Research on 3-DOF Manipulator Based on Motion Analysis and Trajectory Algorithm

Qianhui Tao [1, #], Yafei Li [2, #, *], Xiaoming Wang [3, #]

[1] School of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing, China, 210094

[2] School of Automobile, Chang'an University, Xi'an, China, 710021

[3] School of Automotive Engineering, Shandong Jiaotong University, Jinan, China, 250357

* Corresponding Author Email: 2022900189@chd.edu.cn

#These authors contributed equally.

Abstract. Aiming at the precise control of three-degree-of-freedom manipulator under complex operation, the kinematics model of manipulator and its joints is established by D-H parameter method. Through the analysis of forward kinematics and inverse kinematics of the manipulator, quintic polynomial interpolation algorithm is selected to study the trajectory planning of the manipulator in Cartesian space coordinate system. Through Matlab Robotic Tool and adams simulation analysis, the displacement, velocity and acceleration curves of each joint with time are obtained, and then the accuracy of the algorithm is verified. This method can provide theoretical analysis and basis for other research on kinematics and trajectory planning of multi-joint robots. This study establishes an accurate kinematic model of the 3-degree-of-freedom manipulator and optimizes the trajectory planning algorithm, providing a theoretical and simulation basis for improving its control precision in complex operations. It not only helps solve practical issues such as balancing model accuracy and real-time performance for low-degree-of-freedom manipulators but also provides a reference for research on kinematics and trajectory planning of multi-joint robots, which has positive significance for promoting the engineering application of manipulator automation technology.

Keyword: D-H modeling, manipulator kinematics, quintic polynomial interpolation algorithm, Matlab adams, trajectory planning.

1. Introduction

The advancement of industrial robot technology in China has promoted the widespread application of 3-degree-of-freedom manipulators. However, due to their limited operational range, practical applications often face challenges. Although extensive research has been conducted on the kinematics and trajectory planning of such manipulators, Yin Zhijian[3] proposed the mechanism design and trajectory planning of a 3-DOF forearm rehabilitation exoskeleton; Zhao Xiong et al. [9] focused on the dynamic analysis and lightweight design of a 3-DOF apple-picking manipulator; Lu Enpeng[1] explored the time-optimal trajectory planning of 6-DOF industrial manipulators; and Du Xiaohui and Yu Tao[6] established a UR10 manipulator model using the Robotic Toolbox in MATLAB, conducted simulation studies on it, and realized trajectory planning in joint space through the quintic polynomial interpolation method, proving that the established model operates stably and correctly. Yet, issues still exist in achieving a balance between model accuracy and real-time performance, optimizing trajectories under complex constraints, and ensuring the engineering adaptability of algorithms. To address this, this paper introduces a kinematic analysis method for 3-degree-of-freedom manipulators. This method enhances the flexibility of the model by improving the kinematic solution algorithm and integrates intelligent optimization with traditional technologies to develop an efficient trajectory planning strategy. The proposed method provides a theoretical basis and technical guidance for the precise control and effective operation of 3-degree-of-freedom manipulators in practical scenarios, thereby promoting the advancement and innovation of robotic automation technology. [4].

2. Kinematic modeling and analysis of manipulator

2.1. establishment of coordinate system of manipulato

The industrial robot arm selected in this paper has three degrees of freedom, each degree of freedom is driven by a rotary joint, and mainly includes four parts: base, joint, arm structure and end effector. UG software is used to establish 3D model of robot arm, and its modeling process includes drawing robot arm parts sketch, creating 3D model of parts and assembling parts, etc. Its basic structure is shown in Figure 1. Coordinate system is established at joints of robot arm by standard DH parameter method, as shown in Figure 2.

