Asian Journal Science and Engineering

Vol. 2, No. 2, December 2023
https://attractivejournal.com/index.php/ajse

Integrated Robotic Arm Control: Inverse Kinematics, Trajectory Planning, and Performance Evaluation for Automated Welding

Arif Nur Huda[1], Dwi Pebrianti[2], Zainah binti MD. Zain[3*]
[1] *Politeknik Negeri Malang, Indonesia*
[2,3] *International Islamic University Malaysia, Malaysia*

zainah@umpsa.edu.my*

ARTICLE INFO
Article history:
Received
October 04, 2023
Revised
December 01, 2023
Accepted
December 06, 2023

Abstract

This research delves into the automated functionality of robotic arm manipulators, a hallmark of Industry 4.0, within the manufacturing sector. The study focuses on precise movement adhering to predetermined trajectories, addressing the vital aspects of inverse kinematics and trajectory planning in robotic arm control. Utilizing the Matlab robotic toolbox, the research conducts simulations of inverse kinematic and trajectory planning. An experimental setup involving a robotic arm controlled by an Arduino Mega 2560 microcontroller, embedded with the inverse kinematic algorithm and trajectory planning, is executed. Data acquisition involves inputting coordinates and orientation for automatic welding along a straight path. Joint angles are measured using rotary encoders and converted into Cartesian coordinates to determine the end-effector's position. Discrepancy analysis compares measured joint angles with simulation values, revealing error margins. Movement quality of the robotic arm is assessed through Capability Processes (CP) evaluation. Results indicate disparities between experimental and simulated values. At input coordinates (400mm, 0mm, 300mm), joint angle errors of 2.51º, 0.98º, and 1.48º are observed for joints 2, 3, and 5, respectively. Similarly, at input coordinates (300mm, 0mm, 300mm), joint angle errors of 1.17º, 1.5º, and 2.7º are registered for the same joints. Trajectory error analysis during straight welding reveals average errors of 2.25 mm and 10.57 mm along the x and y axes. Mean absolute errors for joints 2, 3, and 5 are 1.9º, 0.48º, and 1.91º.

Keywords: Robotic Arm Manipulators, Inverse Kinematics, Trajectory Error Analysis

Published by CV. Creative Tugu Pena
ISSN 2963-6752
Website https://attractivejournal.com/index.php/ajse

INTRODUCTION

In the era of Industry 4.0, technological advancements have significantly impacted various sectors, including the manufacturing industry. One technology gaining prominence is the utilization of integrated robotic arm manipulators, which has opened new avenues for automating production processes. The

presence of robotic arm manipulators enables the implementation of advanced automation, enhancing production efficiency and facilitating the development of higher-quality products.

In this context, this research assumes a pivotal role by focusing on the integrated control of robotic arm manipulators. In their more sophisticated applications, robotic arm manipulators can perform not only physical movements but also be configured to execute specific tasks automatically and with precision. Central to this endeavor are the key aspects of inverse kinematics and trajectory planning, which constitute fundamental elements in governing the movement of robotic arm manipulators.

The utilization of inverse kinematics enables robotic arm manipulators to be directed to desired positions and orientations, paving the way for various applications, including automated welding processes. Meanwhile, trajectory planning allows manipulators to follow predetermined paths with high precision, a critical factor in maintaining the quality and accuracy of end results, such as automated welding.

This study also involves the evaluation of the proposed control system's performance. This evaluation encompasses not only the aspects of kinematics and trajectory planning but also delves into how well the robotic arm manipulator can sustain movement quality in the context of automated welding. Performance analysis becomes crucial in ensuring that the robotic arm manipulator is reliable and consistently yields results aligned with production objectives.

Thus, this study aims to investigate and develop integrated control for robotic arm manipulators, with a focus on inverse kinematics, trajectory planning, and performance evaluation in the context of automated welding. By gaining a deeper understanding of these key elements, it is anticipated that this research can positively contribute to delivering more sophisticated and efficient automation solutions in the manufacturing industry.

In the era of the development manufacturing industry, apply several cutting-edge technologies. One of the technologies that is starting to be popularly applied in the manufacturing industry is a robot manipulator of the robotic arm type. The application of this technology is because the industry currently wants to increase effectiveness and productivity. The increase in effectiveness and productivity is because the robot arm can increase speed and accuracy [1]In addition, a robot can replace human work that was previously difficult and dangerous to be easier and safer [2]. The manipulator technology in the form of a robot arm has been used in various processes in the Industry, such as moving goods, cutting, metal casting, welding, painting, assembly and others. This research will specifically discuss a robot arm that is used to do welding work automatically. The welding process is one of the metal joining processes that uses high temperatures for melting [3] Currently, some of these processes play a role in every manufacturing industry, for example the automotive industry, components,

machining etc. Currently, many industrial manufacturing processes, especially welding, have been replaced by a robot arm. This is indicated by the increase in sales of industrial robots / robotic arms by 12%, namely 178,132 units in 2013 [4] In Japan, the automotive industry requires 1436 robot units per 10,000 workers [5] From these data, it shows that a robot arm plays an important role in an industry. In the field of automatic welding and robotic arms, many researchers have conducted research that is also related to what researchers do, such as those conducted by [6] with the title Design of MIG Welding Motion Automation System to Improve the Quality of Welding Joints. Thesis research by [7] with the title Development of GMAW (Gas Metal Arc Welding) Automatic System Prototype for Pipe Welding Based on Machine Vision. Research by [8] with the title Design of 5 Degrees of Freedom Robot Arm with Kinematics Approach. Research by [9] with the title Design and development of a robotic arm. Then Research by [10] with the title Design and Construction of a Robotic Arm For Industrial Automation. Research on robotic arms was conducted by [11] with the title Mathematical Modeling Trajectory Planning of a 5 Axis Robotic Arm for Welding Applications. In this study, a robot trajectory planning algorithm was determined for efficient welding, where the end-effector carrying the welding gun traverses a certain trajectory. The algorithm is tested and simulated for various curve shapes including straight lines, circles and parabolic curves. Research on welding automation by [12] with the title PC-based 3-Axis Automation Controller in Welding Process Simulation. This research produces an automatic tool for welding applications. From the test results, the 3-axis robot can perform several variations of movements in the welding process such as zigzagging according to the G-Code entered into the Mach3 and EMC2 software. In this study, we did not directly use a welding gun but replaced it with a stationery to find out the results of the movement. Research on Robot arms was conducted by [13] with the title Design of 5 Degrees of Freedom Robot Arm with Kinematics Approach. The research produced an output in the form of a 5 degree of freedom robot arm driven by a servo motor with control using Arduino Uno. From the test results using the forward kinematic method the robot can run well and can reach the maximum distance on the x = 425 axis, y = 425 mm and z = 480 mm. Research on robot arms was conducted by [14] with the title Teaching and Learning Robotic Arm Model. This research uses MATLAB Graphical User Interface (GUI) as software, arduino board as control unit, and accelerometer as sensor. All three provide a practical laboratory model for learning robot arm platform programming.

