Robot arm trajectory planning study for a table tennis robot

Bo Zhang[1], *Bingqiang* Chen[2], and *Yansong* Deng[2,*]

[1]College of electrical engineering, Southwest Minzu University Chengdu, Sichuan, China
[2]College of electronic information, Southwest Minzu University Chengdu, Sichuan, China

Abstracts. The 4 degree of freedom robot arm of a table tennis robot has a variety of trajectories. In order to improve the response and the success rate of the shots, we used the joint space trajectory planning method to establish a kinematic model with the robot arm joints as variables, and by combining it with the robot arm kinematics, we obtained the relevant parameters for each joint of the robot arm. Simulation tests and physical tests were carried out to obtain a more accurate trajectory of the robot arm.
Keywords: 4 degrees of freedom robotic arm; Path planning; Joint parameters; Robotic arm simulation.

1 Introduction

In order to promote the intelligence of the sport we have developed a table tennis robot, which in this paper only focuses on the robot arm. A common problem among all types of robots that can be used for human-robot sparring is that they are not accurate enough to hit the ball[1]. Often the racket does not reach the specified position or the racket reaches the specified position after the ball, but the trajectory planning is not perfect. The so-called trajectory planning is: every movement of the robot arm will form a certain trajectory in space. The trajectory refers to the position, velocity and acceleration of each degree of the robot arm's freedom in the predetermined coordinate system of the time course. Trajectory planning can be divided into two cases, Cartesian space planning and joint space planning. Most traditional human-robot sparring robots use Cartesian spatial planning, using the homeopathic coordinate nodes of joints in space to constitute trajectories, which may result in singularities, causing instability of trajectories. In this paper, based on this problem, the method of joint space trajectory planning is used to simulate the robot arm as a kinematic model with joints as reference variables, and by combining the model with the kinematic laws of the robot arm, the relevant parameters of each joint can be obtained, and the best path selection is finally completed[2-5].

2 Principle of trajectory planning

We carry out the path planning of the robot arm by means of joint trajectory planning, where

---

* Corresponding author: 342547288@qq.com

the joints are variable and their position, velocity and acceleration are represented as smooth functions of time by studying the relationship between the motion of the joints and the time angle and velocity. We have chosen a polynomial representation by comparison, which allows a good match between the initial and end boundary positions[6-8]. We use a cubic polynomial interpolation, which requires at least four constraints, the initial angle θ(0) at time t0, the initial angle θ(tf) at time tf, and the joint velocity of zero at the initial and terminal moments in order to ensure that the joint velocity is continuous, so the four constraints are:

$$\theta(0) = \theta_0; \theta(t_f) = \theta_f; \dot{\theta}(0) = 0; \dot{\theta}(t_f) = 0 \tag{1}$$

With these four constraints we can uniquely determine a cubic polynomial which is

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t \tag{2}$$

$$\dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2 \tag{3}$$

$$\ddot{\theta}(t) = 2a_2 + 6a_3 t \tag{4}$$

From the inverse solution of the kinematics of the robot arm we already know that:

$$\theta_3' = atan\,2(s_3, \pm c_3) =$$
$$atan\,2\left(c_{12}a_x + s_{12}a_y, \pm\sqrt{1 - \left(c_{12}a_x + s_{12}a_y\right)^2}\right) \tag{5}$$

$$-s_{12}o_x + c_{12}o_y = -c_4 \tag{6}$$

$$s_4 = \pm\sqrt{1 - c_4^2} = \pm\sqrt{1 - \left(s_{12}o_x - c_{12}o_y\right)^2} \tag{7}$$

$$\theta_4 = atan\,2(s_4, \pm c_4) =$$
$$atan\,2\left(\pm\sqrt{1 - s_{12}o_x - c_{12}o_y}^2, s_{12}o_x - c_{12}o_y\right) \tag{8}$$

Bringing the four constraints into the above equation yields four equations containing four unknown quantities:

$$a_0 = \theta_0; \tag{9}$$

$$a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 = \theta_f \tag{10}$$

$$a_1 = 0 \tag{11}$$

$$a_1 + 2a_2 t_f + 3a_3 t_f^2 = 0 \tag{12}$$

The solution yields:

$$a_i: a_0 = \theta_0; \tag{13}$$

$$a_1 = 0 \tag{14}$$

$$a_2 = 3(\theta_f - \theta_0)/t_f^2 \tag{15}$$

$$a_3 = -2(\theta_f - \theta_0)/t_f^3; \tag{16}$$

If the arm is required to pass each intermediate point in succession, and if the desired velocity of each joint at the intermediate point is known, the constraint is then:

$$\theta(0) = \theta_0; \theta(t_f) = \theta_f; \dot{\theta}(0) = \theta_0; \dot{\theta}(t_f) = \dot{\theta}_f \tag{17}$$

