Joint Control Simulation of A 6-DOF Upper Limb Rehabilitation Manipulator

Qianqian Duan*, Shaojie Xin

Machinery Institute of Shanghai Dianji University, Shanghai China

*Corresponding author e-mail: duanq2@outlook.com, xinsj@sdju.edu.cn

Abstract. In order to realize the accurate control of each joint trajectory of the upper limb rehabilitation manipulator, a simple model of the six degree of freedom upper limb rehabilitation manipulator was established by using SolidWorks. The correctness of the kinematics equation was verified by using MATLAB robot toolbox, and the model was imported into ADAMS software for dynamic analysis to verify the rationality of the mechanism; In view of the poor tracking effect of some joints in the dynamic simulation results, the fuzzy PID control algorithm is applied to the joint control module of MATLAB and Adams. The simulation results show that the fuzzy PID control can effectively improve the control performance of joint trajectory, which proves the controllability of the structure design, and provides the basis for further research.

Keywords: Rehabilitation manipulator, Co-simulation, Dynamic modeling, Trajectory tracking.

1. Introduction

In recent years, the rapid development of robot technology, in addition to today's widely used industrial robots, robot applications in other fields are also more mature. As a combination of medical robot and modern rehabilitation medical technology, rehabilitation robot can effectively help patients recover, save medical resources and improve efficiency.

As a branch of rehabilitation robot, the design of upper limb rehabilitation robot should not only meet the motion characteristics of traditional manipulator, but also consider the physiological structure parameters of human upper limb. The manipulator is a very complex nonlinear and strong coupling dynamic system. At the same time, the single upper limb of human body is a structure with high degrees of freedom. According to the traditional design mode, that is to directly process the prototype and then test it repeatedly, it will not only cause a lot of waste of manpower, material resources and time, but also make it more difficult to improve the performance of the manipulator [1]. Therefore, it is necessary to use the virtual technology to establish the virtual prototype model, and for the kinematics and dynamics analysis of the prototype, we can use the advantages of the system simulation software in this aspect to co simulate the mechanical system and control system in the virtual environment, which can further improve the design efficiency and obtain more superior system performance [2].

Lu Guangda designed a robot for finger rehabilitation by using ADAMS software, and proved the feasibility of finger rehabilitation mechanism through kinematics simulation of the joint [3]; Yu Jiqun and others developed a new type of exoskeleton robot with four degrees of freedom [4]. The kinematics
and dynamics of the robot were simulated and analyzed. The experiments show that the robot can provide effective rehabilitation treatment for patients; A 6-DOF exoskeleton rehabilitation robot is designed, which can fully simulate the human upper limb movement and is suitable for active and passive rehabilitation training, and can provide effective adjuvant treatment for patients [5]. Based on the upper limb rehabilitation robot, a new virtual simple model of 6-DOF upper limb rehabilitation robot is designed and established. The kinematics and dynamics of 6-DOF manipulator are analyzed by using D-H parameter method, and the joint control simulation of virtual prototype is carried out by using ADAMS and MATLAB.

2. Mechanical System Modeling

2.1. Modeling of Manipulator
The main joints of human upper limb are shoulder, elbow and wrist, which can be divided into seven spatial degrees of freedom according to the joint motion posture. After analyzing the physiological structure of human upper limb, six degrees of freedom are selected to create the virtual prototype model of the manipulator. The required parts are created in the three-dimensional Drawing Software SolidWorks, and the overall assembly is shown in Figure. 1.

![Figure 1. Virtual prototype model](image)

The arm consists of seven parts: base, shoulder, arm, elbow, forearm, wrist and end effector. And six degrees of freedom of the model are shoulder adduction / abduction, shoulder flexion / extension, shoulder pronation / valgus, elbow flexion / extension, forearm pronation / supination and wrist finger extension. The parameters of each joint of upper limb rehabilitation manipulator are defined in Table 1.

| Joint | Name                                | Mode of movement | Angle range |
|-------|-------------------------------------|------------------|-------------|
| 1     | shoulder adduction / abduction      | rotate           | (-90-25)    |
| 2     | shoulder flexion / extension        | rotate           | (45-135)    |
| 3     | shoulder pronation / valgus         | rotate           | (0-100)     |
| 4     | elbow flexion / extension           | rotate           | (0-50)      |
| 5     | forearm pronation / supination      | rotate           | (0-150)     |
| 6     | wrist finger extension              | rotate           | (0-80)      |

