Menoufia J. of Electronic Engineering Research (MJEER), Vol. 31, No. 1, Jan. 2022

Embedded Adaptive 2-DOF PID Controller for Robot Manipulator using a Supervisory Fuzzy Logic System

Atef Abdrabou[1], Ahmad M. El-Nagar[2], Emad A. Elsheikh[3], Mohammad El-Bardini[4]

[1] Egyptian German Co., 6th of October City, Egypt. [2,3,4]Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt.
[1] atef_egc@yahoo.com, [2] ahmed.elnagar@el-eng.menofia.edu.eg, [3] emad.elshaikh@el-eng.menofia.edu.eg, [4] dralbardini@el-eng.menofia.edu.eg

Abstract— The objective of this study is to finding a solution to overcome the uncertainties and nonlinearities of a robot manipulator, like disturbances and other variations, in real-time by designing and implementing an adaptive 2- degree of freedom (DOF) proportional-integral-derivative (PID) controller. The proposed controller is performed using a supervisory fuzzy logic system (FLS). The employed controller proves its robustness in trajectory tracking. The performance analysis for PID, 2-DOF PID, and adaptive 2-DOF PID controllers is performed by doing the simulation using Matlab. On the other hand, the proposed controller is implemented practically using a microcontroller to prove that the proposed adaptive 2-DOF PID controller has a better performance compared to other traditional fixed-gain controllers. Also, the controller is more robust in the case of the presence of disturbances or other variations.

Keywords—PID, 2-DOF PID, Adaptive 2-DOF PID controller, Embedded controller, Robot manipulator.

I. Introduction

Robot manipulators became a more efficient substitutional for humans, especially in fields that require repetitive, complex, dangerous, or fast tasks like assembly, medical operations, welding, and materials handling [1, 2]. Achieving a perfect set-point tracking and overcoming the system's intrinsic and extrinsic effects like nonlinearity and disturbances are the main objectives of the controllers. So that designing controllers have many challenges in both theoretical and practical phases [3].

Proportional-integral-derivative (PID) controllers are the most widely used ones as a result of simplicity of design and implementation, but they cannot handle uncertainties and nonlinearities of the process as a result of controller fixed gains like intelligent controllers [4, 5].

To overcome the limitations of the PID controller, the 2-DOF PID controller is introduced to enhance trajectory tracking performance, but on the other hand, the overshoot decreasing has a direct badly effect on the speed of system response [6-8].

Fuzzy logic controllers (FLCs) are much closely human thinking than traditional controllers. FLCs proved their robustness for nonlinear systems due to their effectiveness in control strategy based on expert knowledge [9-12].

In this paper, the adaptive 2-DOF PID controller is introduced, which is composed of two parts; the first one is a nominal controller, which is represented by the 2-DOF PID controller and the second one is a variational algorithm, which is represented by the fuzzy logic system (FLS). The initial gains of the nominal controller can be obtained directly from traditional methods like the Zeigler-Nichols method, and then these gains can be adapted in a variational algorithm to improve performance and robustness against uncertainties and disturbances.

The adaptive 2-DOF PID controller is implemented practically using a microcontroller to control a robot manipulator in real-time. Both simulation and practical results proved that the employed adaptive 2-DOF PID controller has several advantages over traditional fixed gains controllers such as better stability, faster response, and smaller overshoot. Furthermore, the proposed controller is more robust even in the presence of uncertainties and other variations.

The study contributions are summarized as:

1) Proposing an adaptive 2-DOF PID controller.
2) Practical implementation for the proposed controller using a microcontroller to control in a real robot manipulator.
3) Filtering out the noise by adding a second-order digital filter.
4) Ability of the proposed controller to improve system performance.

This study is organized as the following; section 2 introduces the robot manipulator model. Section 3 introduces the proposed adaptive 2-DOF PID controller. The simulation results are presented in section 4. System implementation and different results of experiments are presented for comparison and clarification in section 5. Finally, section 6 gives the conclusion.