Research on robotic arms was conducted by [15] with the title Design and Construction of a Robotic Arm for Industrial Automation. The robot arm is designed from acrylic material where servo motors and stepper motors are used for connecting between the arms to execute the robot arm movements. To control the robot an automatic computer code is used and the serial communication method provides an automatic process for the automatic process.

Research on robotic arms was conducted by [16] with the title Design and Development of a Robotic Arm. The research produced a design and model of a 5 degree of freedom robotic arm. The robot arm movement is driven by a DC motor and the joint position in the form of an angle is monitored by a potentiometer. Arm motion control is done through Matlab Graphic User Interface (GUI) and Arduino Mega2560. Workstation settings of the robot arm can automatically use backward kinematics to move the arm to the desired location.

Robot Arm research was conducted by [17] with the title Inverse Kinematic Solution of a 6 DOF Serial Manipulator. In this study, researchers have compared the analytical inverse kinematic solution for the serial ABB IRB7600-500 Robot arm with the simulation value using Matlab Robotics Toolbox. The result is that the analytical solution either in Cartesian space or joint space shows a very small error value. The kinematics of three DOF was introduced for the lower limb of the humanoid robot. Decoupled closed-form solution for the position and orientation was the solution of kinematics; the joint sequences were presented by Denavit-Hartenberg (DH) transformation matrices. Swing phase equations were developed to avoid matrix inversion problems [18] The kinematic parameters on industrial robot were affected by vibrations disturbance. The error in motion was improved by sensors of accelerometer and gyroscope. The motion profile was analyzed for joint; then, the path tracking of welding task was estimated [19]. Human robot kinematics was identified by geometry kinematics approach to map human arm configuration and stiffness controlled index by hand gesture. The human arm stiffness was estimated within robot experiential stability region. A moving task was implemented to test the performance of geometry kinematics approach on Baxter robot simulator [20] The geometric approach was used to solve the kinematics of the autonomous positioning of a robotic arm. This modeling and analysis approach was tested by using a five DOF arm with a gripper mounted to the iRobot mobile platform [21] Online robot kinematics parameter errors estimation based on inertial measurement unit (IMU) was presented. It obtained the orientation of the manipulator with the orientation of the IMU in real time. This approach incorporated Factored Quaternion Algorithm (FQA) and Kalman Filter (KF) to the orientation of the IMU [22] An analytical solution of inverse kinematics for a five DOF spatial parallel micromanipulator was presented. A geometrical mode and structural of system were introduced for the microrobot's task [23] Forward kinematics and inverse kinematics were calculated and simulation was done for joints and link parameters of six-axis robotic arm. Trajectory planning was described for the requisite motion of the manipulator as a time sequence task [24] Forward and inverse kinematics of a KUKA robotic arm in the application of a simple welding process were introduced. A general DH representation of forward and inverse matrix was obtained. A movement flow planning was designed and developed for the programming of the robot [25] The mobile robot with arm (KUKA youBot) and the solving of inverse kinematics problem were introduced.

The robot was presented as 8 DOF. The kinematics redundancy of the holonomic platform was presented. Including redundancy parameters, the inverse kinematics solution was suggested [26] The end effector position and orientation error of a space robot were studied. A geometric parameter identification method was presented based on a laser ranger attached to the end effector. The independence of the geometric parameters was analyzed. Identification equations were derived by simulation which was implemented for different types of robot configuration [27]

Forward Kinematic

The forward kinematics analysis of the Robot Arm uses a 4x4 Homogeneous Transformation Matrix that refers to each DH parameter. The Homogeneous Transformation Matrix refers to the equation below.

$$A_i^{i-1} = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where: c_(θ_i) is cos(θi) and s_(θ_i) is sin(θi).

$$A_1^0 = \begin{bmatrix} c_1 & 0 & -s_1 & a_1c_1 \\ s_1 & 0 & c_1 & a_1s_1 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_4^3 = \begin{bmatrix} c_4 & 0 & s_4 & 0 \\ s_4 & 0 & -c_4 & 0 \\ 0 & 1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_5^4 = \begin{bmatrix} c_5 & 0 & -s_5 & 0 \\ s_5 & 0 & c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c_3 & 0 & -s_3 & a_3c_3 \\ s_3 & 0 & c_3 & a_3s_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_6^5 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The results of the homogeneous transformation matrix calculation based on the equation above.The final homogeneous transformation matrix from frame 1 to frame 6 is as follows. The matrix below represents the position and orientation of the 6 dof robot arm. The letters px, py, pz represent the coordinate points of the robot while r11 to r33 is a 3 x 3 matrix containing the orientation of the robot end-effector in degrees.

$$A_6^0 = A_1^0 \cdot A_2^1 \cdot A_3^2 \cdot A_4^3 \cdot A_5^4 \cdot A_6^5 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Where:

$$r_{11} = s_6[c_4s_1 - s_4(c_{123} - c_1s_{23})] + c_6[c_5(s_{14} + c_4(c_{123} - c_1s_{23})) - s_5(c_1s_{23} + c_{13}s_2)]$$

$$r_{12} = c_6[c_4s_1 - s_4(c_{123} - c_1s_{23})] - s_6[c_5(s_{14} + c_4(c_{123} - c_1s_{23})) - s_5(c_{12}s_3 + c_{13}s_2)]$$

$$r_{13} = -s_5[s_{14} + c_4(c_{123} - c_1s_{23})] - c_5[c_{123} + c_{123}]$$

$$r_{21} = -s_6[c_{14} - s_4(s_{123} - c_{23}s_1)] - c_6[c_5(c_1s_4 + c_4(s_{123} - c_{23}s_1)) + s_5(c_2s_{13} + c_3s_{12})]$$

