International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-12 Issue-4, November 2023

Obstacle Avoidance Trajectory Planning for Robotic Arm based on Genetic Algorithm

Masood Usama, Xianhua Li, Haohao Yu, Yamin Iqra

***Abstract**: For the problem of obstacle avoidance trajectory planning of a robot arm, a robot arm obstacle avoidance method based on a genetic algorithm is proposed. It is based on two key advantages: the motion process can avoid obstacles and is more stable and efficient. First, the motion of each joint is planned as a sixth-degree polynomial, and the coefficients of the sixth-degree term are set as the pending parameters. The motion of each joint is changed by changing the pending parameters. Then, the fitness function is constructed by calculating the collision detection, angular velocity limit detection, acceleration limit detection, and the total trajectory length and rotation angle for each joint. Finally, the fitness function is optimised using a genetic algorithm to obtain smooth, continuous, and collision-free trajectories. Matlab simulation experiments show that this method can get the optimal or suboptimal trajectory without collision.*

***Keywords:** Genetic Algorithm; Obstacle Avoidance; Robotic Arm; Trajectory Planning*

I. INTRODUCTION

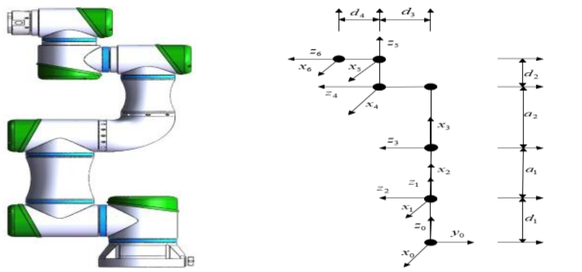
With the continuous progress of science, robotics is being increasingly used in various fields. The primary purpose of robotics is to replace human beings in performing tasks that are dangerous, complex, or heavy, thereby improving work efficiency. To make the robot work more safely and efficiently, firstly, the robotic arm must be controlled to avoid all obstacles, rather than being positioned at a single point. Secondly, the speed and acceleration of each joint should be continuous and not too large. Finally, the amount of movement of the robot arm during operation should be minimized. Currently, a large number of researchers have studied the obstacle avoidance path planning of robot arms. In the literature [1], the A* algorithm, a global path search algorithm, is used. It maps the Cartesian space onto the joint space, which in turn finds the free motion space of the robot arm, and then uses the A* algorithm for path search. Literature [2-3] uses the artificial potential field method. This is the classical algorithm, and the article improves the artificial potential field method. The algorithm applies a suction force to the target, causing the path to attract to the target point, and a repulsion force to the obstacle, causing the path to repel and avoid the obstacle. The literature [4-5], on the other hand, used the random search tree algorithm. With the development of artificial intelligence algorithms, researchers have begun to apply intelligent algorithms to robotic arms, such as the ant colony algorithm, particle swarm algorithm, and genetic algorithm, to name a few. Literature [6] has used ant colony algorithm for robot path planning. All of the above are for path obstacle avoidance planning, but only for path obstacle avoidance, without considering the problem of velocity and acceleration continuity. Literature [7] has used a genetic algorithm for obstacle avoidance of a robot arm using segmented planning, which allows continuity of acceleration as well as velocity, but the segmentation makes too many coefficients to be determined, which makes the computation too complex. To solve the above problems, the article refers to the literature [8] to solve the obstacle avoidance trajectory by changing the coefficients of the six terms of each joint planning, but the design of its fitness function tends to make the algorithm more time-consuming. At the same time, it is easy to fall into the local optimum and fail to achieve the optimal solution. In this article, we add angular velocity and angular acceleration bounds to the fitness function to optimise the algorithm. The acceleration and velocity of the trajectory are more reasonable, and the calculation process is faster.

