Operation Trajectory Control of Industrial Robots Based on Motion Simulation

Chengyi Xu¹,², Ying Liu¹*, Enzhang Jiao¹, Jian Cao², Yi Xiao²

¹College of Mechanical and Electronic Engineering, Nanjing Forestry University, Nanjing 210037, China
²College of Mechanical Engineering, Nantong Vocational University, Nantong 226007, China

*Corresponding author(Ying Liu, E-mail: lying_new@163.com)

Abstract

Aiming at the complex trajectory of industrial robots, an efficient method of robot trajectory control based on motion simulation is proposed in this paper. Firstly, the robot is modeled, the robot mathematical model is compared with the geometric solid modeling, and the control point information of the job trajectory model is extracted from the CAD platform. Then, the robot model and the information of trajectory control point are imported into the ADAMS platform. ADAMS platform is used to simulate the motion of the robot. The rotation function of each robot joint has extracted by the simulation result, and the robot trajectory is controlled by the data of each joint angle function. Finally, the efficiency of the method is verified by the robot operation test.

Key words: Industrial robot, trajectory control, motion simulation, ADAMS

1. Introduction

Industrial robot control mode mainly includes jog of control and continuous trajectory control. The common glue, cutting, arc welding and other operations are continuous trajectory control. Moreover, the control program is a very important factor that affects the accuracy of the track other than the accuracy of the robot body. At present, the generation of robot control program is mainly two kinds of teaching programming and offline programming. However, offline programming can precisely control the trajectory of the operation, and it is one of the hotspots of robot research at home and abroad. Such as the 80-90s of 20th century, most people use the analytic method, geometry, iterative method and numerical method, which need these methods for large amount of calculation, and it will come up with multiple solutions. In addition, it takes some time and the method from which to obtain the optimal solution and the cost of machine, the operation is complex. In recent years, Kim has proposed a robot trajectory generation method based on three-dimensional scanning data, Kwon et al. have proposed a method of robot trajectory generation based on object contours. Hu et al. have proposed a robot using the machine vision scanning object shape trajectory generation method, these methods are generally difficult to accurately plan the trajectory, the extraction of the trajectory is a plane curve, versatility is poor, and the generated trajectory cannot meet the surface defects.

Job graphics information digital graphics, the robot is a digital device; job graphics information and robot trajectory control are respectively, the corresponding input and output relationship. With the rapid development of digital technology, they can achieve better sharing. According to the kinematics theory of robot, the inverse solution of kinematics has generated by the manipulator control program. CAD graphics for continuous job paths can be thought of as a series of points, and the continuous trajectory control of the robot can be seen as a set of set of point controls. The number of control points on a continuous job path is higher; the track accuracy of the robot operation is higher. If the number of control points on the continuous job path is sufficient, you can ignore the effects of the interpolated form, the velocity, and other factors on the accuracy of the robot's time points.

2. Industrial robot modeling

According to the definition of D-H coordinate system and the related four parameters such as rod length hi, offset di, torsion angle αi and angle θi, we can determine the coordinate system, structural parameters and motion variables of the MOTOMAN-UP6 robots according to reference. The D-H coordinate system is shown in Fig 1. The structural parameters and movement variables of each bar are shown in Table 1.
The position and attitude of the end actuator center of the vertex relative to the UP6 robot frame coordinate system are described by matrix X.

\[
X = M_{06} \times E
\]  

\[
M_{06} = [M_{01} \ M_{12} \ M_{23} \ M_{34} \ M_{45} \ M_{56}],
\]

\[
M_{01} = \begin{bmatrix}
\cos \theta_1 & 0 & -\sin \theta_1 & h \cos \theta_1 \\
\sin \theta_1 & 0 & \cos \theta_1 & h \sin \theta_1 \\
0 & -1 & 0 & d_1 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
M_{12} = \begin{bmatrix}
\cos(-90 \times \frac{\pi}{180} + \theta_2) & -\sin(-90 \times \frac{\pi}{180} + \theta_2) & 0 & h \cos(-90 \times \frac{\pi}{180} + \theta_2) \\
\sin(-90 \times \frac{\pi}{180} + \theta_2) & \cos(-90 \times \frac{\pi}{180} + \theta_2) & 0 & h \sin(-90 \times \frac{\pi}{180} + \theta_2) \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
M_{23} = \begin{bmatrix}
\cos \theta_3 & 0 & -\sin \theta_3 & h \cos \theta_3 \\
\sin \theta_3 & 0 & \cos \theta_3 & h \sin \theta_3 \\
0 & -1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
M_{34} = \begin{bmatrix}
\cos \theta_4 & 0 & \sin \theta_4 & 0 \\
\sin \theta_4 & 0 & -\cos \theta_4 & 0 \\
0 & 1 & 0 & d_4 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
M_{45} = \begin{bmatrix}
\cos(90 \times \frac{\pi}{180} + \theta_5) & 0 & -\sin(90 \times \frac{\pi}{180} + \theta_5) & 0 \\
\sin(90 \times \frac{\pi}{180} + \theta_5) & 0 & \cos(90 \times \frac{\pi}{180} + \theta_5) & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
M_{56} = \begin{bmatrix}
\cos \theta_6 & -\sin \theta_6 & 0 & 0 \\
\sin \theta_6 & \cos \theta_6 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & d_6 \\
\end{bmatrix}
\]