$$r_{22} = s_6[c_5(c_1s_4 + c_4(s_{123} - c_{23}s_1)) + s_5(c_2s_{13} + c_3s_{12})] - c_6[c_{14} - s_4(s_{123} - c_{23}s_1)]$$
$$r_{23} = s_5[c_1s_4 + c_4(s_{123} - c_{23}s_1)] - c_5[c_2s_{13} + c_5(c_2s_{13} - c_3s_{12})]$$
$$r_{31} = s_{46}(c_2s_3 + c_3s_2) - c_6[s_5(c_{23} - s_{23}) + c_{45}(c_2s_3 + c_3s_2)]$$
$$r_{32} = s_6[s_5(c_{23} - s_{23}) + c_{45}(c_2s_3 + c_3s_2)] + c_6s_4(c_2s_3 + c_3s_2)$$
$$r_{33} = c_{45}(c_2s_3 + c_3s_2) - c_5(c_{23} - s_{23})$$
$$p_x = a_1c_1 - d_6[s_5(s_{14} + c_4(c_{123} - c_1s_{123})) + c_5(c_{12}s_3 + c_{13}s_2)] - d_4(c_{12}s_3 + c_{13}s_2) + a_2c_{12} + a_2c_{123} - a_3c_1s_{23}$$
$$p_y = a_1s_1 - d_4(c_2s_{13} + c_3s_{12}) + d_6[s_5(c_1s_4 + c_4(s_{123} - c_{23}s_1)) - c_5(c_2s_{13} + c_3c_{12})] + a_2c_2s_1 + a_3c_{23}s_1 - a_3s_{123}$$
$$p_z = d_1 - d_4(c_{23} - s_{23}) - d_6[c_5(c_{23} - s_{23}) - c_4s_5(c_{23} + c_3s_2)] - a_2s_2 - a_3c_2s_3 - a_3c_3s_2$$

After calculating the above equation we will know the position of the end effector in the cartesian diagram (Px, Py, and Pz) as well as its orientation. In addition to calculating the solution analytically, the forward kinematic solution can also be calculated using the Matlab robotic toolbox software.

Inverse Kinematic

Solution for finding the angle values of joint 1 (θ1), joint 2 (θ2), and joint 3 (θ3)

Before finding the values of θ1, θ2, and θ3. it will first find the position of the robot arm wrist point. This method is called kinematic decoupling. This method is to separate the robot kinematics structure into 2 parts, namely, the kinematics structure to form the end-effector position in x, y and z coordinates and the kinematics structure used to form the orientation of the end-effector. The steps to find the wrist point of the robot arm are as follows.

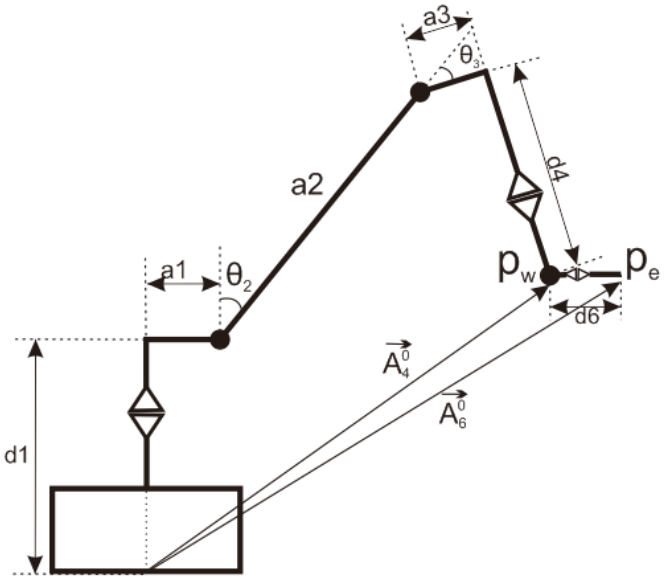


Figure 1 Kinematic Decoupling

The first step to calculate the position of the wrist is to know the homogeneous transformation matrix from frame 0 to frame 6 (end-effector). We can see this matrix in the discussion of forward kinematic.

$$A_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

To find the position of the wrist point is with the following mathematical equation.

$$p_w = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - d_6 \begin{bmatrix} r_{13} \\ r_{23} \\ r_{33} \end{bmatrix} \quad (4)$$

$$p_w = \begin{bmatrix} p_{w,x} \\ p_{w,y} \\ p_{w,z} \end{bmatrix} \quad (5)$$

The search for the p_w point is to make it easier to find the values of θ_1, θ_2, and θ_3. Where the three angles are used to find the position of the robot. Another way that can be used to find the wrist point is to calculate the homogeneous transformation matrix from point 0 - 4 (A_4^0). Namely through the equation:

$$A_4^0 = A_1^0 A_2^1 A_3^2 A_4^3 \quad (6)$$

$$A_4^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_{w,x} \\ r_{21} & r_{22} & r_{23} & p_{w,y} \\ r_{31} & r_{32} & r_{33} & p_{w,z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Joint 1

Before finding the angle value of joint 1, we need to draw the kinematics structure of the robot in the top view as shown in Figure 4.6. In this picture there is a right triangle whose angle is unknown. By knowing the 2 sides of the right triangle we can find out the angle θ_1, The two sides are the value of the wrist position in the x coordinate and the wrist position in the y coordinate.

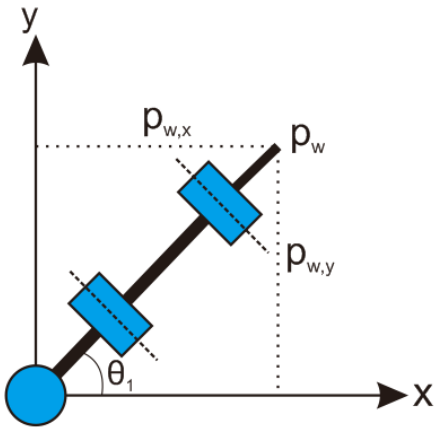


Figure 2 Top view of robotic arm kinematics

Here is the equation to find the value of θ_1 using the arc tangent rule.

$$\boldsymbol{\theta_1 = Atan\left(\frac{p_{w,y}}{p_{w,x}}\right)} \quad \textbf{(8)}$$

a) Joint 3

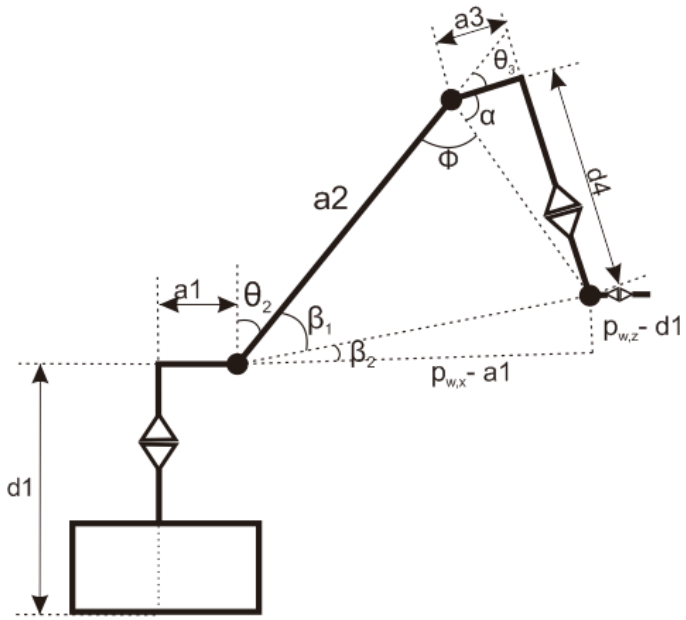


Figure 3 Geometrical Robotic Arm

The first step to find the value of joint 3 is to find the value of α

$$\boldsymbol{\alpha = \text{Atan}\left(\frac{d4}{a3}\right)} \quad \textbf{(9)}$$