Manuscript received on 11 October 2023 | Revised Manuscript received on 14 November 2023 | Manuscript Accepted on 15 November 2023 | Manuscript published on 30 November 2023.
** Correspondence Author (s)*
Masood Usama*, School of Mechanical Engineering, Anhui University of Science and Technology, China. E-mail: engr.automation1@gmail.com, ORCID ID: 0009-0006-8559-0645
Xianhua Li, School of Mechanical Engineering, Anhui University of Science and Technology, School of Artificial Intelligence, Anhui University of Science and Technology, China. E-mail: Xhli01@163.com
Haohao Yu, School of Mechanical Engineering, Anhui University of Science and Technology, China. E-mail: 85483529@qq.com
Yamin Iqra, School of Computer Science and Engineering, Anhui University of Science and Technology, China. E-mail: Iqrak6113@gmail.com

II. COLLISION MODEL

A. Robotic Arm Model

The article studies a six-degree-of-freedom robotic arm, as shown in Fig. 1(a) for the 3D model structure of the arm, and Fig. 1(b) represents the established simplified linkage coordinate system diagram. Table 1 shows the improved D-H parameter table.



(a)Robot Arm Modelling (b)Coordinate Diagram of Connecting Rod of Manipulator

Fig. 1 Coordinate Diagram of the Joint of the Manipulator

Tab.1 DH Parameters of the Manipulator

Connecting rod	$a_i/(mm)$	$\alpha_i/(mm)$	$d_i/(mm)$	$\theta_i/(^\circ)$
1	0	0	144	$\theta_1(0)$
2	0	$\pi/2$	0	$\theta_2(-90)$
3	-264	0	0	$\theta_3(0)$
4	-236	0	106	$\theta_4(-90)$
5	0	$\pi/2$	114	$\theta_5(0)$
6	0	$-\pi/2$	67	$\theta_6(0)$

B. Obstacle Models

Space obstacles are generally irregular geometric shapes which are too complex to be handled directly. Therefore, they need to be simplified. This paper uses a regular body to envelop the obstacle. Although this increases the volume of the obstacle, it dramatically simplifies the obstacle domain, thus simplifying the calculation. The regular body has a cylinder, a rectangle, a sphere, and so on. In general, the robot arm uses the cylinder to simplify the envelope. For obstacles, you need to look at their nearest regular body. In this paper, the obstacle is simplified by using a rectangular body for the envelope, as shown in Fig.2.

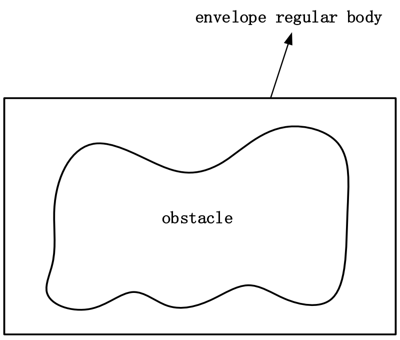


Fig. 2 Irregular Geometric Shape Obstacle Diagram

III. Collision Detection

This journal employs a double-blind review process, which means that both the reviewer(s) and author(s) identities are concealed from the reviewers, and vice versa, throughout the review process. All submitted manuscripts are reviewed by three reviewers, one from India and the other two from overseas. There should be proper comments from the reviewers for acceptance or rejection.

A. Rectangular Obstacle Space Description

Ellipse and ellipsoid are taken as the basic shapes in reference [9], and different shapes can be represented by varying the parameters in the function expression, and in three dimensions, the surface equation can be expressed as：

$$S(x,y,z)=0 \tag{1}$$

The points on this surface which are given by the above equation are those points on the surface where the centre of the obstacle is (x_0, y_0, z_0) The parameters of its geometry. The following analytical expression can describe its spatial surface. $S(x,y,z)=0$;

$$S(x,y,z)=\left(\frac{(x-x_0)}{h_1}\right)^{2m}+\left(\frac{(y-y_0)}{h_2}\right)^{2n}+\left(\frac{(z-z_0)}{h_3}\right)^{2p}-1 \tag{2}$$

