Research on Trajectory Planning of Tobacco Palletizing Robot

Haibo Lin, Yingjun Xiong, Ling Lu

Chongqing University of Posts and Telecommunications, Chongqing 400065, China

1359056282@qq.com

Abstract

This paper presents an industrial robotic trajectory planning method for tobacco palletizing systems. Based on the simple polynomial interpolation algorithm in joint space, the commonly used cubic polynomial interpolation and quintic polynomial interpolation algorithms are combined into a hybrid polynomial interpolation algorithm. This method ensures that the position and velocity curves of the robot of the tobacco have smooth features. This paper uses Matlab to generate the trajectory image of the calculated data, which lays the foundation for further research on the robot control system.

Keywords

Trajectory planning; Palletizing system; Polynomial interpolation.

1. Introduction

With the rapid development of logistics technology and industrial automation, industrial palletizing robots have become emerging technologies in the field of automation. In recent years, it has been widely used in the tobacco logistics industry and widely used to solve the problems of laboriousness, fatigue and error-prone of the existing manual stacking and is of great significance to the improvement of distribution efficiency. The tobacco palletizing robot must perform specific tasks at the end of the tobacco logistics and must ensure its grip trajectory to avoid sudden changes in speed or torque to ensure smooth palletizing operations. In order to obtain a stable and accurate robotic palletizing trajectory, this paper proposes a hybrid interpolation algorithm based on quintic interpolation and cubic interpolation for trajectory planning.

2. Tobacco palletizing robot structure

As shown in Fig. 1, the mechanical model of the tobacco palletizing manipulator has six degrees of freedom. The base adopts a floor-standing type. Although the movement of the wrist is small, it requires an extremely compact structure. In the case of sufficient strength, weight as light as possible, for the use of connecting rod lever claw. The manipulator is composed of a base, a revolving waist, an arm, an arm, a hand bowl and a gripper. There are six rotating joints in total. The position of the wrist reference point can be determined by the first three joints. The posture of the wrist can be determined by the last three joint to determine.

In this paper, the design of the robot arm D-H coordinate system shown in Fig. 2, each joint can be rotated and translated. To model the links and joints between robots using the D-H notation, you must assign a basic reference coordinate system to each joint. The axis of the coordinate system is in the direction of rotation with the right-hand rule. If it is a sliding joint, it is defined as the direction of joint positive linear motion as the axis. If it is a rotating joint, press the right hand rule and the direction of the thumb is the axial direction. The x-axis of the local reference coordinate system is the adjacent two coordinates The vertical axis of the system.