$$\boldsymbol{\phi = \text{Acos}\left(\frac{(a_2{}^2 + (d_4{}^2 + a_3{}^2) - ((P_w,z - d_1)^2 + (P_w,x - a_1)^2)}{2a_2(d_4{}^2 + a_3{}^2)}\right)} \quad \textbf{(10)}$$

so the solution to get the value of joint 2 is:

$$\boldsymbol{\theta_3 = \pi - \phi - \alpha} \quad \textbf{(11)}$$

b) Joint 2

With reference to Figure 4.10, before looking for the value of joint 2, we need the angle values β_1, β_2. The following is the calculation to find the angles β_1, β_2.

$$\boldsymbol{\beta_2 = \text{Atan}\left(\frac{P_w,z - d_1}{P_w,x - a_1}\right)} \quad \textbf{(12)}$$

$$\boldsymbol{\beta_1 = \text{Acos}\left(\frac{a_2{}^2 + ((P_w,z - d_1)^2 + (P_w,x - a_1)^2) - (d_4{}^2 + a_3{}^2)}{2a_2\sqrt{(P_w,x - a_1)^2 + (P_w,x - a_1)^2)}}\right)} \quad \textbf{(13)}$$

The final solution used to calculate the angle of joint 2 is as follows

$$\boldsymbol{\theta_2 = 90 - \beta_1 - \beta_2} \quad \textbf{(14)}$$

Solution to find the angle value of joint 4 (θ4), joint 5 (θ5), joint 2 (θ6)
To get the value of θ4, θ5, θ6, it is necessary to calculate the rotation matrix from frame 3 to 6 ($\boldsymbol{R_6^3}$). The following mathematical equation used to calculate $\boldsymbol{R_6^3}$ is :

$$\boldsymbol{R_6^3 = (R_3^0)^{-1}\, R_6^0} \quad \textbf{(15)}$$

Here are the steps to calculate $\boldsymbol{(R_3^0)^{-1}}$:

$$\boldsymbol{R_3^0 = R_1^0 R_2^1 R_3^2} \quad \textbf{(16)}$$

$$\boldsymbol{R_3^0 = \begin{bmatrix} c_{123} - c_1 s_{23} & s_1 & -c_{12}s_3 - c_{13}s_2 \\ c_{23}s_1 - s_{123} & -c_1 & -c_2 s_{13} - c_3 s_{12} \\ -c_2 s_3 - c_3 s_2 & 0 & s_{23} - c_{23} \end{bmatrix}} \quad \textbf{(17)}$$

After getting the value $\boldsymbol{R_3^0}$ the last step is to invert the value.

$$\boldsymbol{(R_3^0)^{-1} = \begin{bmatrix} c_{123} - c_1 s_{23} & c_{23}s_1 - s_{123} & -c_2 s_3 - c_3 s_2 \\ s_1 & -c_1 & 0 \\ -c_{12}s_3 - c_{13}s_2 & -c_2 s_{13} - c_3 s_{12} & s_{23} - c_{23} \end{bmatrix}} \quad \textbf{(18)}$$

While the value of $\boldsymbol{R_6^0}$ or the overall matrix value, can be seen in the overall homogeneous transformation matrix (3 ...) in columns and rows 1-3 that we have calculated in the discussion of forward kinematics.

$$\boldsymbol{R_6^0 = R_1^0 R_2^1 R_3^2 R_5^4 R_6^5} \quad \textbf{(19)}$$

Then the last step to find R_6^3 through equation 4.9 :

$$\boldsymbol{R_6^3 = \begin{bmatrix} R_6^3{}_{11} & R_6^3{}_{12} & R_6^3{}_{13} \\ R_6^3{}_{21} & R_6^3{}_{22} & R_6^3{}_{23} \\ R_6^3{}_{31} & R_6^3{}_{32} & R_6^3{}_{33} \end{bmatrix}} \quad \textbf{(20)}$$

The solutions to find θ_4, θ_5, and θ_6 are as follows:

$$\theta_4 = Atan\left(-R_6^3{}_{23}, -R_6^3{}_{13}\right) \quad \textbf{(21)}$$

$$\theta_5 = Atan\sqrt{\left[\left(\left({R^3_6}_{13}\right)^2 + \left({R^3_6}_{23}\right)^2\right), {R^3_6}_{23}\right]} \text{ atau } \theta_5 = Acos({R^3_6}_{33}) \quad \textbf{(22)}$$

$$\theta_6 = Atan\left(-{R^3_6}_{32}, -{R^3_6}_{31}\right) \quad \textbf{(23)}$$

METHOD

The initial process of this research activity is studying all theories related to research topic robotics science, Arduino programming, Control Design and Matlab software

1. System Setup and Hardware Configuration:

The design design referred to here is electronic design the mechanical design includes the design of the robot arm and its motor. Making mechanical designs will be done in Solidworks software. The mechanical design is related to the wiring diagram of a control and actuator. Making electronic design will be done in proteus software which will be in the form of a schematic circuit.

2. Inverse Kinematics Implementation:

Mathematical Formulation: Derive the inverse kinematics equations specific to the chosen robotic arm's kinematic structure. Algorithm Development: Implement the inverse kinematics algorithm in software, enabling the calculation of joint angles for desired end-effector positions and orientations.

3. Trajectory Planning:

Path Definition: Define welding paths and trajectories using mathematical representations (e.g., cubic splines, Bézier curves) for smooth and accurate movement. Trajectory Generation: Develop algorithms to generate a sequence of waypoints along the desired path, ensuring adherence to specified speed and acceleration profiles.

4. Hardware and Software Integration:

The design referred to here is Electronic design, mechanical design related to wiring diagrams regarding a control and actuator. Making electronic design will be done in proteus software which will be in the form of a schematic circuit. Simulation aims to find out the description of the tool when implemented in real conditions. As with design making, in the tool simulation process there are also 2 processes, namely mechanical simulation (robot arm motion) and electronic simulation. Simulation of robot arm movements will be carried out with the help of Matlab software while the electronic circuit will be simulated with Proteus software. Electronic simulation includes the control of a stepper motor as the driver of the robot arm by the Arduino Mega2560 microcontroller.

5. Data Acquisition and Performance:

After all the hardware is ready, the next step is to create a program in the form of C language that will be planted on the arduino Mega2560 microcontroller. Making the program aims to run the system on a robot arm. Making this program will be done with the help of 2 software, namely using Arduino IDE and Matlab Simulink.