Figure 1. The overall structure of the robotic arm

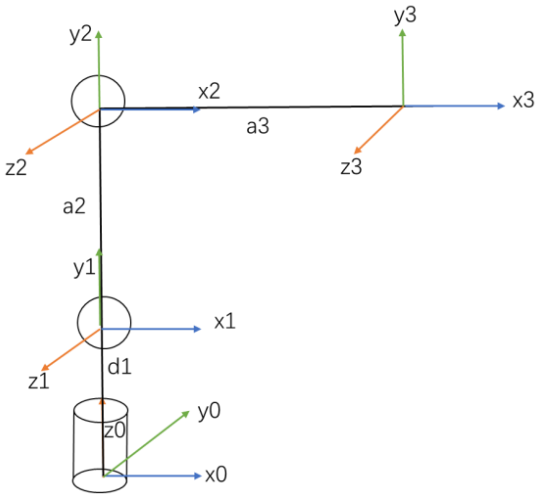

Figure 2. D-H coordinate system

Determine the relevant parameters of the manipulator according to the coordinate relationship in Figure 2, see Table 1. By using homogeneous transformation matrix to construct coordinate transformation chain in solid space, kinematics mapping between robot joint coordinate systems is established, and finally the position and attitude of end tool coordinate system relative to base coordinate system are solved.

Table 1. D-H parameters of the robotic arm

Connecting rodi	θi(°)	di(mm)	ai(mm)	αi(°)
1	θ1	d1(70)	0	90
2	θ2	0	a2(80)	0
3	θ3	0	a3(150)	0

2.2. Forward Kinematic Analysis

According to the coordinate transformation rule, the transformation matrix of the robot arm link (i-1) coordinate system to the adjacent link (i) coordinate system is:

$$ {}_{i}^{i-1}T = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

The transformation matrix from the base to the end effector is obtained by multiplying the transformation matrix of each link in turn, and the pose of the end effector can be obtained according to the joint angle[5].

The pose transformation matrix of the three joints of the robot arm shown in FIG. 1 is established by using equation (1):

$$ {}_{1}^{0}T = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 1 & 0 & d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

$$ {}_{2}^{1}T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

$$ {}_{3}^{2}T = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & a_3\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & a_3\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

The pose transformation matrix of the manipulator end effector relative to the base coordinate system is:

$$ {}_{3}^{0}T = {}_{1}^{0}T\,{}_{2}^{1}T\,{}_{3}^{2}T = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5} $$

Solve the attitude equation and position equation of the terminal:

$$ \begin{aligned} nx &= \cos(\theta_2+\theta_3)cos\theta_1 \\ ny &= \cos(\theta_2+\theta_3)sin\theta_1 \\ nz &= \sin(\theta_2+\theta_3) \end{aligned} \tag{6} $$

$$ \begin{aligned} ox &= -\sin(\theta_2+\theta_3)cos\theta_1 \\ oy &= -\sin(\theta_2+\theta_3)sin\theta_1 \\ oz &= \cos(\theta_2+\theta_3) \end{aligned} \tag{7} $$

$$ \begin{aligned} ax &= sin\theta_1 \\ ay &= cos\theta_1 \\ az &= 0 \end{aligned} \tag{8} $$

$$ \begin{aligned} px &= cos\theta_1(a3\cos(\theta_2+\theta_3) + a2cos\theta_2) \\ py &= sin\theta_1(a3\cos(\theta_2+\theta_3) + a2cos\theta_2) \\ pz &= d1 + a3\sin(\theta_2+\theta_3) + a2sin\theta_2 \end{aligned} \tag{9} $$

2.3. Inverse kinematics analysis

The inverse kinematics solution of manipulator is to solve the joint angles of manipulator for the known pose matrix of manipulator end. The parameters of each link of the manipulator are known. According to the given position and attitude of the end effector, the joint angles θi (i=1, 2, 3) are calculated by inverse kinematics. Due to the complexity of the calculation, some calculation steps are omitted, and the expression results are listed.

