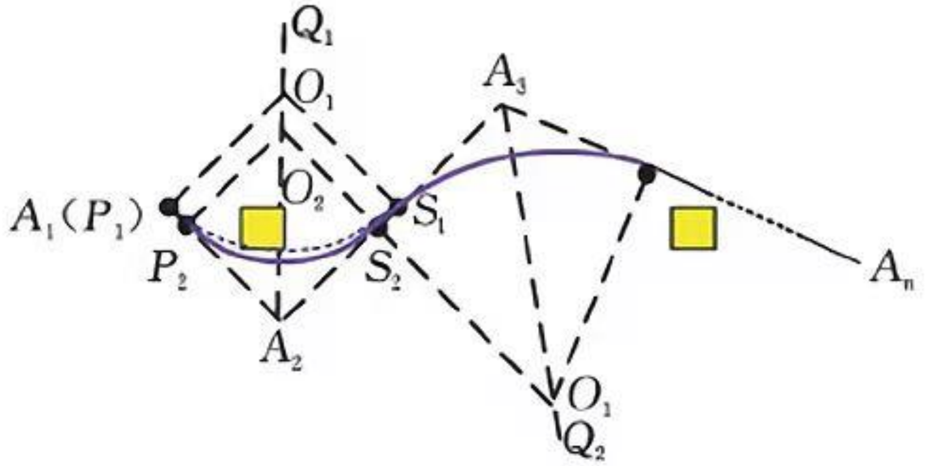

Figure 5: Schematic diagram of dynamic tangent smoothing algorithm

As shown in Figure 5, the initial position of the robot is $A_1(x_1, y_1)$ and the end position is an (x_n, y_n). Starting from the initial position, smooth the turning point of A_i (x_i, y_i) (i = 2, 3……, n-1) in turn. In the figure, monotonously using the fixed tangent point will cause the robot to fall into the dead zone containing obstacles. Therefore, this paper changes the fixed tangent point into a dynamic tangent point and proposes a dynamic tangent point adjustment algorithm.The specific steps are as follows.

Compare the lengths of two-line segments A_0A_i and A_iA_{i+1}, and select the end point P (x_p, y_p) of the shorter side as the initial tangent point. Draw a vertical line through the point P, and intersect the angle bisector A_iQ_{i-1} of $\angle A_{i-1}A_iA_{i+1}$ (i= 2, 3,……, n-1) at the point $O_{i-1}(x_0, y_0)$:

$$x_0 = (x_p + k_{01}y_p + k_{01}(k_0x_2 - y_2))/(1 + k_0k_{01}) \tag{3}$$

$$y_0 = k_0(x_0 - x_2) + y_2 \tag{4}$$

The radius R of the tangent circle can be expressed as

$$R = \sqrt{x_0^2 - 2x_0x_p + y_0^2 - 2y_0y_p + x_p^2 + y_p^2} \tag{5}$$

The tangent circle equation is

$$(x - x_0)^2 + (y - y_0)^2 = R^2 \tag{6}$$

In the formula, k_{01} is the slope of the shorter side, and k_0 is the slope of the angular bisector.

Then through a series of judgments and selection of the value of α, the entire algorithm can be implemented. The flowchart of the algorithm is shown in Figure 6.