6. Experimental Validation:

After all the hardware is ready, the next step is to create a program in the form of C language that will be planted on the arduino Mega2560 microcontroller. Making the program aims to run the system on a robot arm. Making this program will be done with the help of 2 software, namely using Arduino IDE and Matlab Simulink.

Retrieval is done by giving input or commands in the form of 2 coordinate points along with the end-effector orientation. The data that will be observed is in the form of the position value of the robot end-effector with the angle value of each joint.

Data analysis is done by comparing the Experiment value in the form of end-effector coordinate value and angle of each joint with the simulation value. In addition, the data will be analyzed for the capability process of the 6 DOF robot arm to determine whether the robot arm is capable or not.

7. Documentation and Reporting:

Comprehensive Documentation: Document all aspects of the methodology, including algorithms, code snippets, hardware setup, and experimental procedures. - Performance Report: Summarize the findings of the performance evaluation, including error analysis, trajectory tracking results, and CP analysis.

The outlined methodology provides a comprehensive approach to achieving integrated robotic arm control for inverse kinematics, trajectory planning, and performance evaluation in the context of automated welding. The steps outlined above ensure a systematic and rigorous process to develop, implement, and validate the proposed control system.

RESULT AND DISCUSSION

In order to create a modeling and simulation of a robotic arm, in this research project using Matlab software with additional robotic toolbox. The following are the results of modeling and simulation of the robotic arm for welding. To make a robot arm according to the desired design, data is needed in the form of Denavit-Hartenberg parameters. The following is a visualization of the 6 degree of freedom robot arm with Matlab software.

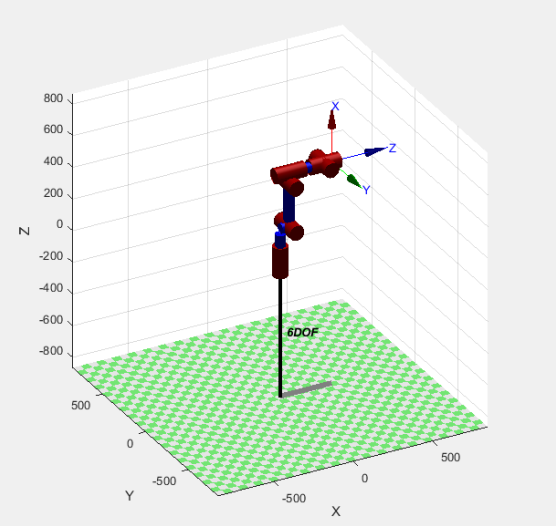


Figure 4 Robot modeling in matlab software

From the picture above, we can visualize the robot arm complete with the position of the end-effector in 3-dimensional space or in the cartesian axis (x, y and z axes).

2. Trajectory Planning:

Table 1 Comparison of Inverse Kinematics Solutions (Analytical and Simulation)

Px=400mm, Py=0mm, Pz=300mm & Sudut Euler (-90º, 180º, 90º)						
Joint Angle	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
Calculation (Analytical)	0º	-59.578º	17.639º	0º	41.918º	0º
Simulation (Numerical)	0º	-59.577º	17.730º	0º	41.847º	0º
Error		0.001º	-0.091º		0.071º	
Px=300mm, Py=0mm, Pz=300mm & Sudut Euler (-90º, 180º, 90º)						
Joint Angle	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
Calculation (Analytical)	0º	-76.377º	43.880º	0º	32.477º	0º
Simulation (Numerical)	0º	-76.379º	43.974º	0º	32.406º	0º
Error		0.002º	0.094º		0.071º	

From the table above, it can be seen that by giving input values, it can be seen that the inverse kinematic solution of geometric and algebraic methods has a value that is realistically the same as the simulation value using numerical methods. This inverse kinematic solution has also been converted into the arduino programming language to be implemented on the robot arm.

Trajectory planning used on this robot arm is a type of LSPB (Linear Segment Parabolic Blend) Trajectory in Cartesian-space. The end-effector

movement planning of the robot arm in the form of a welding torch is expected to follow a straight line trajectory from the coordinate point (Px = 400 mm, Py = 0, Pz = 300) mm to the coordinate point (Px = 300mm, Py = 0mm, Pz = 300mm) by maintaining the end-effector orientation with the euler angle value (90º, -180º, -90º). By using the Matlab Robotics Toolbox simulation, the author has simulated a robot arm end-effector movement from the coordinate points specified above. The visualization of the movement of the robot arm from position 1 to position 2 is shown in Figure 5

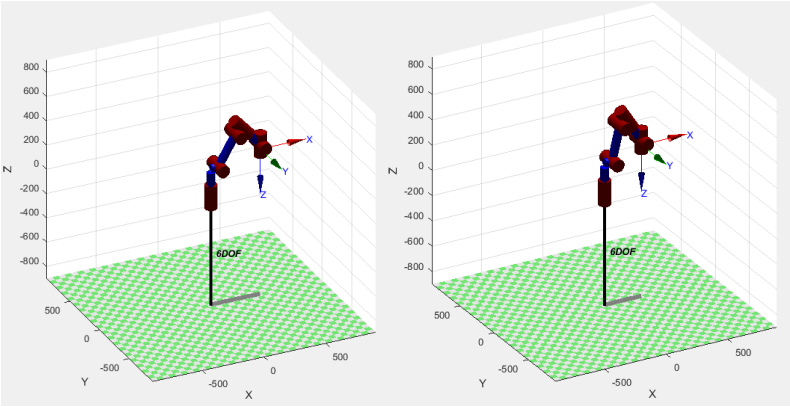


Figure 5 Positions 1 and 2 of the Robotic Arm

In the simulation above the robot arm still maintains the end-effector orientation of 90º, 180º, -90 in euler angle or zyz angle. The value of changes in the position of the end-effector in the x, y, z coordinates and changes in the joint angle of the robot arm can be shown in Figures 2 and 3

