

Design and Construction of a Four-Degree Freedom Robot with PID Controller

¹Oscar Andres Otalora Castro, ²Jorge Luis Aroca Trujillo, ³Miguel Angel Tovar Cardozo and
²Ruthber Rodriguez Serrezuela

¹Electronic Engineering, Universidad Antonio Narino-UAN, Bogota, Colombia

²Industrial Engineering, Corporacion Universitaria del Huila, Corhuila, Huila, Colombia

³Faculty of Business Administration, Corporacion Universitaria Minuto de Dios, Uniminuto, Colombia

Abstract: At present, robotic arms are used in many applications, mostly in the field of industry. For fields such as the medical care area, a high precision control in robotics is used. Therefore, the control method that is applied to these systems is important. In this document, a four-degree freedom robotic robot was designed from solidwork. A Proportional-Integral-Derivative (PID) controller was applied based on its kinematics and dynamics found. Both the robot function and the response of the controller were tested. The process of controlling the robotic arm designed and simulated is obtained with MATLAB.

Key words: Kinematics, dynamics, DH parameters, PID control, four-degree, MATLAB

INTRODUCTION

Robotic arms are commonly used in various industries such as process industries, medical fields, nuclear plants to meet a certain number of tasks such as spray paints, transportation of materials, spot welding and arc welding, deposition of material for 3D printing, calligraphy, etc (Serrezuela *et al.*, 2017a-c; Carvajal *et al.*, 2016; Azhmyakov *et al.*, 2015). Each of these applications requires the end effector of the robotic arm to be performed with an accurate level of positioning and generate a much faster response and robust behavior when external disturbances and certain variable load conditions occur. There is also a set of non-linearities such as the slack due to the mechanical transmission, friction and dynamic variables over time of the robotic manipulator system (Azhmyakov *et al.*, 2014a, b; Serrezuela and Chavarro, 2016). They can generate a large tracking error, especially, when a rapid position tracking is required. Additionally, the parameters used in the mechanical parts as well as the drive systems have a certain degree of uncertainty which is why it is also reflected in its complexity (Trujillo *et al.*, 2017, 2018a, b). Therefore, good control of the robotic arm has always been the challenge of researchers due to its level of difficulty (Trujillo *et al.*, 2018; Benavides *et al.*, 2018a, b; Serrezuela *et al.*, 2016a, b).

The control strategy of robotic arms in industries is the most commonly used, since, it takes into account the computed torque control model (Serrezuela *et al.*, 2016a, b; Montiel *et al.*, 2017). The degree of difficulty of this control is relatively simple to apply and generates efficient results in terms of tracking errors and accuracy. Yet, the

uncertainties that arise in the model are due to poorly estimated or unmodeled parameters which cause difficulties in the design of an efficient algorithm based on an exact Mathematical model. For this, other types of solutions have been designed in order to give the system the desired robustness (Serrezuela *et al.*, 2017a-c; Benavides *et al.*, 2018a, b; Perdomo *et al.*, 2017).

The PID control is a control system that allows varying the speed, pressure, temperature and flow between various variables of a process in general through a feedback loop. The PID controller is responsible for calculating the difference between the real variable and the desired variable. In articulation systems, we are usually interested in keeping the speed constant. Therefore, the PID control measures the difference between the speed in the articulation and the required speed and acts by varying the speed of the motor, so that, we can have our acceleration constant (Serrezuela *et al.*, 2017a-c). The PID control algorithm is based on three fundamental parameters:

- Proportional gain (P)
- Integral (I)
- Derivative (D) Serrezuela *et al.* (2018)

The Proportional parameter (P) consists of calculating the difference between the current data and the set point (%) and applying the respective change. The value normally used for submersibles applications is 50% and for centrifugal applications is 10% (Serrezuela *et al.*, 2018a, b):

$$C_p(s) = K_p$$

The Integral (I) parameter refers to the time it takes to carry out the corrective action. For the adjustment to be much faster the value of this parameter has to be smaller, the adjustment is faster, although, it can cause instability in the system such as oscillations and motor vibration. In this case, the appropriate value for submersible applications is half a second and for centrifugal applications, it is one second:

$$u(t) = K_i \int_0^t (\tau) d\tau \quad C_i(s) = \frac{K_i}{s}$$

The Derivative parameter (D) generates a predictive action that it prevents the error and initializes a timely action. It produces a significant correction by responding to the speed of change of this error before the magnitude of the error becomes increasingly large.

The correct attenuation or programming of these three parameters helps us to effectively control our system. If these parameters are not properly programmed, the system can be unstable and effects such as vibration can be generated in the motor and get it damaged:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t (\tau) d\tau$$

PID control is widely used in systems in order to achieve energy savings. It will be implemented in this robotic arm with 4 degrees of freedom to see its response and verify how efficient it is in this system.

