The International Journal of Engineering and Information Technology

journal homepage:www.ijeit.misuratau.edu.ly

DESIGN And IMPLEMENTATION Of HYDRIC CONTROLLER For TWO-WHEELED ROBOT

Ahmed J. Abougarair
Electrical and Electronics Engineering
University of Tripoli, Libya
a.abougarair@uot.edu.ly

Ifaw F. Buzkhar
Electrical and Electronics Engineering
University of Tripoli, Libya

Abstract— **The two-wheeled robotic machine (TWRM) with five degrees of freedom (DOF) is a type of mobile robot that consists of two wheels and a body that can rotate and move in different directions. The proposed architecture of the TWRM in this paper features five DOFs, which allows for greater flexibility and range of motion. However, this also makes controlling the system more challenging, as the center of mass (COM) changes while performing tasks in multiple directions. To address this issue, the study utilizes a state-space model created by linearizing the non-linear modeling equations at the equilibrium point using the Lagrangian modeling. To stabilize the TWRM, several controllers were evaluated, including PID, LQR, and LQR with PID methods. The PID controller is a popular feedback control strategy that adjusts the control output based on the error between the desired and actual values of the system output. LQR is a state-space control technique that aims to minimize a cost function that expresses the system's performance criteria. The PID with LQR technique combines the strengths of both strategies, with PID used as a feed-forward to control the intermediate body (IB) and end-effector and LQR used as state feedback to regulate all states. The controllers were evaluated under various circumstances, including disturbance signals, tracking pathways, and moving actuators. Simulation results showed that the LQR with LQR controller performed better than the other two control systems in terms of least overshoot, rising time, and applied input forces. This suggests that the PID with LQR technique is a robust and effective control strategy for stabilizing the TWRM and improving its performance for various tasks.**

Index Terms: DOF, TWRM, PID, LQR, Stability.

I. INTRODUCTION

Mobile robots have become increasingly important in various industries, such as manufacturing, healthcare, and transportation, as they can perform tasks more efficiently and accurately than humans. Two-wheeled robots, in particular, have gained popularity due to their ability to maneuver in tight spaces and navigate over uneven terrain. Mobile robots are capable of moving autonomously without human intervention within a designated area to perform various tasks. Two-wheeled robots come in different designs with additional mechanical components for specific activities. Maintaining balance and wheel control are essential characteristics shared by all two-wheeled robots [1,2]. Researchers have proposed various control strategies for regulating integer order systems, including a unique controller architecture based on pole placement fractional PI-state feedback [3,4]. Stability problems for systems with multiple zero eigenvalues near the origin were addressed by introducing a multi-delayed-proportional controller [5]. A nonlinear control approach for moving a four-degree-of-freedom mobile inverted pendulum robotic system while stabilizing the pendulum was also proposed [6]. For a wheeled inverted pendulum model, an adaptive backstepping control was suggested [7]. Researchers have also proposed control approaches for stabilizing and monitoring the trajectory of a self-erecting single inverted pendulum based on robust linear quadratic regulator (LQR) and proportional velocity (PV) controllers [8]. To stabilize two-wheeled self-balancing vehicles, a nonlinear H controller was developed [9]. Additionally, a study focused on improving the performance of a double inverted pendulum system [10]. This paper presents the mechanical description of a two-wheeled robotic machine (TWRM) in Section 2, followed by control system design in Section 3. Simulation results of various methodologies are presented in Section 4, and the paper concludes in Section 5 with a summary of the main findings.

II. THE MECHANICAL DESCRIPTION OF TWRM

The degree of freedom (DOF) of a robot is defined by the number of ways it can move in space. In the case of a particular robot, its DOF is defined by four different types of translation along the X and Z axes. These four types of translation are represented by the rotational displacement of the right and left wheels and by the linear displacement of the robot's intermediate body (IB) and end-effector in

Received 7 Feb , 2023; revised 12 May, 2023; accepted 25 June, 2023.

Available online 28 June, 2023.

the vertical and horizontal directions, respectively as shown in Fig. 1. The fifth DOF is the angular tilt of the robot's IB. To assist in lifting, positioning, and transporting objects from one location to another, a two-wheeled robot with two linear actuators has been designed. This robot is capable of achieving a large workspace due to its DOF, which is defined by the four types of translation along the X and Z axes. The right and left wheels' angular displacement through rotational displacement, the intermediate body's (IB) linear displacement in the vertical direction and the end-linear effector's displacement in the horizontal direction, and the robot's IB's angular tilt serve as representations for these four types of translation. The fifth DOF is the robot's IB's angular tilt. The two-wheeled robot with two linear actuators can move objects from one location to another while lifting and positioning them as required, making it a useful tool in various industries.

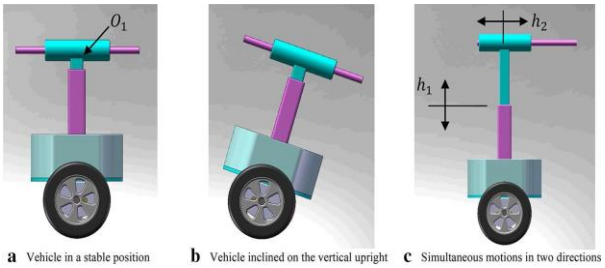


Figure 1. Mobility of the vehicle [1]

Friction plays an important role in the dynamics of a robotic system, especially in the case of a two-wheeled robot that is required to maintain balance in an upright position. The frictional forces that occur at the interface between the wheels and the ground, and between the wheels and the chassis, can affect the robot's motion and stability. To accurately model the dynamics of a two-wheeled robot, it is crucial to consider the effects of friction. The Coulomb friction model is often used to represent the frictional forces between two dry surfaces in contact. The model assumes that the frictional force is proportional to the normal force between the surfaces and is limited by the coefficient of static friction. The coefficient of static friction represents the maximum amount of force that can be applied to the surface without causing the surfaces to slide past each other. Once the force exceeds this limit, the surfaces begin to slide, and the frictional force transitions to the kinetic friction regime, which is typically lower than the static friction coefficient. In the case of a two-wheeled robot with linear actuators, the Coulomb friction model can also be used to represent the frictional forces within the actuators. This is important in the design of the control system, as the frictional forces can affect the accuracy and precision of the robot's motion during the lifting and placing tasks. Overall, the accurate modeling of frictional forces is critical in the design and control of a two-wheeled robot with linear actuators, as it affects the robot's stability, motion, and precision.

III. MATHEMATICAL MODEL OF TWRM

In order to analyze the mechanical system of the two-wheeled robot with linear actuators, various behaviors of its mathematical model are examined. This involves linking the forces or torques applied to the mechanical system's linkages to the kinematics of the system. The equations of motion for the system are extracted from the X, Y, and Z axes, and nonlinear mathematical models are derived from these equations. The equations of motion are similar to those obtained in previous papers [1-3]. The system is represented in a schematic diagram in Figure 2, and the physical parameters of the system are listed in Table 1. The equations of motion for the system are derived in this paper using the Lagrange formulation. The resulting nonlinear second-order differential equations capture the dynamics of the system under investigation and are listed in the motion equations. The mathematical modeling and analysis of the mechanical system of the two-wheeled robot with linear actuators are critical in designing and optimizing the control system for the robot. By understanding the system dynamics, it is possible to develop effective control strategies that ensure the stability, motion, and precision of the robot during its various tasks.

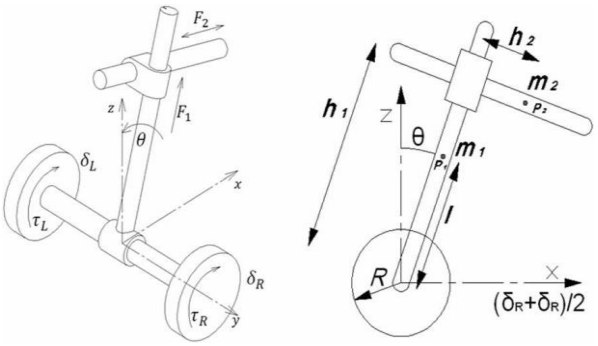


Figure 2. Schematic diagram of TWRM [2]

The equation for h_1

$$\frac{1}{2}m_2\big(2g\cos(\theta) - 2h_1\dot{\theta}^2 - 4\dot{h}_2\dot{\theta} - 2h_2\ddot{\theta} + 2\ddot{h}_1 + (\ddot{\delta}_r + \ddot{\delta}_l)\sin(\theta)\big) = F_1 - \mu_1\dot{h}_1 \quad (1)$$

The equation for h_2

$$\frac{1}{2}m_2\big(2g\sin(\theta) - 2h_2\dot{\theta}^2 - 4\dot{h}_1\dot{\theta} - 2h_1\ddot{\theta} - 2\ddot{h}_2 - (\ddot{\delta}_r + \ddot{\delta}_l)\cos(\theta)\big) = F_2 - \mu_2\dot{h}_2 = F_2 - \mu_2\dot{h}_2 \quad (2)$$