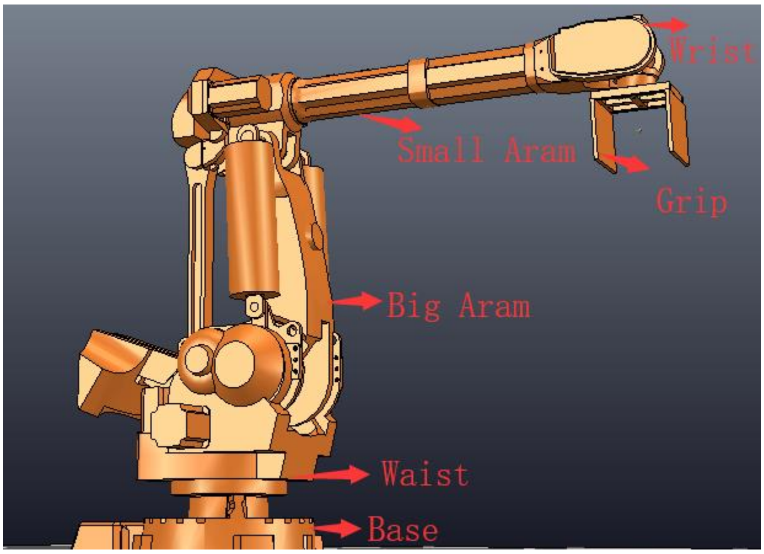

Fig. 1 Tobacco palletizing robot

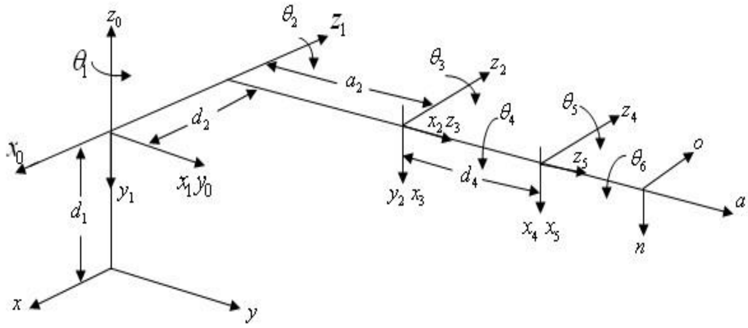

Fig. 2 Robot D-H coordinate system

Determine the parameter values of each connecting rod and joint according to the definition rules of connecting rod parameters to get the D-H parameter table as follows:

Table 1 D-H parameters of robot joints

Joint	θ_i	α_i	a_i	d_i	Joint Variable Range	Joint variable range (Relative to the initial position)
1	θ_1	-90°	0	d_1=190	$-90^\circ \sim 270^\circ$	$-180^\circ \sim 180^\circ$
2	θ_2	0	a_2=215	d_2=32.5	$-180^\circ \sim 0^\circ$	$-180^\circ \sim 0^\circ$
3	θ_3	90°	0	0	$-45^\circ \sim 225^\circ$	$-135^\circ \sim 135^\circ$
4	θ_4	-90°	0	d_4=220	$-180^\circ \sim 180^\circ$	$-180^\circ \sim 180^\circ$
5	θ_5	90°	0	0	$-135^\circ \sim 135^\circ$	$-135^\circ \sim 135^\circ$
6	θ_6	0	0	0	$-180^\circ \sim 180^\circ$	$-180^\circ \sim 180^\circ$

3. Kinematics Solution Based on Matlab

The Robotics toolbox toolbox provided by Matlab provides us with many useful functions including: positive and inverse kinematics solution, trajectory planning. It can also be used to analyze the positive and inverse kinematics of the manipulator. Because of the positive and inverse kinematics of the manipulator, The computational complexity and computational complexity of the solution are large. Therefore, the positive and inverse solution functions provided by matlab Robotics toolbox can be used to obtain the output quickly, and the positive and inverse solutions can be quickly calculated, laying the foundation for the motion control of the manipulator. The first step is to set up a robotic arm model, and then find a positive and inverse solution to it.

To create a robotic object, we can use the link and robot functions in the Robotics Toolbox toolbox to build a robotic object based on the D-H parameter table we set up earlier. The code is as follows:

```
d1=190;     d2=32.5;     a2=215;     d4=220;
L1=link([-pi/2 0 pi/2 d1 0]);
L2=link([0 a2 0 d2 0]);
L3=link([pi/2 0 pi/2 0 0]);
L4=link([-pi/2 0 0 d4 0]);
L5=link([pi/2 0 0 0 0]);
L6=link([0 0 0 0 0]);
r=robot({L1 L2 L3 L4 L5 L6});
r.name='Tobacco Palletizing Robot';
drivebot(r);
```