2.2. Kinematic Analysis
Kinematics analysis is the basis of the analysis of manipulator motion characteristics, including forward and inverse solutions. The forward kinematics solution of the manipulator is that after the establishment of the link coordinate system, the geometric parameters and joint angles of each coordinate system are known, and then these independent transformations are connected to obtain the position and attitude of the end coordinate system relative to the reference coordinate system [6]. Use D-H parameter transformation method is used to establish the linkage coordinate system of 6-DOF upper limb rehabilitation manipulator, as shown in Figure.3.
Figure 2. 6-DOF linkage coordinate system

Table 2. Upper Limb Rehabilitation Robot D-H Parameter

| coordinate | \( a_i \) | \( a_i \) | \( d_i \)(cm) | \( \theta_i \) |
|------------|-----------|-----------|--------------|-------------|
| 1          | 0         | 9         | 0            | \( \theta_1 \) |
| 2          | 5         | \( \pi/2 \) | 18           | \( \theta_2 \) |
| 3          | 10        | \(-\pi/2\) | 0            | \( \theta_3 \) |
| 4          | 3         | \(-\pi/2\) | 8            | \( \theta_4 \) |
| 5          | 2         | \( \pi/2 \) | 0.7          | \( \theta_5 \) |
| 6          | 6         | \( \pi/2 \) | 6            | \( \theta_6 \) |

Given the vector parameters of each coordinate system, the transformation matrix \( i-1 \) is the general expression of \( i-1T^{-1} \) [7].

\[
i^{-1}T = \begin{bmatrix}
c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\
s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\
0 & s\alpha_i & c\alpha_i & d_i \\
0 & 0 & 0 & 1
\end{bmatrix}
\]  

(1)

\( c\theta_i = \cos\theta_i \), \( s\theta_i = \sin\theta_i \), \( c\alpha_i = \cos\alpha_i \), \( s\alpha_i = \sin\alpha_i \).

Add the D-H parameters in Table 2 into equation (1), the pose transformation matrix of the end coordinate system 6 relative to the base coordinate system can be obtained

\[
0_6T = 0_1T_1^{-1}T_2^{-1}T_3^{-1}T_4^{-1}T_5^{-1}T_6
\]  

(2)

The position vector of the end of the manipulator can be obtained:

\[
P = [P_x \ P_y \ P_z]^T
\]  

(3)

In order to verify the kinematics equation of each joint is correct, the robot toolbox is used, according to the D-H parameter table is modeled and simulated in MATLAB. When the manipulator driver executes trajectory planning after receiving the position command, the D-H parameter is also changing. By substituting the changed parameters into equations (1) and (2), the pose vector of the manipulator end relative to the base coordinate system can be obtained. Two groups of variables were selected:

\[
x_1 = [0 \ 0 \ 0 \ 0 \ 0]
\]

\[
x_2 = \begin{bmatrix}
\pi/8 \\
\pi/8 \\
\pi/8 \\
\pi/8 \\
\pi/4
\end{bmatrix}
\]

\( x_1 \) is the starting angle of each joint, \( x_2 \) is the angle of the end point of each joint. The starting position and end pose vector of the wrist finger joint point can be obtained by substituting \( x_1 \) and \( x_2 \) into the end vector equation of the manipulator arm:

\[
P_1 = [26 \ 0.7 \ 32]^T
\]

\[
P_2 = [17.6661 \ 8.8002 \ 35.3198]^T
\]
The two groups of variables are input into Adams model joint drive with step function, $P_1$ and $P_2$ are substituted into MATLAB robotic model, and the running track of the two models is observed. The simulation results are shown in Fig. 4.

![Figure 3. Trajectory of Manipulator in MATLAB Robotic Toolbox](image)

2.3. Dynamics Analysis

Based on the correct derivation of the above kinematic equation, the establishment of the dynamic equation of the manipulator is analyzed as follows:

a) Lagrange function is established by Lagrange equation method

$$L = K - P$$

Where $L$ is the Lagrangian function, $K$ is the total kinetic energy and $P$ is the total potential energy of the system.

b) Kinetic energy of the system. The kinetic energy of each link is:

$$E_{ki} = \frac{1}{2} m_i \dot{\theta}^2 + \frac{1}{2} I_m k_r \dot{\theta}^2$$

Where $1$ is the mass of the connecting rod and $2$ is the mass of the connecting rod. $\dot{\theta}$ is the velocity of the center of mass of the connecting rod, $3$ is the moment of inertia of each connecting rod relative to the center of mass, and $4$ is the iterative constant.