MATERIALS AND METHODS

Kynematic model: The symbolic representation of the robotic arm is shown in Fig. 1. To make this

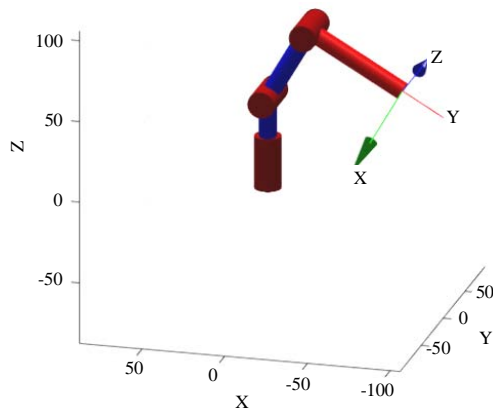


Fig. 1: Symbolic representation of the robotic arm simplified for 3 degrees of freedom

representation, 5 frames of fixed references are used to compose the arm. Because the robotic arm has four degrees of freedom and its four joints are rotational, this manipulator is called angular or articulated and has a spherical work area.

Thanks to the Kinematic model, the movement that is generated in the robot with respect to a fixed reference system can be studied. The direct kynematics is found when you want to determine what the position and orientation of the end of the robot is given the values of the angles of the joints and the measurements of each link of the robot. However, if you want to move the robot to a position and orientation known or specified by the user, there is the inverse kinematics where the configuration that the robotic arm must adopt is resolved. In accordance with the DH agreement, reference frames are placed considering the following guidelines:

- Joints are listed from $i = 1$ to n where i is the i -ith articulation
- Axes z_i are placed along the next joint $i+1$
- If z_i and z_{i-1} are intercepted, the origin of the frame $x_i y_i z_i$ is located at that point. This happens between frames $x_0 y_0 z_0 y x_1 y_1 z_1$ and between frames $x_2 y_2 z_2 y x_3 y_3 z_3$.
- The x_i axis is taken along the common normal between the axes z_i and z_{i-1} with direction from the joint i to $i+1$
- The y_i axis is taken, so that, the frame is completed.
- For the base (frame $x_0 y_0 z_0$) only the direction of the axis z_0 is specified and $x_0 y_0 z_0$ are chosen at convenience
- For the last link there is no frame $i+1$. In general, the joint n is of rotation and the z_n axis is chosen aligned with z_{n-1}

The four parameters of the manipulator are obtained from these frames of reference:

- Link length a_i : distance between the axes a_{i-1} and z_i along the x_i axis
- Joint distance d_i : distance between the axes x_{i-1} and x_i along the axis z_{i-1}
- Angle of articulation θ_i : required rotation of the axis x_{i-1} on z_{i-1} to be parallel to x_i . The rule of the right hand is used to know the meaning
- Torsion angle α_i : required rotation of the z_{i-1} axis on x_i to be parallel to z_i

Direct Kynematic model: As it is seen in Fig. 1, joint No. 4 is not significant which does not affect the end position of the manipulator, thus, it makes the gripper turn,

Table 1: DH parameters

Link (i)	a _i (cm)	α _i	d _i (cm)	θ _i
1	a ₁ = 0	α ₁ = π/2	d ₁ = 230	θ ₁
2	a ₂ = 200	α ₂ = 0	d ₂ = 0	θ ₂
3	a ₃ = 130	α ₃ = 0	d ₃ = 0	θ ₃

therefore, mathematic equations are not as extensive to work with 4 joints. If joint 4 is taken out, there will not be a torsion angle between joints 2 and 3 and their joint distance d₄ is transformed along the link a₃.

Table 1 where the DH parameters were applied for their reference indicates that each one of the rows must be changed in its homogeneous matrix A⁽ⁱ⁻¹⁾_i where it is built upon the composition of 4 basic homogeneous matrices in accordance with Eq. 1:

$$A_i^{i-1} = R_{z,\theta_i} T_{z,d_i} T_{x,a_i} R_{x,\alpha_i} \tag{1}$$

Where:

$$R_{z,\theta_i} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{z,d_i} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{x,a_i} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_{x,\alpha_i} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \cos\alpha_i & -\sin\alpha_i & 0 & 0 \\ \sin\alpha_i & \cos\alpha_i & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrix R_{z,θ_i} indicates rotation in z axis in a θ_i angle, matrix T_{z,d_i} shows movement on the z axis with a d_i path, matrix R_{x,α_i} shows rotation in the x axis with an angle α_i, whereas matrix T_{x,a_i} shows movement on the x axis with a

a_i distance. Doing these matrix operations there can be a result in Eq. 2 which is a homogeneous matrix in function of θ_i because the rest of the parameters are constant:

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

Matrices associated with each link are obtained from Eq. 2:

$$A_1^0 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$A_2^1 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2 \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2 \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$A_3^2 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & a_3 \cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & a_3 \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

The first result A₁⁰ represents the information about the position of axes x₁ y₁ z₁ that refer to the base frame. Both frames are positioned on the same axis one above the other but with a different alignment. The second frame is moved in a [a₂cosθ₂ a₂sinθ₂ 0]^T distance with reference to the first frame and therefore a different distribution.