$$ax = sin\theta_1 \tag{10}$$

$$ay = cos\theta_1 \tag{11}$$

Divide equation (10) and equation (11) and solve for

$$\theta_1 = arctan\frac{ax}{ay} \tag{12}$$

$$nz = \sin(\theta_2 + \theta_3) \tag{13}$$

$$oz = \cos(\theta_2 + \theta_3) \tag{14}$$

Divide equation (14) and equation (15) and solve for

$$\theta_2 + \theta_3 = atctan\frac{nz}{oz} \tag{15}$$

$$pz = d1 + a3\sin(\theta_2 + \theta_3) + a2sin\theta_2 \tag{16}$$

$$nz = \sin(\theta_2 + \theta_3) \tag{17}$$

Equation (16), Equation (17) and Equation (18) can be solved to obtain

$$\theta_2 = arcsin\frac{pz - d1 - a3nz}{a2} \tag{18}$$

$$\theta_3 = atctan\frac{nz}{oz} - arcsin\frac{pz - d1 - a3nz}{a2} \tag{19}$$

3. Implementation of trajectory control algorithm and kinematics simulation of manipulator based on quintic polynomial interpolation

3.1. Manipulator trajectory control algorithm based on quintic polynomial interpolation

Because of its continuous first-order and second-order derivative characteristics, quintic polynomial interpolation is suitable for trajectory planning of robotic arms. It can ensure smooth transition of velocity and acceleration during motion and reduce mechanical impact [6-8].

The specific steps are as follows:

(1) Given n points (n<=6, because there are 6 coefficients in the quintic polynomial), let the polynomial be:

$$P(x) = a0 + a1x + a2x2 + a3x3 + a4x4 + a5x5 \tag{20}$$

(2) Substitute each point (x_i, y_i) into the polynomial to obtain a linear equation system and solve for coefficients a_0 to a_5.

(3) Compared with other methods, quintic polynomial interpolation can optimize the flexibility and adaptability of the manipulator.

3.2. Modeling and Kinematic Simulation

Kinematics simulation of manipulator can directly reflect the spatial trajectory change of end-effector, and the displacement, velocity and acceleration change of each joint.

(1) Establishment of manipulator model

Under Matlab software environment, using Matlab Robotic Tool simulation toolbox, using Link () and robot () according to D-H parameters

The commands to model a 3-DOF robot object are as follows:

```
d1 = 0.10; % Base Height
a1 = 0.15; % first arm
a2 = 0.12; % second arm
L1 = Link('d', d1, 'a', 0, 'alpha', pi/2, 'standard');
L2 = Link('d', 0, 'a', a1, 'alpha', 0, 'standard');
L3 = Link('d', 0, 'a', a2, 'alpha', 0, 'standard');
robot = SerialLink([L1 L2 L3], 'name', '3DOF_Arm_YZ');
```