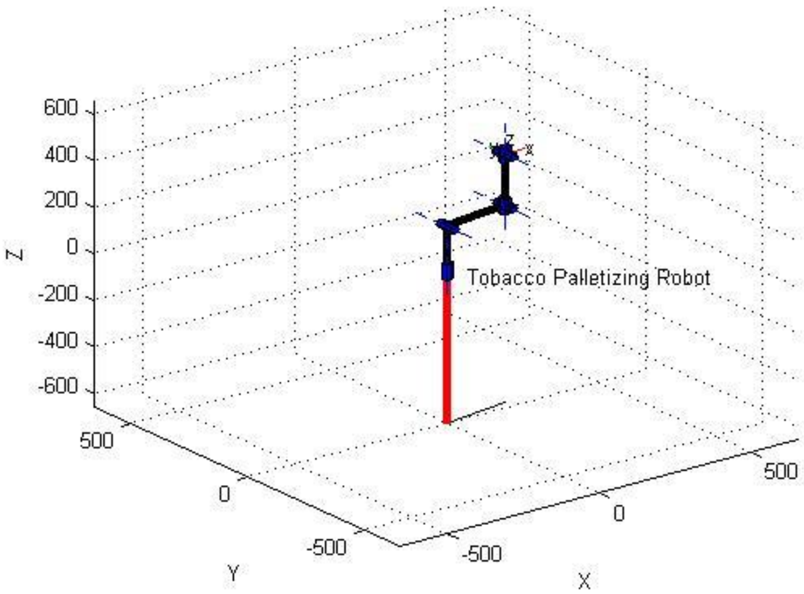

Fig. 3 Manipulator simulation model

The fkine function in Robotics Toolbox can solve the positive solution of the kinematics of the manipulator. The fkine function call format is:

$$TR = FKINE(ROBOT, Q) \tag{1}$$

The parameter ROBOT is the established robot object, TR is the positive solution to each forward kinematics defined by Q, and the angle at which each joint of the manipulator moves is $q = \begin{bmatrix} 180^0 & -45^0 & -45^0 & 30^0 & 0 & 0 \end{bmatrix}$;Use the function fkine (r, q) to find TR:

$$TR = \begin{bmatrix} -0.0000 & -0.0000 & 1.0000 & 23.9720 \\ -0.5000 & -0.8660 & -0.0000 & -32.5000 \\ 0.8660 & -0.5000 & 0.0000 & 342.0280 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{2}$$

TR is the desired posture that the end effector of the manipulator reaches when the joint angle rotates according to $q = \begin{bmatrix} 180^0 & -45^0 & -45^0 & 30^0 & 0 & 0 \end{bmatrix}$.The first three rows in the matrix represent the gesture of the robot grasping the object. The last column $\left[23.9720 \ \ -32.5000 \ \ 342.0280\right]$ represents the three-dimensional coordinates of the target object relative to the robot arm. The ikine function in Robotics Toolbox can solve the inverse problem of robot kinematics. The call format of ikine function is:

$$\mathrm{q} = IKINE\left(ROBOT, TR\right) \tag{3}$$

The parameter ROBOT is a manipulator object, and TR is a transformation matrix that requires an inverse solution, that is, the expected pose of the manipulator end effector. We can use the positive solution matrix TR from the above to try to find the inverse solution to see if it can be restored. By using matlab statement, the solution is:

$$q = \left[3.1415 \quad -0.78537 \quad -0.78537 \quad 0.52358 \ \ 0 \ \ 0\right] \tag{4}$$

Converted into an angle that is:

$$q = \left[180^0 \quad -45^0 \quad -45^0 \quad 30^0 \quad 0 \quad 0\right] \tag{5}$$

Which is the same as the original initial value. It can be proved that the manipulator model built in this paper is correct using the function provided in the Matlab toolbox, which verifies the rationality of the mechanism and verifies the establishment of DH reference frame Correctness.

4. Trajectory Planning of Polynomial Interpolation Functions

4.1 Cubic Polynomial Interpolation

Given the initial and final pose of the manipulator, it is mapped to the joint space through the kinematics equation to obtain the joint angle corresponding to the start and end moments of the manipulator. It is assumed that the angle value of a certain joint of the manipulator at an initial time t_0 is θ_0, and the angle at an ending time t_f is θ_f. By using the four constraints of the angle θ_0 and θ_f between the initial moment and the final moment and the angular velocity $\dot{\theta}_0$ and $\dot{\theta}_f$, a smooth trajectory curve can be obtained. The following three polynomial interpolation will satisfy these four constraints:

$$\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 \tag{6}$$

Find the first derivative of the cubic polynomial (6):

$$\dot{\theta}(t) = c_1 + 2c_2 t + 3c_3 t^2 \tag{7}$$

The initial state into the type (6) and (7):

$$\begin{cases} \theta(t_0) = c_0 = \theta_0 \\ \theta(t_f) = c_0 + c_1 t_f + c_2 t_f^2 + c_3 t_f^3 = \theta_f \\ \dot{\theta}(t_0) = c_1 = \dot{\theta}_0 \\ \dot{\theta}(t_f) = c_1 + 2c_2 t_f + 3c_3 t_f^2 = \dot{\theta}_f \end{cases} \tag{8}$$