For the description of the rectangular body, it is assumed that the rectangular body has side lengths 2a,2b,2c, centre coordinates (x0,y0,z0) and that its power indices and geometric parameters are satisfied;

$$\begin{cases} h_1=a \\ h_2=b \\ h_3=c \end{cases} \quad \begin{cases} m=4 \\ n=4 \\ p=4 \end{cases} \tag{3}$$

$$S(x,y,z)=\left(\frac{x-x_0}{a}\right)^8+\left(\frac{y-y_0}{b}\right)^8+\left(\frac{z-z_0}{c}\right)^8-1 \tag{4}$$

In the simulation, the mechanical arm will be simplified into a straight mechanical arm, but the actual mechanical arm also has a volume. Hence, you need to set the safety distance. The easiest way is to increase the length of the obstacle side, keeping the rectangle centred. However, the length of the arm side should increase the radius of the mechanical arm by a factor of two. As shown in Fig.3.

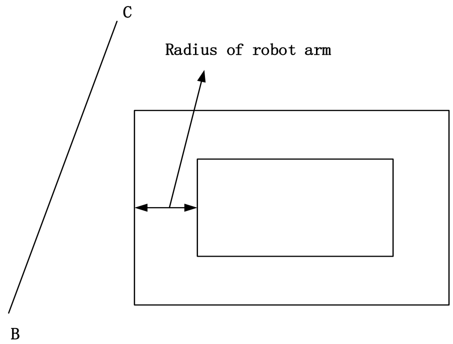


Fig. 3 Position Diagram of Connecting Rod and Obstacle

B. Collision Detection

In Fig.3 b,c are the two articulation points of the robot arm, and bc is the robot arm joint. From the robot kinematics, it can be concluded that the spatial position of the two points b and c is

$$T_B=T_n Rotz(\theta_1)Trans(0,0,d_1)Tans(a_1,0,0)Rots(\alpha_1) \tag{5}$$

$$T_C=T_n T_B Rotz(\theta_2)Trans(0,0,d_2)Tans(a_2,0,0)Rots(\alpha_2) \tag{6}$$

Equations (5) and (6) represent the chi-square articulated transformation matrices of the B and C joint points, so the spatial coordinates of the B and C points are

$$B=[x_b,y_b,z_b]=[T_B(1,4),T_B(2,4),T_B(3,4)] \tag{7}$$

$$C=[x_c,y_c,z_c]=[T_C(1,4),T_C(2,4),T_C(3,4)] \tag{8}$$

From this, we can derive the parametric equation of the space line BC

$$BC:\begin{cases} x=(x_c-x_b)t+x_b \\ y=(y_c-y_b)t+y_b \\ z=(z_c-z_b)t+z_b \end{cases} \tag{9}$$

Substitute the parametric equation for BC into equation (4).

If t has no solution, it means that the arm linkage does not collide.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

If t has a solution, there are two cases: bring t back into the parametric equation of the line BC and find out whether the point is on the line segment BC or not.

(1) If on the line segment BC, indicate a collision.

(2) If not in line segment BC, no collision.

Similarly, all the connecting rods on the arm are used to detect whether a collision occurs; if none of the connecting rods collide during the operation process, it indicates that the arm does not collide. The following figure shows the collision detection flowchart.

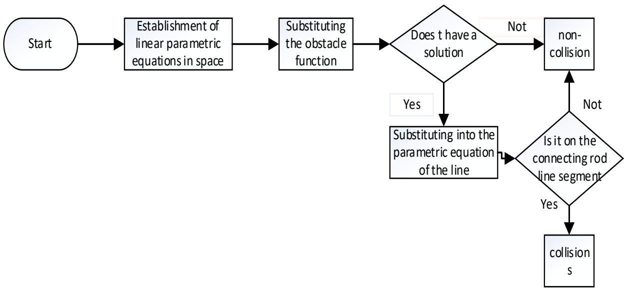


Fig. 4 Flow Chart of Collision Detection

IV. OBSTACLE AVOIDANCE SCHEME BASED ON GENETIC ALGORITHM

A. Fundamentals of Genetic Algorithms

A genetic algorithm is based on the principles of "natural selection" and the "survival of the fittest" law of evolution, simulating the biological evolution process using a random search algorithm. The algorithm has a general direction of application and is simple and easy to understand. The algorithm first randomly generates several individuals, each of which is evaluated through the fitness function to calculate its fitness value. This value can be derived by eliminating the lower fitness and retaining the higher fitness. The resulting population is then followed by a new population that utilises crossover, mutation, and genetic effects on the individuals. This cycle produces the final result. A sketch of the process is shown in Fig.5.