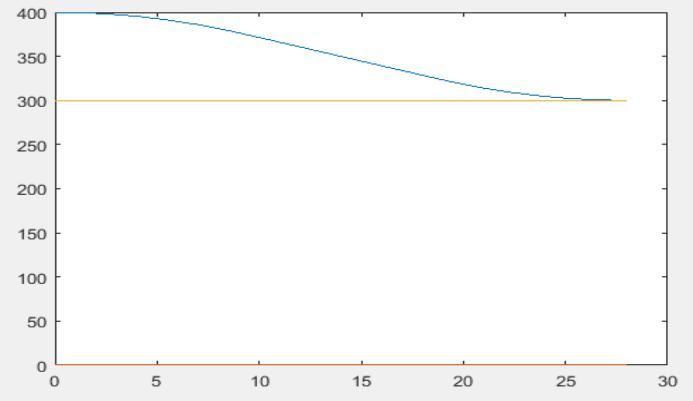


Figure 6 Simulation graph of end-effector movement against time

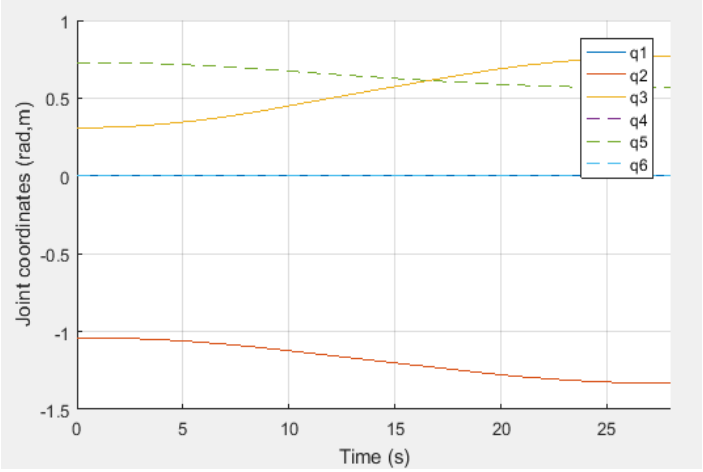


Figure 7 Graph of Joint Angle Change against Time

Trajectory analysis of the robot arm

End-Effector Trajectory

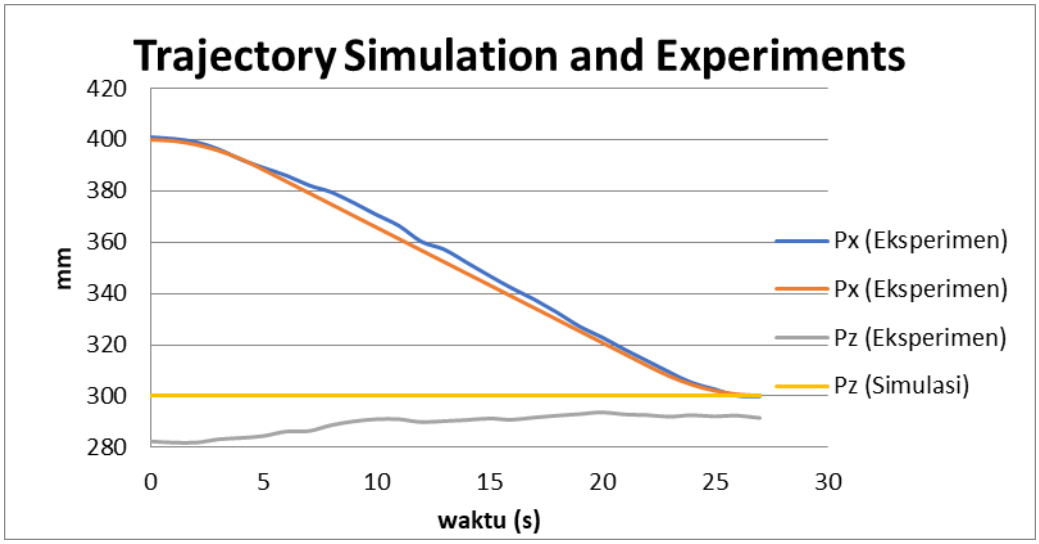


Figure 8 Trajectory Simulation and Experiment Graphic

Figure 8 is a comparison graph of the end-effector trajectory in cartesian coordinates. The red and gray lines are the robot arm trajectory in the x and z axes in the simulation using the Matlab Robotics toolbox, or in other words, the desired trajectory description. While the red and orange lines are the actual trajectory or experimental results of the robot arm on the x and y axes. As discussed in the previous chapter, the robot arm is expected to move smoothly on the x-axis from the 400 mm coordinate point to the 300 mm coordinate, maintaining the orientation of the robot arm. This is intended so that the robot arm can carry out flat welding activities with a straight welding pattern.

We can also see in Figure 8 that the actual movement of the robot arm does not fully follow the desired trajectory. The actual movement of the robot arm on the x-axis when observed is quite smooth when moving from the coordinate point 400 mm to 300 mm, but when compared to the simulation there is still a deviation value. On the z-axis, it can be observed that the robot arm end-effector moves with

a large enough error value. The following is a graph of the error value of the end effector position when moving from the coordinate point (400, 0, 300) mm to the coordinate point (400, 0, 300) by maintaining the euler angle orientation value (-90º, 180º, 90º).

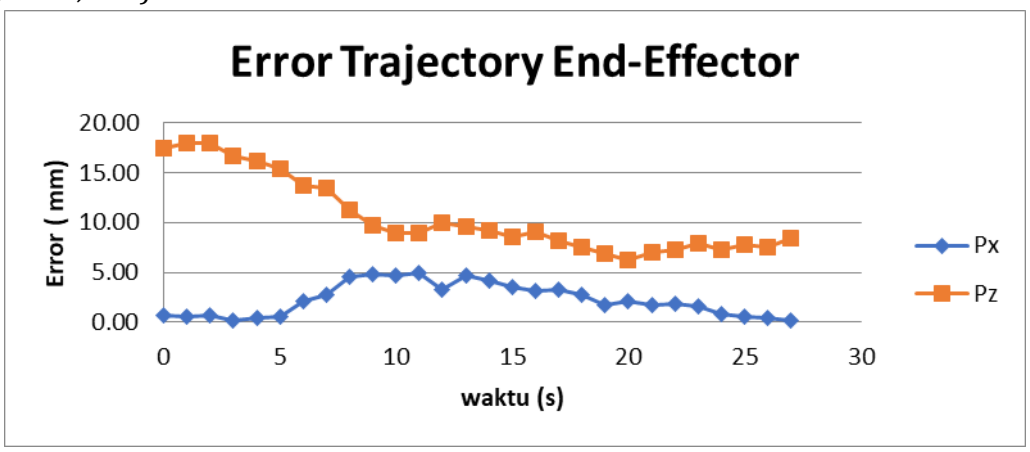


Figure 9 End-effector position error

In Figure 9, it can be seen from the graph above the movement of the robot arm for 27 seconds. It can be seen in the end-effector position that in the end-effector position in the x coordinate, the largest absolute error value that occurs is 4.95 mm, while the largest error that occurs in the y coordinate is 18 mm. The average error value of the x-coordinate is 2.25mm and the average error of the z-coordinate is 10.7 mm.