Lastly, frame x₃-z₃ is moved in a [a₃cosθ₃ a₃sinθ₃ 0]^T distance with reference to the second frame and with a different alignment. The structure of these homogeneous matrices according to Eq. 7 allows finding the PT coordinates of the manipulator in function with the base frame and therefore, the position of this point with reference to the base:

$$T_3^0(q) = A_1^0 A_2^1 A_3^2 \tag{6}$$

Vector q = [θ₁, θ₂, θ₃]^T is called generalized coordinates vector. The result of this operation is:

$$T_3^0(q) = \begin{bmatrix} c_1 c_2 c_3 - c_1 s_2 s_3 & -c_1 c_2 s_3 - c_1 c_3 s_2 & s_1 & a_2 c_1 c_2 - a_3 c_1 s_2 s_3 + a_3 c_1 c_2 c_3 \\ c_2 c_3 s_1 - s_1 s_2 s_3 & -c_2 s_1 s_3 - c_3 s_1 s_2 & -c_1 & a_2 c_2 s_1 - a_3 s_1 s_2 s_3 + a_3 c_2 c_3 s_1 \\ c_2 s_3 + c_3 s_2 & c_2 c_3 - s_2 s_3 & 1 & a_2 s_2 - a_3 c_2 s_3 + a_3 c_3 s_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

where, $\cos\theta_1 = c_1$ and $\sin\theta_1 = s_1$. The direct kinematic model of position for the robotic arm is obtained from matrix (Eq. 7), the terms that relate to the positioning of the end-effector proposed by this matrix correspond to the first three items in the fourth column:

$$\begin{aligned} x &= a_2 c_1 c_2 - a_3 c_1 s_2 s_3 + a_3 c_1 c_2 c_3 \\ y &= a_2 c_2 s_1 - a_3 s_1 s_2 s_3 + a_3 c_2 c_3 s_1 \\ z &= a_2 s_2 - a_3 c_2 s_3 + a_3 c_3 s_2 + d_1 \end{aligned}$$

Whereas its final direction with respect to the base is given by the rotation matrix shown in Eq. 8:

$$R_3^0 = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} c_1 c_2 c_3 - c_1 s_2 s_3 & -c_1 c_2 s_3 - c_1 c_3 s_2 & s_1 \\ c_2 c_3 s_1 - s_1 s_2 s_3 & -c_2 s_1 s_3 - c_3 s_1 s_2 & -c_1 \\ c_2 s_3 + c_3 s_2 & c_2 c_3 - s_2 s_3 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (8)$$

A 4x4 matrix is obtained from Eq. 8.

Inverse Kinematic model of positioning: Opposed to direct kinematics, inverse kinematics searches for angles that must be taken by the joints, so that, the robot is guided and positioned in a specific space location. Calculations for inverse kinematics are much more complex than in direct kinematics because in this case, it depends on the structure of the robot and this leads to unwanted results. In spite of the existence of geometrical methods and computing systems which can calculate the inverse kinematics by using direct kinematics, these complicated methods tend to be slow and unreliable.

Besides in inverse kinematics, solutions may not be just one, therefore, there can be several solutions for the robotic arm that conform to the same positioning and direction. In those cases, some restrictions must be taken into consideration in order to provide such solutions, so that, the best one is applied.

To develop inverse kinematics of the suggested robotic arm, inverse matrices from homogeneous transformation matrices are defined and found in direct kinematics. These are shown in Eq. 9:

$$A_0^1 = \begin{bmatrix} c_1 & s_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ s_1 & -c_1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1^2 = \begin{bmatrix} c_2 & s_2 & 0 & -a_2 \\ -s_2 & c_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^3 = \begin{bmatrix} c_3 & s_3 & 0 & -a_3 \\ -s_3 & c_3 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Once, the matrices representing the reverse rotation are obtained and continuing from (Eq. 9), favored joint variables are based on the spatial position Eq. 10:

$$(A_1^0)^{-1} * T = A_2^1 * A_3^2 \quad (10)$$

The solution of Eq. 10 contains a system of equations which can be the values of the joint by making use of identities and trigonometric equalities Eq. 11:

$$\begin{aligned} \theta_1 &= \arctg\left(\frac{P_y}{P_x}\right) \\ \theta_{2,1} &= \left(2 \arctan\left(\frac{b + \sqrt{b^2 - c^2 + a^2}}{a + c}\right) \right) \\ \theta_{2,2} &= \left(2 \arctan\left(\frac{b - \sqrt{b^2 - c^2 + a^2}}{a + c}\right) \right) \\ \theta_3 &= \arctan\left(\frac{b \cos\theta_2 - a \sin\theta_2}{a \cos\theta_2 + b \sin\theta_2 - a_2}\right) \end{aligned} \quad (11)$$

Where:

$$\begin{aligned} a &= \cos\theta_1 P_x + \sin\theta_1 P_y \\ b &= P_z - d_1 \\ c &= \frac{d_1 * 100 + a^2}{400} \end{aligned} \quad (12)$$

In Eq. 11 two possible solutions for the angle are observed and therefore, there are two solutions for this angle. This is because of the different configurations that can run the robot arm for positioning certain coordinates.

The Jacobian matrix is not nothing but the relationship between the position and orientation of the end with the speeds of the coordinates joint and robot. As well as there is the direct kinematic matrix, there is the direct Jacobian matrix which helps determine the speed of the end of the robot knowing the speeds of each joint. In addition, just as there is the inverse kinematics, there is the inverse Jacobian matrix that allows obtaining speeds joint if the speed of the end of the robot is known.

Direct Jacobian matrix: When making the differentiation of the direct kinematics equations, the direct Jacobian matrix is obtained Eq. 13:

$$\begin{aligned}
 Jd_{[1,1]} &= -130s_1(130c_2c_3 - 200s_2s_3 + 200c_2) \\
 Jd_{[1,2]} &= -130c_1(13c_2s_3 + 200s_2 + 13s_2c_3) \\
 Jd_{[1,3]} &= -130c_1(s_2c_3 + 2c_2s_3) \\
 Jd_{[2,1]} &= -10c_1(-13c_2c_3 + 13s_2s_3 - 20c_2) \\
 Jd_{[2,2]} &= -10s_1(-13c_2s_3 + 20s_2 + 13s_2c_3) \\
 Jd_{[2,3]} &= -130s_1(s_2c_3 + c_2s_3) \\
 Jd_{[3,1]} &= 0 \\
 Jd_{[3,2]} &= 130c_2c_3 - 130s_2s_3 + 200c_2 \\
 Jd_{[3,3]} &= -130s_2s_3 + 130c_2c_3
 \end{aligned} \tag{13}$$

There are two ways to obtain the Jacobian matrix: the direct reverse which consists directly of inverting the direct Jacobian equation and the inverse kinematics inverse Jacobian where a Jacobian matrix is calculated from the inverse kinematics. This study only shows the first case, the direct inverse Jacobian matrix expressed by Eq. 14 (Serrezuela *et al.*, 2017a-c):

$$\begin{aligned}
 Ji_{[1,1]} &= \frac{s_1}{10(-20c_2 - 13c_2c_3 - 13s_2s_3)} \\
 Ji_{[1,2]} &= \frac{-c_1}{10(-20c_2 - 13c_2c_3 - 13s_2s_3)} \\
 Ji_{[1,3]} &= 0 \\
 Ji_{[2,1]} &= -\frac{(s_2s_3 - c_2c_3)s_1}{200s_3} \\
 Ji_{[2,2]} &= \frac{s_1}{10(-20c_2 - 13c_2c_3 - 13s_2s_3)} \\
 Ji_{[2,3]} &= -\frac{s_2c_3 + c_2s_3}{s_3} \\
 Ji_{[3,1]} &= \frac{c_1(-20c_2 - 13c_2c_3 + 13s_2s_3)}{2600s_3} \\
 Ji_{[3,2]} &= \frac{s_1(-20c_2 - 13c_2c_3 + 13s_2s_3)}{2600s_3} \\
 Ji_{[3,3]} &= \frac{13s_2c_3 + 20s_2 + 13c_2s_3}{-2600s_3}
 \end{aligned} \tag{14}$$

Dynamic model: Dynamics is the study of the movement of the bodies taking into account the different internal and external forces of those bodies. If observed from a point of view of the robotics, the relationship between the movement of the links that make up the robotic arm and the forces exerted by actuators that give movement to the joints is analyzed. Once given the dynamics of the robotic arm, a control to each of the actuators of the joints can be applied, so that, the end of the robot can move in a more efficient manner.