Deriving the equation for d_L

$$\frac{1}{2}m_1\left(\frac{1}{2}\ddot{\delta}_R + \frac{1}{2}\ddot{\delta}_L - l\dot{\theta}^2\sin(\theta) + l\ddot{\theta}\cos(\theta)\right) + \frac{1}{2}m_2\left(\ddot{h}_1\sin(\theta) + 2\dot{h}_1\dot{\theta}\cos(\theta) - h_1\dot{\theta}^2\sin(\theta) + h_1\ddot{\theta}\cos(\theta) + \ddot{h}_2\cos(\theta) - 2\dot{h}_2\dot{\theta}\sin(\theta) - h_2\dot{\theta}^2\cos(\theta) - h_2\ddot{\theta}\sin(\theta) + \frac{1}{2}\ddot{\delta}_L + \frac{1}{2}\ddot{\delta}_R\right) + 2m_w\ddot{\delta}_L + \frac{2J_w\ddot{\delta}_L}{R^2} = T_L - \mu_w\left(\frac{\dot{\delta}_L}{R^2}\right) - \mu_c\dot{\delta}_L \quad (3)$$

Deriving the equation for d_R

$$\frac{1}{2}m_1\left(\frac{1}{2}\ddot{\delta}_R + \frac{1}{2}\ddot{\delta}_L - l\dot{\theta}^2\sin(\theta) + l\ddot{\theta}\cos(\theta)\right) + \frac{1}{2}m_2\left(\ddot{h}_1\sin(\theta) + 2\dot{h}_1\dot{\theta}\cos(\theta) - h_1\dot{\theta}^2\sin(\theta) + h_1\ddot{\theta}\cos(\theta) + \ddot{h}_2\cos(\theta) - 2\dot{h}_2\dot{\theta}\sin(\theta) - h_2\dot{\theta}^2\cos(\theta) - h_2\ddot{\theta}\sin(\theta) + \frac{1}{2}\ddot{\delta}_L + \frac{1}{2}\ddot{\delta}_R\right) + 2m_w\ddot{\delta}_R + \frac{2J_w\ddot{\delta}_R}{R^2} = T_R - \mu_w\left(\frac{\dot{\delta}_R}{R^2}\right) - \mu_c\dot{\delta}_R \quad (4)$$

Deriving the equation for θ

$$2m_2\dot{\theta}\left(\dot{h}_2h_2 + \dot{h}_1h_1\right) + \frac{1}{2}m_2(h_1\cos(\theta) - h_2\sin(\theta))\left(\ddot{\delta}_L + \ddot{\delta}_R\right) + \frac{1}{2}m_1l\,cos(\theta)\left(\ddot{\delta}_L + \ddot{\delta}_R\right) - m_2g(h_1\sin(\theta) + h_2\cos(\theta)) + \ddot{\theta}(J_1 + J_2 + m_1l^2 + m_2h_2^2 + m_2h_1^2) + m_2\left(h_2\ddot{h}_1 + h_1\ddot{h}_2\right) - m_1g\,l\sin(\theta) = 0 \quad (5)$$

In order to linearize the system, the equilibrium point in the vertical upright position is taken into account. This is achieved by considering the tilt angle when it is close to zero, typically within the range of 0.3 radians and -0.3 radians. To describe the dynamics of the two-wheeled robot with linear actuators, ten state vectors, X, are used. These state vectors capture the system dynamics and are represented in the following vector [3,11]:

$$X = \left[\delta_R\ \delta_L\ \theta\ h_1\ h_2\ \dot{\delta}_R\ \dot{\delta}_L\ \dot{\theta}\ \dot{h}_1\ \dot{h}_2\right]$$

Where the state vector variables can be identified as follows:

δ_R:Right wheel displacement, δ_l:Left wheel displacement

θ:Chassis pitch angle , h_1:Vertical linear link displacement

h_2:Horizontal linear link displacement

$\dot{\delta}_R$:Right wheel velocity , $\dot{\delta}_L$: Left wheel velocity

$\dot{\theta}$: Chassis angular velocity , $\dot{h}_1$:Vertical linear link velocity

$\dot{h}_2$:Horizontal linear link velocity

State variables of wheels velocity, angular velocity and linear velocities of the links are derivative of wheels displacements, links linear displacements and the pitch angle, respectively, and can be formulated as follows:

$X_1 = \delta_R \quad X_2 = \delta_l \quad X_3 = \theta \quad X_4 = h_1 \quad X_5 = h_2$

$X_6 = \delta_R \quad X_7 = \delta_L \quad X_8 = \dot{\theta} \quad X_9 = \dot{h}_1 \quad X_{10} = \dot{h}_2$

Table 1. TWRM parameters [1,2]

Parameter	Description	Value	Unit
m_1	Mass of the chassis	3.1	kg
m_2	Mass of the linear actuators	0.6	kg
m_W	Mass of wheel	0.14	kg
g	Gravitational acceleration	9.81	m/s^2
l	Distance of chassis′ center of mass for wheel axle	0.14	m
R	Wheel radius	0.05	m
J_1	Chassis moment of inertia	0.068	kg.m^2
J_2	Moving mass moment of inertia	0.0093	kg.m^2
J_W	Wheel moment of inertia	0.00017	kg.m^2
μ_1	Coefficient of friction of vertical linear actuator.	0.3	Ns/m
μ_2	Coefficient of friction of horizontal linear actuator	0.3	Ns/m
μ_C	Coefficient of friction between wheel and ground	0.1	Ns/m
μ_W	Coefficient of friction between chassis and wheel	0	Ns/m
δ_L, δ_R	Angular displacement of right and left wheels	-	m
θ	Tilt angle of the intermediate body around the vertical Z axis	-	rad
h_1	Vertical linear link displacement	-	m
h_2	Horizontal linear link displacement	-	m
F_1	Force generated by the vertical linear actuator	-	N
F_2	Force generated by the horizontal linear actuator	-	N
T_R, T_L	Right and left wheels torque	-	N/m

$$\begin{bmatrix}\dot{X}_1\\ \dot{X}_2\\ \dot{X}_3\\ \dot{X}_4\\ \dot{X}_5\\ \dot{X}_6\\ \dot{X}_7\\ \dot{X}_8\\ \dot{X}_9\\ \dot{X}_{10}\end{bmatrix} = \begin{bmatrix}0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&A_{63}&0&A_{65}&A_{66}&A_{67}&0&0&A_{610}\\ 0&0&A_{73}&0&A_{75}&A_{76}&A_{77}&0&0&A_{710}\\ 0&0&A_{83}&0&A_{85}&A_{86}&A_{87}&0&0&A_{810}\\ 0&0&0&A_{94}&0&0&0&0&0&0\\ 0&0&A_{103}&0&A_{105}&A_{106}&A_{107}&0&0&A_{1010}\end{bmatrix}\begin{bmatrix}\delta_R\\ \delta_L\\ \theta\\ h_1\\ h_2\\ \dot{\delta}_R\\ \dot{\delta}_L\\ \dot{\theta}\\ \dot{h}_1\\ \dot{h}_2\end{bmatrix} + \begin{bmatrix}0&0&0&0\\ 0&0&0&0\\ 0&0&0&0\\ 0&0&0&0\\ 0&0&0&0\\ B_{61}&B_{62}&0&B_{64}\\ B_{71}&B_{72}&0&B_{74}\\ B_{81}&B_{82}&0&B_{84}\\ 0&0&B_{93}&0\\ B_{101}&B_{102}&0&B_{104}\end{bmatrix}\begin{bmatrix}T_R\\ T_L\\ F_1\\ F_2\end{bmatrix} \quad (6)$$

These state vectors enable the representation of the five degrees of freedom of the robot's dynamics. By using these state vectors, it is possible to achieve a better understanding of the robot's behavior and develop effective control strategies to ensure its stability and precision during its various tasks. The state space model is represented by [1,3] . The coefficients of the state space model for the two-wheeled robot with linear actuators are presented in the appendix of Ref. [1]. The state space model captures the dynamics of the system and enables the development of a control system that ensures stability and precision during the robot's various tasks. The necessary torques for the right and left wheels are denoted as T_R and T_L , respectively. The linear actuator's produced forces F_1 and F_2 are used to move the payload in the vertical and horizontal directions, respectively. Figure 3 illustrates the instability of the system, as seen in

Figure 3. Open loop step response

Fig. 4, where certain poles are located on the right side of the complex plane. It has been recognized that the system is unstable, meaning that small perturbations can cause the system to diverge and become uncontrollable. o address the instability of the system, various control strategies can be employed. These include feedback control, feedforward control, and model predictive control, among others. The choice of control strategy depends on the specific requirements of the robot's task and the characteristics of the system dynamics.

