International Journal of Emerging Engineering Research and Technology
Volume 6, Issue 11, 2018, PP 24-31
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

Modeling and Control of 2-DOF Robot Arm

Nasr M. Ghaleb[1] and Ayman A. Aly[1, 2]

[1]*Mechanical Engineering Department, College of Engineering, Taif University, PO Box 888, Taif, Saudi Arabia*

[2]*Mechanical Engineering Department, Faculty of Engineering, Assiut University, PO Box 71516, Assiut, Egypt*

***Corresponding Author:** Nasr M. Ghaleb, Mechanical Engineering Department, College of Engineering, Taif University, PO Box 888, Taif, Saudi Arabia*

ABSTRACT

This paper presents a Modeling, Simulation and Control of a Two Degree of Freedom (2-DOF) robot arm.This Work is taken from the Final Year capstone project. First The Robot specifications , Robot Kinematics with Denavit-Hartenberg parameters (DH)for Forward kinematics and Inverse Kinematicsof 2-DOF robot armwere presented. Then The dynamics of the 2-DOF robot arm was studied to derive the equations of motion based on Eular-Lagrange Equation of motion. A Control Design was performed using PID controller for the modeling and control Technique. The models has been done based on Matlab/Simulink software.

Keywords: *Robotics, 2-DOF Robot arm, Kinematic, Dynamic, PID Control and Modeling.*

INTRODUCTION

Robotics is defined practically as the study, design and use of robot systems for manufacturing and generally are used to perform highly repetitive, unsafe, hazardous, and unpleasant tasks. Robotics many different functions that used either in industry and manufacture or in complex, clatter and changing environment such as pick and place, assembly, drilling, welding, machine tool load and unload functions, painting, spraying, etc. or in A delivery in a hospital and Hotels, Discovering the space As a results of these different tasks there are different robot arm configuration such as rectangular, spherical, cylindrical, revolute and prismatic jointed.

A pick and place robot arm is used to ease process of moving materials and supplying the motion required in the manufacturing processes. The transfer process of the materials is usually being accomplished, using man power and as the transfer process is repeated for a period of time, it can cause injuries to the operator. the robot arm preventing injuries and increasing the efficiency of the work, with reducing the human being errors that cost highly time and martial.

The proportional-integral-derivative (PID) control has simple structure for its three gains. The control performances are acceptable in the most of industrial processes. Most robot manipulators found in industrial operations are controlled by PID algorithms independently at each joint.

There are many control techniques used for controlling the robot arm.

The most familiar control techniques are the PID control, adaptive control, optimal control and robust control. As the final goal is to design and manufacturing real robots, it's helpful doing the simulation before the investigations with real robots, to enhance the final real robot performance and behavior.

ROBOT SPECIFICATION AND KINEMATICS

Robot Specification

A two degree of freedom robot arm is described in Figure(1)

which consists primarily of two links with the following specifications in OXY coordinates:

L_1 = 1 m is the length of the first link.
L_2 = 1 m is the length of the second link.

m_1 = 1 kg is the mass of the first link.
m_2 = 1 kg is the link of the second link.

θ_1 = the rotation angel of the first link.

θ_2 = is the rotation angel of the second link.

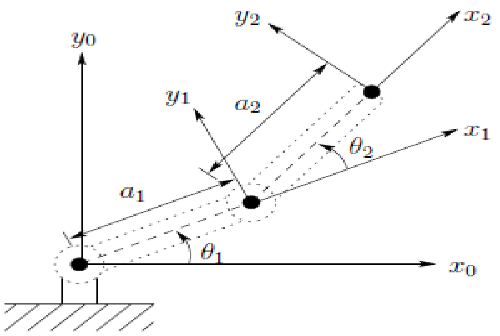

Figure 1. *Two degree of freedom Robot Arm*

Robot Kinematics

Forward Kinematics

The Forward kinematics of a robotic arm is determined a group of parameters called Denavit-Hartenberg (DH) parameters which used for deriving the homogenous transformation matrices between the different frames assigned on the robot arm structure. The DH parameters for a two degree of freedom robotic arm are defined as follows:

Table 1. *DH-parameters for the 2-DOF robotic arm*

Link	$\boldsymbol{a_i}$	$\boldsymbol{\alpha_i}$	$\boldsymbol{d_i}$	$\boldsymbol{\theta_i}$
1	L_1	0	0	θ_1
2	L2	0	0	θ_2