Error Joint Trajectory

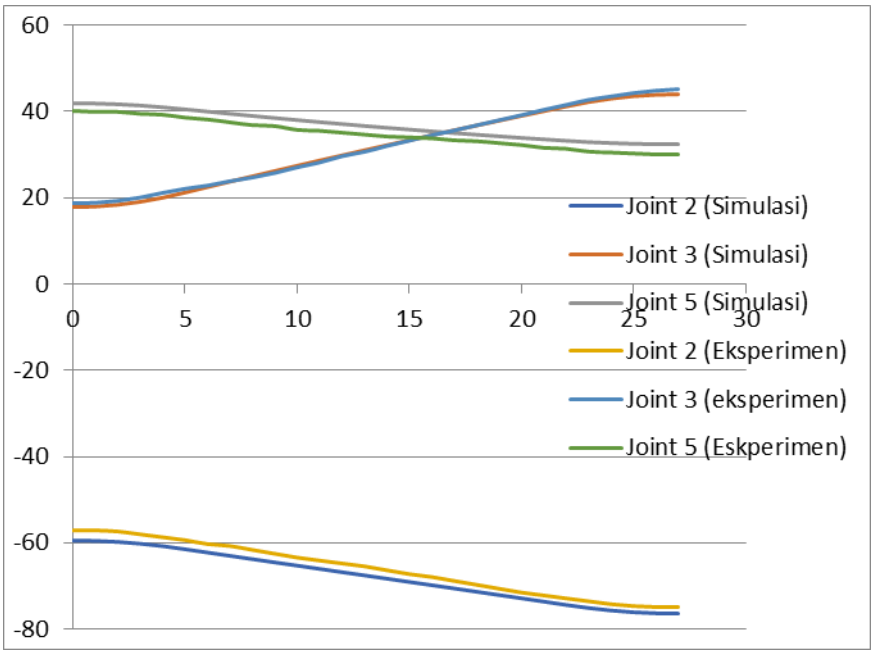
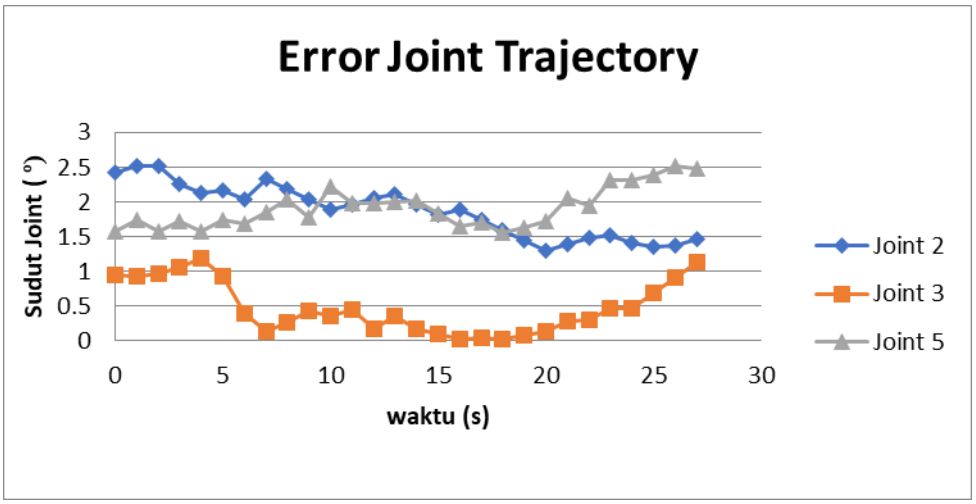


Figure 10 *Joint Trajectory*

As discussed in the previous chapter, that in the planned movement of the robot arm movement across a straight line. The straight line starts from the end-effector position Px = 400mm, Py = 0mm, Pz = 300 to Px = 400mm, Py = 0mm, Pz =

300mm by maintaining the end-effector orientation of -90º, -180º, 90º in euler angles. In Figure 4.20 the graph shows that the joint movement can largely follow the trajectory planning as in the simulation. but there is still an error value in the three joint movements. The average error value by comparing experiments and simulations can be shown in Figure 11.



Gambar 11 *Error Trajectory Joint Angle*

Figure 11 is the joint trajectory error value as the robot moves from coordinate point (400, 0, 300) to coordinate point (300, 0, 300). The error value is the absolute error value by comparing the angular movement value of the simulated joint with the experiment for 27 seconds. The error value at joint 2 tends to decrease until the end of the coordinate position (400, 0, 300) while on the contrary, the angular error value of joint 5 increases when moving towards the coordinate (300, 0, 300). The average error values for joint trajectory 2, 3 and 5 are 1.9º, 0.48º, 1.91º respectively.

CONCLUSION

In this study, we embarked on a comprehensive exploration of integrated robotic arm control with a specific focus on inverse kinematics, trajectory planning, and performance evaluation within the context of automated welding. The amalgamation of these critical elements has yielded valuable insights and tangible advancements in the realm of industrial automation and precision. Our investigation into inverse kinematics unearthed an effective algorithm that translated desired end-effector positions and orientations into accurate joint angles, facilitating precise movement execution. This accomplishment underscores the pivotal role of inverse kinematics in realizing targeted robotic arm maneuvers.
The trajectory planning component of our research exemplified the importance of meticulous path generation. Through systematic algorithms, we were able to orchestrate flawless trajectories that maintained desired speed and acceleration profiles, culminating in the successful automation of welding processes along predefined paths. The harmonious integration of hardware and software

components, as evidenced by our real-time control interface, fortified the usability and applicability of the robotic arm control system. This interface not only simplified user interaction but also facilitated seamless communication between the control interface and the Arduino Mega 2560 microcontroller, enabling streamlined command execution and data acquisition.

The culmination of our efforts materialized in the form of an automated welding process that seamlessly amalgamated the robotic arm and welding apparatus. The system demonstrated remarkable adeptness in adhering to trajectories, resulting in consistent weld quality. Performance evaluation metrics, including joint angle accuracy, trajectory tracking, and weld quality, collectively validated the robustness and precision of the integrated control system. The Capability Process (CP) analysis bolstered our confidence in the system's potential for industrial deployment. The calculated CP values, surpassing the threshold of 1.33, signified the system's capability to consistently achieve welding objectives. This statistical affirmation, coupled with comparative analysis against simulation results, firmly established the system's reliability and efficacy. In conclusion, this study has illuminated the path toward a refined and potent integrated robotic arm control system. The synergy of inverse kinematics, trajectory planning, and rigorous performance evaluation has paved the way for enhanced automated welding processes within the Industry 4.0 landscape. Our findings serve as a testament to the transformative potential of this technology, propelling manufacturing industries toward greater efficiency, precision, and automation.

REFERENCES

[1] "Design and Fabrication of 3-DOF Robot Arm Using Parallelogram Mechanisms," *Int. J. Emerg. Trends Eng. Res.*, vol. 9, no. 9, 2021, doi: 10.30534/ijeter/2021/03992021.