(2) Joint space trajectory planning

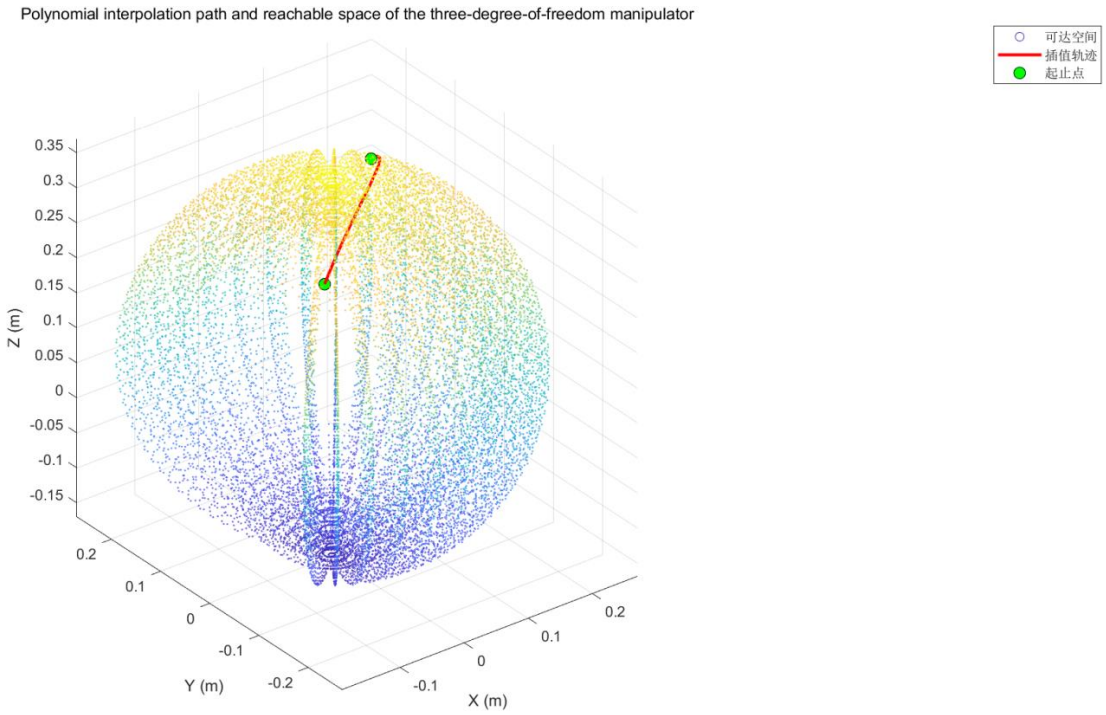

Figure 3. Animation of the robotic arm trajectory

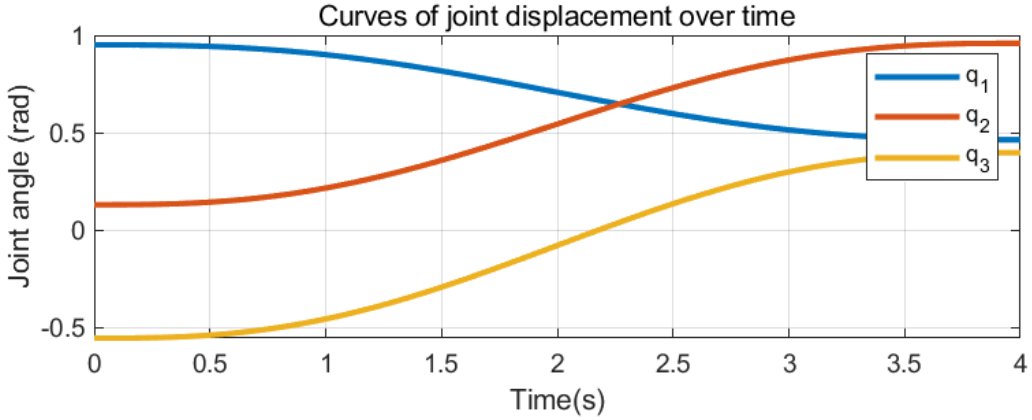

Figure 4. Curve of displacement of each joint over time

From Figure 3, it can be seen that the robotic arm reaches a wide range of positions, and the trajectory optimisation is commendable, proving the rationality of the robotic arm design. Meanwhile, observation from Figure 4 shows that the relationship between the displacements of each joint and time also demonstrates the projection changes of the end effector's position across each coordinate axis.ADAMS dynamics simulation of 3 manipulator

In contemporary industrial automation, the robot arm has emerged as a pivotal apparatus across various domains owing to its exceptional flexibility and precision. This chapter delves into the dynamic response attributes of the three rotating joints of a manipulator under a defined driving function, employing the ADAMS multi-body dynamics simulation platform. Through an assessment of the angular velocity and angular acceleration of each joint, the kinematic efficacy of the manipulator is appraised, thereby furnishing a groundwork for enhancing the manipulator's design [9-10].

3.3. Establishment of simulation model of manipulato

(1) Import and Simplification of Models

The modeling process in ADAMS can be intricate, and the alignment between components may be suboptimal. Therefore, this study opts to transfer the three-dimensional model constructed in UG modeling software to ADAMS for further simulation. Following the construction of the model in UG, it should be exported in Parasolid format (.x_t). Configuring the ADAMS environment is essential prior to importing the model, as it significantly impacts simulation accuracy.

A novel model is constructed using ADAMS software to replicate the real-world physical environment accurately. The gravitational force is aligned with the negative Z-axis to mimic actual gravity conditions. Additionally, the unit system is configured to MMKS (millimeter, kilogram, second), a standard in mechanical engineering for dimensions, mass, and time measurements. This system facilitates seamless integration with real-world engineering data, thereby ensuring the precision of subsequent simulation outcomes.