The homogenous transformation matrices for the 2-DOF robotic arm shown in Figure(1) are derived as follows:

$$ {}_1^0\boldsymbol{T} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & L_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & L_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$ {}_2^1\boldsymbol{T} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Using the Eq. (1) and (2), the homogenous transformation matrix ${}_2^0T$can be derived as follows:

$$ {}_2^0\boldsymbol{T} = \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Therefore,

$$ {}_H^R\boldsymbol{T} = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$ {}_2^0\boldsymbol{T} = {}_H^R\boldsymbol{T} \quad (5)$$

From Eq. (5), the position coordinates of the manipulator end-effector is given by:

$$Px = L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \quad (6)$$

$$Py = L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2) \quad (7)$$

And the end-effector's orientation matrix is defined by the first three rows and three columns of the transformation matrix in Eq.(3).

Inverse Kinematics

The inverse kinematics of a robotic arm is a solution of finding the robot arm joint variables of given the position Cartesian coordinates of the end-effector. The mathematical equations used to solve the inverse kinematics problem can be derived either algebraically or geometrically. The geometrical approach is considered to be much easier for robot arms of high degrees of freedom. In our Case, we solved the inverse kinematics equations for the 2-DOF robotic arm shown in Figure(2) using the geometrical method.

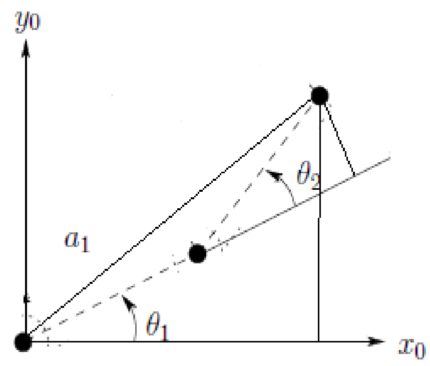

Figure 2. *Two degree of freedom Robot Arm Inverse Kinematic*

From Figure2, a mathematical equation for solving the elbow joint angle $\boldsymbol{\theta}_2$ can be derived using Pythagoras theorem as follows:

$$p_x{}^2 + p_y{}^2 = L_1{}^2 + L_2{}^2 + 2L_1L_2\cos\theta_2 \quad (8)$$

$$\cos\theta_2 = \frac{1}{2L_1L_2}\left(p_x{}^2 + p_y{}^2 - L_1{}^2 - L_2{}^2\right) \quad (9)$$

$$\sin\theta_2 = \pm\sqrt{1 - \cos\theta_2{}^2} \quad (10)$$

Therefore,

$$\theta_2 = \pm\, atan\frac{\sin\theta_2}{\cos\theta_2} \quad (11)$$

For The joint variable θ_1:

$$p_x = (L_1 + L_2\cos\theta_2)\cos\theta_1 - L_2\sin\theta_1\sin\theta_2 \quad (12)$$

$$p_y = L_2\sin\theta_2\cos\theta_1 + (L_1 + L_2\cos\theta_2)\sin\theta_1 \quad (13)$$

$$\Delta = \begin{bmatrix} L_1 + L_2\cos\theta_2 & -L_2\sin\theta_2 \\ L_2\sin\theta_2 & L_1 + L_2\cos\theta_2 \end{bmatrix} \quad (14)$$

$$p_x{}^2 + p_y{}^2 = (L_1 + L_2\cos\theta_2)^2 + (L_2\sin\theta_2)^2 \quad (15)$$

$$\Delta\sin\theta_1 = \begin{bmatrix} L_1 + L_2\cos\theta_2 & p_x \\ L_2\sin\theta_2 & p_y \end{bmatrix} \quad (16)$$

$$\Delta\cos\theta_1 = \begin{bmatrix} p_x & -L_2\sin\theta_2 \\ p_y & L_1 + L_2\cos\theta_2 \end{bmatrix} \quad (17)$$

$$\sin\theta_1 = \frac{\Delta\sin\theta_1}{\Delta} = \frac{(L_1+L_2\cos\theta_2)p_y - L_2\sin\theta_2 p_x}{Px^2+p_y{}^2} \quad (18)$$

$$\cos\theta_1 = \frac{\Delta\cos\theta_1}{\Delta} = \frac{(L_1+L_2\cos\theta_2)p_x + L_2\sin\theta_2 p_y}{p_x{}^2+p_y{}^2} \quad (19)$$

$$\theta_1 = atan\frac{\sin\theta_1}{\cos\theta_1} = atan\frac{(L_1+L_2\cos\theta_2)p_y \pm L_2\sin\theta_2 p_x}{(L_1+L_2\cos\theta_2)p_x \pm L_2\sin\theta_2 p_y} \quad (20)$$