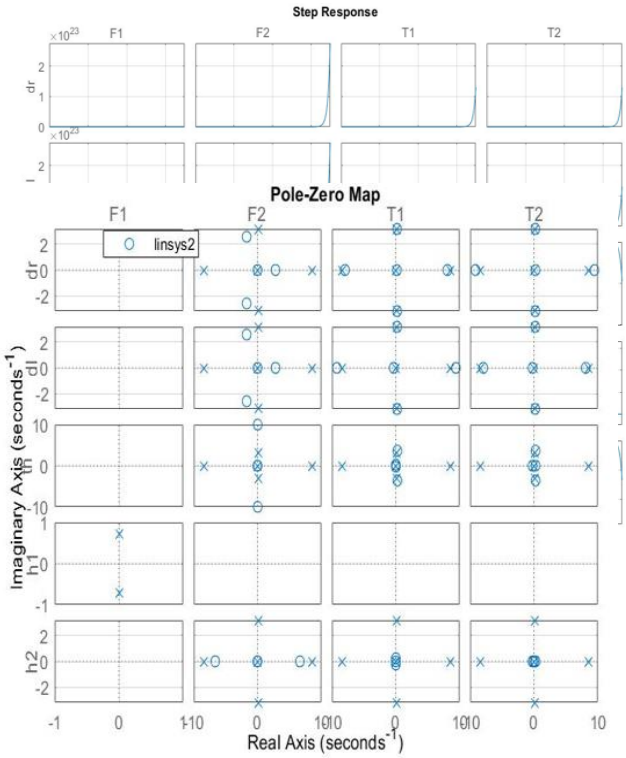


Figure. 4. Pole-zero map for open-loop system

IV. CONTROL SYSTEM DESIGN

In this section, we will demonstrate how to import the state-space model of the two-wheeled robot with linear actuators into Simulink. Next, we will add several controllers [12-15] to the Simulink model in an attempt to control the robot's tilt angle, position, and the length of its intermediate body (IB) and end-effector. Additionally, we will test the 5 DOF two-wheeled robot with linear actuators by introducing various input signals and disturbances. The purpose of these tests is to evaluate the robot's robustness and determine whether the control strategies are effective under different operating conditions. Simulink is a powerful tool that enables us to simulate the dynamics of the system and assess the performance of different control strategies. By using Simulink, we can develop an effective control system that ensures the stability, motion, and precision of the robot during its various tasks.

A. PID Controller

PID (proportional integral derivative) controllers are among the most stable and accurate controllers used to regulate process variables. These controllers utilize a feedback loop mechanism to continuously adjust the control input based on the difference between the desired output and the actual output of the system [16].

The output of a PID controller at any given time t is determined by the following equation:

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de}{dt} \qquad (7)$$

where e(t) represents the tracking error, Kp is the proportional gain, Ki is the integral gain, and Kd is the derivative gain. In this section, five PID controllers are connected in a feedback system as shown in Fig. 5. MATLAB tuning is then used to adjust the PID gains until a good response is achieved. Table I shows the PID gains for each output, which are adjusted based on the performance of the system during testing. PID controllers are an effective control strategy for the two-wheeled robot with linear actuators. By using PID controllers along with MATLAB tuning, it is possible to achieve stable and accurate control of the robot's position, tilt angle, and the length of its intermediate body and end-effector.

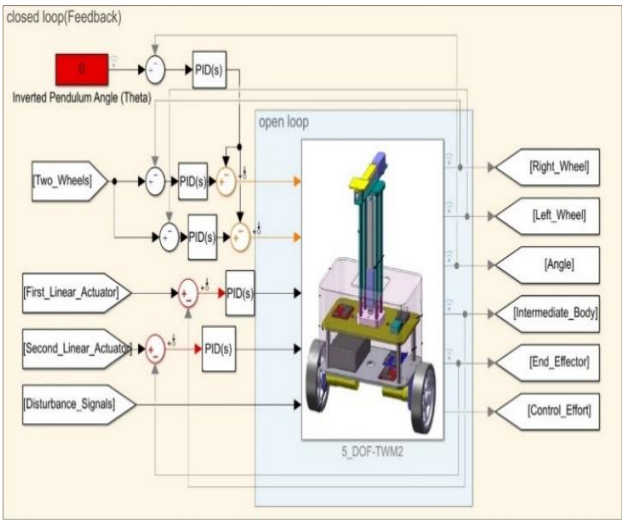


Figure 5. Feedback system with 5 PID controllers

Table 2. PID gains

	P	I	D
PID (dr)	22620	437100	270
PID (dl)	-2730	-510	12
PID (th)	7500	28070	260
PID (h1)	41	12.4	31
PID (h2)	-280230	-23127120	-833

B. Full-State Feedback System

When all state variables are continuously measured and fed back to the controller, it is known as a full-state feedback system. The state-space equations for this type of closed-loop feedback system are represented by equation (8):

$$\dot{x} = Ax + B(-Kx) = (A - BK)x \qquad (8)$$
$$y = Cx$$

where x is the state vector, A is the system matrix, B is the input matrix, K is the gain matrix, and y is the output vector represented by the matrix C x. The stability and performance of the closed-loop feedback system are determined by the location of the eigenvalues of the matrix (A-BK), which are equal to the closed-loop poles. In modern control design approaches for linear multiple-input, multiple-output (MIMO) systems, many techniques have evolved from LQR (Linear Quadratic Regulator), which is a potent design methodology [17,18]. Figure 6 shows a system with all-state output connected with gains matrix. The MATLAB lqr command is used to extract the gain matrix (K) value that will help to move the system poles and achieve desirable results. Although LQR is a powerful approach to control design, setting the values for Q and R (weight matrices for states and inputs) is still a tradeoff solution. Attempts will be made to modify the Q and R matrices to achieve the best response with minimal energy.

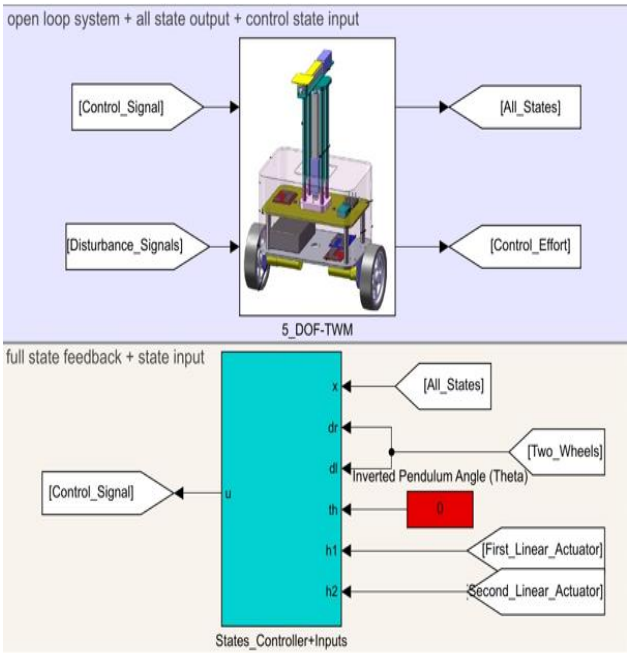


Figure 6. System with full state feedback [19]

C. PID plus LQR method

LQR with PID (Linear Quadratic Regulator with Proportional Integral Derivative) is a control strategy that combines two different control algorithms to achieve better performance in controlling a system. LQR is a state-space control technique that aims to minimize a cost function that expresses the system's performance criteria, while PID is a feedback control technique that adjusts the control output based on the error between the desired and actual values of the system's output. LQR is a robust and optimal control strategy that can handle system uncertainties and disturbances, but it can be computationally expensive and may not be suitable for real-time control applications. On the other hand, PID is a simple and widely-used control strategy that can provide fast and stable control of a system, but it may not be able to handle complex dynamics and disturbances. By combining LQR and PID, the resulting control strategy can take advantage of the strengths of both techniques and overcome their limitations. The LQR component of the control strategy provides an optimal control law that minimizes the cost function and takes into account the system dynamics and uncertainties. The PID component of the control strategy provides a fast and stable response to changes in the system output and can handle disturbances and noise. The LQR with PID control strategy is typically implemented by cascading the two controllers, with the LQR controller as the outer loop and the PID controller as the inner loop. The output of the LQR controller is used to set the desired setpoint for the PID controller, which adjusts the control output based on the error between the desired and actual values of the system output. The objective of the control strategy is to try to get the best results to track a path through the LQR method and the best results to change the lengths of intermediate body (h1) and end-effector (h2) through the PID controller. Figure. 7, shows the PID plus gain matrix connection. Table 2 shows the PID gains , matrix states and input matrix.

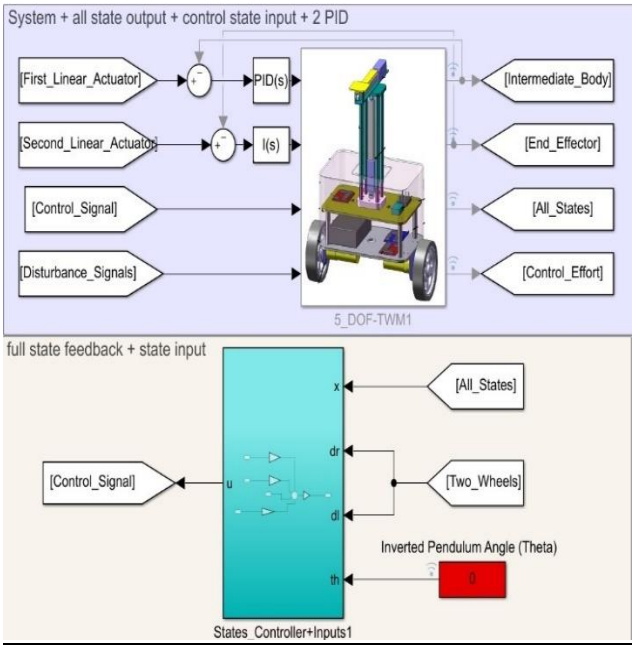


Figure. 7. Control ssystem using PID with LQR strategy [19]

Table 3. PID gains for intermediate body (h1), end-effector (h2) and (Q,R) matrixes

	P	**I**	**D**	
PID (h1)	1890	2700	110	
PID (h2)	0	230	0	
Q	diag[1000 1000 1000 10 1000 1 1 1 0.2 1]			
R	[0.1 0.1 0.1 0.01]			

