Robotic Arm Simulation by using Matlab and Robotics Toolbox for Industry Application

Article · October 2020

DOI: 10.14445/23488549/IJECE-V7I10P101

Le Van Anh
Tru?ng Đ?i h?c Nông Lâm Hu?

SSRG International Journal of Electronics and Communication Engineering Volume 7 Issue 10, 1-4, October 2020
ISSN: 2348 – 8549 /doi:10.14445/23488549/IJECE-V7I10P101 © 2020 Seventh Sense Research Group®

Robotic Arm Simulation by using Matlab and Robotics Toolbox for Industry Application

Dinh Tho Long[#], To Van Binh[#], Roan Van Hoa[#], Le Van Anh[#], Nguyen Van Toan[#]

[#]*University of Economics - Technology for Industries, Viet Nam*

Received Date: 12 September 2020
Revised Date: 28 October 2020
Accepted Date: 01 November 2020

Abstract

In the era of 4.0 technology, robotic arms are becoming more and more popular in modern industries. Therefore, the research and simulation of robotic arms mean a lot in improving the efficiency of using this tool in all sectors. This paper mainly deals with robot arm modeling for industrial polishing applications. We introduce the simulation of dynamics and kinematics model of multiple degrees of freedom robotic arm based on the Matlab tool and Robotics platform. From this model, we analyze the forward kinematics of the multi-step free controller and calculate the angle of the robot arm's joints based on the inverse kinematic model, thus deducing the controller's dynamics solution. The test results demonstrate the simulation method's effectiveness, thereby providing a potential basis for industrial applications, ensuring product quality, improving productivity, and labor efficiency.

Keywords — *Robotic arm control, Industrial applications, Modeling, Matlab/Robotics.*

I. INTRODUCTION

Today, along with the development of information technology and mechanics, robots have become more widely used in all production processes, improving manufacturing efficiency, increasing productivity, enhancing working conditions, and accelerating industrial automation [1], [2]. The robotic arm has also been widely used in the medical [3] field, besides its applications in high temperature, high pressure, dust, noise, radiation, and polluted environments. Additionally, robotic arms are used in the automotive industry for welding, painting, loading, and unloading [4]. Moreover, the robot arm can also replace workers in harsh environments, replace human labor who do heavy, monotonous, and repetitive work, improve labor productivity, and ensure product quality. Its applications have been extended to space exploration, marine bed research, nuclear science research, or automated medicine operations. Therefore, it can be confirmed that the robotic arm is a key technology in automatic control and has become an important part of modern industrial systems [5]. The robotic arm contains an operator, a controller, a servo drive system, detection, and sensors. This is a type of automated production equipment with human-like features, automatic control, reprogramming, and completing various operations in three dimensions. Nowadays, the industrial robot arm is used in manufacturing and is usually driven in an open loop, which means the position trajectory and the speeds are calculated only once and then executed by the robot arm. There is no adjustment in operation. To perform the tasks correctly, a closed-loop controller is required. The sensor will measure the errors between the target and current position, calculate and execute the new trajectory and velocity, and then modify the next control command [6]. In the study of Mohammed Abu Qassem et al. [9], Matlab/Simulink was used as a tool to test the motion properties of the AL5B Robot arm. A robot model will be developed; "Jacobian" forward, inverse, velocity motion, and path planning problems will be performed and tested. The author studied Matlab's tools to simulate robot control systems in the industry. The simulation will help the test process save on initial investment as well as avoid unnecessary risks. It is also a useful tool for training and scientific research in robotics, automation, and control. It can be seen that domestic research on the mechanical arm started relatively late, and the technology in automation control is not advanced. Current robotic arm control systems typically use single-core processors and electrical equipment for control. The system performance is relatively stable, but the architecture of single-core processors and complex instruction systems results in slow performance and limited processing power. Therefore, in this paper, we set up the robotic arm's mathematical description to model the dynamics and dynamics. On this basis, we propose a plan for arms control planning—robots for industrial polishing applications.

The rest of the paper is organized as follows. The second section simulates a robot arm using the Robotics toolkit. The dynamic and kinetic modeling of the robot arm will be discussed in the third section. Finally, the fourth section is the conclusion of this paper.