The total kinetic energy of the system is:

$$E_k = \sum_{i=1}^{6} E_{ki}.$$  

c) System potential energy. The components of potential energy of each link on the coordinate axis are only affected by gravity:

$$E_{pi} = m_i g y_i$$

Where, $gy_i$ is the acceleration of gravity in $y_i$ component on the Axis. The total potential energy of the system is

$$E_p = E_{p1} + \cdots + E_{p6}.$$  

d) The dynamic equation of the system. By substituting the total potential energy and the total kinetic energy into the Lagrange equation, the driving torque of each joint can be obtained $\tau_i$ [8]. There is a certain relationship between the driving torque and the rotation angle, angular velocity and angular acceleration of the joint. Because the expression of the torque required by each joint in the 6-DOF manipulator is too complicated, the dynamic equation can be simplified as:

$$\tau_i = M(\theta_i) \ddot{\theta}_i + V(\theta_i, \dot{\theta}_i) \dot{\theta}_i + G(\theta_i)$$

Where $M(\theta_i)$ is the moment of inertia matrix of $6 \times 6$, $V(\theta_i, \dot{\theta}_i)$ is the vector matrix of Coriolis force and centrifugal force of $6 \times 6$. In this formula, $\theta_i$, $\dot{\theta}_i$, $\ddot{\theta}_i$ is the rotation angle, angular velocity and angular acceleration of each joint are expressed respectively.

The dynamic analysis in ADAMS is different from the kinematics analysis. In kinematics analysis, the rotation drive is added to each joint. After defining the drive function for six rotation drives, the number of degrees of freedom is 0; In the dynamic analysis, torque drive is added to the joint to keep the 6-DOF of the manipulator.
The traditional upper limb rehabilitation manipulator usually has a variety of training modes. The following motion planning is based on the passive training mode, that is, the robot arm drives the affected limb to do the specified movement training. Considering that the patient's arm has no ability of autonomous movement in the early stage, it can only do small amplitude movement. In the sagittal plane Y-Z, the wrist finger joint at the end is in the shape of holding a pen, and the whole mechanical arm does repeated arc drawing movement. The joint torque driving function is defined as $0.1 \sin(2 \pi t \times \text{time})$. The simulation driving torque and angle tracking characteristics of each joint are observed, as shown in Figure.5.

![Simulation results of driving torque and angle tracking for joints](image)

**Figure 4.** Simulation results of driving torque and angle tracking for joints

Figure.5 shows the maximum value and change trend of the driving torque of each joint of the manipulator. The driving torque of the rotating joint is the product of the applied force and the stressed area. Combined with the D-H parameter table, it can be seen that the stressed area of the shoulder adduction / abduction joint is the largest, and the driving torque is the largest, which is 0.365N/m, followed by the elbow, and the wrist; When the sinusoidal signal is input to drive each joint to make arc motion, the output of driving torque of each joint is not completely sinusoidal due to the interference of high coupling among the joints of the manipulator. The wrist finger joint is affected by its end effector. Compared with other joints, the Coriolis force and centrifugal force are significantly greater than gravity, The results show that the angular velocity is faster, but the response speed to torque tracking is slower, the tracking effect is poor, and it is easy to fluctuate. When the robot arm does not assist the movement of the affected limb, the arc drawing action is carried out alone. Because the rotation axis directions of the shoulder adduction / abduction, shoulder pronation / valgus and forearm pronation / valgus are consistent with the gravity direction, when the robot arm is in low speed motion, the Coriolis force and centrifugal force of its dynamic coefficient matrix are not obvious, and the gravity acceleration component of its rotation axis is too large, The results show that the rotation angle has poor effect on torque tracking. In order to improve the fluctuation phenomenon and the tracking accuracy, a control module can be added, that is to say, MATLAB and Adams are used for the co-simulation of dynamics.

### 3. Integrative Control Simulation

Because the manipulator is a multi-input and multi-output, strong coupling complex electromechanical system, it is difficult to achieve accurate control [9]. Moreover, the control structure of Adams is simple and the algorithm is single, so it is difficult to achieve the effect of accurate control. The MATLAB
software with more powerful control function can be used to establish the joint control, and the better control algorithm can be used.

3.1. Establishment of Co-simulation system
The input and output interface of the model is established in ADAMS, and the data communication with MATLAB is realized by its control export module. The output variable in ADAMS is the input variable of MATLAB control system, and the output variable of MATLAB control system is the input variable returned to Adams, which realizes a closed-loop control of the two co-simulation [10]. According to the dynamic equation (7), the driving torque input of the joint affects the joint angle and angular velocity. Therefore, the driving torque of each joint can be taken as the control object, and the angular displacement of each joint can be taken as the output object of Adams. Through the control module, the rotation angle of each joint can accurately approach the joint angle of the reference trajectory.

