Hindawi
Journal of Robotics
Volume 2020, Article ID 1701943, 12 pages
https://doi.org/10.1155/2020/1701943

Research Article

Improved Manipulator Obstacle Avoidance Path Planning Based on Potential Field Method

Ming Zhao[1] **and Xiaoqing Lv**[2]

[1] *School of Applied Technology, University of Science and Technology Liaoning, Anshan 114051, China*
[2] *School of Electronics and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China*

Correspondence should be addressed to Xiaoqing Lv; 2471346039@qq.com

Received 6 August 2019; Revised 1 November 2019; Accepted 13 November 2019; Published 22 January 2020

Academic Editor: Gordon R. Pennock

Aiming at the existing artificial potential field method, it still has the defects of easy to fall into local extremum, low success rate and unsatisfactory path when solving the problem of obstacle avoidance path planning of manipulator. An improved method for avoiding obstacle path of manipulator is proposed. First, the manipulator is subjected to invisible obstacle processing to reduce the possibility of its own collision. Second, establish dynamic virtual target points to enhance the predictive ability of the manipulator to the road ahead. Then, the artificial potential field method is used to guide the manipulator movement. When the manipulator is in a local extreme or oscillating, the extreme point jump-out function is used in real time to make the end point of the manipulator produce small displacements and change the action direction to effectively jump out of the dilemma. Finally, the manipulator is controlled to avoid all obstacles and move smoothly to form a spatial optimization path from the start point to the end point. The simulation experiment shows that the proposed method is more suitable for complex working environment and effectively improves the success rate of manipulator path planning, which provides a reference for further developing the application of manipulator in complex environment.

1. Introduction

The manipulator obstacle avoidance path planning [1] refers to: planning a feasible path from the starting point to the end point of the manipulator end. During the operation, the manipulator is required to not collide with any obstacles. This is a more complicated three-dimensional obstacle path planning problem in space.

Common manipulator obstacle avoidance path planning methods include grid method [2], free space method [3, 4], rapidly-exploring random tree (RRT) algorithm [5], probabilistic road map algorithm [6], particle swarm optimization [7], ant colony algorithm [8], genetic algorithm [9], artificial potential field method [10], and so on. Compared with other methods, the artificial potential field method has the advantages of simple configuration and high operating efficiency, but the algorithm belongs to the local optimization method. It is used in the manipulator of multi-link structure, sometimes there are problems such as stagnation and oscillation. In view of the shortcomings of the artificial potential field method in the obstacle avoidance path planning of manipulators, relevant scholars have done a lot of research. Literature [11] introduced the laws of dynamics to improve the artificial potential field method, and used the attraction velocity, and the repulsive velocity to construct the joint velocity of the joint space, and planned an obstacle avoidance path for the manipulator. However, the improved algorithm still has the problem of being easily trapped in local extremum. The literature [12] uses the navigation potential function method to deal with the local extremum problem, but the inverse kinematics solution is applied in the overall path planning, which is computationally complex, and inefficient. In literature [13], the solution of adding virtual obstacles is applied to solve the minimum value problem. However, before this scheme, a reasonable set of angle values needs to be obtained, and the related operations are not easy to implement. Literature [14] attracts the current point away from the local extremum by adding a virtual target point. However, geometric methods are used to determine virtual target points, and its applicability is not strong in complex environments. Literature

Table 1: The link parameters of the PUMA560.

Link i	α_{i-1} [°]	a_{i-1}	d_i	θ_i	Joint angle range [°]
1	0	0	0	θ_1	−160~160
2	−90	0	d_2	θ_2	−225~45
3	0	a_2	0	θ_3	−45~225
4	−90	a_3	d_4	θ_4	−110~170
5	90	0	0	θ_5	−100~100
6	−90	0	0	θ_6	−266~266

Among them $a_2 = 431.8$ mm, $a_3 = 20.32$ mm, $d_2 = 149.09$ mm $d_4 = 433.07$ mm.

[15, 16] combined with the random advantage of RRT algorithm to generate virtual target points to escape local extremum, but there are too many uncertain factors in the process, and the path is not guaranteed to be optimal.

Based on the above analysis, it can be seen that the defects of the artificial potential field method restrict the completion quality of the manipulator obstacle avoidance path planning task. This paper starts from reducing the possibility of falling into local extremum and path optimization, by establishing the dynamic virtual target points to improve the ability of the manipulator to predict obstacles in front; by setting the extreme point jump function to help the manipulator to jump out of local extremum or oscillation. Among them, the random principle of the dynamic virtual target point reduces the possibility that the manipulator is trapped in the local extremum, and the extreme point jump-out function not only effectively solves the problem of extreme point, and oscillation but also reduces the unnecessary moving distance as much as possible, which is beneficial to improve the overall path quality.

2. Kinematics Modeling and Analysis of Manipulator

2.1. Kinematics Modeling of Manipulator. This paper takes PUMA560 as the research object, and its specific structure diagram is available from reference [17]. The PUMA560 is an articulated robot whose first three joints determine the position of the end and the last three joints determine the orientation of the end. Since this paper studies the path planning problem at the end of the manipulator, it does not involve the grasping action, so only the angle changes of the first three joints are concerned. The degrees of the last three joints are fixed to zero in the discussion that follows.

According to the structural diagram of the manipulator and the coordinate system of the link parameters established by the D-H [18, 19] parameter method, the link parameters of the PUMA560 [17, 20] can be obtained, as shown in Table 1.

2.2. Kinematics Analysis of the Manipulator. After the parameters are determined, the manipulator can be analyzed for positive kinematics. In the case of a given base coordinate, the end position, and attitude can be obtained by the transformation formula. Its transformation formula is as follows:

$$ {}^{i-1}_{\ \ \ i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -ds\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & dc\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1} $$

Substituting the link parameters information of PUMA560 into Equation (1), the homogeneous transformation matrix between the links can be calculated. By multiplying the link transformation matrices, a matrix of end positions and poses can be obtained. That is, its positive kinematics equation, as shown in Equation (2).

$$ {}^0_6T = {}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_4){}^5_6T(\theta_6). \tag{2} $$

3. Collision Detection

3.1. Modeling of Obstacles. Obstacle modeling is one of the important steps in obstacle avoidance path planning. In order to ensure that there is no collision and to simplify the calculation as much as possible, this paper uses the circumscribed ball method [21] to establish the obstacle model.

(i) First wrap the obstacle in the smallest cube.

(ii) When the long side of the cube is less than 3 times the short side, make the circumscribed ball of the cube at this time. Otherwise, the cube is divided into two parts by a plane passing through the midpoint of each long side.

(iii) The wrapped information of the two parts is released, and the two parts are treated separately as obstacles. That is, steps (i), (ii), and (iii) are repeated until the obstacle is completely wrapped by the circumscribing ball.

(iv) After converting the obstacle into one or more circumscribed balls, obtain the center and radius of each circumscribed ball.

3.2. Collision Detection. The links of the manipulator have a certain cross-sectional area. In order to conveniently establish a collision detection system, each link is enveloped into a cylinder. The radius of the section of the cylinder is measured as the radius of the link of the manipulator. In order to reduce the complexity of the calculation, the radius of the link of the manipulator is added to the radius of each circumscribed ball of the obstacle. At this time, the positional relationship between the obstacle and the link of the manipulator is transformed into a simple ball-to-wire relationship. The collision detection process is as follows:

When the position coordinates of the two ends of the link are (x_i, y_i, z_i) and $(x_{i+1}, y_{i+1}, z_{i+1})$, a two-point equation of the straight line is established.

$$ \frac{x - x_i}{x_{i+1} - x_i} = \frac{y - y_i}{y_{i+1} - y_i} = \frac{z - z_i}{z_{i+1} - z_i}. \tag{3} $$

Let $(x - x_i)/(x_{i+1} - x_i) = (y - y_i)/(y_{i+1} - y_i) = (z - z_i)/(z_{i+1} - z_i) = t$. After sorting, a parametric equation is obtained, which is shown in Equation (4).

Table 2: Table of collision analysis.