The dynamic model is responsible for mathematically relating the location of the robot through the velocity and acceleration produced by forces and pairs applied to joints. In addition, once known the dynamics of the robot, the movements of the robot can be generated, so that, the end perform an action or follow a path determined by the user. Based on the balance of forces established by Newton's second Law or equivalent for movements of rotation that is Euler's Law, the dynamic model of a robot is generated:

$$\begin{aligned}
 \Sigma F &= m\dot{v} \\
 \Sigma T &= L\dot{\omega} + \omega x(l\omega)
 \end{aligned}$$

The dynamic model based on energy considerations of the type can even be developed as the Lagrangian approach. This research was carried out by using Newton-Euler approach and starting with the balance of forces and pairs for each link of the robotic arm. It uses an algorithm whose order of complexity is based on the following formula O^n , i.e., that the complexity depends directly on the number of degrees of freedom. Forces and acting pairs on a link i with reference to the base of the robot are obtained with this method. Carvajal *et al.* (2016) Eq. 15 based on vector type operations which are more efficient in comparison to the matrix operations necessary in the Lagrangian formulation (Serrezuela *et al.*, 2017a-c):

$$\begin{aligned}
 T_1 &= 82582.37\ddot{q}_1 - 200051.39\dot{q}_2^2 - \\
 &\quad 35.19(\dot{q}_2 + \dot{q}_3)(-78.89\dot{q}_2 - 78.8\dot{q}_3) - \\
 &\quad 35.19(\dot{q}_2 + \dot{q}_3)(130\dot{q}_2 + 130\dot{q}_3) \\
 T_2 &= 15717.79\ddot{q}_2 + 1967.8\ddot{q}_3 + 91.98gz \\
 T_3 &= 1967.8\ddot{q}_2 + 401.8\ddot{q}_3 + 91.98gz
 \end{aligned} \tag{15}$$

Computed torque PID controller: When using the PD controller, it is likely that there is an error in steady state monitoring and control theory recommends adding an integrator to the controller at each joint in the external arm of the computed PD torque controller. Thus, a computed torque PID controller is produced when selected $u_{(t)}$ as a proportional, integral and derivative control:

$$\begin{aligned}
 \dot{e} &= e \\
 \tau &= M(\ddot{q} + K_v\dot{e} + K_p e) + N
 \end{aligned}$$

This controller does not change the dynamics of the system and however, it adds stability on the tracking provided that k_i does not have a very large value (Fig. 2). This controller was simulated on the dynamic model of the robotic arm which uses the computed torque PID equation and the control signal is given by:

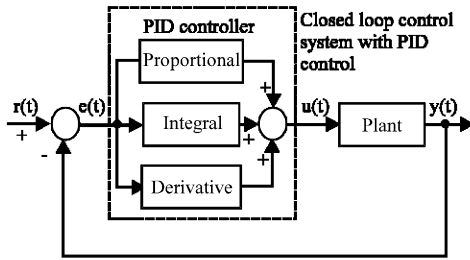


Fig. 2: Computed torque PID control diagram

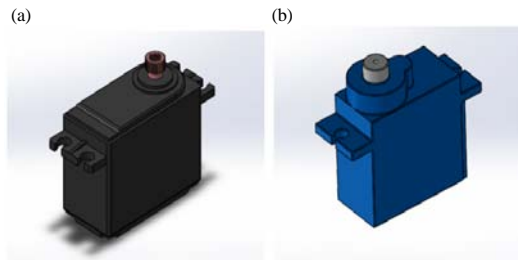


Fig. 3: Servo design: a) mg995 and b) sg90

$$\tau = \hat{M}(\ddot{q} + K_v \dot{e} + K_p e) + \hat{C}q + \hat{G}$$

Where:

$$\dot{e} = e$$

For the parameters of controller constants, the following values are taken $k_v = 20$, $K_p = 100$ and $K_i = 10$.

Mechanical design of the robotic arm: The design of the robotic arm was made through the Solidwork program which is a CAD Software (Computer-Aided Design) for mechanical modeling in 2D and 3D. This allows to model parts and assemblies and generates technical drawings as well as other necessary information for its construction.

Initially, the servos for the work were selected, simulating each of the corresponding articulations from the base to the clamp. Since, the first 3 articulations exert greater force in their respective order, servos mg 995 were used whereas servos sg90 were used for the last joint and the clamp (Fig. 3).

Subsequently, each of the parts that make up the robotic arm were created in entirety with measurements in relation to the servos mg995 and sg90 for a suitable coupling. They were created in such a way that a robotic arm could be robust and of good appearance and whose model was not having problems when giving it movement as generation of trajectories or simulating its kinematics, dynamics and applying the PID control (Fig. 4-6).