II. Robot Manipulator Model

The dynamic modeling of a robot manipulator is introduced in this stage for simulation, controlling, and mechanical designing. Dynamic modeling is used to define parameters and the relationship between displacement, velocity, and acceleration, to force and torque acting upon the robot manipulator joints [13]. The system model for an n-rigid link, revolute, direct-drive robot is assumed to be of the following form [14, 15]:

$$M(p)\ddot{p}+V(p,\dot{p})\dot{p}+G(p)+F\dot{p}=\tau \quad (1)$$

where p, $\dot{p}$, $\ddot{p}$ are a $nx1$ vectors of joint position, velocity, and acceleration, respectively, and n represents the robot's degree of freedom number [16]. $M(p)$ is a nxn symmetric and positive definite inertia matrix, $G(p,\dot{p})$ is a $nx1$ centrifugal and Coriolis torques vector, $V(p)$ is a $nx1$ vector of gravitational torque, F is a nxn constant diagonal positive definite viscous friction coefficient matrix, and τ is a $nx1$ joint torque vector. As all p, $\dot{p}$, $\ddot{p}$ are $nx1$ vectors, then the dynamic model (1) will become:

$$M(p)\ddot{p}+V(p,\dot{p})\dot{p}+G(p)+F\dot{p}=Y(p,\dot{p},\ddot{p})\theta \quad (2)$$

where $Y(p,\dot{p},\ddot{p})$ is a nxr matrix of the dynamic regressor and θ is a $rx1$ vector of constant parameter consisting from two parameters, which are robot parameters and payload.

III. The Employed Adaptive 2-DOF PID Controller

As stated previously, the adaptive 2-DOF PID controller is composed of two parts; the first part is the 2-DOF PID controller, which is the main controller while the second part is the supervisory FLS, which is used to adjust the parameters of the main controller as shown in Fig. 1.

A. Algorithm Structure

The 2-DOF PID controller output can be found by [17]:

$$U(s)=\left(\frac{q_2 s+q_1}{s}\right)R(s)-\left(\frac{k_d s^2+k_p s+k_i}{s}\right)Y(s) \quad (3)$$

where q_1 and q_2 represent the weighting parameters, and k_p, k_i, and k_d represent the proportional, integral, and derivative gains, respectively. $R(S)$, $Y(S)$, and $U(S)$ represent the set-point, output signal, and control signal, respectively. If the two weighting parameters are set as; $q_1=k_i$, $q_2=\beta k_p$, where β is the weighting gain and its value is between $0\le\beta\le1$ then:

$$U(s)=k_p\left(\beta R(s)-Y(s)+\left(\frac{1}{T_i s}\right)\left(R(s)-Y(s)\right)-T_d sY(s)\right) \quad (4)$$

where $T_i=\frac{k_p}{k_i}$ and $T_d=\frac{k_d}{k_p}$ represent integral and derivative time constants, respectively. Using the Euler method, we can get a discrete form for the above equation as follows;

$$u(k+1)=u(k)+k_p e_p(k+1)-k_p e_p(k)+\left(\frac{T}{T_i}\right)e_i(k)$$
$$+\left(\frac{T_d}{T}\right)e_d(k+2)-2\left(\frac{T_d}{T}\right)e_d(k+1)+\left(\frac{T_d}{T}\right)e_d(k) \quad (5)$$

where $e_p(k)=\beta r(k)-y(k)$, $e_i(k)=r(k)-y(k)$, and $e_d(k)=-y(k)$ represent proportional, integral, and derivative error signals, respectively, T represents the sampling period and k represents the sampling instants.

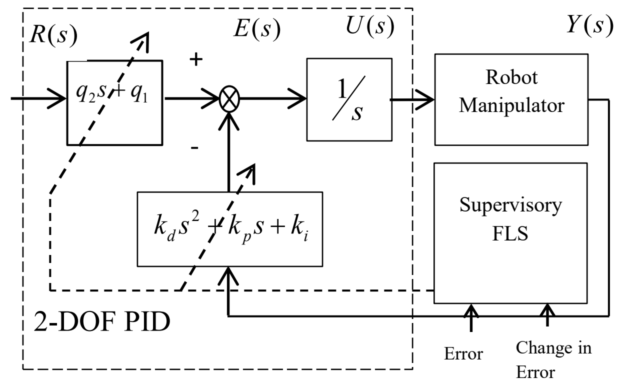

Fig. 1. Adaptive 2-DOF PID controller block diagram.