Solving equation (8) yields the coefficients of polynomial (9):

$$\begin{cases} c_0 = \theta_0 \\ c_1 = \dot{\theta}_0 \\ c_2 = \dfrac{3}{{t_f}^2}(\theta_f - \theta_0) - \dfrac{2}{t_f}\dot{\theta}_0 - \dfrac{1}{t_f}\dot{\theta}_f \\ c_3 = -\dfrac{2}{{t_f}^3}(\theta_f - \theta_0) + \dfrac{1}{{t_f}^2}(\dot{\theta}_f + \dot{\theta}_0) \end{cases} \tag{9}$$

Through the above formula to determine the robot arm to meet the speed requirements of the two poses when moving between the various joint axis angle curve. The following will use MATLAB simulation to analyze cubic polynomial interpolation: a robot arm angle in 4 seconds from the initial point A through the middle point B to reach the end of C changes shown in Fig. 4. The velocities and angular velocities at three locations are shown in the equation:

$$\begin{array}{lll} \theta_A = 30 & \theta_B = 60 & \theta_C = 40 \\ \dot{\theta}_A = 20 & \dot{\theta}_B = 30 & \dot{\theta}_C = 20 \end{array} \tag{10}$$

In Fig. 4, the solid line is the angle curve, and the dotted line is the curve of the angular velocity. The joint angle curve is smooth, while the speed curve shows a sudden change at the middle point B.

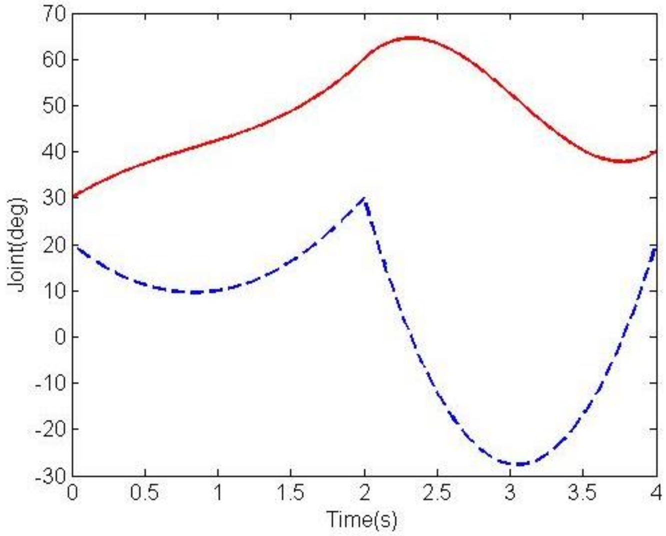

Fig. 4 Cubic polynomial interpolation

4.2 Quintic Polynomial Interpolation

Quintic polynomial interpolation method will solve the cubic polynomial interpolation in the joint angular velocity is not smooth. The quintic polynomial interpolation needs to determine the position, velocity and acceleration of the robot arm at the start point and the end point of the path segment. At this moment, the constraint condition reaches six. The quintic polynomial and its first derivative, the second derivatives are:

$$\begin{cases} \theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 \\ \dot{\theta}(t) = c_1 + 2c_2 t + 3c_3 t^2 + 4c_4 t^3 + 5c_5 t^4 \\ \ddot{\theta}(t) = 2c_2 + 6c_3 t + 12c_4 t^2 + 20c_4 t^3 \end{cases} \tag{11}$$