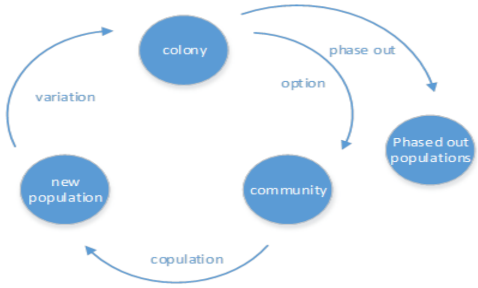


Fig. 5 Schematic Diagram of Genetic Algorithm Process

B. Sixth-Degree Polynomial Programming

Currently, most researchers use fifth-degree polynomials for planning robot arm trajectories. The advantage of fifth-degree polynomial planning is the smooth and continuous motion of the robot arm. The disadvantage is that the motion trajectory is unique, which is due to the six boundary conditions in the solution: start and stop angles, start and stop angular velocities, and start and stop accelerations. As a result, the robotic arm cannot avoid obstacles. Therefore, the article includes the six terms, allowing for an infinite number of solutions. The optimal trajectory that can avoid obstacles is found by solving for the coefficients of the sixth term. This is written as the following equation.

$$\theta_i = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 + a_5 T^5 + G T^6 \tag{10}$$

where G Is the parameter to be determined and θ_i Is the i-th joint angle. Let the start time be T_0 And the end time is T_f. Generally, the start time is 0, and then given six boundary conditions, the start and stop angles, start and stop angular velocities, and start and stop angular accelerations are θ_0, θ_f, $\dot{\theta}_0, \dot{\theta}_f, \ddot{\theta}_0, \ddot{\theta}_f$,r, respectively. This results in the following equation

$$\begin{cases} \theta_0 = a_0 \\ \theta_f = a_0 + a_1 T_f + a_2 T_f^2 + a_3 T_f^3 + a_4 T_f^{4.} + a_5 T_f^5 + G T_f^6 \\ \dot{\theta}_0 = a_1 \\ \dot{\theta}_f = a_1 + 2a_2 T_f + 3a_3 T_f^2 + 4a_4 T_f^3 + 5a_5 T_f^4 + 6G T_f^5 \\ \ddot{\theta}_0 = 2a_2 \\ \ddot{\theta}_f = 2a_2 + 6a_3 T_f + 12a_4 T_f^2 + 20a_5 T_f^3 + 30G T_f^4 \end{cases} \tag{11}$$

The coefficient $a_0 \sim a_5$ can be found when the parameter G The article sets the start and stop velocities, as well as the start and stop accelerations, to 0. The following equation

$$\begin{cases} a_0 = \theta_0 \\ a_1 = 0 \\ a_2 = 0 \\ a_3 = -(10\theta_0 - 10\theta_f + G T_f^6) / T_f^3 \\ a_4 = (15\theta_0 - 15\theta_f + G T_f^6) / T_f^4 \\ a_5 = -(6\theta_0 - 6\theta_f + G T_f^6) / T_f^5 \end{cases} \tag{12}$$

Equation (12) yields the $a_0 \sim a_5$ Coefficients, which can be substituted into equation (10) to obtain its trajectory.