II. SIMULATION OF ROBOT ARM

The degree of freedom is an essential technical indicator of the robot, determined by its structure and closely related to its flexibility. In general, the more degrees of freedom the robot is, the more flexible and close to a human it is, but the more degrees of freedom, the more complex the structure. As shown in Figure 1, we present the robot model with six degrees of freedom. All joints that are rotated, joints 2, 3, and 4, are in the same plane, and the last three joint axes of the model intersect at one point.

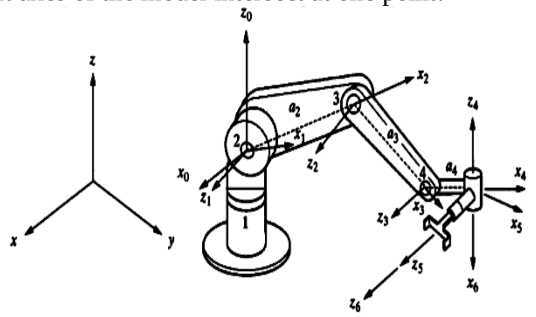

Fig 1: Model of 6 degrees of freedom robot with the reference coordinate system on each coupling

This simulation focuses on handling the robot's trajectory and control force when the robot arm encounters the environment's problems. Therefore, the first stage aims to design a polishing robot based on a multi-degree robotic arm and perform kinetic analysis to establish the robot's forward and reverse equations. In this study, we use Matlab for 3D modeling and dynamic analysis, and mechanical mechanisms are used to simulate and analyze robotics dynamics.

In this study, we also use Robotics tools [7]. These are tools that support the modeling functions of many different robot models. These tools also provide inverse kinetics and robot motion modeling, dynamic solution functions, including Cartesian space and shared space. In Figure 2, we present the simulation model established by the Matlab/Robotics toolkit.

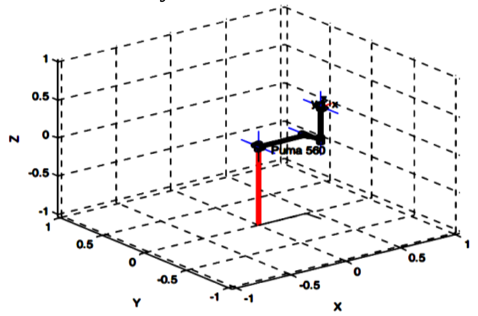

Fig 2: Simulate robotic arm using Robotics toolkit

The mechanical three-degree robot model is built on Simulink/SimMechanics simulation software to simulate the robot in a realistic control environment. The outputs are the coupling angle, the angular velocity, and the vector angular acceleration; The sensor output is the x, y, z position coordinates in the Cartesian coordinate system at the end of the arm. The jacobian control model uses the output matching angle variable as the Jacobian matrix's feedback, gives real-time matching variables, and calculates the match position at the next time based on the current matching variables. As illustrated in Figure 3, the general robot trajectory is calculated using Jacobian matrices. T2 is the desired trajectory; T1 is the actual spatial position of the returned joint position. The robot's general control trajectory is obtained using the Jacobian matrix through the error between the actual line and the line desire.