Number of solutions	Range of solutions	Collision or not
0	No solution	No
1	$0 \leq t \leq 1$	Yes
1	$t < 0$ or $t > 1$	No
2	$0 \leq t_1 \leq 1$ or $0 \leq t_2 \leq 1$ or ($t_1 < 0$ and $t_2 > 1$) or ($t_1 > 1$ and $t_2 < 0$)	Yes
2	($t_1 < 0$ and $t_2 < 0$) and ($t_1 > 1$ and $t_2 > 1$)	No

$$\begin{aligned} x &= t(x_{i+1} - x_i) + x_i \\ y &= t(y_{i+1} - y_i) + y_i \\ z &= t(z_{i+1} - z_i) + z_i. \end{aligned} \tag{4}$$

The standard equation for the ball is shown in Equation (5).

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2, \tag{5}$$

among them, (x_0, y_0, z_0) is the coordinates of the center of the circumscribed ball; r is the sum of the radius of the circumscribed ball and the radius of the manipulator link. Bring Equation (4) into Equation (5), and get a quadratic equation about t. The value of parameter t can be used to determine whether the intersection of the line and the ball is within the range of (x_i, y_i, z_i) to $(x_{i+1}, y_{i+1}, z_{i+1})$. When $0 \leq t \leq 1$, its intersection is within the range of the line segment, otherwise it is not. Therefore, by analyzing the solutions t_1 and t_2 of the equation, the positional relationship between the ball and the line can be judged. The collision analysis [16] is shown in Table 2.

4. Manipulator Obstacle Avoidance Path Planning

4.1. Theoretical Basis of Artificial Potential Field Method in Three-Dimensional Space. The artificial potential field method is an extremely important method in path planning. Its outstanding advantages are high efficiency and operability. The basic principle of the artificial potential field method is to place the controlled object in an abstract artificial potential field environment. The target point in the environment produces "attractive force" on it, and the obstacle produces a "repulsive force" on it, and the combined force of the two forces guides the controlled object to move.

In three-dimensional space, the attractive potential field function of the artificial potential field method is:

$$U_{\text{att}}(X_j) = \frac{1}{2} k \rho^2 (X_j, X_g), \tag{6}$$

among them, $X_j = (x_j, y_j, z_j)$ is the position coordinate of the jth step of the controlled object, that is, the current position point; $X_g = (x_g, y_g, z_g)$ is the position coordinate of the target point; k is the gain coefficient of the attractive force; $\rho(X_j, X_g) = \| X_j - X_g \|$ is the shortest distance between X_j and X_g. For Equation (6) to find a negative gradient [22] for X_j, the attraction function is:

$$F_{\text{att}}(X_j) = -\nabla(U_{\text{att}}) = k\rho(X_j, X_g). \tag{7}$$

Then the attractive component of point X_j on the three coordinate axes is:

$$\begin{aligned} F_{\text{att}x}(X_j, X_g) &= F_{\text{att}}(X_j, X_g) \times \cos \alpha_{j,x}, \\ F_{\text{att}y}(X_j, X_g) &= F_{\text{att}}(X_j, X_g) \times \cos \alpha_{j,y}, \\ F_{\text{att}z}(X_j, X_g) &= F_{\text{att}}(X_j, X_g) \times \cos \alpha_{j,z}, \end{aligned} \tag{8}$$

among them, $\alpha_{j,x}$, $\alpha_{j,y}$, and $\alpha_{j,z}$ are the angles between the line connecting the point X_j and the obstacle center and the x, y, and z axes, respectively.

The obstacle is treated by the method in Section 3.1, and the radius of the ith $i = (1, 2, \ldots, n)$ circumscribed ball is r_i, and the center of the ball is $X_{0,i} = (x_{0,i}, y_{0,i}, z_{0,i})$. The direction vector pointed by the obstacle to X_j is denoted as e_i, as shown in Equation (9). At this time, the position coordinate $X_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})$ of the closest point [21] of X_j and the circumscribed ball can be calculated, as in the Equation (10).

$$e_i = \frac{(x_j - x_{0i}, y_j - y_{0i}, z_j - z_{0i})}{\sqrt{(x_j - x_{0i})^2 + (y_j - y_{0i})^2 + (z_j - z_{0i})^2}}, \tag{9}$$

$$X_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) = (x_{0i} + r_i a_i, y_{0i} + r_i b_i, z_{0i} + r_i c_i), \tag{10}$$

among them,

$$\begin{bmatrix} a_i \\ b_i \\ c_i \end{bmatrix} = \begin{bmatrix} \frac{x_j - x_{0i}}{\sqrt{(x_j - x_{0i})^2 + (y_j - y_{0i})^2 + (z_j - z_{0i})^2}} \\ \frac{y_j - y_{0i}}{\sqrt{(x_j - x_{0i})^2 + (y_j - y_{0i})^2 + (z_j - z_{0i})^2}} \\ \frac{z_j - z_{0i}}{\sqrt{(x_j - x_{0i})^2 + (y_j - y_{0i})^2 + (z_j - z_{0i})^2}} \end{bmatrix}. \tag{11}$$

The acquisition of the relevant repulsion is transformed into a point-to-point calculation. The repulsive potential field function is:

$$U_{\text{rep}}(X_j) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{\rho(X_j, X_{i,j})} - \frac{1}{\rho_0} \right)^2 & \rho(X_j, X_{i,j}) \leq \rho_0 \\ 0 & \rho(X_j, X_{i,j}) < \rho_0, \end{cases} \tag{12}$$

among them, η is the gain coefficient of the repulsive force; ρ_0 is the influence distance of the obstacle; $\rho(X_j, X_{i,j}) = \| X_j - X_{i,j} \|$ is the shortest distance between X_j and $X_{i,j}$. For Equation (12) to find a negative gradient for X_j, the repulsion function is:

$$\begin{aligned} F_{\text{rep}}(X_j) &= -\nabla(U_{\text{rep}}) \\ &= \eta \left(\frac{1}{\rho(X_j, X_{i,j})} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2(X_j, X_{i,j})}. \end{aligned} \tag{13}$$

Then the repulsive component of point X_j on the three axes is:

$$\begin{aligned} F_{\mathrm{rep}x}\left(X_j, X_{i,j}\right) &= F_{\mathrm{rep}}\left(X_j, X_{i,j}\right) \times \cos\beta_{j,x}, \\ F_{\mathrm{rep}y}\left(X_j, X_{i,j}\right) &= F_{\mathrm{rep}}\left(X_j, X_{i,j}\right) \times \cos\beta_{j,y}, \\ F_{\mathrm{rep}z}\left(X_j, X_{i,j}\right) &= F_{\mathrm{rep}}\left(X_j, X_{i,j}\right) \times \cos\beta_{j,z}, \end{aligned} \tag{14}$$

among them, $\beta_{j,x}$, $\beta_{j,y}$, and $\beta_{j,z}$ are the angles between the line connecting the point X_j and the obstacle center and the x, y, and z axes, respectively.

After combining the attractive force and the repulsive force, the resultant force F_j at point X_j can be obtained.

$$F_j = F_{jx} + F_{jy} + F_{jz}, \tag{15}$$

among them, F_{jx}, F_{jy}, and F_{jz} are the resultant force components on the three coordinate axes to point X_j, as shown in Equation (16).

$$\begin{aligned} F_{jx} &= F_{\mathrm{att}x}\left(X_j, X_g\right) + \sum_i F_{\mathrm{rep}x}\left(X_j, X_{i,j}\right), \\ F_{jy} &= F_{\mathrm{att}y}\left(X_j, X_g\right) + \sum_i F_{\mathrm{rep}y}\left(X_j, X_{i,j}\right), \\ F_{jz} &= F_{\mathrm{att}z}\left(X_j, X_g\right) + \sum_i F_{\mathrm{rep}z}\left(X_j, X_{i,j}\right). \end{aligned} \tag{16}$$

According to the resultant force F_j and the step size l, the position $X_{j+1} = \left(x_{j+1}, y_{j+1}, z_{j+1}\right)$ of the next step can be determined, and the components are as shown in Equation (17).

$$\begin{aligned} x_{j+1} &= x_j + l \times \frac{F_{jx}}{F_j}, \\ y_{j+1} &= y_j + l \times \frac{F_{jy}}{F_j}, \\ z_{j+1} &= z_j + l \times \frac{F_{jz}}{F_j}. \end{aligned} \tag{17}$$

According to the above theory, obstacle avoidance path planning in a three-dimensional space with respect to one point can be achieved. This laid the foundation for the more complex manipulator's obstacle avoidance path planning.