C. Design of the Fitness Function

The adaptation function is designed to account for collision detection, acceleration, angular velocity limit detection, trajectory length, and rotation angle. Equation (13) is the adaptation function.

$$F_G = -\frac{f_p f_j f_s}{n_1 f_\theta + n_2 f_c} \tag{13}$$

Where f_p This represents collision detection, where a collision is denoted by zero, and no collision is represented by one. f_i The acceleration limit detection is as follows: exceeding the acceleration limit is 0, not exceeding is 1. f_s The velocity limit detection, exceeding the velocity limit is 0, not exceeding is 1. f_θ It is the sum of the joint angular variables. f_c It is the sum of the end-effector motions. n_1, n_2 The weighting coefficients are 1 and 0.005, respectively. The inclusion of f_θ and f_c The purpose of the above equations is to reduce the movement of its robotic arm while avoiding obstacles, which not only makes the robotic arm energy-efficient but also allows it to complete the operation efficiently. Equations (14) and (15) are solved for f_θ and f_c.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

L is solved by using the kinematic positive solution to obtain the final chi-square matrix, which leads to its final spatial coordinate points.

$$f_\theta = \sum_{i=1}^{n}\sum_{j=1}^{6}\left|\theta_{i+1,j} - \theta_{i,j}\right| \tag{14}$$

$$f_c = \sum_{i=1}^{n}\sqrt{(L_{i+1,x} - L_{i,x})^2 + (L_{i+1,y} - L_{i,y})^2 + (L_{i+1,z} - L_{i,z})^2} \tag{15}$$

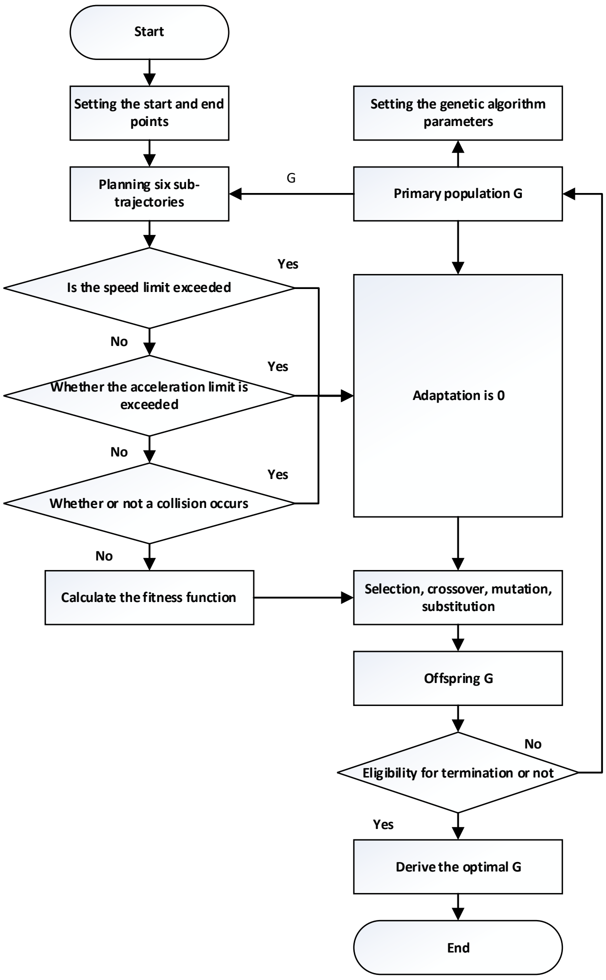


Fig. 6 Flowchart of Obstacle Avoidance Scheme

The addition of q and e to the fitness function has the following advantages: firstly, the fitness is directly set to 0 if some unreasonable values are calculated, thus skipping the collision detection, which requires complex calculations, and can significantly shorten the arithmetic process. Secondly, considering the lifetime of the robot arm, a limit is imposed on its speed and acceleration to extend its lifetime. Finally, when the genetic algorithm runs, it may lead to local optimisation. By adding constraints, it can escape from this local optimisation and achieve global optimisation. Fig.6 shows the flowchart of the entire obstacle avoidance algorithm.