3.2. The design of the controller
For a 6-DOF upper limb rehabilitation manipulator, the higher the number of degrees of freedom, the stronger the nonlinear relationship, which directly affects the motion control of the whole system. Therefore, in order to reduce the strong coupling relationship of the system, each joint can be controlled independently. The fuzzy PID strategy is used to realize the precise control of each joint position, and compared with the most classical PID control method in the passive control training mode. The block diagram of fuzzy PID position control is shown in Figure.6.

![Figure 5. Block diagram of fuzzy PID control system](image)

3.3. Analysis of Co-simulation experiment
In order to verify the control performance of the system under the co-simulation control mode, taking the elbow joint as an example, the sinusoidal signal is used to test the response ability and tracking characteristics of the system under two different control modes, as shown in Figure.7.

![Figure 6. Response tracking characteristics of sinusoidal signal](image)

Input composite signal, that is, sinusoidal signal and step signal are input together, and the simulation time is extended to 50 seconds.

![Figure 7. Response tracking characteristics of composite signal](image)
Through the analysis of co-simulation experiment, it can be concluded that when the input is sinusoidal signal, the response speed of elbow joint under Fuzzy PID control is faster than that of traditional PID control, and the tracking error of the trajectory is controlled between -0.3 and 0.3; After adding the step signal, the system only has a small-time delay, and can accurately track the given trajectory curve. It can be seen from Figure 8 that in a long simulation time, there is no obvious vibration of the elbow joint, and the tracking error of fuzzy PID control has been stable within 0.02, while the maximum tracking error of traditional PID control can be as high as 0.04. This shows that the performance of fuzzy PID control method is better than that of traditional PID control, and it can achieve more stable trajectory tracking.

As above, fuzzy PID control can achieve more accurate control of other joints. According to the relationship between the driving torque and the angle in the dynamic equation of the manipulator, the ideal torque signal is input to obtain the change curve of the angle of each joint, and the optimal trajectory tracking performance is obtained by feedback control.

4. Conclusions

Based on the analysis of the physiological structure of human upper limb and the theoretical research of different degrees of freedom manipulator, a simple prototype model of six degrees of freedom upper limb rehabilitation manipulator is designed by using SolidWorks, Adams and MATLAB Simulink. After the kinematics and dynamics simulation verification, the fuzzy PID algorithm is applied to the joint simulation control, the tracking ability of joint trajectory is improved, which lays a foundation for further control optimization research in the future.

References

[1] Yin Xiang, Dong Zhengzhu, Wang Jiyan. Positioning control of manipulator based on ADAMS and MATLAB interactive simulation [J]. Journal of Changchun University of Technology (NATURAL SCIENCE EDITION), 2005, 26 (4): 332-334. (references)

[2] Li Feng, Zhang Weijun, Yuan Jianjun, Song Yongtao. Research on co-simulation of virtual prototype modeling and solution [J]. Computer simulation, 2012, 29 (08): 282-286.

[3] Lu Guangda, Zhang Qiuyue, Anning, Feng Yu, Ying Tongtong. Kinematics simulation analysis of hand rehabilitation robot based on ADAMS [J]. Mechanical design, 2020, 37 (06): 87-90.

[4] Yu Jiqun, Liu Ning, Lu qiongying. Kinematics analysis and Simulation of rehabilitation wearable upper limb exoskeleton based on ADAMS [J]. Mechanical transmission, 2020, 44 (11): 89-93.

[5] Zhang Hui. Design of 6-DOF exoskeleton upper limb rehabilitation robot [D]. Donghua University, 2014.

[6] Tadej Bajd, Matjaž Mihelj, Marko Munih. Introduction to Robotics[M]. Springer, Dordrecht: 2013-01-01.

[7] Yang Kai, Huang Jinying. Kinematic and Workspace Analysis of an 8-DOF space manipulator [J]. Mechanical transmission, 2021, 45 (03): 147-152.

[8] Shi Xiaohua, Wang Hongbo, Sun Li, Gao Feng, Xu Zhen. Structural design and dynamic analysis of exoskeleton type lower limb rehabilitation robot [J]. Chinese Journal of mechanical engineering, 2014, 50 (03): 41-48.

[9] Ma Ruqi, Hao Shuanghui, Zheng Weifeng, Song Baoyu. Joint simulation research of mechanical arm based on MATLAB and ADAMS [J]. Mechanical design and manufacturing, 2010, 4 (4): 93-95.

[10] Li Ke, Jiang Dikai, Wang Juan. Dynamic analysis and co-simulation of a manipulator joint based on ADAMS-SIMULINK [J]. Electronic technology and software engineering, 2017 (12): 62-64.