Substituting the constraints into Eq. (11):

$$\begin{cases} \theta_0 = c_0 \\ \theta_f = c_0 + c_1 t_f + c_2 t_f{}^2 + c_3 t_f{}^3 + c_4 t_f{}^4 + c_5 t_f{}^5 \\ \dot{\theta}_0 = c_1 \\ \dot{\theta}_f = c_1 + 2c_2 t_f + 3c_3 t_f{}^2 + 4c_4 t_f{}^3 + 5c_5 t_f{}^4 \\ \ddot{\theta}_0 = 2c_2 \\ \ddot{\theta}_f = 2c_2 + 6c_3 t_f + 12c_4 t_f{}^2 + 20c_4 t_f{}^3 \end{cases} \tag{12}$$

Equation (12) is a system of linear equations with 6 equations and 6 unknowns, the solution is:

$$\begin{cases} c_0 = \theta_0 \\ c_1 = \dot{\theta}_0 \\ c_2 = \dfrac{\ddot{\theta}_0}{2} \\ c_3 = \dfrac{20\theta_f - 20\theta_0 - (8\dot{\theta}_f + 12\dot{\theta}_0)t_f - (3\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^3} \\ c_4 = \dfrac{30\theta_0 - 30\theta_f + (14\dot{\theta}_f + 16\dot{\theta}_0)t_f + (3\ddot{\theta}_0 - 2\ddot{\theta}_f)t_f^2}{2t_f^4} \\ c_5 = \dfrac{12\theta_f - 12\theta_0 - (6\dot{\theta}_f + 6\dot{\theta}_0)t_f - (\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^5} \end{cases} \tag{13}$$

The simulation of quintic polynomial interpolation by Matlab is similar to the simulation of cubic polynomial interpolation. The change of an articulated arm angle from the initial point A through the middle point B to the end point C in 4 seconds is analyzed. In order to compare the two interpolation methods, the velocities and angular velocities of the three positions are the same as those of the cubic polynomial interpolation, and the constraint of increasing the angular acceleration is as shown in formula (14):

$$\begin{array}{lll} \theta_A = 30 & \theta_B = 60 & \theta_C = 40 \\ \dot{\theta}_A = 20 & \dot{\theta}_B = 30 & \dot{\theta}_C = 20 \\ \ddot{\theta}_A = 2 & \ddot{\theta}_B = 4 & \ddot{\theta}_C = 2 \end{array} \tag{14}$$

The solid line in the figure is the curve of angle change, the broken line is the curve of angular velocity, and the dotted line is the curve of angular acceleration.

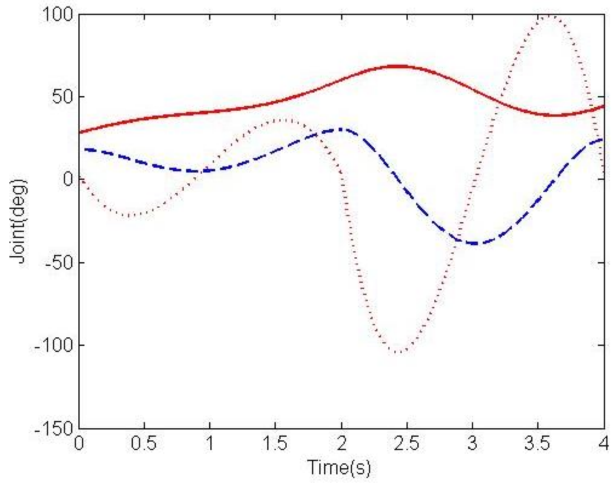

Fig. 5 Quintic polynomial interpolation

The use of high-order polynomial interpolation function to plan trajectories can make robots run more smoothly. The quintic polynomial interpolation method can obtain the smoother curves of angle and angular velocity by adding more constraints, which can solve the problem of the sudden change of the joint angular velocity in cubic polynomial interpolation. Because of the increase of constraints, the computational complexity also increases and the derivation process becomes more complicated.