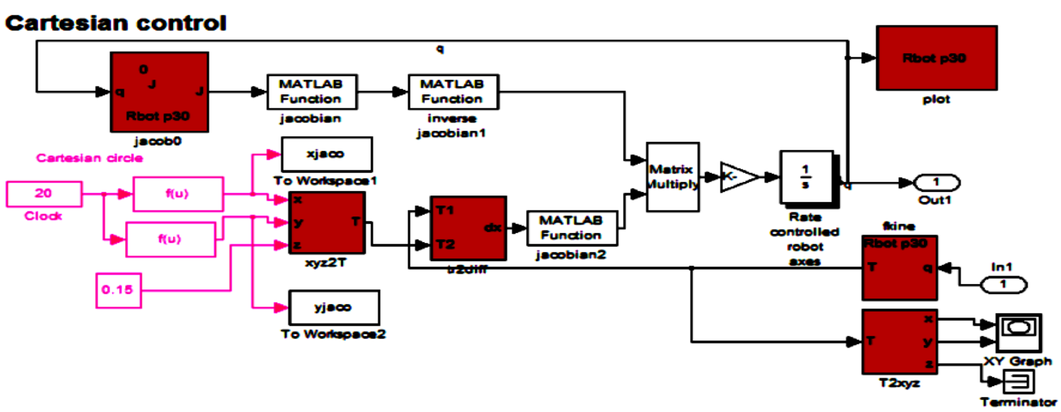

Fig 3: Robot's path control model based on the Jacobian matrix

III. DYNAMISM MODELING AND KINETICS OF A ROBOT ARM

A. Kinetics Model

To realize industrial robot control for polishing, it is necessary to set the robot arm's kinematic model. The kinetic description of the robot establishes the variable relationship between the coordinate axes of each coupling. Here, D-H parameters are used to describe the connection of each joint of the robot [8]. The D-H method uses four parameters to determine the connecting rod (i) with a connecting rod (i-1). In this paper, we also use transformation matrices, D-H parameter description, and simple robot bonding expression. As shown in Figure 1, the spatial position relationship between two adjacent robot coupling and coordinates is defined on each coupling. After setting the coordinate system of each link of the robot, the change matrix between the two adjacent joints' coordinate systems is indicated by the parameters D-H.

Assuming that the robot system has five degrees of freedom, based on the transformation matrix $A_i(i = 1, 2, ..., 5)$ described by the symbol D-H, we can obtain the state posture. End of the robot using the robot's coupling values and bar length parameters.

$$A_1A_2A_3A_4A_5 = {}^RT_H \tag{1}$$

The transformation matrix Ai(i = 1, 2, ..., 5) of robot is:

$$A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & \alpha_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & \alpha_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Solving the above equation, we obtain the solution of the kinetic equation:

$$\begin{bmatrix} n_x & a_x & o_x & p_x \\ n_y & a_y & o_y & p_y \\ n_z & a_z & o_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^RT_H = A_1A_2A_3A_4A_5 \tag{3}$$

The inverse kinetic problem of the corresponding coupling variable is reversed from the position of a known robot terminal, and equation (1) can be converted to:

$$A_2A_3A_4 = A_1^{-1}\,{}^RT_H A_5^{-1} \tag{4}$$

$$A_i^{-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 & -\alpha_i \\ -\sin\theta_i\cos\alpha_i & \cos\theta_i\cos\alpha_i & \sin\alpha_i & -d_i\sin\alpha_i \\ \sin\theta_i\sin\alpha_i & -\cos\theta_i\sin\alpha_i & \cos\alpha_i & d_i\cos\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

According to the calculation and transformation of the matrix A, the inverse kinetic equation's solution can be obtained by (6).

$$\begin{cases} \theta_1 = \arctan\left(p_y / p_x\right) \\ \theta_2 = \arctan\dfrac{p_x\left(z_2 - d_4s\theta_3\right) - d_4c\theta_3\left(p_xc\theta_1 + p_ys\theta_1\right)}{\left(p_xc\theta_1 + p_ys\theta_1\right)\left(a_2 - d_4s\theta_3\right) + d_4p_xc\theta_3} \\ \theta_3 = \arctan\left(s\theta_3 / c\theta_3\right) \\ \theta_4 = \arctan\dfrac{c\theta_5\left(n_xs\theta_1 - n_yc\theta_1\right) + s\theta_5\left(a_xs\theta_1 - a_yc\theta_1\right)}{\left(o_xs\theta_1 - o_yc\theta_1\right)} \\ \theta_5 = \arctan\dfrac{a_xs\theta_1 - a_yc\theta_1}{n_xs\theta_1 - n_yc\theta_1} \end{cases} \tag{6}$$

There are two problems in the motion dynamics of industrial polishing robots: one is to solve the kinetics of polishing motion points by adjusting the different joint rotation angles. Another way is to show the end position of the batting action point, and the robot is required to follow a certain path. Then, constraints are obtained when solving the inverse kinetics problem at a given location. Kinetics solves the problem by calculating the position and direction in Cartesian coordinates by the kinematics and spatial coordinates of a given coupling. Robotics Toolbox supports modeling functions for various robots, providing forward and reverse robot motion and motion solution functions, including Cartesian space and spatial orbital planning. It supports both standard D-H systems and revision systems. The simulation test results of the robot kinetic model are shown in Fig 4.