ROBOT DYNAMICS

The dynamic model of a robot is concerned with the movement and the forces involved in the robot arm and establishes a mathematical relationship between the location of the robot joint variables and the dimensional parameters of the robot. There are two methods for performing the dynamics equations of a robot arm: Eular-Lagrange method and Newten-Eular method. in this work , We used the Eular-Lagrange method which depends on calculating the total Kineatic and Potential Energies of the robot arm to determine the Lagrangian ($\mathcal{L}$) of the whole system in order to calculate the force or torque applied of each joint.With the Lagrangian$\mathcal{L}$, we can solve the Euler- Lagrange equation which relies on the partial derivative of kinetic and potential energy properties of mechanical systems to compute the equations of motion and is defined as follow:

$$F = \frac{d}{dt}\left[\frac{\partial\mathcal{L}}{\partial_{\dot{\theta}}}\right] - \frac{\partial\mathcal{L}}{\partial_{\theta}} \quad (21)$$

where F is the external force acting on the generalized coordinate, represents the torque applied to therobot and $\mathcal{L}$ is the Lagrangian equation of the motion, given in Eq.(22)

$$\mathcal{L}(q(t), \dot{q}(t)) = K_E(q(t), \dot{q}(t)) - P_E(q(t)) \quad (22)$$

To solve the Lagrangian Eq.(22),we need first to calculate the kinetic Energy KE and potential Energy PE as follow:

$$KE = \frac{1}{2}m\dot{x}^2$$

$$PE = mgl$$

The velocity is determined by taking the derivative of the position respect to time. so the position in the end of the link is known using known familiar variables :

$$x_1 = L_1 sin\theta_1$$

$$y_1 = L_1\cos\theta_1$$

$$x_2 = L_1 sin\theta_1 + L_2\sin(\theta_1 + \theta_2) \quad (23)$$

$$y_2 = L_1\cos\theta_1 + L_2\cos(\theta_1 + \theta_2)$$

Then substitute in the kinetic's energy equation:

$$KE = \frac{1}{2}m_1\dot{x}_1{}^2 + \frac{1}{2}m_1\dot{y}_1{}^2 + \frac{1}{2}m_2\dot{x}_2{}^2 + \frac{1}{2}m_2\dot{y}_2{}^2 \quad (24)$$

After simplification,

$$KE = \frac{1}{2}(m_1 + m_2)l_1{}^2\dot{\theta}_1{}^2 + \frac{1}{2}m_2 l_1{}^2\dot{\theta}_1{}^2 + m_2 l_2{}^2\dot{\theta}_1\dot{\theta}_2 + \frac{1}{2}m_2 l_2{}^2\dot{\theta}_2{}^2 + m_2 l_1 l_2 cos\theta_2\left(\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1{}^2\right) \quad (25)$$

The Potential energy equation is defined as:

$$PE = m_1 g l_1\cos\theta_1 + m_2 g(l_1\cos\theta_1 + l_2 cos(\theta_1 + \theta_2)) \quad (26)$$

Substitute Eq.(25) and(26) in Eq.(21) to form the lagrangian equation as:

$$\mathcal{L} = \frac{1}{2}(m_1 + m_2)l_1{}^2\dot{\theta}_1{}^2 + \frac{1}{2}m_2 l_1{}^2\dot{\theta}_1{}^2 + m_2 l_2{}^2\dot{\theta}_1\dot{\theta}_2 + \frac{1}{2}m_2 l_2{}^2\dot{\theta}_2{}^2 + m_2 l_1 l_2 cos\theta_2\left(\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1{}^2\right) - m_1 g l_1\cos\theta_1 + m_2 g(l_1\cos\theta_1 + l_2 cos(\theta_1 + \theta_2)) \quad (27)$$

To calculate the force applied to the robot, we form the Lagrange-EularEq.(21)with the Lagrangian($\mathcal{L}$)

$$F_{\theta_{1,2}} = \frac{d}{dt}\left[\frac{\partial\mathcal{L}}{\partial_{\dot{\theta}_{1,2}}}\right] - \frac{\partial\mathcal{L}}{\partial\theta_{1,2}}$$