5. Industrial Robots Polynomial Hybrid Interpolation Algorithm

Combining the advantages of cubic polynomials and quintic polynomial interpolation algorithms, a continuous path from tobacco palletizing includes the process of picking up packets of cigarettes from the palletizing robot to the specified positions of the pallets and then lowering the packets of cigarettes. We need to limit the acceleration at both the beginning and the end so that the cigarette packs can be smoothly lifted, the middle section we use a smaller amount of cubic polynomial interpolation algorithm.

Known planning starting point A, intermediate path points B, C and end D. The corresponding joint angles are shown in Fig. 6, respectively. The position structure consisting of the first three joints mainly realized the spatial position of the working point, and the last three joints determined the posture of the manipulator. For the control of the end position of the manipulator, only the trajectory of the interpolation algorithm must be planned for the first three joints.

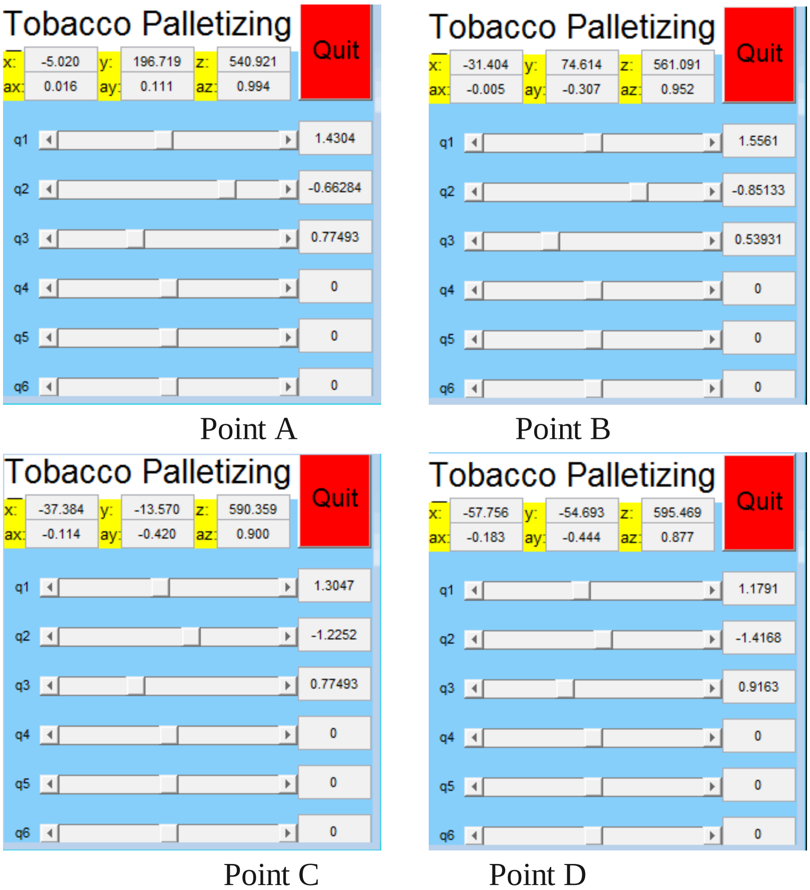

Fig. 6 Starting point, Path point, End point

Using 5-3-5 trajectory planning, AB segment interpolation with the quintic polynomial, BC segment with cubic polynomial interpolation, CD segment with quintic polynomial interpolation, the starting point, the target point speed, acceleration is zero, at the middle point B, C Department of continuous, can seek derivatives. Using MATLAB simulation to analyze the results of the trajectory planning of joint cubic and quintic polynomial algorithm in joint space, as shown in Fig. 7 ~ 9:

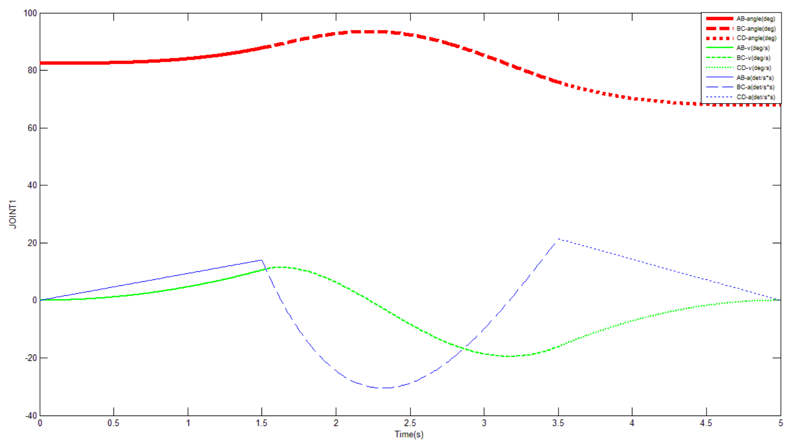

Fig. 7 Angle, speed, acceleration of joint one

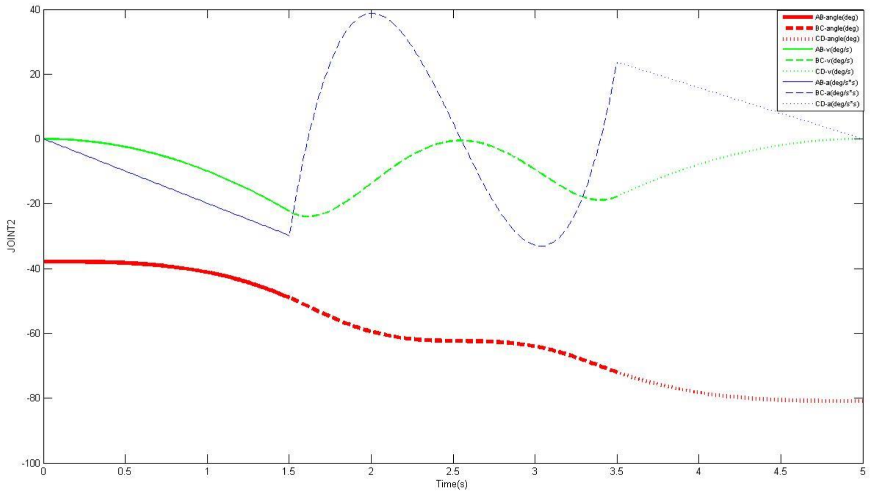

Fig. 8 Angle, speed, acceleration of joint two

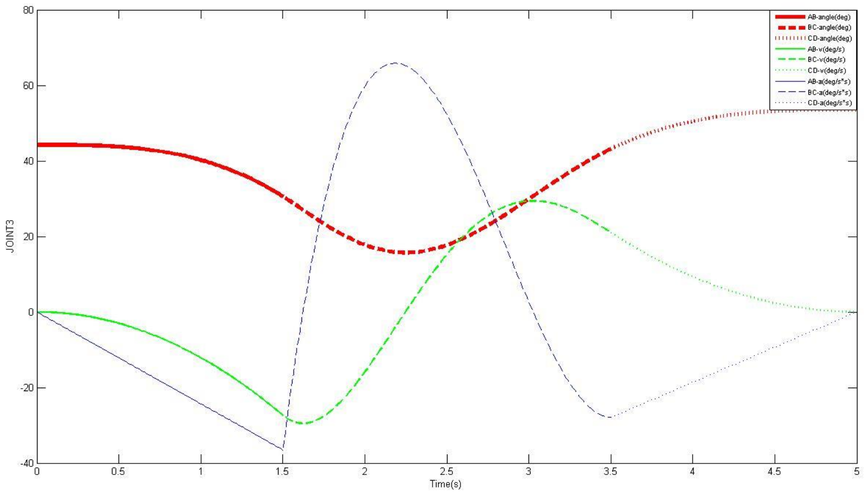

Fig. 9 Angle, speed, acceleration of joint three

Due to the small calculation error, the curve in the graph fails to show perfect smoothness and smoothness, but the angle curve and velocity curve are relatively smooth, and the acceleration is constrained in the initial section and the final section so that the acceleration Continuous, in line with our tobacco palletizing requirements.