V. EXPERIMENTAL SIMULATION

A. Establishment of Robotic Arms and Obstacles

In this paper, the Robotic Toolbox and Genetic Algorithm Toolbox are jointly established in MATLAB. Firstly, the robot arm model is set up in MATLAB, and the start joint angle is set to [2*pi/5, pi/4, pi/2, 0, 0, 0], and the end joint angle is set to [-pi/5, pi/4, pi/4, 0, pi/4, 0]. The obstacle position centre coordinates are [-400, -200, 0], and the side length of the rectangular obstacle is [300, 200, 500]. The movement time is 10 seconds. To determine the typical situation of the robot arm, we will consider whether it will collide. The Monte Carlo method is used to determine the working space of the robotic arm to determine that the obstacle is within the working space, while the trajectory is planned with a 5th-degree polynomial to find that the trajectory passes through the obstacle, as shown in Fig.8.

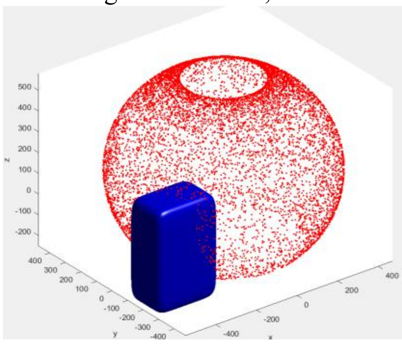


Fig.7 Monte Carlo workspace

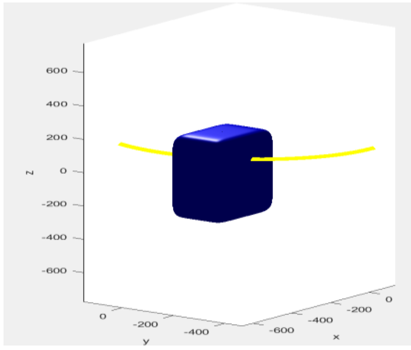


Fig. 8 Relationship between Quintic Polynomial Trajectory and Obstacles

B. Genetic Algorithm for Solving Optimal G and Analyzing the Results

First, the genetic parameters were set, and to reduce the amount of computation, the G values were artificially determined to range from [-12 -12 -12 -12 -12 -12 -12] * 1e-5 to [12 12 12 12 12 12 12 12] * 1e-5, G is a matrix with one row and six columns, because the robot arm has six joints, so the G value should be six.

The angular velocity limit is set to 0.5rad/s, the angular acceleration limit is set to 0.4rad/s2, the number of populations is set to 50, the probability of hybridisation is set to 0.8, the number of elites is set to 3, the termination condition is iterated to 20 generations, and the function is set to the roulette wheel method, The optimal value of G was derived after approximately 10 minutes as [0.2027 0.5488 -0.0171 -0.3314 0.0952 0.2860]*1e-4, with a fitness function of -0.0853, a final movement distance of 1071.7/mm and a joint angle increment of 6.3689/rad. The above iterations were performed 20 times. The average value of its fitness and the optimal value change by iteration are shown in Fig. 9, and it can be seen that the optimal value approaches a smooth curve after seven generations.

The process of the robot arm's movement and the trajectory of the final movement are shown in Fig. 10. From Fig. 10, it can be seen that this trajectory can avoid obstacles.