After simplification, the force applied at joint 1is given by

$$F_{\theta_1} = (m_1 + m_2)l_1{}^2 + m_2 l_2{}^2 + 2m_2 l_1 l_2 \cos\theta_2)\ddot{\theta_1} + \left(m_2 l_2{}^2 - m_2 l_1 l_2 \cos\theta_2\right)\ddot{\theta_2} - m_2 l_1 l_2 \sin\theta_2 (2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2{}^2) - (m_1 + m_2) l_1 g \sin\theta_1 - m_2 l_2 g sin(\theta_1 + \theta_2) \quad (28)$$

and the force applied at joint 2 is given by

$$F_{\theta_2} = (m_2 l_2{}^2 + m_2 l_1 l_2 \cos\theta_2)\ddot{\theta_1} + m_2 l_2{}^2\ddot{\theta_2} - m_2 l_1 l_2 \sin(\theta_2)\,\dot{\theta}_1\dot{\theta}_2 - m_2 l_2 g sin(\theta_1 + \theta_2) \quad (29)$$

The motion of the systemis given by the following formof a nonlinear equation**:**

$$F = B(\ddot{q}) + C(\dot{q}, q) + g(q) \quad (30)$$

where,

$$F = \begin{bmatrix} F_{\theta_1} \\ F_{\theta_2} \end{bmatrix}$$

$$B(\ddot{q}) = \begin{bmatrix} ((m_1 + m_2)l_1{}^2 + m_2 l_2{}^2 + 2m_2 l_1 l_2 \cos\theta_2) & (m_2 l_2{}^2 - m_2 l_1 l_2 \cos\theta_2) \\ m_2 l_2{}^2 + m_2 l_1 l_2 \cos\theta_2 & m_2 l_2 \end{bmatrix}$$

$$C(\dot{q}, q) = \begin{bmatrix} -m_2 l_1 l_2 \sin\theta_2 (2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2{}^2) \\ -m_2 l_1 l_2 \sin(\theta_2)\,\dot{\theta}_1\dot{\theta}_2 \end{bmatrix}$$

$$g(q) = \begin{bmatrix} -(m_1 + m_2) l_1 g \sin\theta_1 - m_2 l_2 g sin(\theta_1 + \theta_2)) \\ -m_2 l_2 g sin(\theta_1 + \theta_2) \end{bmatrix}$$

$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

MATHEMATICAL MODELING OF ACTUATING SYSTEM

The Permanent Magnet Direct Current PMDC motor is used to actaute the system, which has an Electrical part and a Mechanical part, as seen in Eq.(44) and Eq.(45), Describing the Electrical and Mechanical characteristics of PMDC motor, respectively, Using Newton's law, Kirchhoff's law and Ohm's law.

$$[V_{in}(s) - K_b \omega_m] * \frac{1}{(L_a s + R_a)} = I_a(s) \quad (31)$$

$$\omega(s) = K_t * I_a(s) * \frac{1}{(J_m\ s + b_m)} \quad (32)$$

The PMDC motor Transfer Function is derived, Subtituting the electrical part characteristics Eq.(44) in Eq.(45), gives Eq.(46):

$$K_t \frac{1}{(L_a s + R_a)} [V_{in}(s) - K_b \omega_m] = J_m\ s\omega + b_m \omega \quad (33)$$

Rearranging Eq.(46), The PMDC motor *open* loop transfer function is obtained, given by Eq.(47), This equation shows that the PMDC motor is a second order system.

$$G_{speed}(s) = \frac{\omega(s)}{V_{in}(s)} = \frac{K_t}{\left[(L_a J_m)s^2 + (R_a J_m + b_m L_a)s + (R_a b_m + K_t K_b)\right]} \quad (34)$$

PIDCONTROLLER DESIGN

The nonlinear equation that are derived from Euler-Lagrange Eq. (30), where inputvariable F which represents the torque applied to the robot is unknown, so it's requires a control in the Force applied of the joints to reach the final position.In our case, we use the classical linear PID, Particularly we need Two PID controls since the first arm motion is dependent from the second arm motion. In fact, still having a strong interaction between the two arms. The classical linear PID law is performed;

$$F = K_P\, e + K_D\, \dot{e} + K_I \int e\, dt \quad (35)$$

wheree $e = q^d - q$, q^d is desired joint angle, Kp, Ki and Kd are proportional, integral and derivative gains of the PID controller, respectively. This PID control law can be expressed via the following equations.