6. Conclusion

This article describes the mechanics of tobacco palletizing robots. Using its D-H parameters, modeling and kinematics analysis in matlab, and finally trajectory planning of tobacco palletizing robot. Based on the process of industrial robots palletizing cigarette packages, we use the quintic interpolation and the cubic interpolation algorithm to combine. Finally, the trajectory planning of the polynomial hybrid interpolation algorithm is simulated in Matlab to generate the corresponding trajectory image. The polynomial hybrid interpolation algorithm proposed in this paper provides a reference for the research of trajectory planning of industrial robot palletizing robot.

References

[1] Guanbin Gao,Guoqing Sun,Jing Na,Yu Guo,Xing Wu. Structural parameter identification for 6 DOF industrial robots[J]. Mechanical Systems and Signal Processing,2017.

[2] Wei You,Minxiu Kong,Lining Sun,Yanbin Diao. Control system design for heavy duty industrial robot[J]. Industrial Robot: An International Journal,2012,39(4).
[3] Torgny Brogårdh. Present and future robot control development—An industrial perspective[J]. Annual Reviews in Control,2007,31(1).
[4] Xia Zhengya, Hong Liang.Trajectory planning of palletizing robot based on polynomial fitting interpolation function [J]. Shandong Science,2016,29(05):117-123.
[5] GUO Ruifeng, LI Yanhe, SHI Shihuai. Trajectory Planning and Kinematics of Joint Palletizing Robot [J]. Mechanical design and manufacturing,2017(01):76-78+82.
[6] SUN Yue, WEI Xin. Trajectory Planning of Palletizing Robot Based on Fifth Polynomial [J]. Packaging Engineering,2017,38(21):159-163.
[7] Xiong Yanmei, Yang Yandong.Kinematics analysis and simulation of palletizing robot [J]. Mechanical,2015,42(01):62-66.
[8] CHEN Xiao-Ju, LI Hui, WU Xian-Gang.Motion Planning of a Pliant Robotic Palletizing Robot with Optimal Time [J]. COMPUTER MACHINERY & PROCESSING AUTOMATION, 2017(04):18-21+25.
[9] Cao Chengtao, Xu Lunhui, Sun Yongjia, Zhao Xue.Study on Kinematics and Trajectory Interpolation Algorithm of Industrial Palletizing Robot [J] .Machinery Design and Manufacturing, 2017(03): 254-257.
[10] Dong Hang, Du Guangsheng, Liu Dong, Cong Ming.Particular path planning of palletizing robot based on genetic algorithm [J]. Process Technology of Combined Machine Tool and Automation, 2017(05):29-31+35.
[11] ZHANG Yu, DING Lei-yu.Study on Structural Design and Trajectory Planning of Six-Joint Industrial Robot [J] .Mechanical Engineer,2017(11):8-10.
[12] Gao Jun-tao, Yuan Si-cong, Cao Ying-hui, Li Jian-wei.Optimal design of industrial palletizing robot trajectory based on ISIGHT platform [J] .Machinery & Automatic Manufacturing Technology,2015(09):55-59.
[13] Gao Shuai, Du Shanshan. Research on trajectory planning of industrial robot [J] .Computer Simulation,2015,32(10):409-413.
[14] Li Chengwei, Zhu Xiuli, Jia Chao.Study on mechanism design and control system of palletizing robot [J].Electromechanical Engineering,2008,25(12):81-84+99.
[15] Li Hao, Huang Yan, Ma Yanwei.A Smoothing Compression Algorithm Based on Cubic Polynomial Curves [J]. COMPUTER MACHINERY AND PROCESSING AUTOMATION, 2016(06): 12-15.
[16] SUN Ying, CHENG Wen-tao, LI Gong-fa, KONG Jian-yi, JIANG Guo-zhang, LI Zhe. Research status of trajectory planning algorithm and trajectory planning for articulated robots [J] .Journal of Yangtze University2016,13(28):32-38+5.
[17] LI Ju-chen, ZENG Yi-hui. Simulation of Working Space and Trajectory Planning of Six-Joint Industrial Robot [J]. FOOD AND MECHANICAL,2017,33(10):98-103.