Upon importing the model, it is imperative to streamline it to enhance simulation efficiency and accuracy. Initially, non-contributory elements like bolts should be removed. Subsequently, employing Boolean operations to amalgamate stationary parts into components devoid of relative motion simplifies the model and diminishes computational intricacies during simulation.

(2) add constraints

Once the simulation model is created, the kinematic pairs can be defined. In ADAMS View, it is possible to incorporate multiple kinematic pairs. Initially, the fixed pair linking the manipulator's base to the ground is established. Subsequently, connections between each component of the manipulator are established using rotating and translating pairs. The finalized model, post the establishment of kinematic pairs, is illustrated in the figure below:

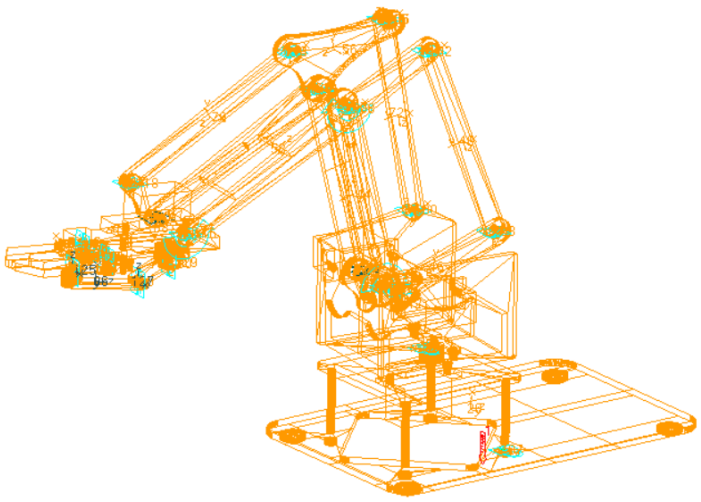

Figure 5. Model after the kinematic pair is established

3.4. Configuration of drive function

Joint 1 employs a three-stage motion process to achieve positive-reverse-positive rotation. The driving mechanism is defined as follows: STEP(time,0,0,5,90d)+STEP(time,5,0,10,-45d)+STEP(time,10,0,15,30d). This function precisely regulates the rotational angle of joint 1 during distinct time intervals by utilizing a composite of STEP functions. Such a segmented drive function design enables the emulation of intricate motion sequences that the manipulator could potentially encounter during practical operations.

The motion of joint 2 is sinusoidal motion superimposed on uniform rotation, and the driving function is as follows:

30d*SIN(0.5*time)+STEP(time,8,0,12,60d), where the first half represents sinusoidal oscillation motion with an oscillation period of 4 seconds and an amplitude of 30°. The sinusoidal function simulates the swing motion required by the robot arm in some working scenarios. The second half shows that joint 2 rotates at a constant speed during a period of 8-12 seconds, simulating the stable rotational movement of the robot arm at a specific stage.

Joint 3 adopts piecewise linear motion, and the driving function is IF(time<3:0,time<6: 45d *(time-3)/3,time< 9: 45d-30d *(time-6)/3,time<15: 15d, 15d). This function realizes the linear angle change of joint 3 in different time periods through IF conditional statements. Joint 3 linearly increases to 45° in 3-6 seconds, linearly decreases to 15° in 6-9 seconds, and remains stationary at 15° in 9-15 seconds. This piecewise linear drive function design can accurately control the motion state of the joint 3 in different stages according to actual work requirements.

3.5. Simulation parameter setting

First, the total simulation time is set to 15s, which can fully simulate the motion process of the manipulator in a typical working cycle. The step size is set to 0.01 seconds. Smaller step size can improve the accuracy of simulation results and capture the state changes of each joint of the robot arm in the movement process more carefully. Then it is time to define the measurement items: establish measurement items at each joint, make angular velocity and angular acceleration measurements, and explicitly specify the units as deg/sec and deg/sec². These two parameters are very important for evaluating the kinematic performance of the manipulator.

3.6. Simulation results and analysis

(1) The trajectory of the end effector is shown in the figure.