$$F = K_P\, e + K_D\, \dot{e} + \xi \quad (36)$$

$$\dot{\xi} = K_I\ e\ ;\ \xi(0) = \xi_0$$

where ξ represents the additional state variable that is the integral action of the PID control law formed and it's time derivative is $\dot{\xi} = K_I e$. The closed-loop equation is obtained by substituting the control action F from Eq.(32) into the robot model Eq.(30), gives:

$$B(\ddot{q}) + C(\dot{q}, q) + g(q) = K_P\, e + K_D\, \dot{e} + \xi \quad (37)$$

From the system'smodel Eq.(30), we can have

$$\ddot{q} = B(q)^{-1}[-C(\dot{q}, q) - g(q)] + F \qquad (38)$$

with

$$\hat{F} = B(q)^{-1}\, F \ \Leftrightarrow\ F = B(q)\hat{F} \qquad (39)$$

so, we decoupled the system to have the new (non-physical)input

$$\hat{F} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \qquad (40)$$

however, the physical torque inputs to the system are

$$\begin{bmatrix} f_{\theta 1} \\ f_{\theta 2} \end{bmatrix} = B(q) \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \qquad (41)$$

The error signals of the system are

$$\mathrm{e}(\theta_1) = \theta_{1f} - \theta_1$$

$$\mathrm{e}(\theta_2) = \theta_{2f} - \theta_2 \qquad (42)$$

where θ_f is the final positions. The final position is given by

$$\begin{bmatrix} \theta_{1f} \\ \theta_{2f} \end{bmatrix} = \begin{bmatrix} \pi/2 \\ -\pi/2 \end{bmatrix}$$

and the initial position is given by

$$\theta_o = \begin{bmatrix} -\pi/2 \\ \pi/2 \end{bmatrix}$$

so, in our case:

$$f_1 = K_{P1}\left(\theta_{1f} - \theta_1\right) + K_{D1}\,\dot{\theta}_1 \ + K_{I1}\ \int e(\theta_1)\, dt$$
$$f_2 = K_{P2}\left(\theta_{2f} - \theta_2\right) + K_{D2}\,\dot{\theta}_2 \ + K_{I2}\ \int e(\theta_2)\, dt$$

However, the complete system equations with control would be

$$\ddot{q} = B(q)^{-1}[-C(\dot{q}, q) - g(q)] + \hat{F} \qquad (43)$$

with

$$\hat{F} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} K_{P1}\left(\theta_{1f} - \theta_1\right) + K_{D1}\,\dot{\theta}_1 \ + K_{I1}\ \int e(\theta_1) dt \\ K_{P2}\left(\theta_{2f} - \theta_2\right) + K_{D2}\,\dot{\theta}_2 \ + K_{I2}\ \int e(\theta_2) dt \end{bmatrix} \quad (44)$$

recalling the physical actual torques

$$\begin{bmatrix} F_{\theta 1} \\ F_{\theta 2} \end{bmatrix} = B(q) \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \qquad (45)$$

RecallingEq.(32) and Eq.(38), gives:

$$\xi_1 = K_{I1}\ \int e(\theta_1) dt \ \ \Leftrightarrow \dot{\xi}_1 = K_{I1}\, e_1$$

$$\xi_2 = K_{I2}\ \int e(\theta_2) dt \ \ \Leftrightarrow \dot{\xi}_2 = K_{I2}\, e_2 \qquad (46)$$

So, the system equations are

$$\dot{\xi}_1 = K_{I1}\, e_1$$

$$\dot{\xi}_2 = K_{I2}\, e_2$$

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = B(q)^{-1}[-C(\dot{q}, q) - g(q)] + \begin{bmatrix} K_{P1}\left(\theta_{1f} - \theta_1\right) + K_{D1}\,\dot{\theta}_1 \ + \dot{\xi}_1 \\ K_{P2}\left(\theta_{2f} - \theta_2\right) + K_{D2}\,\dot{\theta}_2 \ + \dot{\xi}_2 \end{bmatrix} \qquad (47)$$

SIMULATION AND RESULTS

Approximate mathematical models can be obtained and then simulated in combination with the designed control law, for providing a more realistic validation of the system behavior and control performance. First we made a mathematical model of the PMDC motor to determine the transfer function and then start the simulation of the actuator system without controller using MATLAB /Simulink as shown in Figure (3)to study the system performance and then improving the design by adding PID Controller as shown in Figure(4)and(5) for a more realistic validation of the system performance.