V. SIMULATION RESULTS

To further evaluate the performance of the controllers in the two-wheeled robot with linear actuators, a different set of input signals will be used. Table 3 shows the shape and description of each signal used in the scenarios. The results of each scenario will be represented by a number in all controllers. To compare the performance of the controllers, the control effort E will be calculated. The control effort refers to the amount of energy or power required by the controller to perform its intended task. By calculating the control effort, it is possible to determine which controller is most efficient in achieving the desired performance [19, 20].

$$E = \int U^2(\mathrm{t})\ dt \qquad (9)$$

Table 4. Description of the signals Scenarios

Scenario number	**Description of the signal**
I	A disturbance signal, an impulse signal, is a push signal for the system.
II	The disturbance signal varies by increasing the number of impulses at different time by different amplitude.
III	Input signal increases IB length (h1), length increases from 0m to 0.1m.
IV	The length of the IB (h1) changed from 0.1 m to 0.3 m in 5 sec.
V	Change the end-effector (h2) length from 0 m to 0.005 m.
VI	Input signal increases end-effector (h2), length increases from 0.1m to 0.15m in 5 sec.
VII	A change in the robot position by adding ramp signal
IX	A change in the robot position by sin wave signal

A. Simulation results of PID

Scenario I: Figuge. 8 shows that all robot outputs are affected by disturbers signal (push) but it can be observed that all output take a short time until reaches set-point, except the left wheel (d_r) is tack a long time.

Scenario II: Figure. 9, even an increase in disturbance signals and putting it at different times does not significantly affect a robot output except the left wheel (d_r) which still takes long time to reaches set-point

Scenario III: A change in the length of the IB (h_1) is observed from Figure. 10 this change does not affect the other output, also observed that h1 has a higher response, as it moved from 0m to 0.1m and settled in a 0.1m Within a short time.

Scenario IV: In Figure. 11, the system still conserves the output results even if the h1 value changes from 0.1 to 0.3

Scenario V: Figure. 12 it can be seen that the robot has a number of changes, not considered large in the output angle (th) and right wheel (d_r), but considered large on the left wheel (dl) where the value is far from required and become unstable, which made the robot orbiting itself (and the center of rotation Is the right wheel). It is also observed that the robot's IB (h1) was not affected due to a change in the length of the end-effecter (h_2), and is also noted that the end-effecter is stable, where it moves to 0.05m and settles in a short time.

Scenario VI: From Figure. 13 it has been observed that even with the change of the end-effect length, the end-effect output remains stable as it moves from 0.1 to 0.2 within a short time, also the left wheel is affected by the increase causing the system is unstable.

Scenario VII: In this scenario it can be noted from Figure. 14, that the robot follows the path well in the right wheel (d_r), but the left wheel (d_l) starts with a high value and then take a long time to reach the desired value. Also, it has been noted that h_1 and h_2 were not affected by a change, and the angle took time to settle at set-point.

Scenario VIII: In Figure. 15, it can be observed that the same problem in the left wheel is still, and it is noted that the results of the right wheel and E and B is still good, and the reason for the instability of the angle is due to the constant movement of the robot.

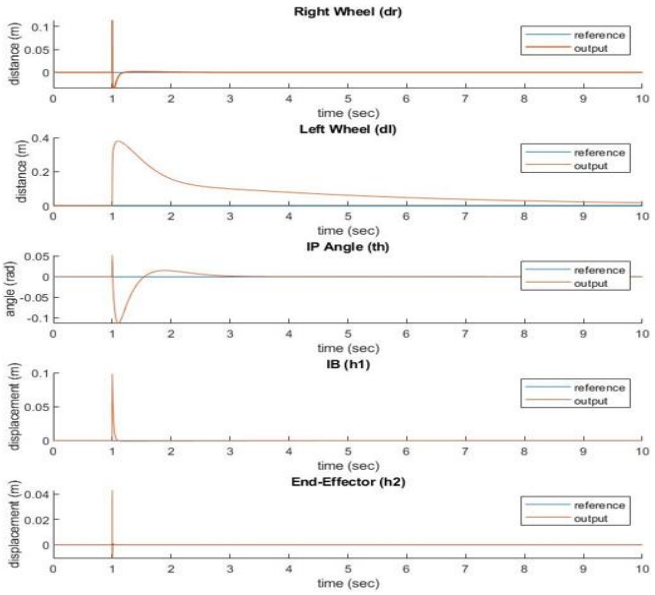


Figure 8. Robot Response Using the PID Controller (Scenario I)

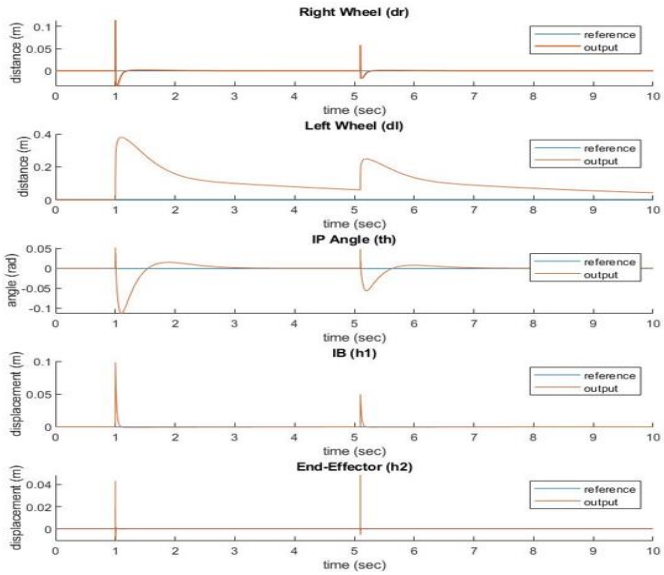


Figure 9. Robot Response Using the PID Controller (Scenario II)

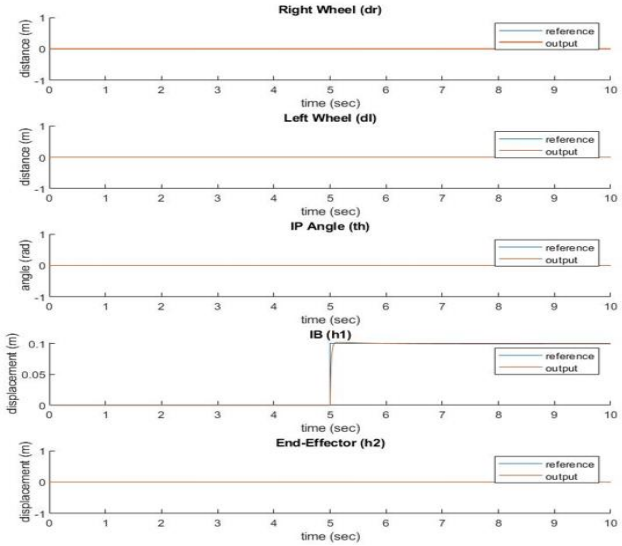


Figure. 10. Robot Response Using the PID Controller (Scenario III)

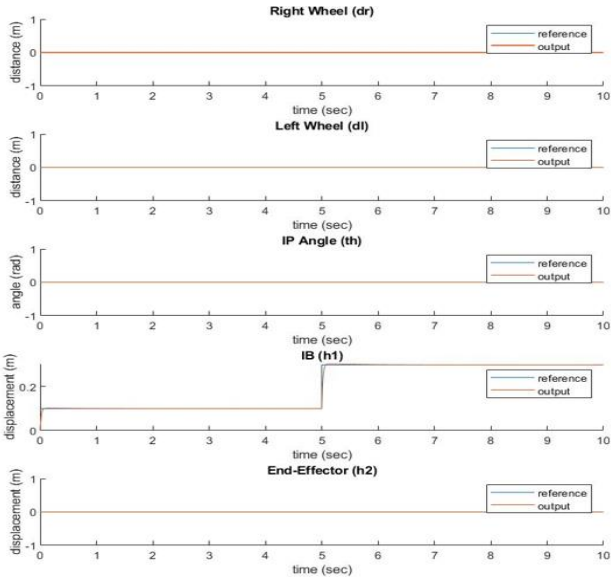


Figure. 11. Robot Response Using the PID Controller (Scenario IV)

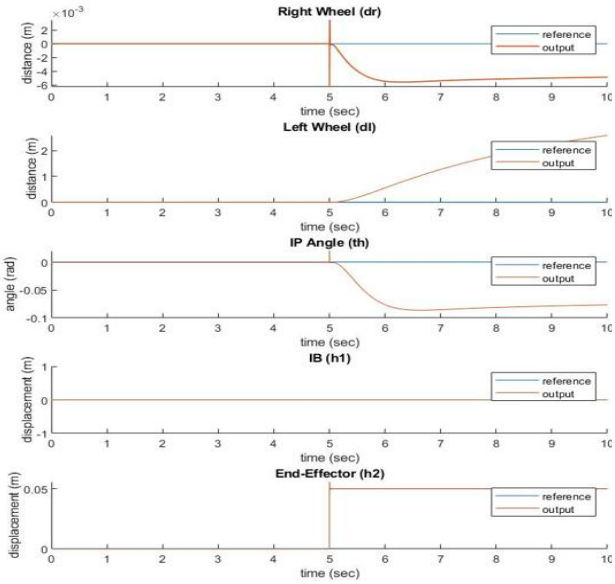


Figure 12. Robot Response Using the PID Controller (Scenario V)