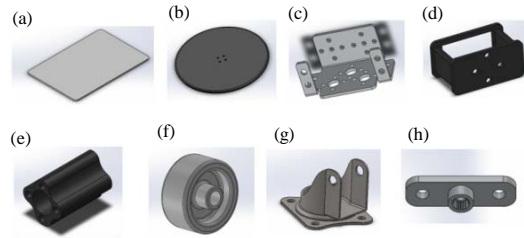


Fig. 4: a-h) Design of each of the pieces that make up the robotic arm

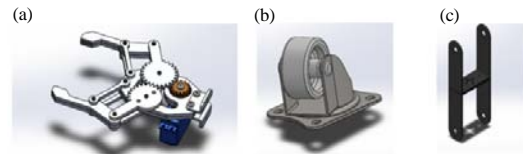


Fig. 5: a-c) Assembly of the parts that make up the robotic arm

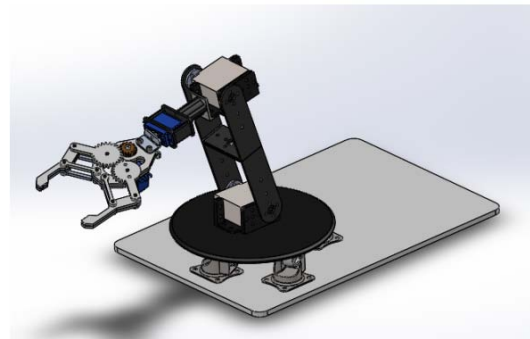


Fig. 6: Final Mechanical design of the robotic arm

Once all the pieces used in the design of the robotic arm were created, the next thing was to make the respective assemblies where the robotic arm was being shaped little by little (Fig. 5). Taking the small assemblies, the total assembly is carried out where the entire desired robotic arm is assembled. Figure 6 shows the final constructed design.

RESULTS AND DISCUSSION

Figure 7 shows that the angular errors with respect to the references do not disappear that is the system is not stabilized. It can be seen in all the graphs that this controller is not as accurate as the PID controller of computed torque, since, the desired trajectories are not followed as faithfully as stipulated where the desired path in the simulation is the green one and the real one in the simulation is red (Fig. 8-10).

As described in the document and observed in Fig. 12, the movement of the last joint corresponding to