The system reaches steady state value of 0.778 rad/s in 1.27 s, with small overshoot and the system response is very fast as shown in Figure(3), so we need to add controller to improve the system behavior. PID controller simulinkis built to control the angular speed at a desired set point value ($K_{a,t} = 1$)and improving the system performance as illustrate in Figure4.

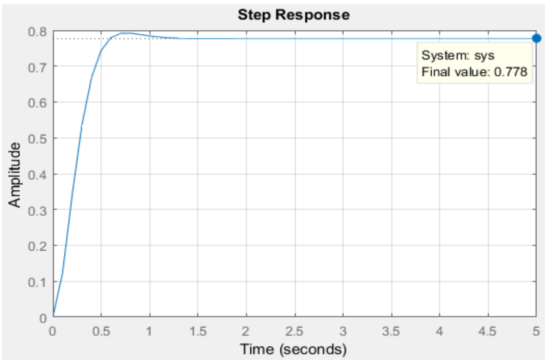

Figure3. *Step Response of PMDC without controller.*

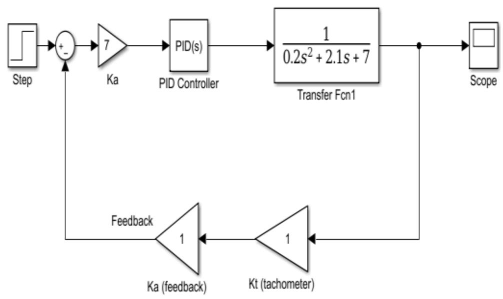

Figure 4. *Simulink model of the actuator system(PMDC).*

We have noticed from Figure(5) and (6) that the system is second order system (Under damped) and the values of the controller are getting with PID tuning by trial and error with system characteristicsvalues for the best performance and behavior:K_p=13, K_i=39, K_d=1.056, OS (%) < 5, settling time < 2 sec, steady-state error <1% and K_{DC} = 1.

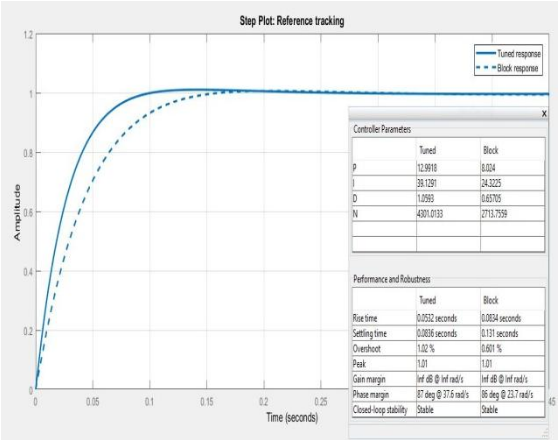

Figure 5. *System characteristics(With PID controller tuning).*

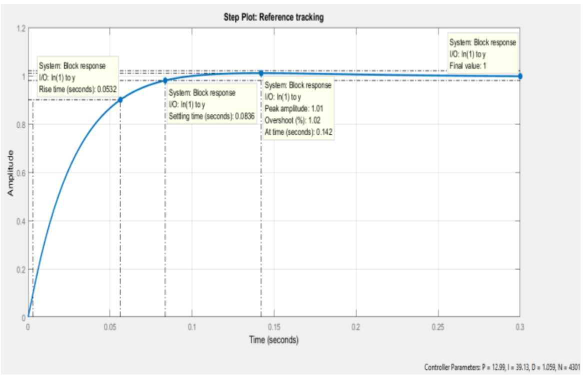

Figure6. *Step response of the system with PID controller.*

Then we have built and studied the steady state dynamic model with PID control of single link by MATLAB simulation as illustrate in Figure(7).

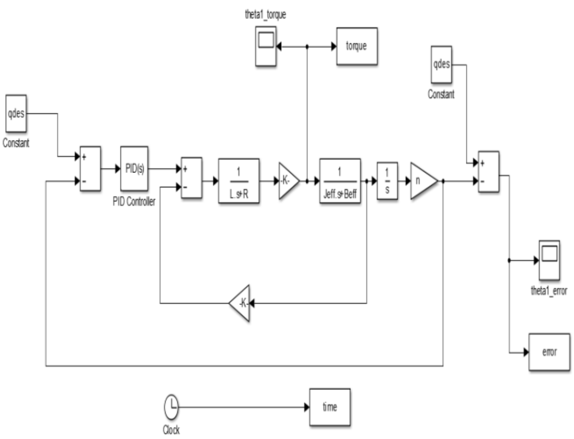

Figure7. *PID Controller of One Single Link.*

After tuning by trial and error we got with the PID controller values :kp=40, ki=13,kd=5for the best performance, Also we have noticed from Figure(8) that the position error reaches at a considerable time to zero steady-state of constant input.