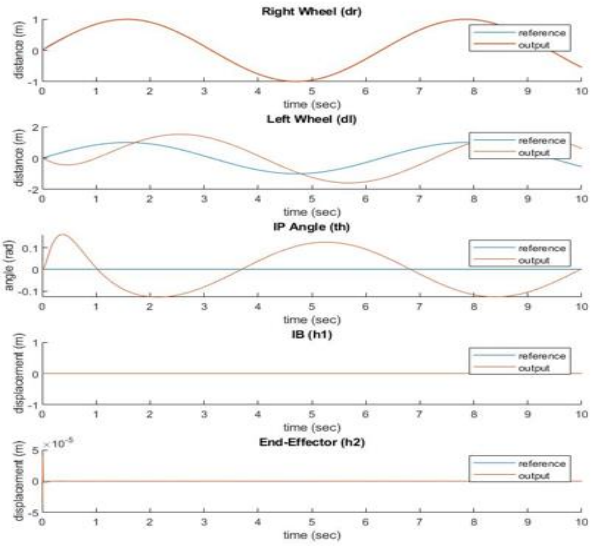


Figure 13. Robot Response Using the PID Controller (Scenario VI)

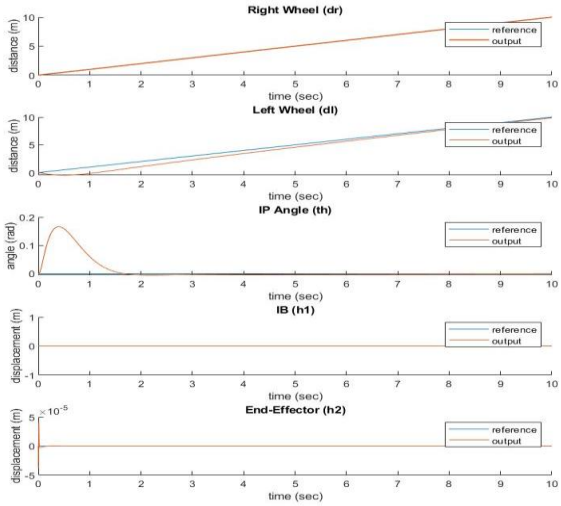


Figure 14. Robot Response Using the PID Controller (Scenario VII)

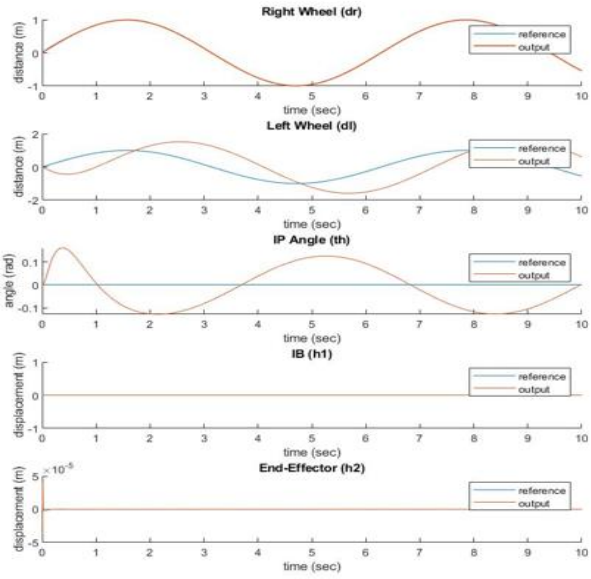


Figure 15. Robot Response Using the PID Controller (Scenario VIII)

B. Simulation results of LQR

Scenario I: Figure. 16 shows that all robot outputs are affected by disturbers signal (push) but it can be observed that all output take approximately 1 sec until reaches set-point.

Scenario II: The results from Figure. 17, are close to those of the first scenario (I) however, the disturbance signal was introduced at different time.

Scenario III: From Figure. 18, it has been noted that the system was not significantly affected due to a change in the length of h1, and also the change of length of h1 was in a short time.

Scenario IV: The same results obtained in the third scenario (III) are seen in Figure. 19.

Scenario V: From Figure. 20, it can be noted that the system has had several problems due to the introduction of a signal to change the length of end-effector (h2), and this signal affected all outputs, but did not change the length of h2.

Scenario VI: In Figure. 21, the same problem is observed in the fifth scenario (V).

Scenario VII: From Figure. 22, it has been noted that the robot is good in terms of tracing the path with a slight delay between the input and output signals, and it can be identified that there is harmony between the two wheels right (d_r) and left (d_l). The tilt of the robot angle (th) is due to the continuous movement of the robot. The change in the length of h_2 is considered problem and it can be noted the IB (h1) output has small error.

Scenario VIII: From Figure. 23, it has been seen that the angle instability is due to the backward and forward movement of the robot, and it can note h_1 and h_2 output were affected.

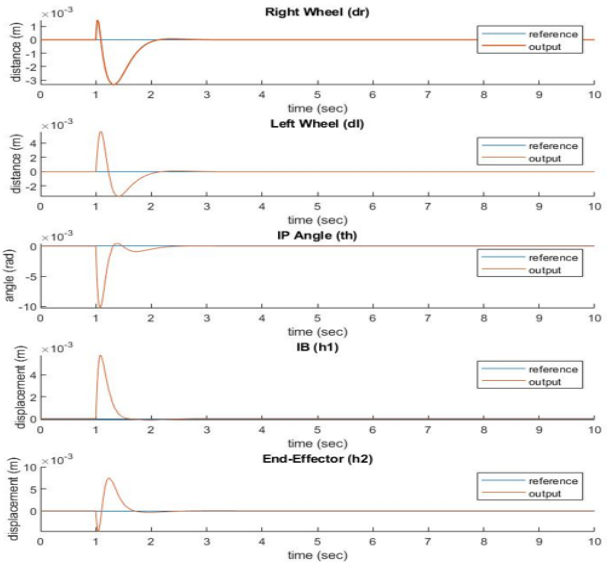


Figure 16. Robot Response Using the LQR Controller (Scenario I)

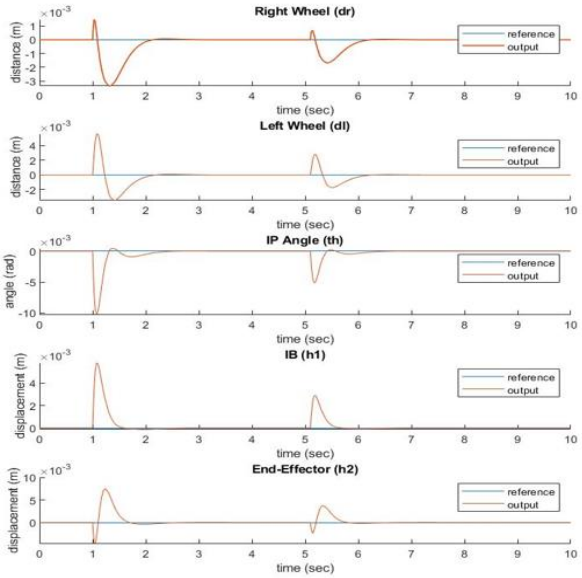


Figure 17. Robot Response Using the LQR Controller (Scenario II)

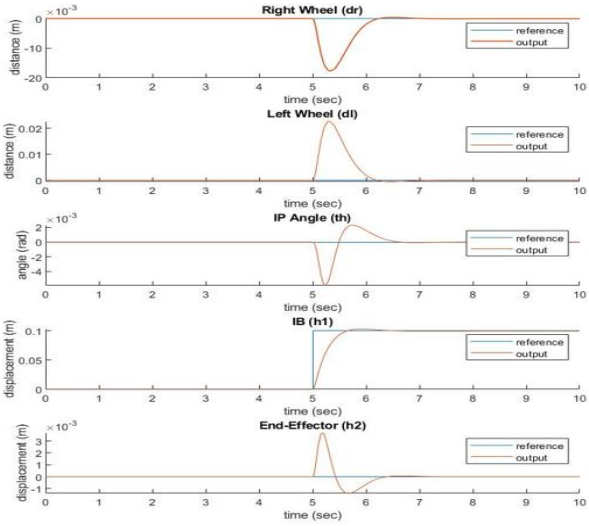


Figure 18. Robot Response Using the LQR Controller (Scenario III)

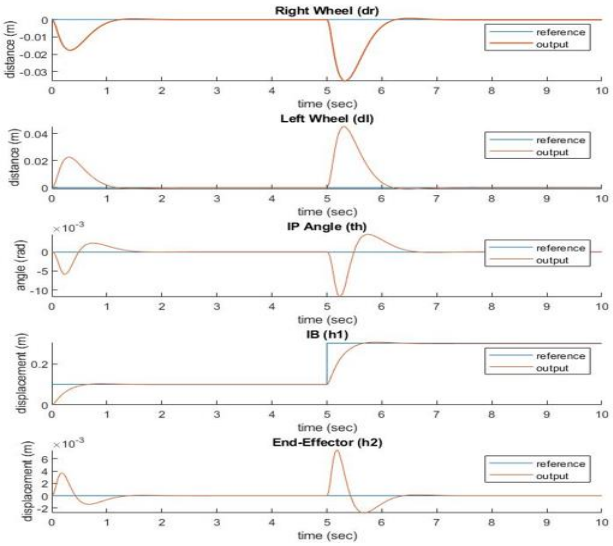


Figure 19. Robot Response Using the LQR Controller (Scenario IV)