4.2 Obstacle Avoidance Path Planning Method for Manipulator

4.2.1. Establish Dynamic Virtual Target Points. The artificial potential field method belongs to a kind of local optimization algorithm. In order to achieve better global optimization effect, this paper uses dynamic virtual target points to guide the manipulator movement. That is, before each step of action, a virtual target point is determined according to a certain principle, which attracts the end of the manipulator and guides the arm to move. The selection process of the virtual target point is as follows. First consider the end of the manipulator as a point. Then, according to the theory of Section 4.1, taking the current coordinates of the point as the starting point and the target point as the end point, an optimal path for obstacle avoidance is planned. Calculate the total length of the path. When its length is small, the virtual target point is equal to the target point. Otherwise, a point on the path (which is a random point within the specified range) is selected as the virtual target point of the manipulator's current action. Record the virtual target point here as X_{gg}.

4.2.2. Variable Parameter ρ_0 Is Used. In the normal case, ρ_0 is set to a fixed value, that is, the influence distance in the whole process adopts the same standard. However, when there are obstacles of different sizes in the space and the difference in the radius of the circumscribed sphere is large, if a small ρ_0 is used at this time, the phenomenon that the controlled points are too close to the large obstacles is likely to occur. At this time, it is necessary to make a large adjustment to avoid obstacles, which will greatly increase the difficulty of control or lengthen the path length. In response to this problem, this paper proposes that the radius of the obstacle itself determines its influence distance. When the radius of the circumscribed ball of the ith obstacle is r_i, its influence distance $\rho_{0,i}$ is $\rho_{0,i} = b * r_i$, where b is a constant.

4.2.3. Handling of Invisible Obstacles. Due to the complex structure of the manipulator, its end may collide with itself during operation. In order to reduce this possibility, the initial joint angle portion of the manipulator is treated as an invisible obstacle. The circumscribed ball center of the obstacle is the position coordinate of the starting joint angle, and the radius is set to 1.5 times the link radius. It participates in the calculation of all relevant repulsions.

4.2.4. Set the Extreme Point Jump-Out Function. If the current joint angle has caused the resultant force at the end of the manipulator to be the smallest among all adjacent joint angles [23], but the end position corresponding to the joint angle does not reach the target point, the manipulator at this time has fallen into a local minimum point. In order to effectively solve this problem, this paper sets the extreme point jump-out function. This function is called in time when it is detected that the end position of the manipulator is repeated with a position that has passed before (including the case of oscillation). First, Bloch Quantum Genetic Algorithm (BQGA) [24] is used to generate a small displacement at the end of the manipulator. Then, the RRT algorithm [5, 15, 16] is used to change the direction of motion of the manipulator. Finally, under the joint action of the two algorithms, the local extremum is jumped out.

The process of using BQGA to generate small displacements is as follows:

(1) Parameter initialization of BQGA. Current iteration number t, initial group $Q(t)$, maximum number of iterations $M - g$, variables λ_1, λ_2, λ_3, and so on.

(2) The spatial transformation of the solution yields an approximate solution set $X(t)$.

(3) Calculate fitness. Collision detection of the end of the manipulator. If it collides, the fitness function is set to infinity. Otherwise, calculate the fitness function (the resultant force at the end of the manipulator). Obtain contemporary optimal solutions BX and contemporary optimal chromosomes BC.

(4) Take BX as the global optimal solution GX and BC as the global optimal chromosome GC.

(5) In the loop, set $t \leftarrow t + 1$, and obtain a new population $Q(t)$ by updating and mutating $Q(t - 1)$.

(6) Transform the optimization result in the unit space into the solution space to get the solution $X(t)$ of the optimization problem.

(7) If $fit(BX) < fit(GX)$, update the contemporary optimal solution $BX \leftarrow GX, BC \leftarrow GC$. Otherwise $GX \leftarrow BX, GC \leftarrow BC$.

(8) If $t > M - g$, output the global optimal solution $GX = (\lambda_1, \lambda_2, \lambda_3)$, and end. Otherwise, return to step (5).

(9) At this time, the angle of each joint angle of the arm is $T = [\ T(1) + \lambda_1 \quad T(2) + \lambda_2 \quad T(3) + \lambda_3 \quad 0 \quad 0 \quad 0\ ]$, so that the corresponding end position TT can be obtained according to the kinematics of the manipulator, that is, the process of completing the small displacement of the end of the manipulator.

The process of changing the direction of motion using the RRT algorithm is as follows:

(1) Start with the current position of the end of the manipulator and record it as q_{start}. Given the separation distance Δl.

(2) A random point q_{rand} is generated in the working space of the manipulator.

(3) Find the node q_{near} closest to q_{rand} in the RRT and calculate the Euclidean distance l from q_{near} to q_{rand}. If $l < \Delta l$, the new node q_{new} is equal to q_{rand}. Otherwise, q_{new} is equal to the point after q_{new} extends Δl in the direction of q_{rand}.

(4) Determine whether the space segment of q_{near} to q_{new} collides with an obstacle. If it collides, round off q_{new} and return to step (2). Otherwise, add q_{new} to the tree.

(5) Determine whether the maximum number of iterations has been reached. If not, return to step (2) to continue the loop. Otherwise, draw the path and mark the node at the end of the path as the virtual target point, denoted as X_{ggg}.

(6) The direction in which q_{start} points to X_{ggg} is taken as the next action direction of the manipulator.

In the commonly used path planning method [16], the virtual target point method is also used to jump out of the local extremum. Its process is to create a virtual target point near the end, after the end needs to reach the virtual target point, and then move to the final target point. In contrast, the method in this paper only changes the direction of the action through the randomly generated virtual target point, and does not need to arrive, that is, after running one step, the virtual target point disappears instantly. This can improve work efficiency and ensure path quality.

In general, the random principle of dynamic virtual target points reduces the likelihood that the manipulator will fall into local extremum. The dynamic virtual target point and the appropriate influence distance make it difficult for the end of the manipulator to reach a dangerous area that is too close to the obstacle, which improves the obstacle prediction ability of the manipulator to some extent. The invisible obstacle treatment better overcomes the problem of the structural structure of the manipulator. Setting the extreme point jump-out function solves the stagnation and oscillation of the manipulator. The application of these measures improves the effectiveness and applicability of the manipulator's obstacle avoidance path planning system, and is conducive to planning a more ideal obstacle avoidance path.

4.3. Manipulator's Obstacle Avoidance Path Planning Process. According to the above theory, the basic steps of the obstacle avoidance path planning of the manipulator are summarized as follows:

(1) The manipulator's kinematics modeling and obstacle modeling using the methods of Sections 2.1 and 3.1, respectively.

(2) Parameter initialization. For example, the initial angle is denoted as T_0, the starting point is marked as X_0, the target point is marked as X_{g0}, the step size is recorded as λ, the gain coefficient of the attractive force is denoted as k, the gain coefficient of the repulsive force is denoted as η, the joint angle combination is recorded as T (initialized as $T = T_0$), the corresponding end position is denoted by TT (initialized as $TT = X_0$), the dynamic virtual target point is denoted by X_g, and the switching instruction is denoted by sg (initialized to $sg = 0$).

(3) Is it judged that $sg = 0$ is established? If it is established, $X_g = X_{gg}$, if it is not established, then $X_g = X_{ggg}$ and $sg = 0$, where X_{gg} is derived from the method of Section 4.2.1 and X_{ggg} is derived from the method of Section 4.2.4.

(4) Traverse the adjacent joint angles. There are three possible values for each joint angle. They are $\theta_i - \lambda, \theta_i$ and $\theta_i + \lambda$, where $i = 1, 2, \ldots, m$, and m is the number of joint angles that can change the end position. Set the joint angle that does not affect the end position to zero.