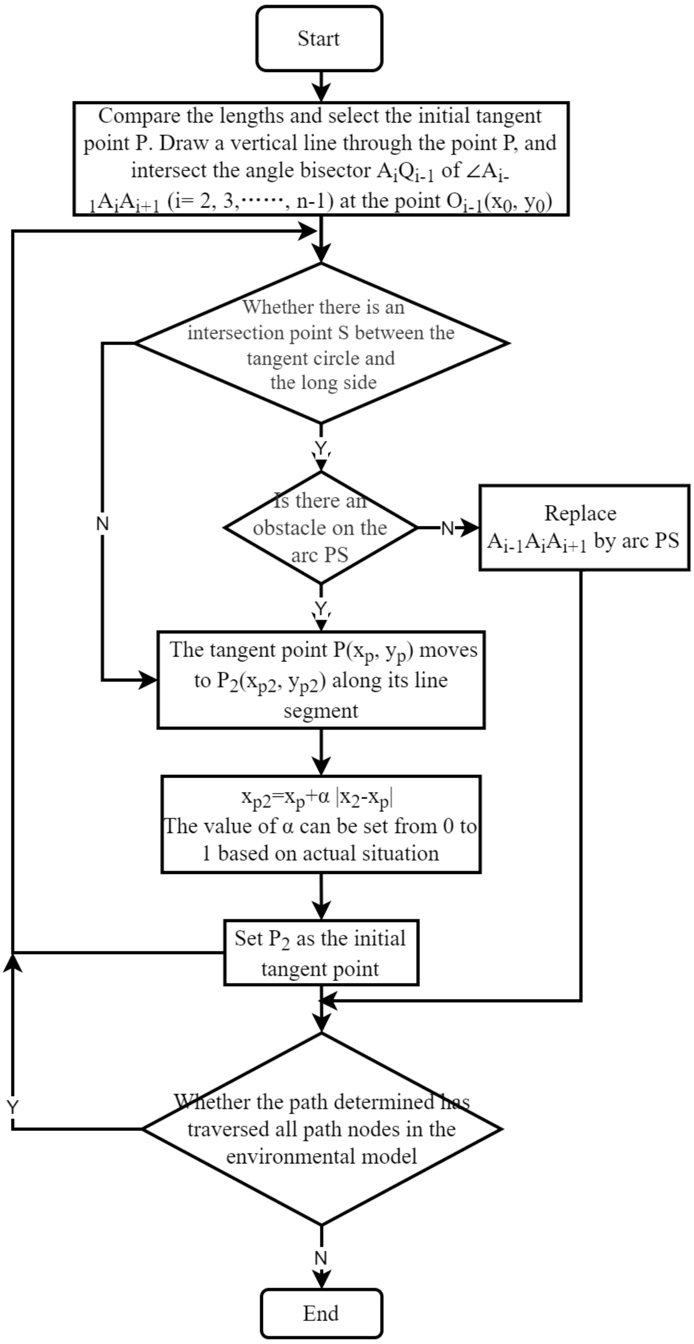

Figure 6: Flowchart of the dynamic tangent smoothing algorithm

3. Experimental results and analysis

3.1. Inverse verification of kinematics solution

No matter what kind of rehabilitation period patients belong to, rehabilitation robot assisted patients with rehabilitation training need to develop a certain trajectory to drive patients to exercise, or provide reference function. In this paper, the six-degree-of-freedom robot is modeled by MATLAB Robotic Toolbox, and the ikine function is used to solve the inverse kinematics of each path point of the manipulator. The jtraj function is used to plan the trajectory by quintic polynomial interpolation. The angular displacement curve, angular velocity curve and angular acceleration curve of each moving joint in the working process of the robot are shown in Figure 7.

ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

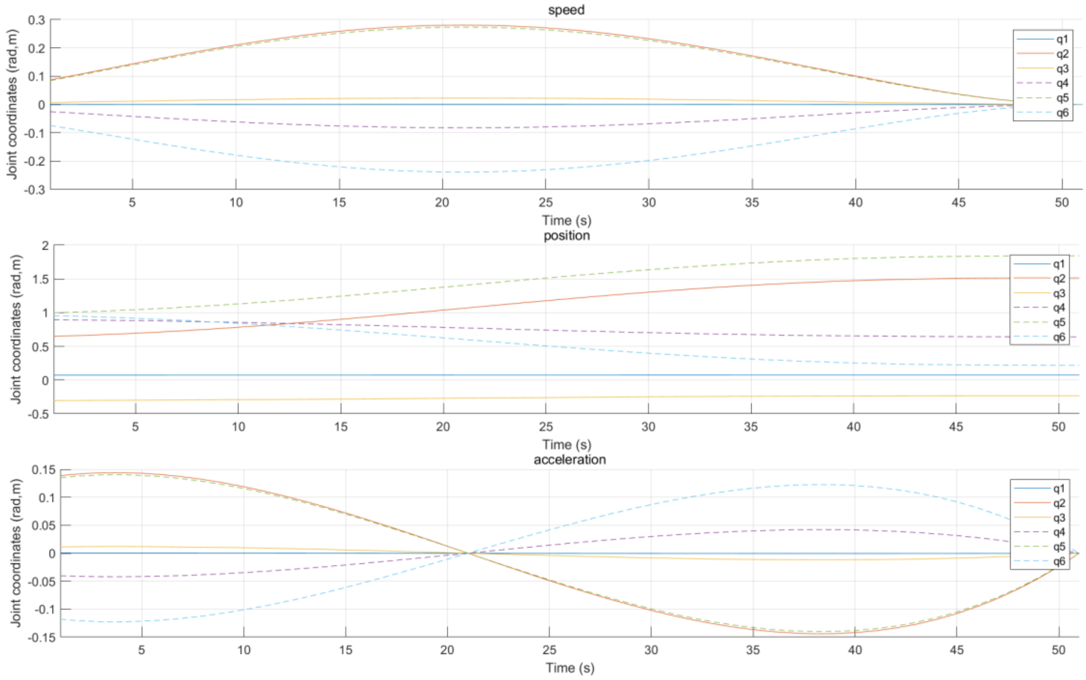

Figure 7: Changes of angle, angular velocity and angular acceleration of each joint

The trajectory change and end-point pose of the 6 - DOF robot are shown in Figure 8. The simulation of joint space trajectory of 6 DOF robot proves the correctness and effectiveness of inverse kinematics. The trajectory change diagram of the robot end and the angle, angular velocity and angular acceleration change diagram displayed in MATLAB are of great significance to analyze the stability and continuity of the robot end motion, and also lay the foundation for the optimization of trajectory planning below.

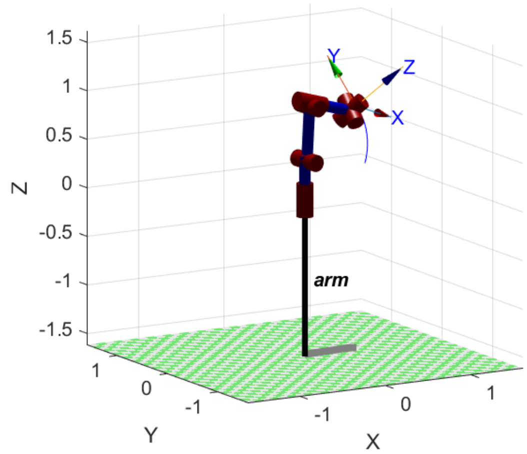

Figure 8: Robotic track

3.2. A Star Comparison of optimization algorithms

In order to further improve the safety and compliance of rehabilitation training, this paper intends to use the optimized A-star algorithm to optimize the trajectory of the robot end. This method can from a smooth curve with curvature continuity, which can better meet the needs of human motion training.

The obstacle, the target node and the starting node are marked in the grid map of 30 * 30, and the optimal path shown in Figure 9 is generated by using the traditional A star algorithm. In Figure 9, the black grid represents obstacles, the red solid line represents the optimal path searched by the traditional A Star Algorithm, the blue circle represents the target nodes, and the gray grid represents the nodes in the Close List after the A Star Algorithm search is completed. It can be seen that the corner angle of the path near the obstacle is large and the smoothness is not good, and there are redundant inflection points.