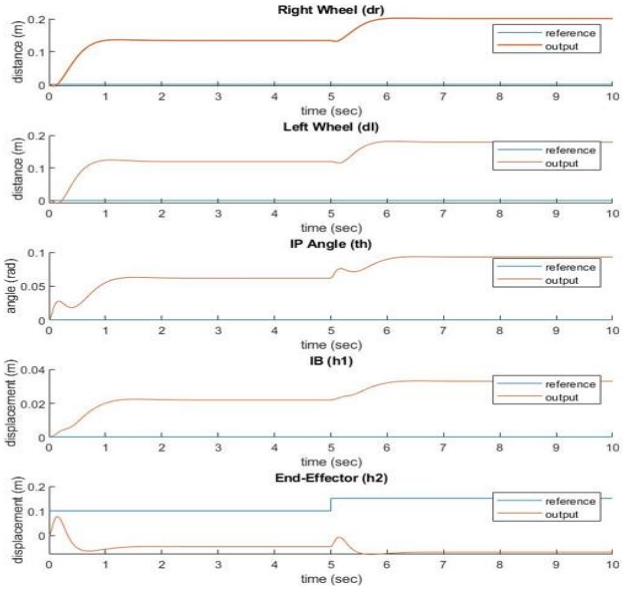


Figure 20. Robot Response Using the LQR Controller (Scenario V)

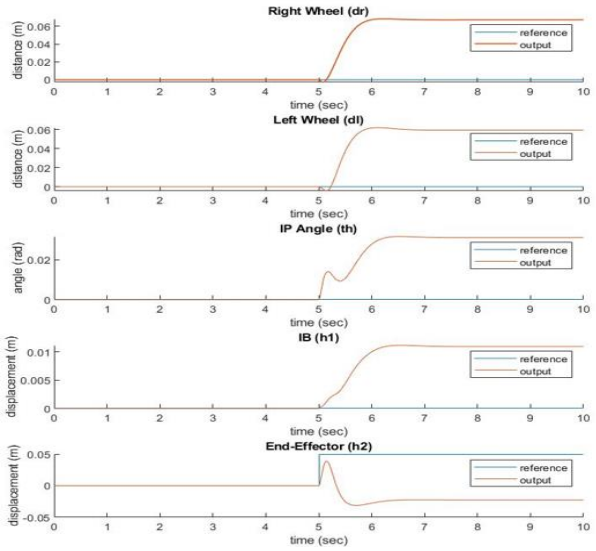


Figure 21. Robot Response Using the LQR Controller (Scenario VI)

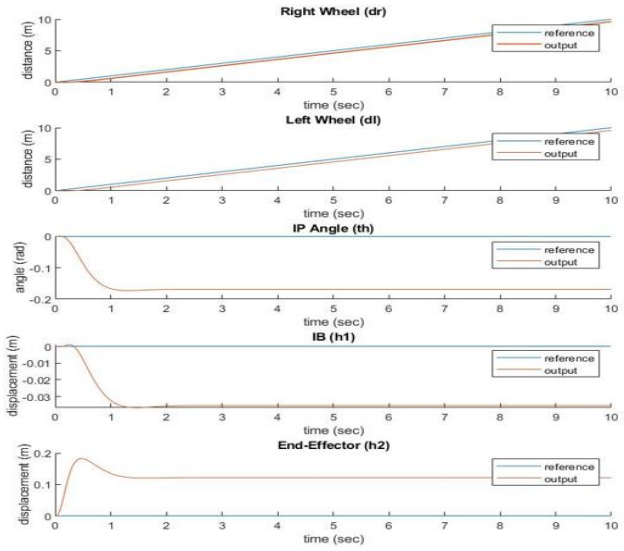


Figure 22. Robot Response Using the LQR Controller (Scenario VII)

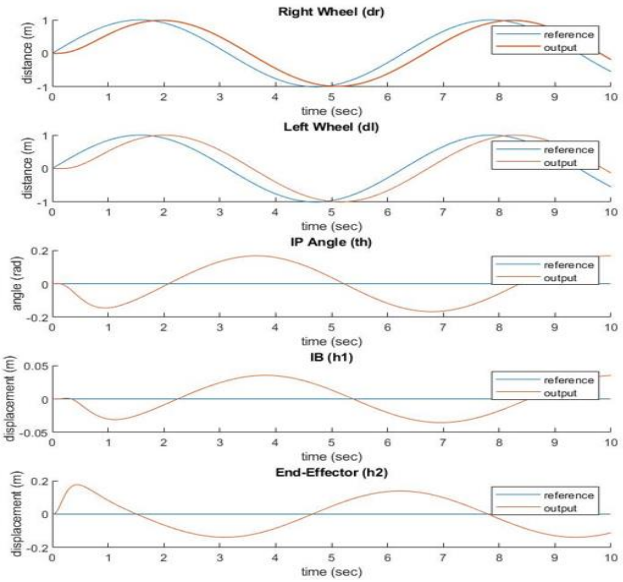


Figure 23. Robot Response Using the LQR Controller (Scenario VIII)

C. Simulation results of LQR with PID

Scenario I: In Figure. 24, it can be observed that the disturbance signal affected the robot, where the output IB (h_1) takes approximately 1 sec to reach set-point, and the rest of the outputs required approximately 3 sec to reach set-point.

Scenario II: It can be noted from Figure. 25, that the robot's results are the almost the same as in the first scenario but disturbance signal was introduced at different time.

Scenario III: In the figure. 26, it has been shown that a change in the length h_1 affected d_r and dl output. The change in right wheel (d_r) and left wheel (d_l) output was in opposite direction which caused the robot angle return to set-point in approximately 1 sec. The end-effector (h_2) output of the robot has minimally been affected by change in h1 length and return to set-point in approximately 1 sec, and also it can be noted that h1 output has a fast response.

Scenario IV: It can be seen from Figure. 27, that the two wheels are affected by a change in the length of the h1. It can be seen that the change in the two wheels and the tilt of the angle (th) is still close to the results of the previous scenario but in different time. The output of h2 in the first seconds the same as the third scenario, after 5 seconds the response was similar to the third scenario but with more length. It can be seen that output of h1 has a fast response.

Scenario V: Figure. 28, shows that the robot was affected by the change of the length of end-effector (h2). This change of the length caused the two wheels to move at a distance of up to -0.1m. This movement of the wheels had small impact on the robot angle (th) with a value approximately -0.05 rad. The output of IB (h1) is close to the set-point (zero). The h2 response is acceptable because within 3 sec it reaches the set-point.

Scenario VI: From Figure. 29, it can be seen that the right and the left wheels show an increase in the distance towards the negative side from the set-point. This movement of the wheels caused the angle to tilt towards the negative side. The output of h1 is very close to the set-

point. The h2 output is still as good as the response in the fifth scenario.

Scenario VII: From Figure. 30, it has been noted that the robot is acceptable in terms of tracing the path. It can also be identified that there is harmony between the two wheels right (dr) and left (dl). There is a very slight tilt in the angle (th) in the first few seconds and the angle become approaching zero. It is observed that there is a change in the length of h2 in first few second and then the length become settling close to zero. The IB (h1) length showed insignificant change in the length (approaching zero).

Scenario VIII: Figure 31. It can be observed that the robot still has acceptable result in terms of tracing the path as in the eighth scenario. The tilt of the robot angle is logic, it is due to continuous movement of the robot. The IB (h1) length showed insignificant change in the length (approaching zero). The h2 output showed changes in its length due to robot movement. The tabulated results in Table 3 displayed summarizes a comparison between all the control strategies by the values of rise time, settling time, and peak time. In general, the PID with LQR produces much better system performance. Also, PID with LQR give the best control effort as presented in Figure. 32.