(5) According to the kinematics analysis of the manipulator in Section 2.2, obtain the position information of the end and each joint angle.

(6) Collision detection is performed on each link using the method in Section 3.2. If it collides, it is rounded off, otherwise, the information of the angle combination and its corresponding end position information are saved.

(7) Calculate the resultant force at all end positions in step (6), find the end position where the resultant force is the smallest, and save the end position (denoted as TT_{next}) and its corresponding angle combination (denoted as T_{next}).

(8) It is judged whether TT_{next} in step (7) reaches the vicinity of the target point X_{g0}, and if so, it ends, otherwise, it goes to step (9).

(9) Determine whether the TT_{next} at this time is the same as the position that has passed before, and if so, call the extreme point jump-out function and set $sg = 1$, then return to step (3). Otherwise, the manipulator is driven according to the angle information at this time, that is, $T = T_{\text{next}}$, $TT = TT_{\text{next}}$, and then returns to step (3).

The corresponding flow chart is shown in Figure 1:

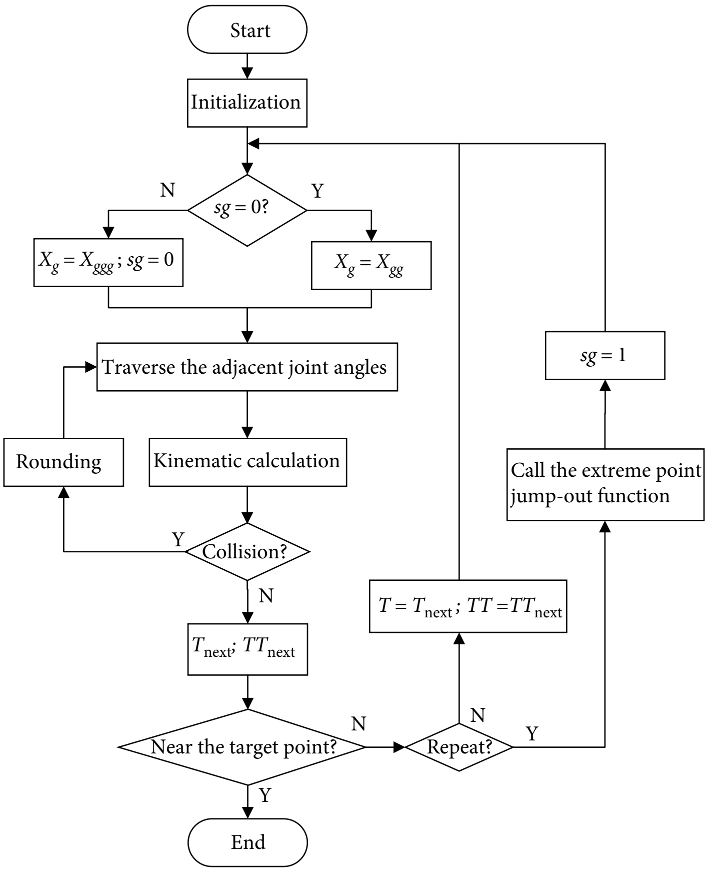

Start
Initialization
N
$sg = 0?$
Y
$X_g = X_{ggg}$; $sg = 0$
$X_g = X_{gg}$
Traverse the adjacent joint angles
Rounding
Kinematic calculation
Y
Collision?
N
T_{next}; TT_{next}
Near the target point?
N
Y
End
Repeat?
N
Y
$T = T_{\text{next}}$; $TT = TT_{\text{next}}$
Call the extreme point jump-out function
$sg = 1$

FIGURE 1: Flow chart of the obstacle avoidance path planning of the manipulator.

5. Simulation Experiment

In order to verify the effectiveness of the path planning method proposed in this paper, simulation experiments were carried out with Matlab R2016b. The simulation experiment is from single obstacle to multi-obstacle, and compared with the simulation results of common methods. The initialization of the parameters is shown in Table 3.

Record the total number of steps required for the manipulator to complete the task as s_{sum}. During the movement, the angle changes of the three joints are respectively recorded as $\Delta\theta_{1,s}$, $\Delta\theta_{2,s}$, and $\Delta\theta_{3,s}$, where $s = 1, 2, \ldots, s_{\text{sum}}$, then their cumulative angle changes are:

$$\begin{aligned}\Delta\theta_1 &= \left|\Delta\theta_{1,1}\right| + \left|\Delta\theta_{1,2}\right| + \cdots + \left|\Delta\theta_{1,s_{\text{sum}}}\right|,\\ \Delta\theta_2 &= \left|\Delta\theta_{2,1}\right| + \left|\Delta\theta_{2,2}\right| + \cdots + \left|\Delta\theta_{2,s_{\text{sum}}}\right|,\\ \Delta\theta_3 &= \left|\Delta\theta_{3,1}\right| + \left|\Delta\theta_{3,2}\right| + \cdots + \left|\Delta\theta_{3,s_{\text{sum}}}\right|.\end{aligned} \tag{18}$$

Therefore, the cumulative angle change of the entire manipulator is:

$$\Delta\theta_{\text{sum}} = \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3. \tag{19}$$

During the movement, the coordinate value of the end position of the manipulator is recorded as X_s $(s = 1, 2, \ldots, s_{\text{sum}})$, and the change of the end position is as follows:

$$\Delta X_s\left(X_s, X_{s-1}\right) = \left\|X_s - X_{s-1}\right\|. \tag{20}$$

TABLE 3: Table of manipulator initialization parameters.

Parameter	Value
T_0/rad	(0, 0, 0, 0, 0, 0)
λ/rad	0.01π
k	10
η	1

That is, $\Delta X_s\left(X_s, X_{s-1}\right)$ is the distance between two points X_s and X_{s-1}. X_0 is the starting position coordinate. Therefore, the end moving distance during the entire movement is approximately:

$$\Delta X_{\text{sum}} = \Delta X_1 + \Delta X_2 + \cdots + \Delta X_{s_{\text{sum}}}. \tag{21}$$

5.1. Simulation Experiment of Single Obstacle Path Planning. Let the coordinates of the target point be $X_{g0} = (0.3500,\ -0.2500,\ 0.4000)$, and calculate the spherical center coordinates and radius of the obstacle according to the established obstacle model as $X_{ob} = (0.4000,\ -0.3000,\ -0.1500)$ and $r = (0.2500)$, respectively. According to the flow chart of the obstacle avoidance path planning of the manipulator, the simulation results are shown in Figures 2 and 3.

It can be seen from the Figure 2 that the manipulator avoids the obstacle and accurately reaches the vicinity of the target point, forming a relatively smooth path. It can be seen

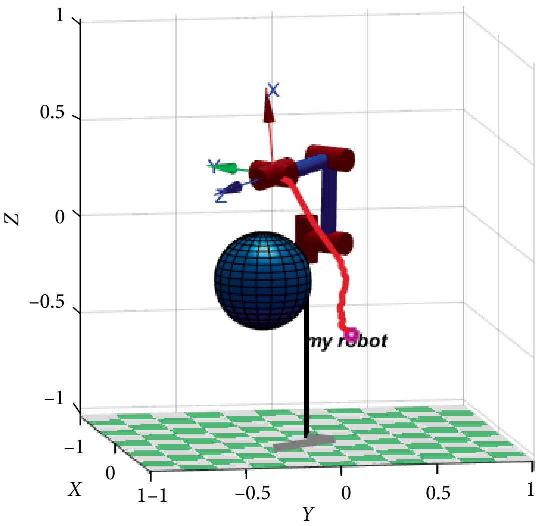


FIGURE 2: Path plan of an obstacle.