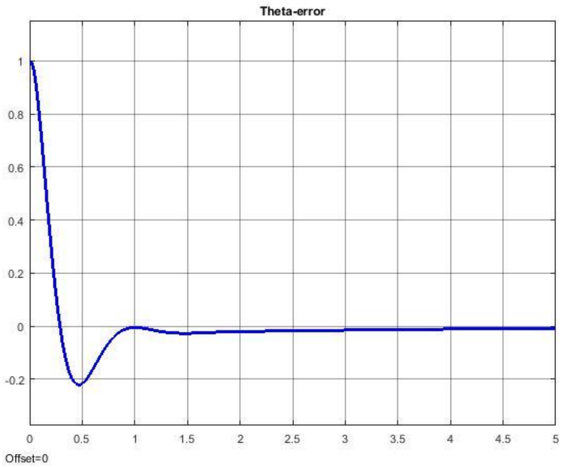

Figure8. *The position error.*

The force or torque applied can be shown in Figure(9) is zero at steady state.

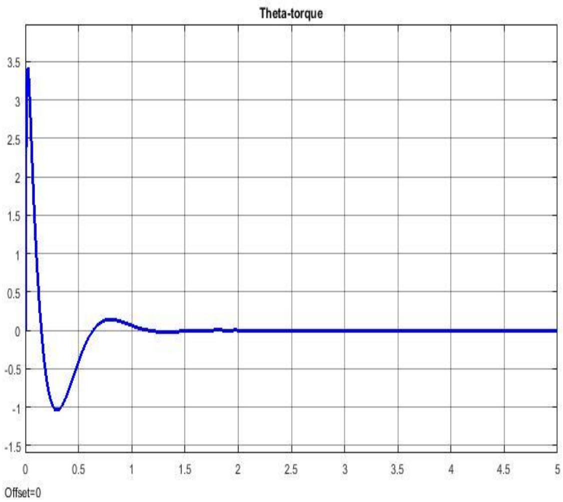

Figure 9. *The torque error.*

Then we have studied the two degree of freedom with PID control as a whole system by trial and error,

The PID parameters are manually tuned to get the best performance of the system. The best behavior and performance of the controller parameters values are as follows :

$K_{p1} = 250$ $\quad K_{p2} = 250$

$K_{i1} = 200$ $\quad K_{i2} = 200$

$K_{d1} = 30$ $\quad K_{d2} = 30$

By our approach trial and error, we have noticed that K_{p1} is related to direct error and to speed of evolution, K_{p1} is related to speed of interaction with change in states and K_{p1} is related to overall error cancellation.

We have seen in the Figures (10) and (11) that the position error reaches zero at a reasonably fast time.