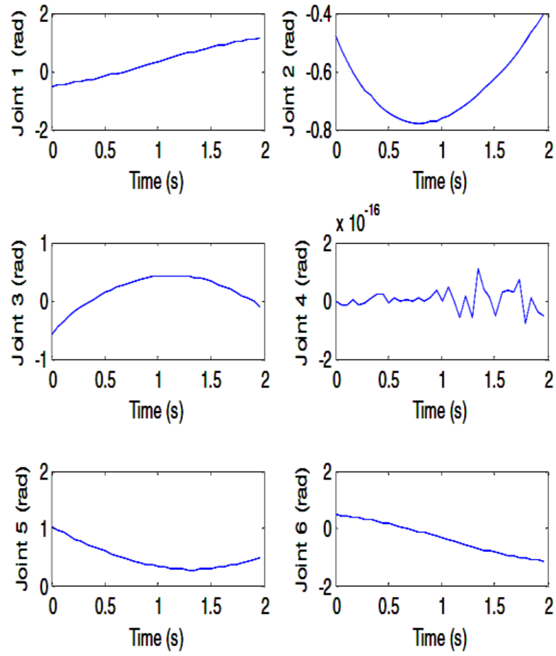

Fig 4: Results of robot kinetic simulation

B. Dynamic Model

If external disturbances such as friction are not taken into account, the kinematic equation of the n degrees of freedom controller is:

$$\tau = M(q)\ddot{q} + h(q,\dot{q}) + G(q) \tag{7}$$

Where M(q) is the positive inertia matrix, $h(q,\dot{q})$ is the centrifugal force, and the Coriolis force vector, *G(q)* is the gravity vector, *q* is the joint position, and τ is the joint torque.

An example of the two-degree-of-freedom operator model in the plane, supposing that we have a stiff arm with both joints being a swivel, and each joint's mass is concentrated at the end of the joint. The two joints' mass is m_1 and m_2, respectively; the lengths are l_1 and l_2, respectively; the matching angle is $q = \begin{bmatrix}\theta_1 & \theta_2\end{bmatrix}^T$ and the matching moment is $\tau = \begin{bmatrix}\tau_1 & \tau_2\end{bmatrix}^T$. The dynamic equation for a machine operating two degrees of freedom is:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad (8)$$

In the Cartesian coordinate system, when the arm's end comes in contact with the outside, the arm, and the outside will create a mutual force F_e. To maintain arm balance, a certain driving torque must be applied to each joint $\tau = J^T F_e$. J is the Jacobian matrix. The Cartesian force acting at the end of the arm can be mapped to an equivalent joint torque. On exposure to the medium, the kinetic equation of the actuator is:

$$\tau = M(q)\ddot{q} + h(q,\dot{q}) + G(q) - J^T F_e \quad (9)$$

Forward dynamics are used to calculate joint acceleration based on a given joint position, velocity, and moment. The integral speed and the position/angle of the joint can be calculated based on this. Inverse dynamics are the calculation of joint torque through joint position, velocity, and acceleration information. The torque required for each joint can reach the expected position, speed, and acceleration. Inverse kinetics are calculated by Newton's Euler method. Computing the inverse kinetics requires understanding each joint's inertial, mass, and kinetic parameters and expanding the matrix that describes each joint's kinetics by adding mass and inertia parameters. The simulation test results of the dynamic model are shown in Figure 5.