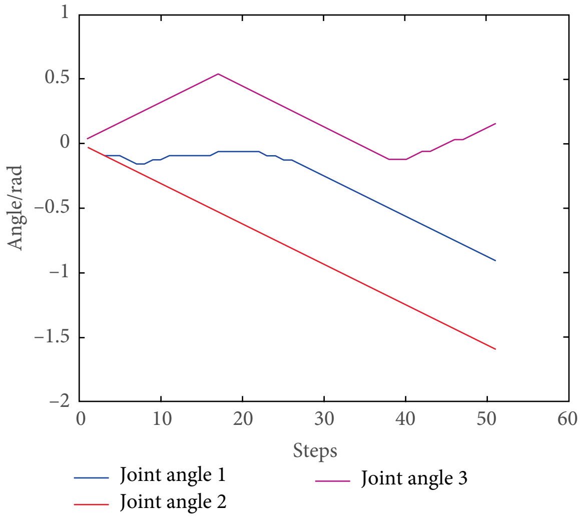


FIGURE 3: Changes in the joint angle of each.

from Figure 3 that the angles of the joint angles of the manipulators change continuously in small amplitude during the stepping process, which ensures the stability of the movement of the manipulator. Observing the whole movement process of the manipulator, the joints and links do not collide with the obstacle. Under a certain safety margin, the end of the manipulator advances step by step toward the target point. s_{sum} is only 52 when the process is completed. The cumulative angular change of the entire manipulator $\Delta\theta_{\text{sum}}$ is 4.1783 rad, and the cumulative end moving distance ΔX_{sum} is 1.2232 m. In the experiment of single obstacles, there was no situation of falling into local extremum or oscillation, and the path planning task was completed smoothly.

5.2. Simulation Experiment of Multiple Obstacle Path Planning. Let the number of obstacles be three, $X_{g0} = (0.3000, -0.1500, 0.7200)$, $X_{ob} = (0.6000,\ 0.3500,\ -0.1000;\ 0.7000,\ -0.4000,\ 0.3000;\ 0.5000,\ -0.2000,\ -0.3000)$, $r = (0.1500,\ 0.2500,\ 0.1200)$, and other parameters remain unchanged. The simulation results from the starting point to the target point are shown in Figures 4 and 5. Table 4 records the coordinate information of each path point where the end position of the manipulator travels.

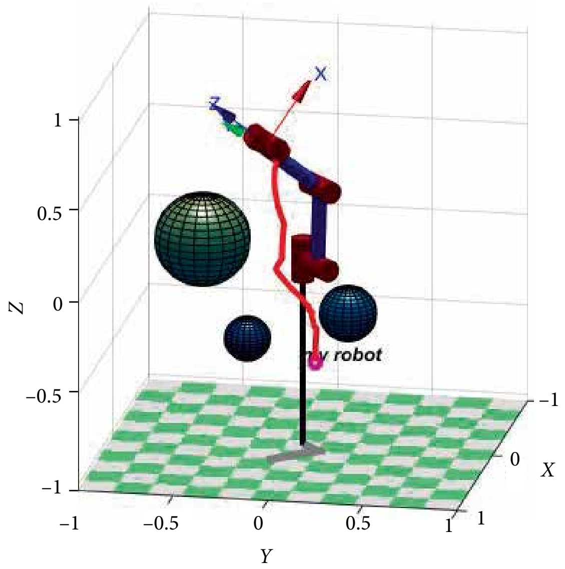


FIGURE 4: Path plan of three obstacles.

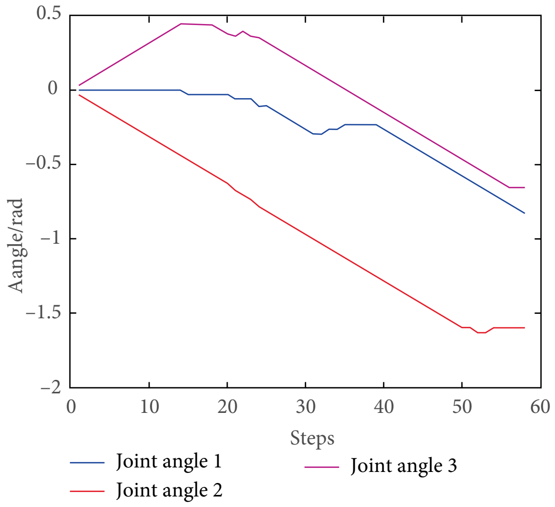


FIGURE 5: Changes in the joint angle of each.

It can be seen from Figure 4 that the manipulator can still avoid obstacles successfully reaching the vicinity of the target point in a complicated working environment with three obstacles. During the operation, the whole manipulator did not collide with the obstacles, and it was not too close to the obstacles, which proved that the manipulator can predict the obstacles in advance. It can be seen from Figure 5 that the joint angle of the manipulator changes continuously without abnormal fluctuation or static phenomenon. The overall movement of the manipulator is stable, and the end running path is smooth. s_{sum} is only 59 when the process is completed. The cumulative angular change of the entire manipulator $\Delta\theta_{\text{sum}}$

Table 4: Coordinate information of the path point of the manipulator.

Steps	$TT(x \quad y \quad z)$	Steps	$TT(x \quad y \quad z)$
1	(0.4519 0.1491 − 0.4195)	23	(0.4730 0.1343 − 0.1498)
2	(0.4513 0.1491 − 0.4060)	24	(0.4775 0.1342 − 0.1350)
3	(0.4502 0.1491 − 0.3924)	25	(0.4775 0.1342 − 0.1350)
…	…	26	(0.4468 0.1351 − 0.1569)
19	(0.4627 0.1346 − 0.1789)	…	…
20	(0.4812 0.1340 − 0.1606)	64	(0.3251 − 0.1284 0.7131)
21	(0.4812 0.1340 − 0.1606)	65	(0.3150 − 0.1318 0.7233)
22	(0.4549 0.1349 − 0.1678)	66	(0.2000 − 0.1500 0.7200)

Table 5: Comparison of data between two planning methods for an obstacle.

S/N	s_{sum}	$\Delta\theta_{\text{sum}}$ [rad]	ΔX_{sum} [m]	Extreme point?
1	44 (45)	3.5500 (3.5500)	1.4616 (1.5082)	No (No)
2	52 (53)	4.2412 (4.2097)	1.3254 (1.5204)	No (No)
3	44 (46)	3.6757 (3.6128)	1.4493 (1.4825)	No (No)
4	51 (56)	3.8642 (4.0527)	1.4639 (1.5969)	No (No)
5	48 (48)	3.2358 (3.3615)	1.5348 (1.5828)	No (No)
6	50 (54)	3.8327 (4.0527)	1.5411 (1.4013)	No (No)

The corresponding histogram is shown in Figures 6–8.

Table 6: Comparison of data between two planning methods for two obstacles.

S/N	s_{sum}	$\Delta\theta_{\text{sum}}$ [rad]	ΔX_{sum} [m]	Extreme point?
1	64 (113)	5.5292 (8.4195)	1.4867 (2.2876)	No (Yes)
2	58 (53)	4.4726 (4.3354)	1.3254 (1.5356)	Yes (No)
3	70 (119)	5.9062 (8.8593)	1.5150 (2.1116)	No (Yes)
4	70 (∞)	5.8434 (∞)	1.5056 (∞)	No (∞)
5	72 (113)	5.9690 (11.0898)	1.5247 (2.5900)	No (Yes)
6	128 (∞)	6.4088 (∞)	1.5740 (∞)	Yes (∞)

The corresponding histogram is shown in Figures 9–11, where the graph reaching the maximum represents ΔX_{sum}.

Table 7: Comparison of data between two planning methods for three or more obstacles.

S/N	s_{sum}	$\Delta\theta_{\text{sum}}$ [rad]	ΔX_{sum} [m]	Extreme point?
1	78 (105)	6.5973 (8.9850)	1.5905 (2.2011)	No (Yes)
2	84 (150)	6.6602 (12.4093)	1.6221 (2.7519)	Yes (Yes)
3	91 (∞)	6.6403 (∞)	1.7230 (∞)	No (∞)
4	114 (130)	6.8801 (10.4301)	1.8434 (2.1753)	Yes (Yes)
5	∞ (∞)	∞ (∞)	∞ (∞)	∞ (∞)
6	143 (∞)	7.2257 (∞)	1.7650 (∞)	Yes (∞)

The corresponding histogram is shown in Figures 12–14, where the graph reaching the maximum represents ΔX_{sum}.

is 4.2158 rad, and the cumulative end moving distance ΔX_{sum} is 1.4400 m. Observing the coordinate information of each path point at the end of the manipulator in Table 4, the end position information is different. However, when observing the running process of the program, it was found that the manipulator had a phenomenon of repeating with the previous position in the 21th step and the 24th step, that is, the manipulator was caught in the local extreme value. After this happens, the main program calls the extreme point jump-out function twice, and adjusts the position of the end of the manipulator and its action direction in time. After only two adjustments, the arm jumped out of the local extremes that were trapped. The reason why the repeated position information does not appear in the process is because the adjusted information covers the repeated information when the extreme point jump-out function adjusts the manipulator. In this way, the manipulator will not stop moving during the whole operation. In this operation, the manipulator not only effectively addresses the extreme point and oscillation problems but also ensures the path quality as much as possible. Eventually, the manipulator runs stably along a more desirable path to the vicinity of the target point.