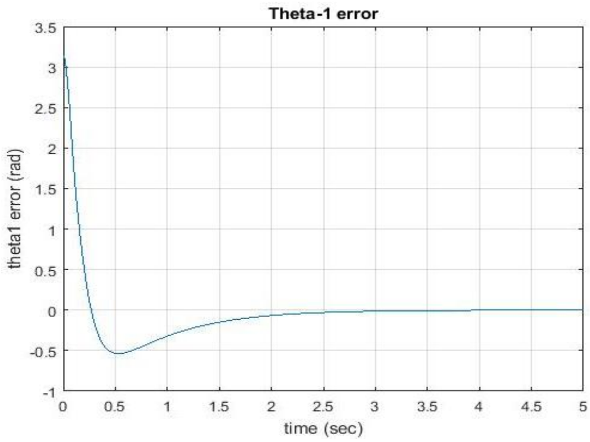

Figure10. *Theta 1 error.*

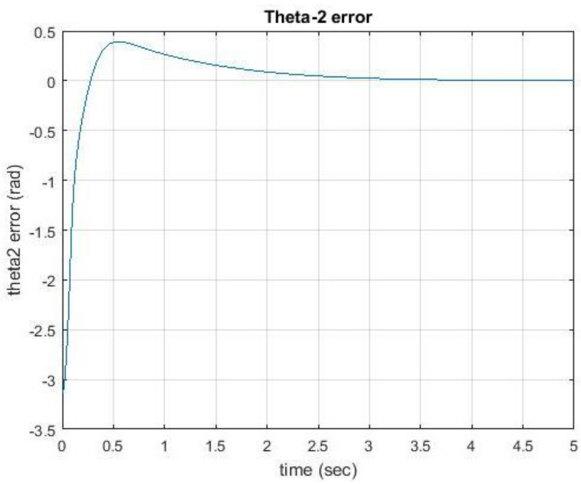

Figure11. *Theta 2 error.*

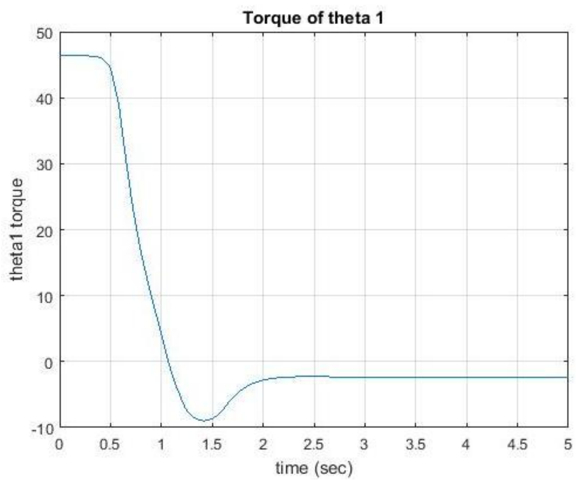

Figure12. *Torque of Theta 1.*

And from Figures (12) and (13), we have seen the force or torque applied slight overshot and quickly stabilize.

By the experiment, we got that the slightest change in the controller parameters yields more overshoots and oscillations due to highly sensitive to initial and final positions.

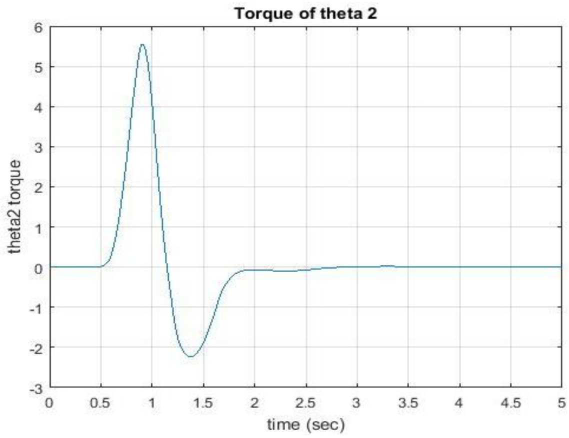

Figure13. *Torque of Theta 1.*

Conclusion

The main topic of this paper is modeling, simulation and control of two degree of freedom robot arm.

The mathematical models of the actuator and whole system are illustrated. Lagrangian and Euler-Lagrange used to derive a dynamic model that simulated the actual robot movement in real life and obtain sufficient control over the robot joint positions. The system was controlled in order to reach a desired joint angle position through simulation of PID controllers using MATLAB/Simulink.

Also, the result showed that a slight changes in initial joint angle positions of the robot arm resulted in different desired joint angle positions and this necessitated that the gains of the PID controllers need to be adjusted and turned at every instant in order to prevent overshoot and oscillation that associated with the change in parameters values.

References

[1] S. H. M. V. Mark W. Spong, Robot Modeling and Control, WILEY, December 2005.

[2] N.Jazar, R. (2010). Theory of Applied Robotics.

[3] Okubanjo, A. A., Modeling of 2-DOF Robot Arm and Control. (FUTOJNLS, Volume-3 , December 2017 .

[4] David I. Robles G., PID control dynamics of a robotic arm manipulator with two degrees of freedom, August 17, 2012.

[5] KhalilIbrahim, Ayman A. Aly and Ahmed Abo. Ismail ,"Position Control of A Single Flexible Arm", International Journal of Control,

Automation and Systems, (IJCAS), Vol.2, No.1, pp.12-18, 2013

[6] Aalim M. Mustafa & A AL-SAIF, Modeling, Simulation and Control of 2-R Robot. Global Journal of Researches in Engineering, Volume 14, 2014.

Citation: *Nasr M. Ghaleb and Ayman A. Aly, "Modeling and Control of 2-DOF Robot Arm", International Journal of Emerging Engineering Research and Technology, 6(10), pp.24-31*