[2] J. D. P. Aguilar and J. V. Padilla, "Control de un brazo robótico usando el hardware kinect® de microsoft," *Prospectiva*, vol. 11, no. 2, 2014, doi: 10.15665/rp.v11i2.43.

[3] S. Broota, "Building of Inmoov Robotic Arm for Performing Various Operations," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 1, 2022, doi: 10.22214/ijraset.2022.39804.

[4] J. Cai, J. Deng, W. Zhang, and W. Zhao, "Modeling Method of Autonomous Robot Manipulator Based on D-H Algorithm," *Mob. Inf. Syst.*, vol. 2021, 2021, doi: 10.1155/2021/4448648.

[5] C. Canali *et al.*, "Design of a Novel Long-Reach Cable-Driven Hyper-Redundant Snake-Like Manipulator for Inspection and Maintenance," *Appl. Sci.*, vol. 12, no. 7, 2022, doi: 10.3390/app12073348.

[6] L. Cheng, D. Li, G. Yu, Z. Zhang, and S. Yu, "Robotic arm control system based on brain-muscle mixed signals," *Biomed. Signal Process. Control*, vol. 77, 2022, doi: 10.1016/j.bspc.2022.103754.

[7] G. Chowdhary, M. Gazzola, G. Krishnan, C. Soman, and S. Lovell, “Soft robotics as an enabling technology for agroforestry practice and research,” *Sustain.*, vol. 11, no. 23, 2019, doi: 10.3390/su11236751.

[8] R. Dong, J. Du, Y. Liu, A. A. Heidari, and H. Chen, “An enhanced deep deterministic policy gradient algorithm for intelligent control of robotic arms,” *Front. Neuroinform.*, vol. 17, 2023, doi: 10.3389/fninf.2023.1096053.

[9] Y. Gao *et al.*, “A Dual-Armed Robotic Puncture System: Design, Implementation and Preliminary Tests,” *Electron.*, vol. 11, no. 5, 2022, doi: 10.3390/electronics11050740.

[10] A. Ibarguren, I. Eimontaite, J. L. Outón, and S. Fletcher, “Dual arm co-manipulation architecture with enhanced human–robot communication for large part manipulation,” *Sensors (Switzerland)*, vol. 20, no. 21, 2020, doi: 10.3390/s20216151.

[11] A. Kawamura, B. Gang, M. Uemura, and S. Kawamura, “Mechanism and control of robotic arm using rotational counterweights,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2015. doi: 10.1109/ICRA.2015.7139567.

[12] H. J. Kim, A. Kawamura, Y. Nishioka, and S. Kawamura, “Mechanical design and control of inflatable robotic arms for high positioning accuracy,” *Adv. Robot.*, vol. 32, no. 2, 2018, doi: 10.1080/01691864.2017.1405845.

[13] T. Kitago *et al.*, “Robotic therapy for chronic stroke: General recovery of impairment or improved task-specific skill?,” *J. Neurophysiol.*, vol. 114, no. 3, 2015, doi: 10.1152/jn.00336.2015.

[14] E. Kopperger, J. List, S. Madhira, F. Rothfischer, D. C. Lamb, and F. C. Simmel, “A self-assembled nanoscale robotic arm controlled by electric fields,” *Science (80-.).*, vol. 359, no. 6373, 2018, doi: 10.1126/science.aao4284.

[15] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario, “Soft robot arm inspired by the octopus,” *Adv. Robot.*, vol. 26, no. 7, 2012, doi: 10.1163/156855312X626343.

[16] F. S. Lee, C. I. Lin, Z. Y. Chen, and R. X. Yang, “Development of a control architecture for a parallel three-axis robotic arm mechanism using CANopen communication protocol,” *Concurr. Eng. Res. Appl.*, vol. 29, no. 3, 2021, doi: 10.1177/1063293X211001956.

[17] S. H. Lee *et al.*, “Robotic Manipulation System Design and Control for Non-Contact Remote Diagnosis in Otolaryngology: Digital Twin Approach,” *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3259539.

[18] J. Li, A. Samoylov, J. Kim, and X. Chen, “Roman: Making Everyday Objects Robotically Manipulable with 3D-Printable Add-on Mechanisms,” in *Conference on Human Factors in Computing Systems - Proceedings*, 2022. doi: 10.1145/3491102.3501818.

[19] L. Liu, Q. Liu, Y. Song, B. Pang, X. Yuan, and Q. Xu, “A collaborative control method of dual-arm robots based on deep reinforcement learning,” *Appl. Sci.*,

vol. 11, no. 4, 2021, doi: 10.3390/app11041816.
[20] A. N. W. Qi, K. L. Voon, M. A. Ismail, N. Mustaffa, and M. H. Ismail, "Design and Development of a Mechanism of Robotic Arm for Lifting," *2nd Integr. Des. Proj. Conf.*, no. December, 2015.
[21] F. Scotto Di Luzio *et al.*, "Bio-cooperative approach for the human-in-the-loop control of an end-effector rehabilitation robot," *Front. Neurorobot.*, vol. 12, no. October, 2018, doi: 10.3389/fnbot.2018.00067.
[22] S. Shirafuji, S. Ikemoto, and K. Hosoda, "Development of a tendon-driven robotic finger for an anthropomorphic robotic hand," *Int. J. Rob. Res.*, vol. 33, no. 5, 2014, doi: 10.1177/0278364913518357.
[23] N. Singh, C. Huyck, V. Gandhi, and A. Jones, "Neuron-based control mechanisms for a robotic arm and hand," *Eng. Technol.*, vol. 11, no. 2, 2017.
[24] A. Suarez, M. Perez, G. Heredia, and A. Ollero, "Cartesian aerial manipulator with compliant arm," *Appl. Sci.*, vol. 11, no. 3, 2021, doi: 10.3390/app11031001.
[25] M. Tang, Y. Yan, Y. Zhang, W. Wang, and B. An, "Motion Control Of Photovoltaic Module Dust Cleaning Robotic Arm Based On Model Predictive Control," *J. Ind. Manag. Optim.*, vol. 19, no. 10, 2023, doi: 10.3934/jimo.2023002.
[26] K. Van Wyk, M. Culleton, J. Falco, and K. Kelly, "Comparative Peg-in-Hole Testing of a Force-Based Manipulation Controlled Robotic Hand," *IEEE Trans. Robot.*, vol. 34, no. 2, 2018, doi: 10.1109/TRO.2018.2791591.
[27] H. Wei, Y. Bu, and Z. Zhu, "Robotic arm controlling based on a spiking neural circuit and synaptic plasticity," *Biomed. Signal Process. Control*, vol. 55, 2020, doi: 10.1016/j.bspc.2019.101640.