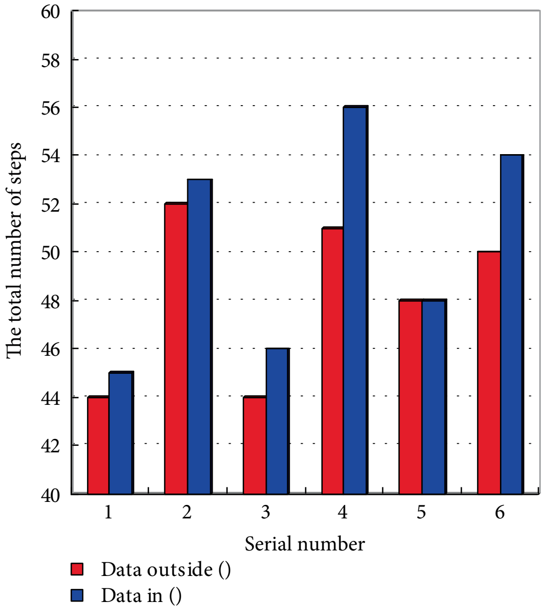


Figure 6: A comparison chart of s_{sum} when there is an obstacle.

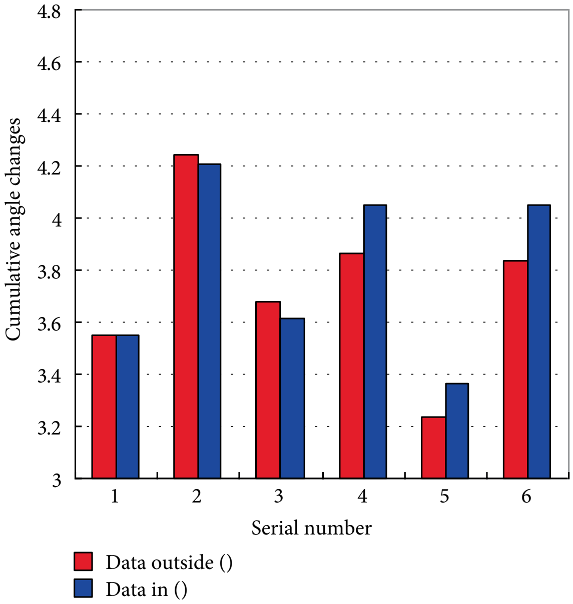


Figure 7: A comparison chart of $\Delta\theta_{\mathrm{sum}}$ when there is an obstacle.

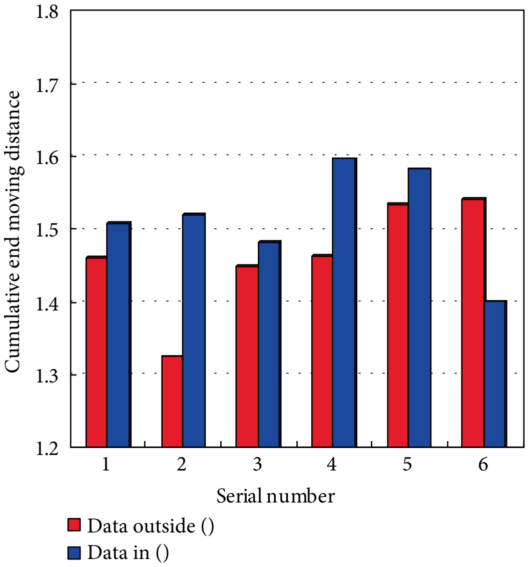


Figure 8: A comparison chart of ΔX_{sum} when there is an obstacle.

5.3. Comparison with Commonly Used Path Planning Methods. Literatures [15, 16] use the hybrid algorithm of artificial potential field method and RRT algorithm to realize the obstacle avoidance path planning of the manipulator. The main idea of the method is as follows: the artificial potential field method is used for path planning; when the manipulator is trapped in the local extremum, the RRT algorithm is used to establish the virtual target point to jump out of the local extremum; when the manipulator jumps out of the local extremum, the artificial potential field method continues to be used for path planning. This method is common in the existing literature. Therefore, this paper refers to it as the commonly used method for manipulator obstacle avoidance path planning. The disadvantage of using this method is that because the target point is fixed, the manipulator is more likely to fall into local extremum, and the path quality cannot be guaranteed during the process of jumping out of the local extremum. In order to further prove the effectiveness and superiority of the method in this paper, the following comparative experiments under the same conditions were carried out. The comparison data of the six groups are shown in Tables 5–7. Among them, the data results obtained by the method of this article are placed outside "()", the data results obtained by the commonly used method are placed inside "()", and ∞ indicates that the planning fails.

It can be obtained from Table 5 (Figures 6–8) that in the case of an obstacle, the difference between the two methods is not large, and the path planning task can be completed well. However, it can be seen from the Table 6 (Figures 9–11) of the two obstacles that the probability of occurrence of extreme points in the commonly used methods begins to increase, and even the phenomenon of planning failure occurs. From the simulation data of three or more obstacles shown in Table 7 (Figures 12–14), when the environment is more complicated, the improved method proposed in this paper shows a big advantage in terms of the number of steps, the cumulative angle change, and the cumulative end moving distance. From the comprehensive analysis of the data in Tables 5–7, it can be seen that when the improved obstacle avoidance path planning method is applied, the manipulator can complete the task with fewer steps, and the overall angle change is small, and the end moving distance is obviously shortened. The success rate of path planning has also increased significantly. It can be calculated that the improved method proposed in this paper reduces the possibility of the manipulator falling into the extreme point by about 27.78%, and reduces the possibility of planning failure by about 22.22%. Observe the relevant simulation results of two or more obstacles and analyze the data that both methods are planned successfully. It can be obtained that compared with the

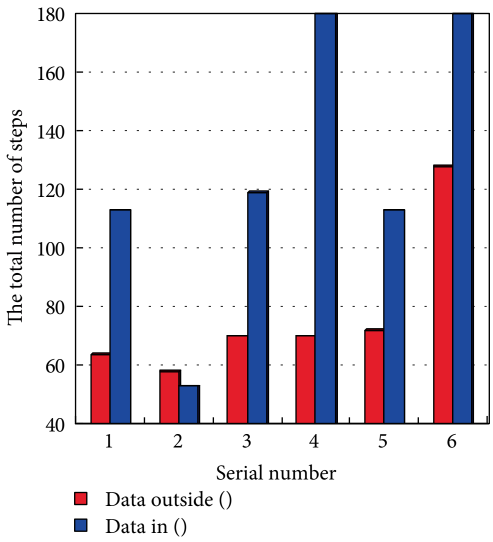


Figure 9: A comparison chart of s_{sum} when there are two obstacles.