The four equations described are:

$$a_0 = \theta_0; \tag{18}$$

$$a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 = \theta_f \tag{19}$$

$$a_1 = \dot{\theta}_0 \tag{20}$$

$$a_1 + 2a_2 t_f + 3a_3 t_f^2 = \dot{\theta}_f \tag{21}$$

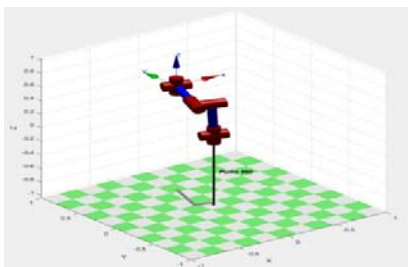The solution yields:

$$a_i: a_0 = \theta_0; \tag{22}$$

$$a_1 = \dot{\theta}_0 \tag{23}$$

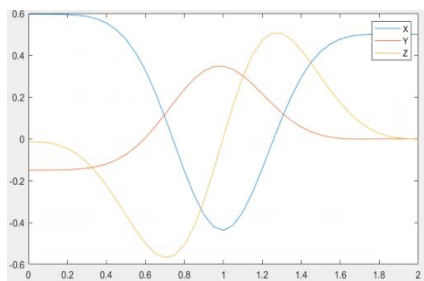$$a_2 = 3(\theta_f - \theta_0)/t_f^2; \tag{24}$$

$$a_3 = -2(\theta_f - \theta_0)/t_f^3 \tag{25}$$

3 Simulation model construction and data analysis

3.1 Simulation and data logging

Fig. 1. Simulation Model of manipulator.

Fig. 2 End trajectory of manipulator.

Table 1. Relationship between joint angle and time.

	0.5s	1s	1.5s	2s
Joint 1 angle	0 rad	0 rad	0 rad	0 rad
Joint 2 angle	0.2 rad	0.7 rad	1.4 rad	1.6 rad
Joint 3 angle	0 rad	0 rad	0 rad	0 rad
Joint 4 angle	-0.2 rad	-0.75 rad	-1.4 rad	-1.6 rad

Table 2. Relationship between joint speed and time.

	0.5s	1s	1.5s	2s
Joint 1 speed	0 rad/s	0 rad/s	0 rad/s	0 rad/s
Joint 2 speed	1.5 rad/s	3.0 rad/s	1.5 rad/s	0 rad/s
Joint 3 speed	0 rad/s	0 rad/s	0 rad/s	0 rad/s
Joint 4 speed	-1.5 rad/s	-3 rad/s	-1.5 rad/s	0 rad/s

Table 3. Joint acceleration as a function of time.

	0.5s	1s	1.5s	2s
Joint 1 acceleration	0 rad/s^2	0 rad/s^2	0 rad/s^2	0 rad/s^2
Joint 2 acceleration	8.0 rad/s^2	0 rad/s^2	-8.0 rad/s^2	0 rad/s^2
Joint 3 acceleration	0 rad/s^2	0 rad/s^2	0 rad/s^2	0 rad/s^2
Joint 4 acceleration	-8.0 rad/s^2	0 rad/s^2	8.0 rad/s^2	0 rad/s^2

We have used MATLAB to build a simulation model of the robot arm and set up corresponding detection points to record the position, angle and acceleration of the joints of the arm in different states and plot them as images. The main ideas in MATLAB are as follows: firstly, we build a spatial model, establish a coordinate system, integrate the robot arm model into the coordinate system and limit the rotation angle of the robot arm by means of an algorithm, so that the data can be better detected. During the detection process, an error data reminder is also set up and an error message is output when the robot arm is switched off at a negative angle. In the simulation it is also necessary to substitute the positive and negative solution formulas for the kinematics of the robot arm so that the individual parameters of the joint can be solved. To ease of observation, we plot the obtained robot arm parameters versus time as an image to better aid the analysis. The following figure shows the simulation model of the manipulator in MATLAB and the image of the position of the end effector of the manipulator[9-11].