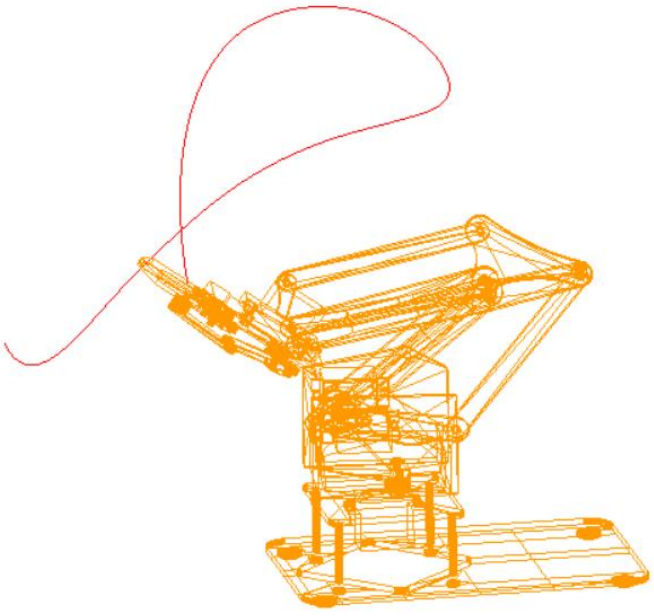

Figure 6. End effector trajectories

From the figure we can see that the trajectory of the end effector is a smooth curve, without obvious turns, abrupt changes or sharp points. It can be seen that the stability of the whole robot arm is very good, which can ensure the stability in the actual application process.

(2) Angular velocity variation characteristics are shown in the figure.

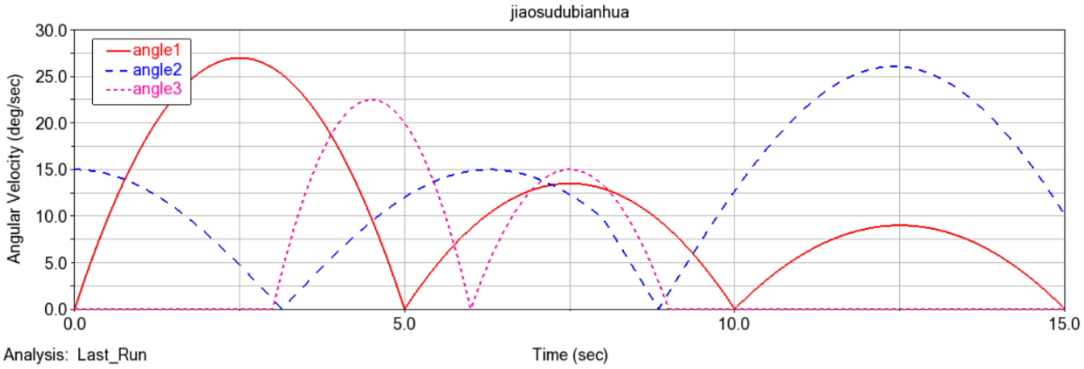

Figure 7. Angular velocity variation characteristics

According to the data, the angular velocity variation law of the three joints follows the law of uniform acceleration and uniform deceleration, and presents good stability in these processes, and the angular velocity changes continuously without fracture phenomenon. It can be seen that the stability of the robot arm is very good.