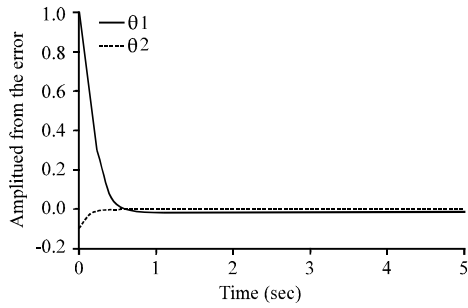


Fig. 7: Angular error PD of computed torque

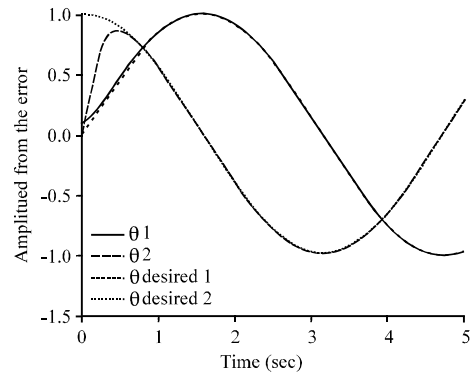


Fig. 8: Real and desired trajectories

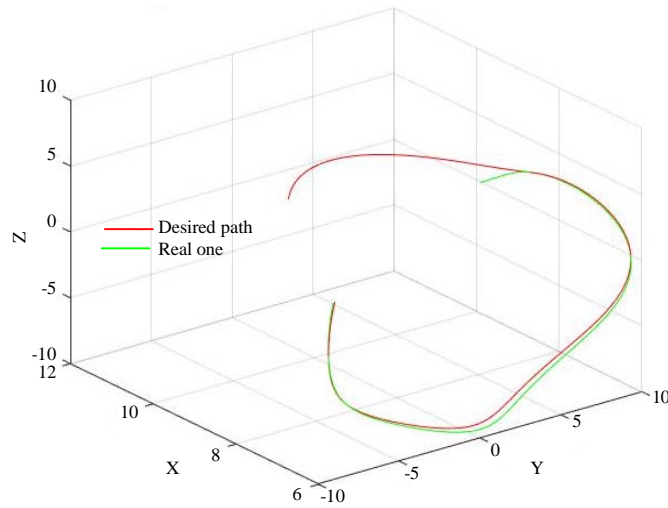


Fig. 9: End path with a PD controller

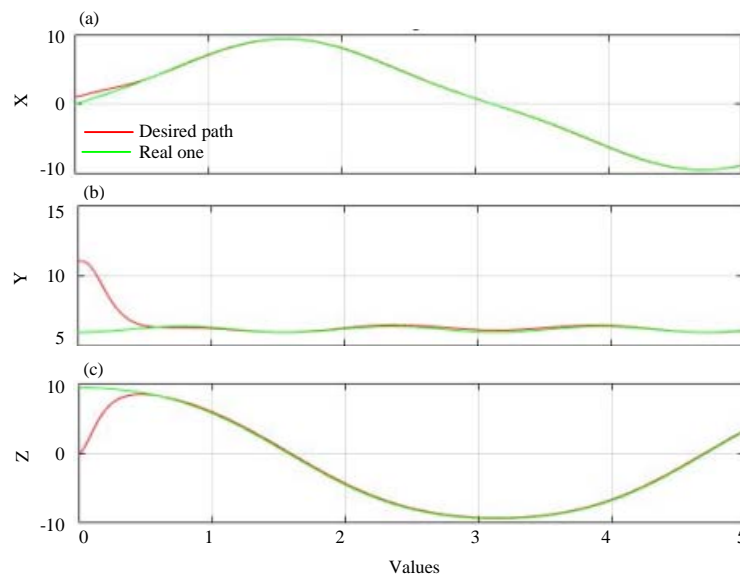


Fig. 10: a-c) Movement in cartesian axes

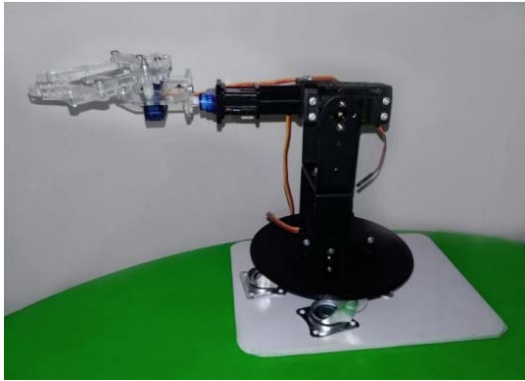


Fig. 11: Physical construction of the robotic arm

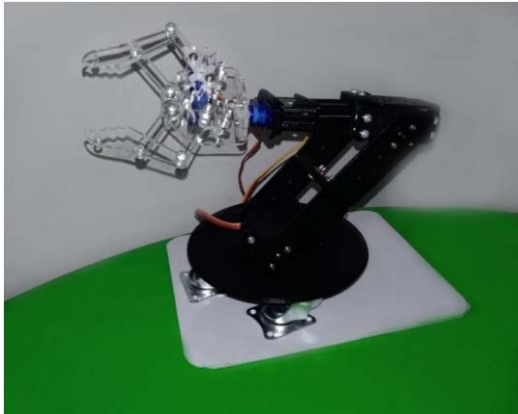


Fig. 12: Movement of the fourth joint of the robotic arm

joint number 4 only serves to rotate the clamp, therefore, it does not affect at all the final positioning of the robotic arm (Fig. 11 and 12).

CONCLUSION

It can be concluded that the smaller value of K_p , the better behavior of this controller. That is, we reach the conclusion that the best behavior will be given in $K_p = 0$, converting it into a PD control.

Using CAD-type tools is a great help for the mechanical design of the robotic arms, since, it allowed an evaluation and verification of different aspects of it before its manufacture. It relatively reduces manufacturing times and the need for more material for construction of the robotic arm which helps to generate less costs in the development of the project.

The use of software tools such as MATLAB for mathematical calculation also helped considerably in reducing the development time of the system modeling. It

allowed different simulations and variations of the proposed models to be carried out in order to obtain the final design of the robotic arm.