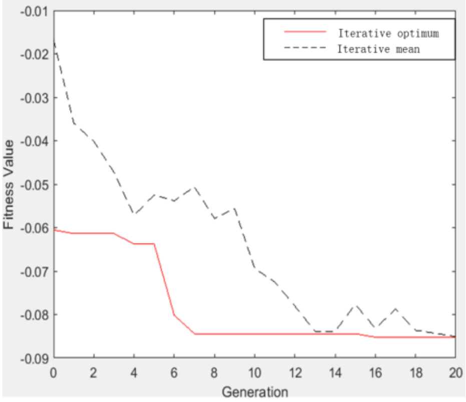


Fig. 9 Iterative Changes of Mean and Optimal Fitness Values

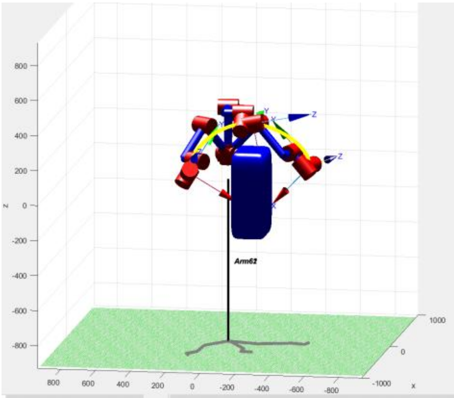
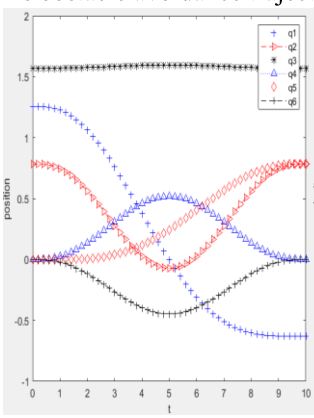


Fig. 10 Motion Process Diagram of the Manipulator

The motion of each joint during the movement is shown in Fig.11, the angular velocity of each joint is shown in Fig.12, the angular acceleration of each joint is shown in Fig.13, and the units of each vertical coordinate in Fig.11, Fig.12, and Fig.13 are rad, rad/s, and rad/s2, respectively. The units of the horizontal coordinate are all s. From Figs. 11, 12, and 13, it is easy to see that the motion of the robot arm is smooth and continuous. At the same time, the limit requirements of the robot arm are not exceeded. The result proves that the robot arm's obstacle avoidance trajectory planning is reasonable.



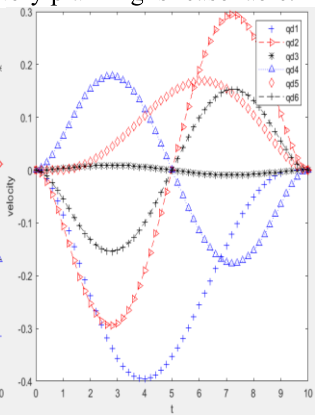


Fig.11 Angles of Each Joint Fig.12 Angular Velocity of Each Joint

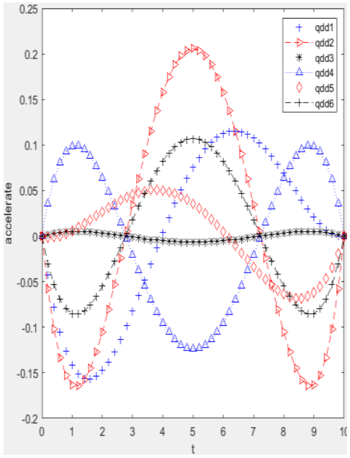


Fig. 13 Angular Acceleration of Each Joint

VI. CONCLUSION

The design of the fitness function in this paper significantly reduces the computation time and can quickly approximate the optimal solution at the beginning of the operation. And the obtained trajectory results are more reasonable, so the life of the robot arm can be strengthened.

The robot arm's obstacle avoidance method uses a genetic algorithm, and the resulting trajectory is not an optimal solution, but a near-optimal solution that meets the requirements. Finding the optimal solution requires numerous iterations and increases the population size. This results in the optimum taking a lot of time to calculate, which is not very meaningful.

The collision detection model is configured to accommodate a variety of regular bodies. The direct function solution method is simple and does not require consideration of the spatial geometric relationship between the connecting rod and the obstacle, thereby avoiding complicated calculations.

ACKNOWLEDGMENTS

This work was supported by the Key Research and Development Program of Anhui Province (2022i01020015)

DECLARATION STATEMENT

Funding	No, I did not receive.
Conflicts of Interest	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval or consent to participate, as it presents evidence.
Availability of Data and Material	Not relevant.
Authors Contributions	All authors have equal participation in this article.