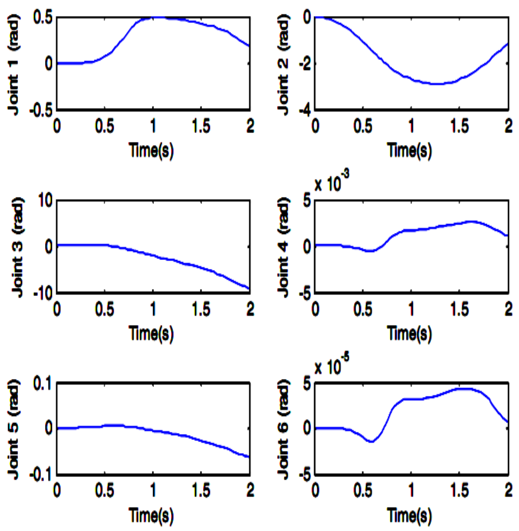

Fig 5: Results of the simulation of the dynamic model on Matlab

IV. CONCLUSIONS

Currently, a robotic arm system can provide greater reliability in polishing workpieces and smart manufacturing. In this article, we discussed the use of robotic automation systems instead of manual operations for polishing. Researchers have presented different methods for integrating information technology and manufacturing technology to adapt to global manufacturing markets growing competition. This paper investigated a multidimensional robotic arm, and the standard D-H modeling method was used to model the mechanical arm of this study. The robot controller dynamics and dynamics are then analyzed and simulated in the Matlab/Simulink environment with the Robotics toolkit. In the future, we will study the creation of robotic trajectories and control forces in industrial applications.

ACKNOWLEDGMENT

This study was supported by the University of Economics - Technology for Industries, Viet Nam; http://www.uneti.edu.vn/.

REFERENCES

[1] Iqbal, Jamshed, R. Ul Islam, and Hamza Khan, "*Modeling and analysis of a 6 DOF robotic arm manipulator*", Canadian Journal on Electrical and Electronics Engineering 3, no. 6, pp. 300-306, 2012.

[2] H. C. Nguyen, "*Research on the application of Neural Networks in identification and control of the robotic arms-A nonlinear dynamic object*," Journal of Science and Technology – the University of Da Nang, vol. 5, pp. 14-18, 2016.

[3] Meng, Wei, Quan Liu, Zude Zhou, Qingsong Ai, Bo Sheng, and Shengquan Shane Xie, "*Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation*," Mechatronics 31, pp. 132-145, 2015.

[4] Žlajpah, Leon, "*Simulation in robotics*," Mathematics and Computers in Simulation 79, no. 4, pp. 879-897, 2008.

[5] Topaz, Metin, and Serdar Kucuk, "*Dynamics simulation toolbox for industrial robot manipulators*," Computer Applications in Engineering Education 18, no. 2, pp. 319-330, 2010.

[6] Wang, Zhi-Xing, Wen-Xin Fan, Bao-Cheng Zhang, and Yuan-Yuan Shi, "*Kinematical analysis and simulation of an industrial robot based on Matlab*," Jidian Gongcheng/Mechanical & Electrical Engineering Magazine 29, no. 1, pp. 33-37, 2012.

[7] Kang, Hee-Jun, and Young-Shick Ro, "*Robot manipulator modeling in Matlab-SimMechanics with PD control and online gravity compensation*," In International Forum on Strategic Technology 2010, pp. 446-449. IEEE, 2010.

[8] Kim, Suseong, Seungwon Choi, and H. Jin Kim, "Aerial manipulation using a quadrotor with a two DOF robotic arm," In 2013 IEEE/RSJ International Conference Intelligent Robots and Systems, pp. 4990-4995. IEEE, 2013.

[9] Mohammed Abu Qassem, I.M. Abuhadrous, H. Elaydi, "*Modeling and Simulation of 5 DOF educational robot arm*", In Conference: Advanced Computer Control (ICACC), 2010 2nd International Conference on, Volume. 5 April 2010.

[10] P.Hemalatha, C.K.Hemantha Lakshmi, and Dr.S.A.K.Jilani, "*Real-time Image Processing based Robotic Arm Control Standalone System using Raspberry pi"* SSRG International Journal of Electronics and Communication Engineering 2.8 (2015): 12-15.

[11] Ramanideepthi Tanneeru, Ajay Jandrajupalli, B. Kiran Kumar, "Design and Analysis of Robot Arm using Matlab & ANYSYS" SSRG International Journal of Mechanical Engineering 2.3 (2015): 13-18.

[12] Annamareddy Srikanth , M.Sravanth , V.Sreechand , K.Kishore Kumar. "KINEMATIC ANALYSIS OF 3 D.O.F OF SERIAL ROBOT FOR INDUSTRIAL APPLICATIONS". International Journal of Engineering Trends and Technology (IJETT). V4(4):1000-1004 Apr 2013.

[13] Lakshmigandan.B, Janakiram.V, Jothi Raj.S, Salai Selvam.V "EEG Based Brain Controlled Mobile Arm Pick and Place robot", International Journal of Engineering Trends and Technology (IJETT), V45(8),416-418 March 2017.