REFERENCES

- Azhmyakov, V., J.C. Martinez, A. Poznyak and R.R. Serrezuela, 2015. Optimization of a class of nonlinear switched systems with fixed-levels control inputs. Proceedings of the International Conference on American Control Conference (ACC), July 1-3, 2015, IEEE, Chicago, Illinois, pp: 1770-1775.
- Azhmyakov, V., R.R. Serrezuela and L.G. Trujillo, 2014b. Approximations based optimal control design for a class of switched dynamic systems. Proceedings of the IEEE 40th Annual Conference on Industrial Electronics Society (IECON), October 29-November 1, 2014, IEEE, Dallas, Texas, USA., ISBN:978-1-4799-4032-5, pp: 90-95.
- Azhmyakov, V., R.R. Serrezuela, A.M.R. Gallardo and W.G. Vargas, 2014a. An approximations based approach to optimal control of switched dynamic systems. Math. Prob. Eng., 2014: 1-9.
- Benavides, L.C.L., L.A.C. Pinilla, J.S.G. Lopez and R.R. Serrezuela, 2018b. Electrotoxic biodegradation study of the carbofuran insecticide in soil. Intl. J. Appl. Eng. Res., 13: 1776-1783.
- Benavides, L.C.L., L.A.C. Pinilla, R.R. Serrezuela and W.F.R. Serrezuela, 2018a. Extraction in laboratory of heavy metals through rhizofiltration using the plant *Zea mays* (maize). Intl. J. Appl. Environ. Sci., 13: 9-26.
- Carvajal, R.J.H., R.R. Serrezuela, J.A.Q. Lopez and K.L.R. Perdomo, 2016. LQR hybrid approach control of a robotic arm two degrees of freedom. Intl. J. Appl. Eng. Res., 11: 9221-9228.
- Montiel, J.J.G., R.R. Serrezuela and E.A. Aranda, 2017. Applied mathematics and demonstrations to the theory of optimal filters. Global J. Pure Appl. Mathe., 13: 475-492.
- Perdomo, E.G., M.A.T. Cardozo, C.A.C. Perdomo and R.R. Serrezuela, 2017. A review of the user based web design: Usability and information architecture. Intl. J. Appl. Eng. Res., 12: 11685-11690.
- Serrezuela, R.R. and A.F.C. Chavarro, 2016. Multivariable control alternatives for the prototype tower distillation and evaporation plant. Intl. J. Appl. Eng. Res., 11: 6039-6043.
- Serrezuela, R.R., A.F. Chavarro, M.A.T. Cardozo, A.G. Rey and C.A. Cabrera, 2017a. Audio signals processing with digital filters implementation using MyDSP. J. Eng. Appl. Sci., 12: 5-10.

- Serrezeuela, R.R., A.F.C. Chavarro, M.A.T. Cardozo and J.B.R. Zarta, 2016b. An optimal control based approach to dynamics autonomous vehicle. *Intl. J. Appl. Eng. Res.*, 11: 8841-8847.
- Serrezeuela, R.R., A.F.C. Chavarro, M.A.T. Cardozo, A.L. Toquica and L.F.O. Martinez, 2017b. Kinematic modelling of a robotic arm manipulator using matlab. *J. Eng. Appl. Sci.*, 12: 2037-2045.
- Serrezeuela, R.R., J.A. Trujillo, A.N. Ramos and J.R. Zarta, 2018b. Applications Alternatives of Multivariable Control in the Tower Distillation and Evaporation Plant. In: *Advanced Engineering Research and Applications*, Ajaykumar, B.S. and D. Sarkar (Eds.). Research India Publication, Delhi, India, pp: 452-465.
- Serrezeuela, R.R., M.A.T. Cardozo, D.L. Ardila and C.A.C. Perdomo, 2018a. A consistent methodology for the development of inverse and direct kinematics of robust industrial robots. *ARNP. J. Eng. Appl. Sci.*, 13: 293-301.
- Serrezeuela, R.R., N.C. Sanchez, J.B.R. Zarta, D.L. Ardila and A.L.P. Salazar, 2017c. Case study of energy management model in the threshing system for the production of white rice. *Intl. J. Appl. Eng. Res.*, 12: 8245-8251.
- Serrezeuela, R.R., O.F. Villar, J.R. Zarta and Y.H. Cuenca, 2016a. The K-exponential matrix to solve systems of differential equations deformed. *Global J. Pure Appl. Math.*, 12: 1921-1945.
- Trujillo, J.A., R.R. Serrezeuela, J.R. Zarta and A.N. Ramos, 2018b. Direct and Inverse Kinematics of a Manipulator Robot of Five Degrees of Freedom Implemented in Embedded System-CompactRIO. In: *Advanced Engineering Research and Applications*, Ajaykumar, B.S. and D. Sarkar (Eds.). Research India Publication, Delhi, India, pp: 405-419.
- Trujillo, J.L.A., A. Perez-Ruiz and R. Rodriguez Serrezeuela, 2017. Generation and control of basic geometric trajectories for a robot manipulator using compact RIO®. *J. Rob.*, 2017: 1-11.
- Trujillo, J.L.A., J.B.R. Zarta and R.R. Serrezeuela, 2018c. Embedded system generating trajectories of a robot manipulator of five Degrees of Freedom (DOF). *KnE Eng.*, 3: 512-522.
- Trujillo, J.L.A., R.R. Serrezeuela, V. Azhmyakov and R.S. Zamora, 2018a. Kinematic model of the scorbot 4PC manipulator implemented in Matlabs Guide. *Contemp. Eng. Sci.*, 11: 183-199