\[
E = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

The position and attitude of the vertex center coordinate system of the end effector relative to the base coordinate system can be described as follows.

\[
X = M_{01} \times M_{12} \times M_{23} \times M_{34} \times M_{45} \times M_{56} \times E
\]  

0 - base; 1 - swivel; 2 - lower arm; 3 - upper arm; 4,5 - wrist; 6 - end actuator
In the initial state, the motion variable $\theta_i$ is all substituted into 0, then the pose of the UP6 end can be represented by the following $4 \times 4$ D-H matrices.

$$
X = 
\begin{bmatrix}
    n_x & o_x & a_x & p_x \\
    n_y & o_y & a_y & p_y \\
    n_z & o_z & a_z & p_z \\
    0 & 0 & 0 & 1
\end{bmatrix} = 
\begin{bmatrix}
    1 & 0 & 0 & 915 \\
    0 & -1 & 0 & 0 \\
    1 & 0 & -1 & 835 \\
    0 & 0 & 0 & 1
\end{bmatrix}
$$

(4)

Then the upper three elements of the last column in the right matrix are the position of the UP6 robot end actuator center in the initial state, such as the coordinates of the center in the corresponding base coordinate system. The coordinates of the vertex center (915, 0, 835) of the work tool in (4) are consistent with the coordinates measured by the geometric solid model established in AutoCAD in Figure 2, and the established geometric solid model and the mathematical model are verified.

![Figure 2 AutoCAD in the work tool vertex position coordinates](image)

3. Job trajectory model information extraction

CAD graphics are vectors that represent the path of the robot as a direct image. We can use CAD software related functions that will be continuous operating path CAD model, and it is divided into a series of points, the number of points are depending on the job accuracy requirements. Then use the relevant functions of CAD software to directly export the coordinates of the information file, that is, the robot operating track on all the control point data information is extracted.

4. Robot motion simulation

4.1. Generate a spline function from a graphic data file

$(X, y, z) = f(t)$ of the coordinate value versus time of the robot, the function of the coordinate value versus time of $X, Y$ and $Z$ is $x = x(t), y = y(t)$ and $z = z(t)$. Firstly, the 3D solid model of the robot has established on the ADAMS platform, and the motion constraint of each joint has implemented. Then, the data file extracted from the CAD drawing has imported to the platform. Finally, the relevant function of the simulation software is used to generate the end effector. The three spline function on the trajectory is $x = x(t), y = y(t)$ and $z = z(t)$. This builds a robot simulation environment based on the ADAMS platform.

4.2. Realization of robot motion simulation

When the robot reaches the control point, the movement angle of each joint corresponds to the inverse kinematics of the robot. The traditional mathematical solution is very cumbersome, and the solution is not unique. If all the control points of the job path are solved, the computation is large and time-consuming, and the related function modules on the ADAMS simulation platform can solve the above problems. First, we can add a multi-degree of freedom point driver to the robot end effector in the job simulation environment. Then we set the three spline functions of $x = x(t), y = y(t)$ and $z = z(t)$ as the basis of the point driver, which performs the differential calculation on the input discrete data according to Akima difference method. Finally, the motion of the robot is simulated. The robot end operator runs according to the working path in the CAD drawing. The trajectory of the robot is an optimal trajectory. Before the implementation of the robot motion simulation, the sensor function provided by ADAMS can use to define the angle range of each joint in the robot simulation environment. In the process of job simulation, if a joint reaches its rotational limit, the simulation automatically...
terminates, and prompts the sensor number to be alerted, which is providing a great help in checking the rationality of the job trajectory planning.

4.3. Extraction and Verification of Inverse Solution of Robot Motion Simulation

According to the constraint relation between the components in the kinematic model of the ADAMS simulation platform, the multi-degree of freedom point driving function is used to simulate the operation of the robot. The driving of each joint is initiated by the end effector. The driving of each joint is actually inverse, That is, each joint to complete the robot end of the corresponding actuator trajectory to be output corresponding to the angular displacement movement. After the simulation of the job, the post-processing module ADAMS/Post- Processor can automatically generate the motion angle curve of each joint of the robot motion simulation process. This set of curves is actually a combination of kinematic inverse solutions for each control point of the robot motion simulation process. By using these curves to generate the spline function of the corresponding driving joint, we can get the discrete kinematics of the robot corresponding to the control points on the robot trajectory. The inverse solution of any of the control points has substituted into the kinematic positive solution equation of the robot to verify its accuracy and accuracy.