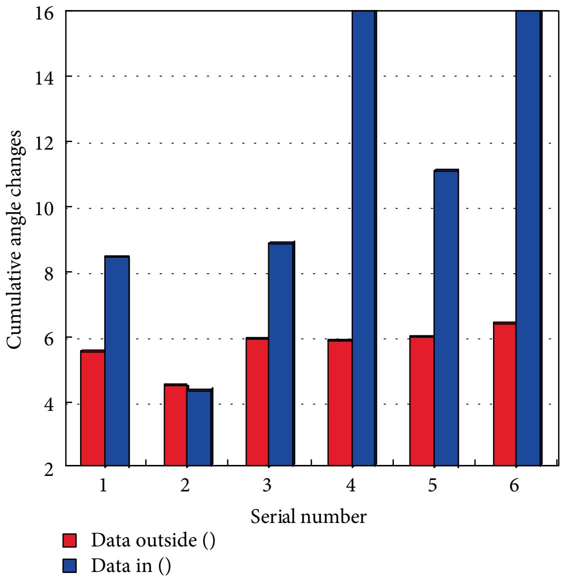


Figure 10: A comparison chart of $\Delta\theta_{\text{sum}}$ when there are two obstacles.

commonly used method, the improved method of this paper reduces the planned step number by about 27.37%, reduces the cumulative angle change by about 31.66%, and shortens the cumulative end moving distance by about 28.77%, which fully reflects the effectiveness and superiority of the method.

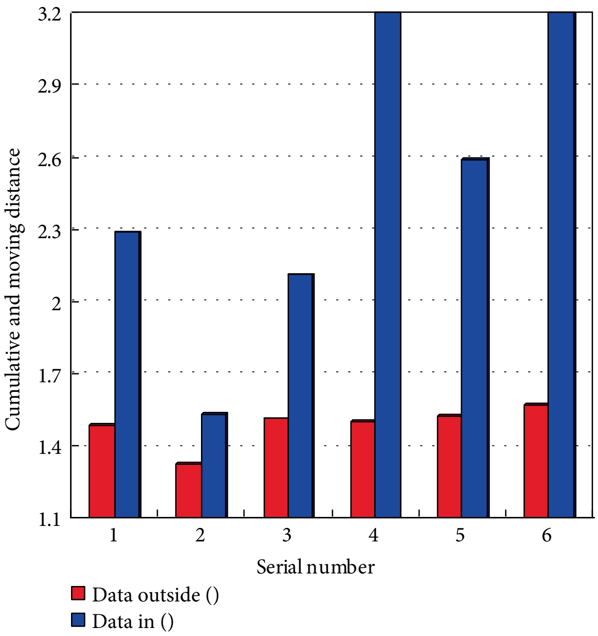


Figure 11: A comparison chart of ΔX_{sum} when there are two obstacles.

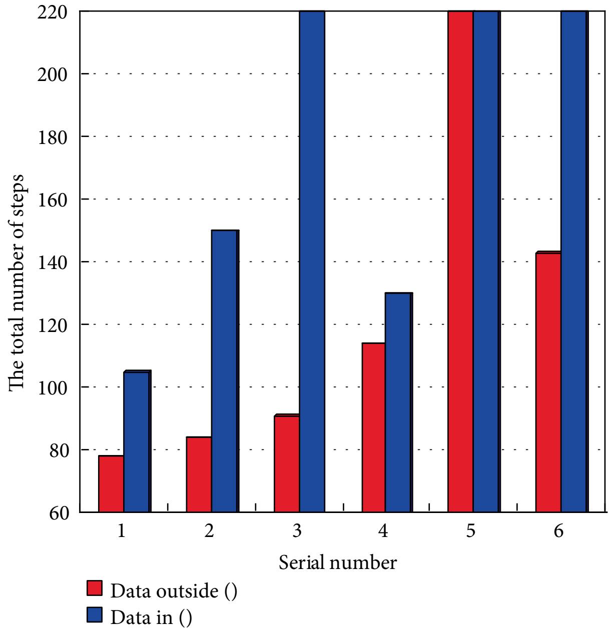


Figure 12: A comparison chart of s_{sum} when there are 3 or more obstacles.

6. Summary

Aiming at the problems existing in the commonly used path planning methods, this paper establishes a relatively complete mechanical arm obstacle avoidance path planning system by adopting a series of improvement measures such as establishing dynamic virtual target points and setting extreme point jump-out functions. The simulation results show that when dealing with complex working environment, the system still has the advantages of smooth running of the manipulator, continuous change of joint angle, less control steps, smaller cumulative angle change, and shorter cumulative end moving distance. It improves the success rate of path planning and also ensures path quality. The improved method proposed in this paper solves the problem of obstacle avoidance path planning of mechanical arm in three-dimensional space, which provides reference for manipulator control in saving energy and finding optimized path.

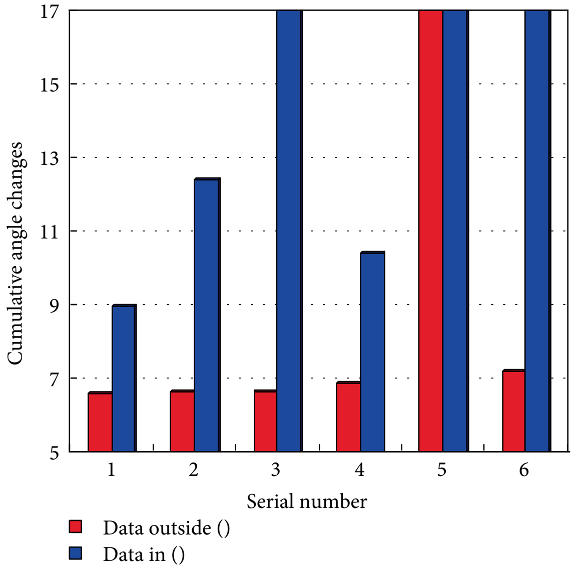


FIGURE 13: A comparison chart of $\Delta\theta_{\mathrm{sum}}$ when there are 3 or more obstacles.

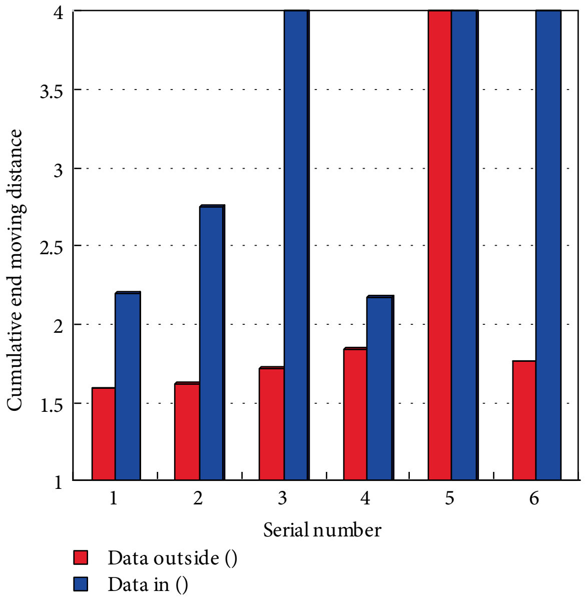


FIGURE 14: A comparison chart of ΔX_{sum} when there are 3 or more obstacles.

Data Availability

The data used to support the findings of this study are included within the article. The source program used to support the findings of this study is included within the supplementary information file.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Supplementary Materials

The source program used to support the findings of this study is included within the supplementary information file. (*Supplementary Materials*)

References

[1] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*, China Machine Press, Beijing, 2016.

[2] D. F. Zhang, H. Zhao, L. Xu, M. L. Liu, and J. Zhang, “Research on workspace analytic method of improved grid method based on industriaI robotic arm,” *Manufacturing Automation*, vol. 41, no. 4, pp. 69–70+90, 2019.

[3] P. Lozano and T. Rez, “Spatial planning: a configuration space approach,” *IEEE Transaction on Computers*, vol. 32, no. 2, pp. 108–120, 1983.

[4] P. Lozano and T. Rez, “A simple motion-planning algorithm for general robot manipulator,” *IEEE Journal of roboties and Automation*, vol. 3, no. 3, pp. 224–238, 1987.

[5] S. M. Lavalle, “Rapidly-exploring random trees: a new tool for path planning,” *Algorithmic & Computational Robotics New Direction*, pp. 293–308, 1998.

[6] L. Kavraki, P. Svestka, J. Latommbe, and M. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces,” *IEEE Transactions on Robotics & Automation*, vol. 12, no. 4, pp. 566–580, 1994.

[7] X. C. Chen and J. Huang, “Collision free path planning of robot manipulator based on artificial potential field and quantum-based particle swarm optimization,” in *Proceedings of the 36th Chinese Control Conference*, pp. 26–28, Dalian, China, 2017.