REFERENCES

1. JIA Qingxuan, CHEN Gang, SUN Hanxu, ZHENG Shuangqi. Obstacle avoidance path planning for spatial robotic arm based on A~** algorithm[J]. Journal of Mechanical Engineering,2010,46(13):109-115.

https://doi.org/10.3901/JME.2010.13.109
2. Chen Jinfeng, Huang Weihua, Zhang Zheng, Li Legend. Path planning algorithm based on improved artificial potential field method in dynamic environment[J]. Combined machine tools and automated machining technology,2020(12): 6- 9+14.
3. ADELI Hossein，TABRIZI M H N，MAZLOOMIAN Alborz. et al ．Path Planning for Mobile Robots Using Iterative Artificial Potential Field Method[J]． International Journal of Computer Science Issue，2011，8(44) : 28-32
4. Biyun Xie, Jing Zhao, Yu Liu. Motion planning of 7R robotic arm obstacle avoidance reach point based on fast extended random tree[J]. Journal of Mechanical Engineering,2012,48(03):63-69. https://doi.org/10.3901/JME.2012.03.063
5. VERAS Lgdo，MEDEIROS Fll，GUIMARAES Lnf． Rapidly Exploring Random Tree with a Sampling Method Based on Sukharev Grids and Convex Vertices of Safety Hulls of Obstacles,[J], International Journal of Advanced Robotic Systems，2019，16(1): https://doi.org/10.1177/1729881419825941
6. Jiaojiao Wang, Jianlin Mao. Mobile robot path planning based on improved ant colony algorithm in a dynamic environment[J]. Information and Communication Technology,2019,13(05):28-32.
7. QI Ruolong, ZHOU Weijia, WANG Tiejun. A spatial robotic arm obstacle avoidance trajectory planning method based on genetic algorithm[J]. Robotics,2014,36(03):263-270.
8. Ma Yuhao, Liang Yanbing. A robot arm obstacle avoidance algorithm based on sixth-degree polynomial trajectory planning[J]. Journal of Northwestern Polytechnical University,2020,38(02):392-400. https://doi.org/10.1051/jnwpu/20203820392
9. by W. F. Xu, B. Liang. Redundant space robot manipulator arm Kinematics, trajectory planning and control [M]. [Science Press]. [2017.11]. [336 pp]. [14345225]

AUTHORS PROFILE

Masood Usama, a master's student at Anhui University of Science and Technology, specialising in the School of Mechanical Engineering, which is deeply immersed in the field of robotics with a specific focus on AI-powered Adaptive Rehabilitation Exoskeletons. His research area encompasses a broad spectrum of robotics, including Robot Design and Control, Kinematics, and Dynamics, particularly in the context of robot arms coordinated operation. He explores the intricate interplay of mechanical design, advanced control systems, and the application of artificial intelligence to create cutting-edge rehabilitation exoskeletons. The student aims to enhance the versatility and adaptability of these systems, ultimately contributing to the advancement of AI-powered robotics in the realm of rehabilitation and beyond.

Xianhua Li is a Professor in Mechanical Engineering at the University of Science and Technology. He obtained a Ph.D. majoring in Mechatronics Engineering at Shanghai University in China. He taught mechanical engineering-related subjects at the School of Mechanical Engineering and the School of Artificial Intelligence. His research area includes robotics, specifically robot design and Control, Robot Kinematics and Dynamics, Coordinated Operation Control of Robot Arms, and Rehabilitation Robotics, including tip-over and Slip Avoidance.

Yu Haohao is a master's student in the School of Mechanical Engineering at Anhui University of Science and Technology. His research direction focuses on robots and rehabilitation robot arms. He is primarily interested in mechanical structures, kinematics simulation, and related topics. He has published papers.

Yamin Iqra is a master's student in the School of Mechanical Engineering at Anhui University of Science and Technology. Her research direction focuses on face detection using machine learning (artificial intelligence). She is mainly interested in artificial intelligence.