(3) Angular acceleration variation characteristics are shown in the figure

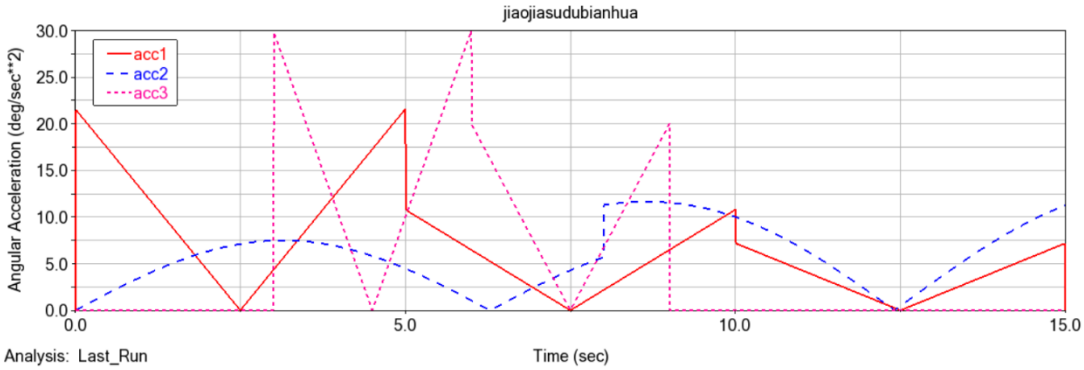

Figure 8. Angular acceleration change characteristics

The acceleration profile of joint 3 exhibits smoothness, suggesting appropriate kinematic constraints without vibration or instability due to redundant degrees of freedom in the manipulator's motion. In contrast, joints 1 and 2 display acceleration spikes, indicating a need for attention despite being relatively minor. Mitigation of these spikes can be achieved through enhancements in the drive mechanism and other relevant strategies.

4. Conclusions

This paper use D-H parameter method to define coordinate system of a 3-degree-of-freedom manipulator and analyze forward and inverse kinematics of manipulator. Quintic polynomial interpolation algorithm and examples are presented. The robot model is constructed with Matlab Robotic Tools, trajectory planning simulation is done followed by spatial trajectory image of the robot end-effector and velocity curve of each joint. Finally, the angular velocity and angular acceleration characteristic curves of manipulator are obtained by Adams simulation. Simulation results show that the trajectory planning requirements are fulfilled. Future research may incorporate physical prototype experiments to validate the consistency between simulations and real-world operations, optimize the trajectory planning algorithm by accounting for load variations and external disturbances in practical scenarios, and explore its application in multi-manipulator collaboration to better meet the demands of complex industrial automation.

References

[1] Lu Enpeng. Research on Time-Optimal Trajectory Planning of 6-DOF Industrial Manipulator [D]. Shenyang University of Technology, 2024. DOI: 10.27322/d.cnki.gsgyu.2024.000653.

[2] Zhao Fei. Design and Analysis of 6-DOF Underwater Manipulator [J/OL]. Mechanical Design and Manufacturing, 1248-1258 [2025-06-19].

[3] Yin Zhijian, Zhu Shengqiang, Zhao Zijun. Mechanism Design and Trajectory Planning of 3-DOF Forearm Rehabilitation Exoskeleton [J]. Electromechanical Engineering Technology, 2024, 53 (07): 102-106+166.

[4] Chu Xiaojing, Ren Hongchuang. Trajectory Planning and Workspace Analysis of 6-DOF Manipulator [J]. Engineering Machinery Abstracts, 2025, (03): 18-24.

[5] Fan Qi, Cai Yuqiang, Jia Sinan. Trajectory Planning and Simulation of Industrial Robots Based on Multiple Methods [J]. Journal of North China University of Science and Technology (Natural Science Edition), 2022, 44 (04): 60-67.

[6] Du Xiaohui, Yu Tao. Study on Trajectory Planning of 6-DOF Manipulator [J]. Electromechanical Technology, 2021, (04): 10-13+64. DOI: 10.19508/j.cnki.1672-4801.2021.04.003.

[7] Tu Kong, Zhang Hua. Joint Space Planning of Robot with Variable Interpolation Period Quintic Polynomial [J]. Journal of Zhejiang University of Science and Technology (Natural Science Edition), 2017, 37 (03): 376-380.

[8] Wang Ningguo, Shi Baoyu, Cheng Maosheng. Research on Kinematics of PUMA560 Robot Based on ADAMS [J]. Journal of Tangshan Normal University, 2024, 46 (03): 73-77.

[9] Zhao Xiong, Cao Gonghao, Zhang Pengfei, et al. Dynamics Analysis and Lightweight Design of a 3-DOF Apple Picking Manipulator [J]. Journal of Agricultural Machinery, 2023, 54 (07): 88-98.

[10] Li Jingyang, Zhao Sihai, Liu Zongsheng, et al. Dynamic Simulation of 6-DOF Mining Hydraulic Manipulator Based on ADAMS [J]. Coal Technology, 2024, 43 (09): 244-248. DOI: 10.13301/j.cnki.ct.2024.09.050.