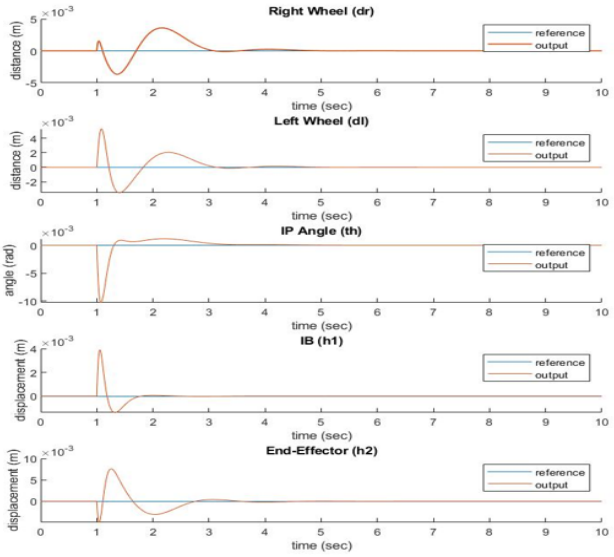


Figure 24. Robot Response Using the LQR with PID (Scenario I)

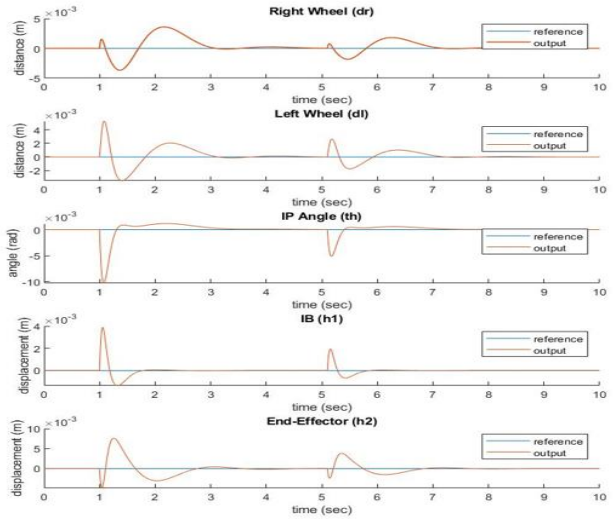


Figure 25. Robot Response Using the LQR with PID (Scenario II)

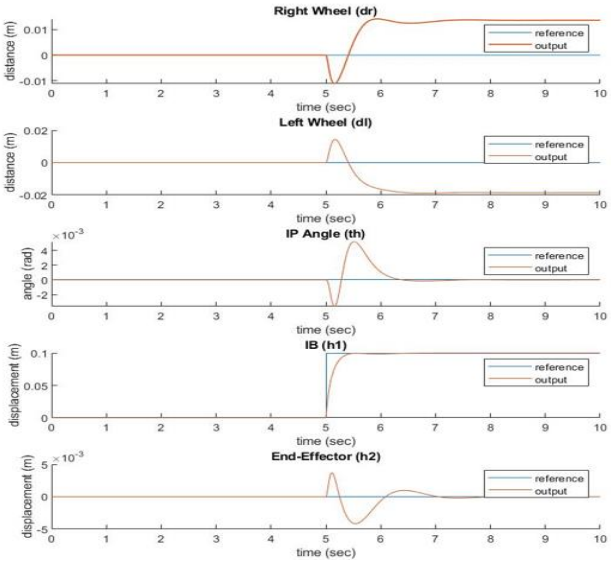


Figure 26. Robot Response Using the LQR with PID (Scenario III)

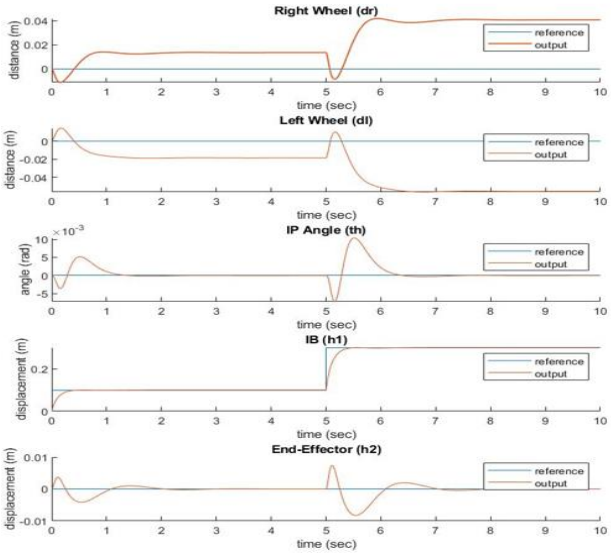


Figure 27. Robot Response Using the LQR with PID (Scenario IV)

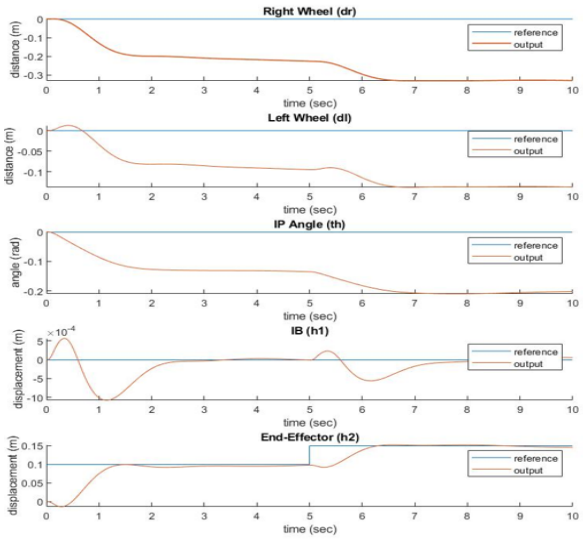


Figure 28.Robot Response Using the LQR with PID (Scenario V)

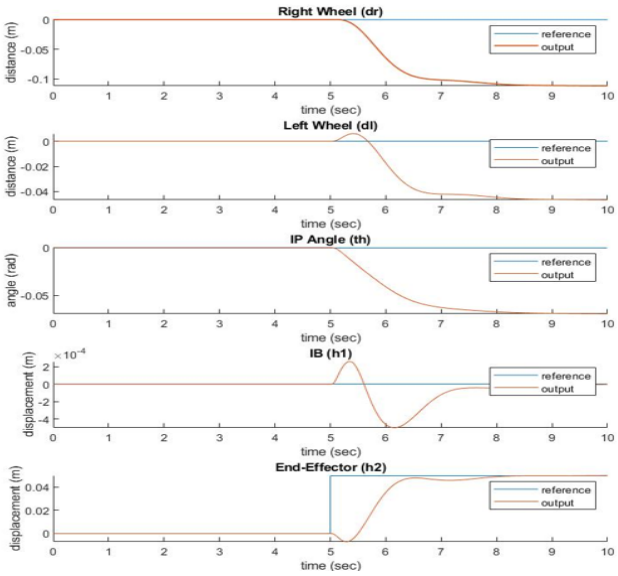


Figure 29. Robot Response Using the LQR with PID (Scenario VI)

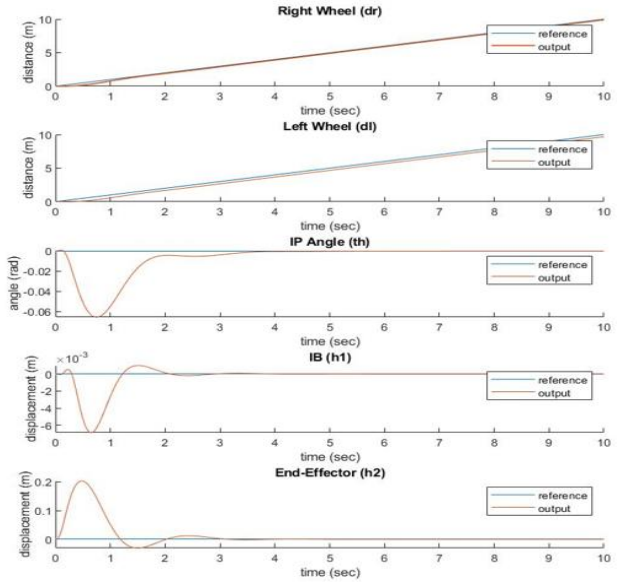


Figure 30. Robot Response Using the LQR with PID (Scenario VII)

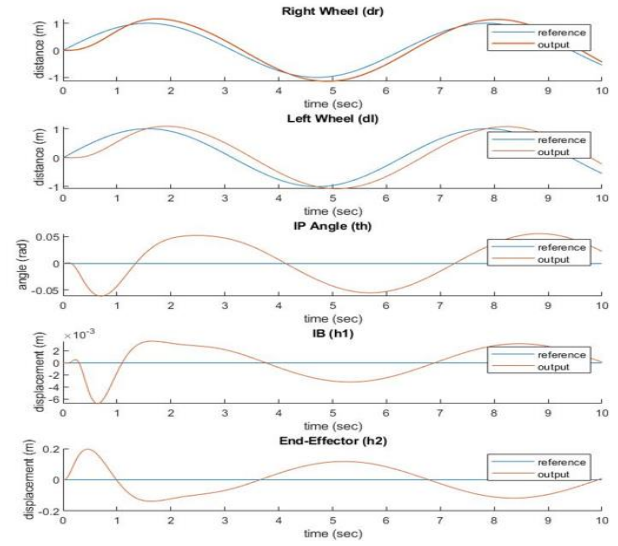
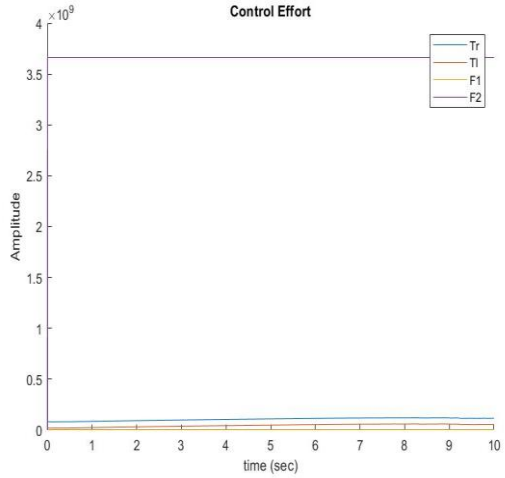
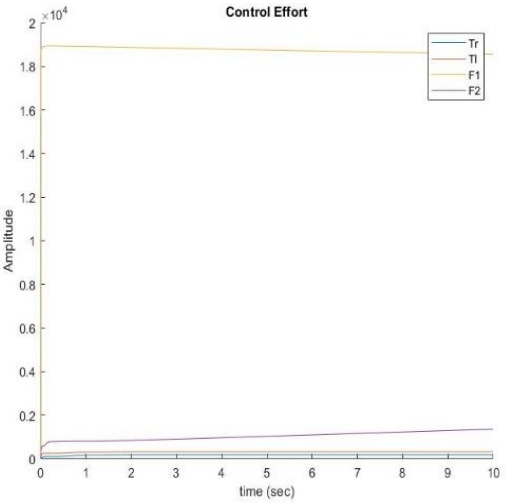