3.2 Data analysis

By graphing the data obtained from the experiments, we can conclude that
1. The velocity and time curves for joints 1 and 3 are approximately straight, the reason for the fluctuation is that the error caused by overshoot during velocity adjustment, after repeated experiments, this error does not affect the trajectory planning.
2. The angular image of joint 2 is similar to a peak shape, rising at moment 0 but stabilising after moment 1.5, while joint 4 is a downhill shape, achieving a minimum and stabilising after moment 1.5. There is a symmetrical relationship between joint 2 and joint 4 in terms of mechanical structure, which also makes the images of both symmetrical. There is also a symmetrical relationship between joint 2 and joint 4 in terms of the velocity image.
3. In terms of acceleration the acceleration of joint 1.3 tends to be stable, but the acceleration of joint 2.4 is sinusoidal and varies by approximately half a cycle.

3.3 Method innovation

We observe the swing model of the manipulator, and get that the complete range of motion of the whole manipulator is a hemisphere, and the manipulator can be retracted, so the range of motion is a solid sphere with the radius of the manipulator. However, the research on the whole manipulator is too large. We have locked the joints of the manipulator. We can regard the motion of each joint as the "circle drawing" motion in the air. Because we know the angular velocity of each joint, we can calculate the linear velocity of part of the manipulator through the relationship between angular velocity and linear velocity. At this time, we can know the length of the arc drawn by the manipulator, Because the length of each part of the manipulator is known, the radius of the sphere in which the manipulator is located can be calculated by using the known angle, and according to the circle of each joint, the radius of the sphere in which each segment of the manipulator is located can be obtained by the same operation mentioned above. At this time, there are three spheres with known radius. The coordinates of each manipulator joint can be obtained by establishing the coordinate system between the planes, and the coordinates of each manipulator fixed point can be obtained by cooperating with the angle calculation. Compared with the traditional kinematic trajectory, this method has less computation and more efficient, and can be more widely used.

4 Testing of the actual model

We changed the robot arm several times in the simulation and wrote corresponding parameters into microcontroller, through comparative analysis, we can see the correctness of the trajectory, but there is still an little error, which may due to the existence of feedback adjustment overshoot factors, but in general, the trajectory planned by the above method can be better for completing the set task, the response speed in the process of hitting and the accuracy rate and the accuracy of the striking process are greatly improved.

5 Summary

In this paper, we establish a spatial coordinate system and use joint space trajectory planning to plan the trajectory of the robotic arm. Through this method, we can obtain a more accurate path, and with the control of the drive we can better complete the hitting of the table tennis ball. After many experiments, it is found that although it is very convenient to use our spherical coordinate solution method to plan the trajectory, there are some hidden dangers in the stability, which need to be improved in our follow-up research.

References

1. Shi Xiangling, Fang Honggen. Time-energy-pulsation optimal trajectory planning for industrial robots[J]. Machine Design and Manufacture,2018(06):254-257.
2. Fu R,Jui HH. Time-optimal robot arm trajectory planning algorithm based on particle swarm optimization[J]. Information and Control,2011,40(06):802-808.
3. Zhou S, Kong JY, Hou Y, et al. Improved ABC algorithm for multi-objective optimization of tandem robotic arms[J]. Combined machine tools and automated machining technology,2019(5):31-35.
4. JIN Wenke, ZHANG Hua, LIU Manlu. NURBS algorithm-based redundant robotic arm trajectory planning[J]. Science, Technology and Engineering,2014,14(28):226-230.
5. Song Xinyi. Research on robot intelligent handling system based on machine vision[D]. Harbin: Harbin Institute of Technology, 2018.
6. Zhou B,Qin L, Gong W. Weighted guided map filtering stereo matching algorithm based on Gaussian Laplacian operator[J]. Advances in Lasers and Optoelectronics, 2019, 56(10):234-240.
7. Cao Y, Zhang Yabin, Zhou Y, et al. Research on improved RRT-Connect-based space manipulator arm obstacle avoidance path planning (in English)[J]. Machine Tools and Hydraulics,2020,48(12):177-183.
8. Qin Chao, Liang Xifeng, Lu J, et al. Trajectory planning and simulation of a seven-degree-of-freedom tomato harvesting manipulator [J]. Journal of Zhejiang University (Engineering Edition), 2018,52(7):1260-1266+1274.
9. Wang Zhaofei. kinematic simulation analysis of 6R articulated robot [J]. Combined machine tools and automated machining technology, 2018(11):62-66.
10. Tang JY, Zhang JJ, Wang XH, et al. An improved robot trajectory planning method [J]. Machine Design, 2017,34(3):31-35.
11. Teng Rumin, Li Yuxin, Wang Xin, Zhao Zhe. Time-optimal trajectory planning of a lifted fire truck based on convex optimization[J]. Journal of Mechanical Engineering,2019,55(6):138-144