ISSN 2616-5767 Vol.5, Issue 6: 33-41, DOI: 10.25236/AJETS.2022.050607

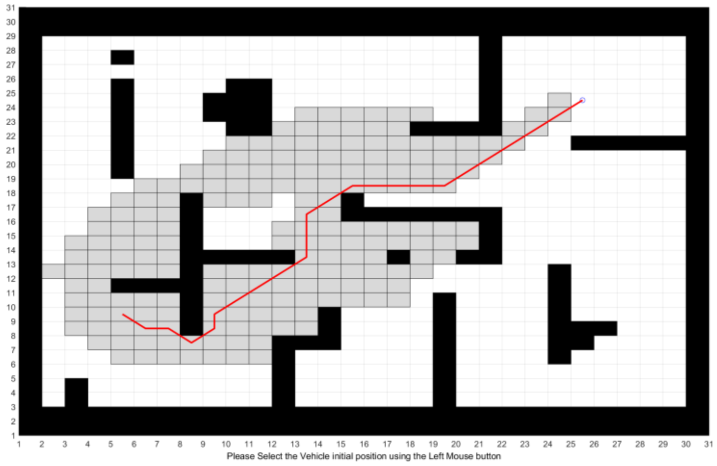

Figure 9: Traditional a Star Algorithm planning path

Use the collision detection algorithm in 2.4 to judge the collision of paths generated by traditional algorithms, and delete redundant inflection points by deleting nodes and reconnecting them. Mark the remaining nodes with red circles, and connect the remaining nodes with solid blue lines in turn to form a path to eliminate redundant nodes. The experimental results are shown in Figure10.

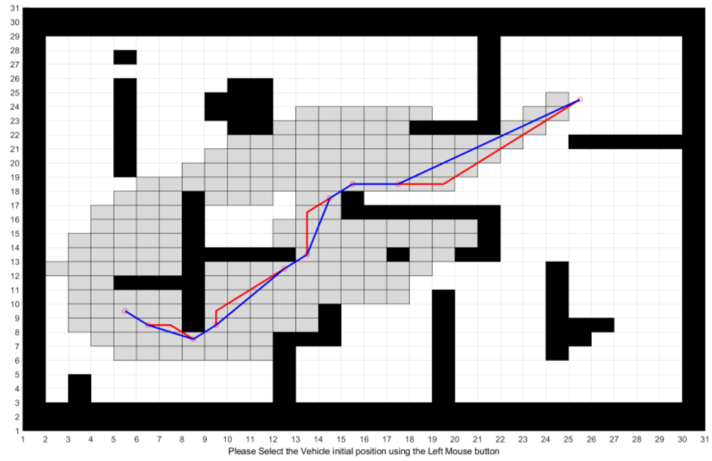

Figure 10: Path after eliminating redundant inflection points

Use the dynamic tangent smoothing algorithm in 2.5 to further smooth the barrier-free path area and the edge part of the map obstacle of the optimized path in Figure 10, and use the green solid line to represent the new path with curvature continuity generated at the corner, as shown in Figure 11.