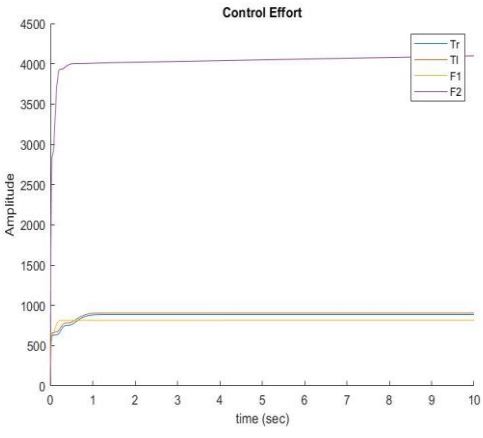
Figure 31. Robot Response Using the LQR with PID (Scenario VIII)



a. PID control effort



b. LQR control effort



c. LQR with PID control effort

Figure 32(a-c). Control effort for different controllers

Table 5. Comparison between different controllers

Output	Specification	PID				LQR				PID with LQR			
		Input				Input				Input			
		Tr	Tl	F1	F2	Tr	Tl	F1	F2	Tr	Tl	F1	F2
dr	Rise time	0.7	0.3	0.2	0.40	0.24	0.2	0.43	0.39	0.03	0.03	0.15	0.29
	Settling time	1.11	1.17	0.76	1.5	1.22	1.25	0.7	1.25	0.5	0.85	0.66	0.73
	Peak time	0.5	0.6	0.38	0.8	0.62	0.5	0.41	0.56	0.3	0.3	0.31	0.33
dl	Rise time	0	0.42	0.25	0.33	0.42	0.23	0.24	0.31	0.20	0.4	0.19	0.24
	Settling time	1.23	1	0.88	1.3	1.5	0.74	0.7	1.22	0.9	0.63	0.65	0.97
	Peak time	0.7	0.49	0.5	0.68	0.86	0.64	0.36	0.58	0.77	0.33	0.28	0.49
h1	Rise time	0.05	0.1	0. 42	0.46	0.36	0..30	0.35	0.43	0.01	0.1	0.31	0.46
	Settling time	0.8	0.9	0.7	0.88	1.1	0.7	0.66	0.77	0.7	0.6	0.5	0.61
	Peak time	0.3	0.4	0.35	0.37	0.5	0.37	0.44	0.31	0.2	0.2	0.20	0.25
h2	Rise time	0.4	0.5	0.2	0.7	0.38	0. 4	0.26	0.66	0.1	0.3	0.19	0.55
	Settling time	0.6	0.95	0.6	1.2	0.9	1.22	0.55	1.30	0.3	0.75	0.51	0.92
	Peak time	0.2	0.6	0.3	0.66	0.70	0.59	0.26	0.69	0.2	0.34	0.19	0.54
theta	Rise time	0.09	0.88	0.6	0.55	0.07	0.67	0.8	0.45	0.05	0.5	0.5	0.43
	Settling time	1.5	1.3	1.3	1.3	0.97	1.03	1.6	1.1	0.9	0.8	0.9	0.87
	Peak time	0.5	0.77	0.5	0.61	0.8	0.54	0.66	0.63	0.56	0.39	0.41	0.39

VI. CONCLUSION

The importance of effective control strategies for achieving optimal performance and safety in complex mobile robots is demonstrated in different scenarios. These strategies can ensure that the robot moves and interacts with its environment in a natural and safe way, maintains stability and balance while moving over complex terrain, coordinates the actions of individual robots in a swarm, and navigates through complex environments while avoiding obstacles. However, PID controller tuning with the MIMO system can be a significant challenge since it is a SISO controller. This investigation encountered the same issue when tuning a PID controller using MATLAB. The results showed that when the robot was controlled by a PID controller, there was no coordination between the two wheels to help the robot follow the path, resulting in poor performance. To resolve this issue, the LQR approach was applied which showed that every input had an impact on every output. As a result, the set point was not reached, and errors were large or tiny for the different outputs. To overcome these issues, this robot was controlled using a combination of PID with LQR methods. Although all inputs still have an impact on all outputs, the final results were better than expected. The idea of combining the two controllers (LQR with PID) resulted in fair energy consumption, good path tracking, and a responsive IB and end-effector. The findings were acceptable and outperformed PID controllers and the LQR approach in terms of the tilt of the robot angle.

REFERENCES

[1] Goher, Khaled M. "*A two-wheeled machine with a handling mechanism in two different directions." Robotics and biomimetics* 3.1 (2016): 22-29.

[2] Goher, K. M., and S. O. Fadlallah. "*PID, BFO-optimized PID, and PD-FLC control of a two-wheeled machine with two-direction handling mechanism: a comparative study*." *Robotics and biomimetics* 5.1 (2018): 10-11.

[3] Goher, K. M., and S. O. Fadlallah. "*Bacterial foraging-optimized PID control of a two-wheeled machine with a two-directional handling mechanism*." *Robotics and biomimetics* 4.1 (2017): 5-6.

[4] M. Bettayeb, C. Boussalem, R. Mansouri, U. M. Al-Saggaf. Stabilization of an inverted pendulum-cart system by fractional PI-state feedback. ISA Transactions, vol.53,no.2, pp.508–516, 2014.

[5] I. Boussaada, I. C. Morărescu, S. I. Niculescu. Inverted pendulum stabilization: characterization of codimension three triple zero bifurcation via multiple delayed proportional gains. Systems & Control Letters, vol.82, pp.1–9, 2015.

[6] R. M. Brisilla, V. Sankaranarayanan. Nonlinear control of mobile inverted pendulum. Robotics and Autonomous Systems, vol.70, pp.145–155, 2015.

[7] R. X. Cui, J. Guo, Z. Y. Mao. Adaptive backstepping control of wheeled inverted pendulums models. Nonlinear Dynamics, vol.79, no.1, pp.501–511, 2015.

[8] A. Elmolihi and A. Abougarair, , Robust control and optimized parallel control double loop design for mobile robot, Journal International Journal of Robotics and Automation (IJRA), Volume 9, Issue 3, PP. 160-170, 2020..

[9] G. V. Raffo, M. G. Ortega, V. Madero, F. R. Rubio. Two wheeled self-balanced pendulum workspace improvement via underactuated robust nonlinear control. Control Engineering Practice, vol.44, pp.231–242, 2015.

[10] D. H. Al-Janan, H. C. Chang, Y. P. Chen, T. K. Liu. Optimizing the double inverted pendulum′s performance via the uniform neuro multi objective genetic algorithm. International Journal of Automation and Computing, vol.14, no.6, pp.686–695, 2017.

[11] Ahmed J. Abougarair, *"Real Time Classification for Robotic Arm Control Based Electromyographic Signal*", 2022 IEEE 2st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA2022), 23-25 may,2022, Sabrata, Libya.

[12] M. Aburakhis N. Shashoa and A. Abougarair, Performance of Anti-Lock Braking Systems Based on Adaptive and Intelligent Control Methodologies, Indonesian Journal of Electrical Engineering and Informatics (IJEEI), 2022.

[13] A. Aboguarair,"Intelligent Control Design for Linear Model of Active Suspension System, 30th International Conference on Microelectronics (IEEE), 2018, Tunisia.

[14] M. Aburakhis, A. Abougarair and M. Edardar, Adaptive Neural Networks Based Robust Output Feedback Controllers for Nonlinear Systems, International Journal of Robotics and Control Systems, Vol. 2, No. 1, 2022, pp. 37-56, 2022.

[15] A. Abougarair, " *Neural Networks Identification and Control of Mobile Robot Using Adaptive Neuro Fuzzy Inference System*", ICEMIS'20: Proceedings of the 6th International Conference on Engineering & MIS 2020, Sep. 2020.

[16] Ren, Tsai-Jiun, Tien-Chi Chen, and Chun-Jung Chen. "*Motion control for a two-wheeled vehicle using a self-tuning PID controller*." *Control Engineering Practice* 16.3 (2008): 365-375.

[17] H. Gnan, A. Oun and A. Abougarair, Implementation of a Brain-Computer Interface for Robotic Arm Control, 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering MI-STA, 25-27 May 2021, Libya.

[18] Goher, Khaled M., and Sulaiman O. Fadlallah. "*Control of a Two-wheeled Machine with Two-directions Handling Mechanism Using PID and PD-FLC Algorithms.*" *International Journal of Automation and Computing* (2019): 516-517.

[19] Ahmed J. Abougarair and Ifaw Buzkhar, Modeling and Control of a Two-Wheeled Robot Machine with a Handling Mechanism, 2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA2023), 21-23 May,2023, Benghazi, Libya.

[20] Ahmed J. Abougarair, Adaptive Neural Networks Based Optimal Control for Stabilizing Nonlinear System , 2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA2023), 21-23 May,2023, Benghazi, Libya.