5. Experiment and analysis

The experimental content for the simulation of robot cutting operations to cut the path CAD model is shown in Figure 3 as an example. Experimental software for AutoCAD, ADAMS, industrial robot control program automatic programmer and so on. The experimental hardware is a MOTOMAN-UP6 robot, and the robot end effector uses a pen instead. A series of data files are extracted from the CAD model. The robot simulation platform is simulated. The robot movement simulation cutting track is shown in Fig. 4. After the motion simulation is completed, the motion angle curve of each robot joint movement and its simulation process are shown in Fig. 5, which can be used to extract the discrete spline data. Part of the data is shown in Table 2.
Table 2. Discs Discrete data for each joint of the robot

<table>
<thead>
<tr>
<th>t/S</th>
<th>JOINT_1/°</th>
<th>JOINT_2/°</th>
<th>JOINT_3/°</th>
<th>JOINT_4/°</th>
<th>JOINT_5/°</th>
<th>JOINT_6/°</th>
</tr>
</thead>
<tbody>
<tr>
<td>156</td>
<td>25.0256</td>
<td>36.7268</td>
<td>1.2458</td>
<td>0.0000</td>
<td>-40.5267</td>
<td>25.0256</td>
</tr>
<tr>
<td>157</td>
<td>25.1864</td>
<td>36.9123</td>
<td>1.0211</td>
<td>0.0000</td>
<td>-40.7652</td>
<td>25.1864</td>
</tr>
<tr>
<td>158</td>
<td>25.3752</td>
<td>37.1203</td>
<td>0.8521</td>
<td>0.0000</td>
<td>-40.7854</td>
<td>25.3752</td>
</tr>
<tr>
<td>159</td>
<td>25.5677</td>
<td>37.3506</td>
<td>0.6752</td>
<td>0.0000</td>
<td>-40.9802</td>
<td>25.5677</td>
</tr>
<tr>
<td>160</td>
<td>25.3012</td>
<td>37.5633</td>
<td>0.4568</td>
<td>0.0000</td>
<td>-41.1235</td>
<td>25.3012</td>
</tr>
</tbody>
</table>

Taking the 158th control point of the job path CAD model as an example, the coordinate values are as follows, X = 1113.2852mm, Y = 468.7479mm, Z = 304.1234mm. According to the spline data of Table 2, the six joint angles are \( \theta_1 = 25.3752, \theta_2 = 37.1203, \theta_3 = 0.8521, \theta_4 = 0, \theta_5 = -40.7854, \theta_6 = 25.3752 \). This data is substituted into the robot kinematics positive solution equation (2), which is established by the D-H matrix for calculation. The results are X = 1113.2854mm, Y = 468.7483mm, Z = 304.1228mm, the error is \( \Delta X = 0.0002 \text{mm}, \Delta Y = 0.0004 \text{mm}, \Delta Z = -0.0006 \text{mm} \). This accuracy fully satisfies the requirements of robot trajectory control. According to the robot joint angle discrete data control robot operation trajectory, the robot executes the program to control the pen tip to draw the curve on the glass plate, that is, the robot trajectory, as shown in Fig. 6, the operating point of the control point is 270. The movement speed of linear interpolation mode is set to 30 mm/s, the robot to complete the operation time is about 1 minutes. Robots throughout the job process is stable, job trajectory also appears smooth, which conforms to the required working path CAD model.

6. Conclusions

The mathematical model of the robot has established by the D-H coordinate system of each component, and the geometric solid model and the mathematical model are verified.
The function of each joint spline function extracted by the robot motion simulation platform is a method to obtain the inverse kinematics data by using the motion simulation results. It has proved that the inverse solution of the simulation and the mathematical model of the kinematics of the robot are equally accurate. At the same time, the method has a small amount of computation, there is no cumbersome solution process, and the optimal solution can be obtained directly. The efficiency of the robot is greatly improved.

The robot continuous trajectory can be regarded as a set of trajectory control points. The method is operable and the generated control program can be applied directly to the complex continuous trajectory control of the robot. Moreover, the method of the three-dimensional operation is the same path, but also it is suitable for a variety of industrial robot offline programming.

Acknowledgements

This work was supported by the Nantong Key Laboratory of Industrial Robotics Applied Research (CP12015008); Nantong Applied Research Project (BK2014027).

References