[8] Y. Dai and M. Zhao, “Manipulator path-planning avoiding obstacle based on screw theory and colony algorithm,” *Journal of Computational and Theoretical Nanoscience*, vol. 13, pp. 922–927, 2016.

[9] Y. Tan, “Robotic arm trajectory planning method by inserting optimal intermediate point searched by improved genetic algorithm,” *Manufacturing Technology & Machine Tool*, vol. 5, pp. 139–143+148, 2019.

[10] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” in *IEEE International Conference on Robotics and Automation*, pp. 90–98, St. Louis, MO, 2003.

[11] L. Xie and S. Liu, “Dynamic obstacle-avoiding motion planning for manipulator based on improved artificial potential filed,” *Control Theory & Applications*, vol. 35, no. 9, pp. 1239–1249, 2018.

[12] S. K. Wang, L. Zhu, and J. Z. Wang, “Six-degree-of-freedom mechanical arm obstacle avoidance path planning based on navigation potential function method,” *Transactions of Beijing Institute of Technology*, vol. 35, no. 2, pp. 186–191, 2015.

[13] J. L. Wang, G. L. Zhang, F. Yang, and B. Jing, “Improved artificial field method on obstacle avoidance path planning for manipulator,” *Computer Engineering and Applications*, vol. 49, no. 21, pp. 266–270, 2013.

[14] W. Ji, F. Y. Cheng, D. A. Zhao, Y. Tao, S. Ding, and J. D. Lü, “Obstacle avoidance method of apple harvesting robot manipulator,” *Transactions of The Chinese Society of Agricultural Machinery*, vol. 44, no. 11, pp. 253–259, 2013.

[15] Z. C. He, Y. L. He, and B. Zeng, “Obstacle avoidance path planning for robot arm based on mixed algorithm of artificial

potential field method and RRT," *Industrial Engineering Journal*, vol. 20, no. 2, pp. 56–63, 2017.

[16] J. Zhu and M. Y. Yang, "Path planning of manipulator to avoid obstacle based on improved artificial potential field method," *Computer Measurement & Control*, vol. 26, no. 10, pp. 205–210, 2018.

[17] Z. X. Cai, *Robotics Foundation*, Mechanical Industry Press, China, 2015.

[18] B. Q. Tian, J. C. Yu, and A. Q. Zhang, "Dynamic modeling of wave driven unmanned surface vehicle in longitudinal profile based on D-H approach," *Journal of Central South University*, vol. 22, no. 12, pp. 4578–4584, 2015.

[19] H. M. C. M. Herath, R. A. R. C. Gopura, and T. D. Lalitharatne, "An underactuated linkage finger mechanism for hand prostheses," *Modern Mechanical Engineering*, vol. 8, no. 2, pp. 121–139, 2018.

[20] X. Q. Lv and M. Zhao, "Application of improved BQGA in robot kinematics inverse solution," *Journal of Robotics*, vol. 2019, Article ID 1659180, p. 7, 2019.

[21] Y. Peng, W. Q. Guo, M. Liu, J. X. Cui, and S. R. Xie, "Obstacle avoidance planning based on artificial potential field optimized by point of tangency in three-dimensional space," *Journal of System Simulation*, vol. 26, no. 8, pp. 1758–1762+1768, 2014.

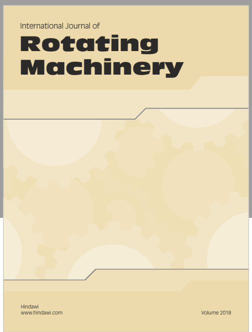
[22] L. Z. Chen and B. Q. Yu, *Mechanical Optimization Design Method*, Metallurgical Industry Press, Beijing, 2014.

[23] P. Y. Ma, *Research on Obstacle Avoidance Path Planning of Six-Degree-of-Freedom Mechanical Arm*, South China University of Technology, China, 2010.

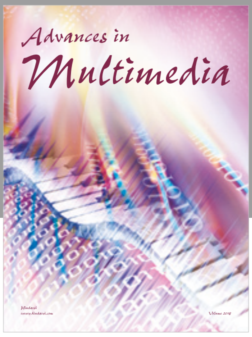
[24] S. Y. Li and P. C. Li, *Quantum Computation and Quantum Optimization Algorithm*, Harbin Institute of Technology Press, Harbin, 2009.

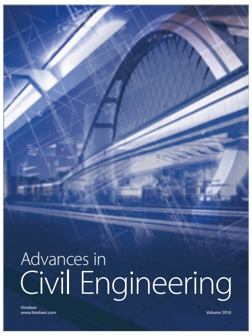
Journal of Engineering
Hindawi www.hindawi.com Volume 2018

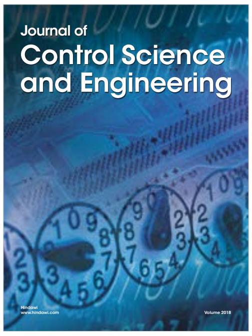
The Scientific World Journal
Hindawi www.hindawi.com Volume 2018


International Journal of Rotating Machinery
Hindawi www.hindawi.com Volume 2018

Journal of Sensors
Hindawi www.hindawi.com Volume 2018

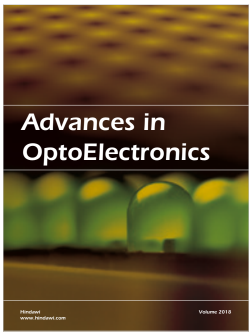

Advances in Multimedia
Hindawi www.hindawi.com Volume 2018


Advances in Civil Engineering
Hindawi www.hindawi.com Volume 2018


Journal of Control Science and Engineering
Hindawi www.hindawi.com Volume 2018

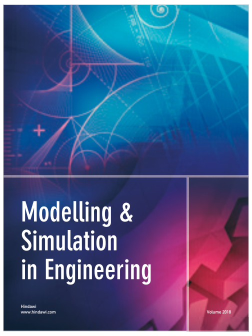
Journal of Robotics
Hindawi www.hindawi.com Volume 2018

Journal of Electrical and Computer Engineering
Hindawi www.hindawi.com Volume 2018

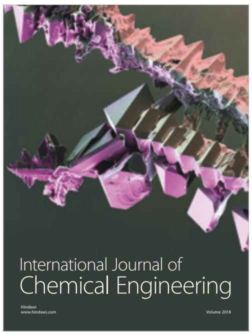

Advances in OptoElectronics
Hindawi www.hindawi.com Volume 2018

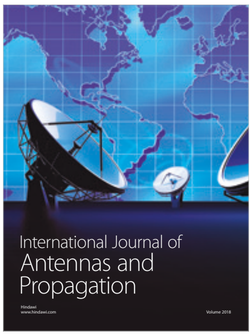
VLSI Design
Hindawi www.hindawi.com Volume 2018

International Journal of Navigation and Observation
Hindawi www.hindawi.com Volume 2018

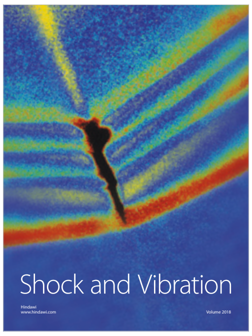

Modelling & Simulation in Engineering
Hindawi www.hindawi.com Volume 2018

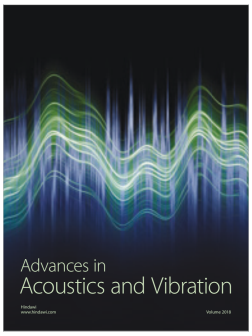
International Journal of Aerospace Engineering
Hindawi www.hindawi.com Volume 2018


International Journal of Chemical Engineering
Hindawi www.hindawi.com Volume 2018


International Journal of Antennas and Propagation
Hindawi www.hindawi.com Volume 2018

Active and Passive Electronic Components
Hindawi www.hindawi.com Volume 2018


Shock and Vibration
Hindawi www.hindawi.com Volume 2018


Advances in Acoustics and Vibration
Hindawi www.hindawi.com Volume 2018