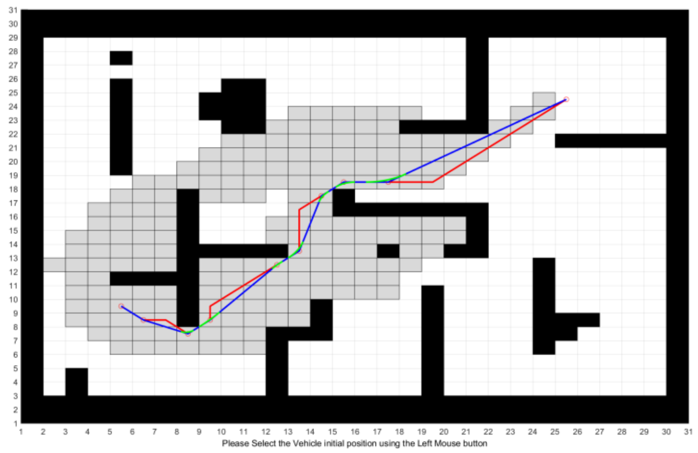

Figure 11: Path after using dynamic tangent smoothing algorithm

In order to facilitate the unified comparison of the generated broken line path and curve path, the approximate circle radius less than 4 obtained by the smoothing algorithm in Figure 11 is defined as a turning point. As shown in Table 1, the paths of Figure 9, Figure 10 and Figure 11 are compared in terms of path length, maximum turning angle and turning times. It can be seen from Table 1 that compared with Fig. 9; the improved algorithm has a shorter path. Moreover, the maximum corner angle of the path in Figure 10 is reduced by 29.5 % compared with that in Figure 9, and the path cohesion is significantly improved compared with the original plan.

Table 1: Path comparison

Map	Path length	Maximum turning angle	Number of track changes
Figure 9	30.2132	90.0000 °	9
Figure 10	28.6336	63.4332 °	7
Figure 11	28.4164	/	5

It can be further concluded from table 1 that compared with the initial path represented by red real lines, the turning times of the path connected by blue and green real lines in Figure 11 are reduced by 44.4 % and have curvature continuity, and the smoothness and safety are significantly increased. The simulation results show that the robot end trajectory planning based on the optimized A-star algorithm can effectively avoid the potential safety hazard caused by the severe steering of the patient in the rehabilitation movement. Its smooth trajectory can also further improve the rehabilitation effect of the upper limb rehabilitation robot and the comfort of the patient during rehabilitation.

4. Conclusion

In this paper, the modeling and trajectory planning simulation of the upper limb rehabilitation robot are carried out, and a trajectory planning scheme for the end of the robot with good safety and compliance is proposed. MATLAB is used to model the robot and solve the kinematic degree pose, and by combining the traditional A Star Algorithm with collision detection algorithm and dynamic tangent point smoothing algorithm with variable tangent point, a planning path with curvature continuity and high degree of cohesion is formed. It can effectively assist the rehabilitation movement, and has high practical value, which is helpful to the design of various passive rehabilitation training tracks and the research of active control strategies of human-computer interaction control in the next step.

References

[1] Zhang Li. Take advantage of the situation to promote the high-quality development of the robot industry [J]. Software and IC, 2022 (01): 16-18. DOI: 10.19609/J. CNKI. CN10-1339/TN. 2022.01.004.

[2] Discussion on innovative technology of medical robots [J]. Standardization and measurement of instruments and meters, 2021(05):13.

[3] Yang Guilin, Wang Chongchong. Overview of compliant motion control of cooperative robots [J]. Automation Expo, 2019(02):66-73.

[4] Wang Qiuhui, Wei Yukun, Liu Limeng. Progress in research and application of rehabilitation robot [J]. Packaging Engineering, 2018,39(18):83-89.

[5] Xu Dong, Xu Han, Li Yibin, Zhang Xie, Li Xiaolong. Review of research progress of upper limb rehabilitation robot [J]. Modern information technology, 2020,4(16):142-144+148.

[6] Yang Qizhi, Cao Dianfeng, Zhao Jinhai. Analysis of research status of upper limb rehabilitation robot [J]. Robot, 2013,35(05):630-640.

[7] Chen Pengju. Design of upper limb rehabilitation robot system [D]. Chang 'a university, 2014.

[8] Liu Ning, Li Nanzhou, Liang Zhou. Kinematics analysis and trajectory planning of upper limb rehabilitation robot [J]. Journal of Qingdao University of Technology, 2020,41(06):120-125.

[9] Yang Jian, Shen Linyong, Ding Hang. Motion trajectory planning and simulation of four-degree-of-freedom upper limb rehabilitation robot [J]. Computer Simulation, 2016,33(08):332-337.

[10] Xia Pengpeng, Han Jianhai, Li Liaoyuan, Li Xiangpan, Guo Bingjing, Du Ganqin. Research on trajectory planning of upper limb rehabilitation robot with three-dimensional space end traction [J]. Chinese Journal of Rehabilitation Medicine, 2021,36(11): 1414-1419.