The fixed gains of the 2-DOF PID controller make the controller unable to overcome the nonlinearities and uncertainties of the process. So that FLS is proposed to adopt the controller gains to overcome these limitations.

The supervisory FLS structure is shown in Fig. 2 where the input variables are $e(k)=r(k)-y(k)$ and $\Delta e(k)=e(k)-e(k-1)$ which represent the error term and the change in error term, respectively, and the output variables are $\beta(k)$, $k_p(k)$, $k_i(k)$, and $k_d(k)$ which represent the weighting, proportional, integral, and the derivative gains, respectively.

For adapting the weighting gain $\beta(k)$, there are three scaling factors, the first and second SFs are $k_{e\beta}$ and $k_{ce\beta}$ are utilized for scaling $e(k)$ and $\Delta e(k)$ as in Eqs. (6) and (7), respectively, the third scaling factor is $k_{u\beta}(k)$ which is used to scale $\mathrm{B}(k)$ are given by Eq. (8) as follows:

$$E_\beta(k)=k_{e\beta}\,e(k) \quad (6)$$

$$\Delta E_\beta(k)=k_{ce\beta}\,\Delta e(k) \quad (7)$$

$$\beta(k)=k_{u\beta}\,\mathrm{B}(k) \quad (8)$$

For adapting the proportional gain $k_p(k)$, there are three scaling factors, the first and second SFs are k_{ep} and k_{cep} are utilized for scaling $e(k)$ and $\Delta e(k)$ as in Eq. (9)

and (10), respectively, the third SF is $k_{up}(k)$ which is used to scale $K_p(k)$ are given by Eq. (11) as follows:

$$E_p(k)=k_{ep}\, e(k) \quad (9)$$

$$\Delta E_p(k)=k_{cep}\, \Delta e(k) \quad (10)$$

$$k_p(k)=k_{up}\, K_p(k) \quad (11)$$

For adapting the integral gain $k_i(k)$, there are three SFs, the first and second SFs are k_{ei} and k_{cei} are utilized for scaling $e(k)$ and $\Delta e(k)$ as in Eqs. (12) and (13), respectively, the third SF is $k_{ui}(k)$ which is used to scale $K_i(k)$ are given by Eq. (14) as follows:

$$E_i(k)=k_{ei}\, e(k) \quad (12)$$

$$\Delta E_i(k)=k_{cei}\, \Delta e(k) \quad (13)$$

$$k_i(k)=k_{ui}\, K_i(k) \quad (14)$$

For adapting the derivative gain; $k_d(k)$, there are three SFs, the first and second SFs are k_{ed} and k_{ced} are utilized for scaling $e(k)$ and $\Delta e(k)$ as in Eqs. (15) and (16), respectively, the third SF is $k_{ud}(k)$ which is used to scale $K_d(k)$ are given by Eq. (17) as follows:

$$E_d(k)=k_{ed}\, e(k) \quad (15)$$

$$\Delta E_d(k)=k_{ced}\, \Delta e(k) \quad (16)$$

$$k_d(k)=k_{ud}\, K_d(k) \quad (17)$$

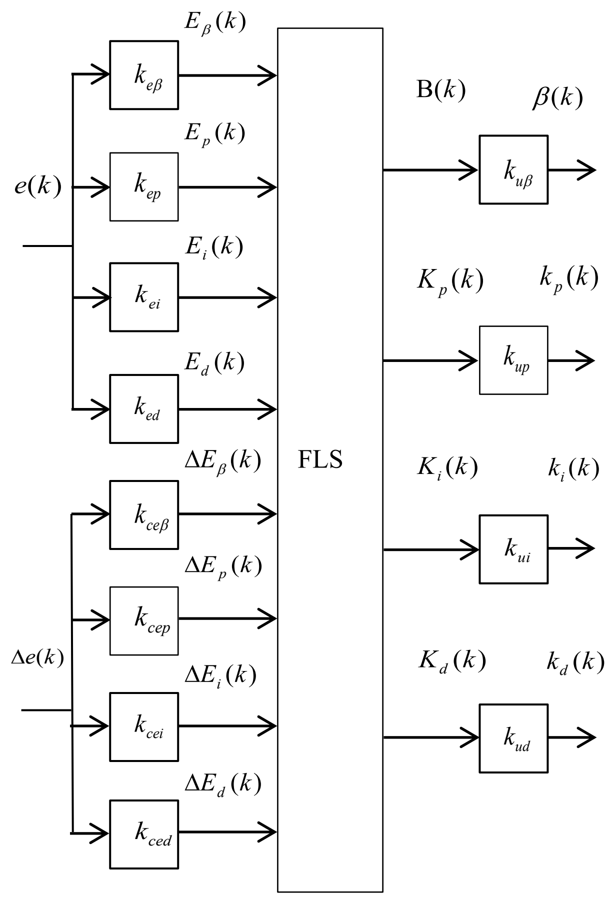

Fig. 2. Supervisory Fuzzy logic controller block diagram.

The Fuzzy Inference System (FIS) has three stages; fuzzification, rule evaluation, and defuzzification stage.

In the first stage, the fuzzy input values can be obtained from the crisp input by using membership functions (MFs), which is called the fuzzification process. The MFs of input linguistic variables $E_x(k)$ and $\Delta E_x(k)$, and the MFs of output linguistic variable $X_x(k)$ are illustrated in Fig. 3. They are consisting of three triangular MFs named as N (Negative), Z (Zero), and P (Positive).

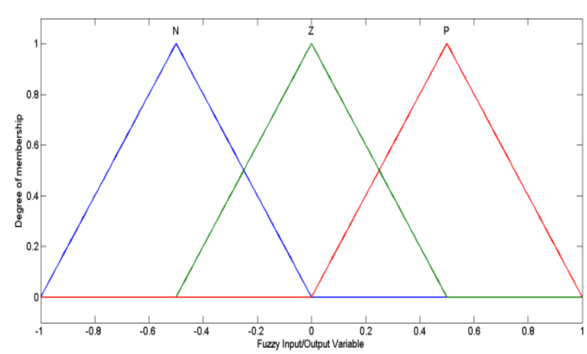

Fig. 3. Membership functions for Input/ Output variables.

In the second stage, the fuzzy outputs can be determined by using predefined linguistic rules, which is called the fuzzy inference stage. Tables 1, 2, 3, and 4 [18] represent the rules of two-inputs four outputs FIS. The rules symbolic description is given as:

$$IF\ E_x(k)\ is\ "linguistic value"\ AND\ \Delta E_x(k)\ is\ "linguistic value"\ THEN\ X_x(k)\ is\ "linguistic value"$$

TABLE 1: FUZZY RULES FOR B

B(k)		$E_\beta(k)$		
		N	**Z**	**P**
$\Delta E_\beta(k)$	**N**	P	P	Z
	Z	P	Z	N
	P	Z	N	N

TABLE 2: FUZZY RULES FOR K_p

$K_p(k)$		$E_p(k)$		
		N	**Z**	**P**
$\Delta E_p(k)$	**N**	P	P	Z
	Z	P	Z	N
	P	Z	N	N

TABLE 3: FUZZY RULES FOR K_i

$K_i(k)$		$E_i(k)$		
		N	**Z**	**P**
$\Delta E_i(k)$	**N**	P	P	Z
	Z	P	Z	N
	P	Z	N	N

TABLE 4: FUZZY RULES FOR K_d

$K_d(k)$		$E_d(k)$		
		N	**Z**	**P**
$\Delta E_d(k)$	**N**	P	P	Z
	Z	P	Z	N
	P	Z	N	N

In the third stage, the crisp output of the FLS can be calculated, which is called defuzzification stage. The center of gravity (COG) method is used for the proposed FIS [19]:

$$COG = \frac{\sum_{i=1}^{m} \mu(f_i) f_i}{\sum_{i=1}^{m} \mu(f_i)} \quad (18)$$

where m is the fired fuzzy rules number, f_i is the i^{th} fuzzy set center of gravity, and $\mu(f_i)$ is the i^{th} fuzzy set area.

IV. Simulation Results

The adaptive 2-DOF PID controller is applied for controlling one link robot manipulator by using MATLAB 7.9.0 (R2009b) to visualize the enhancements of the proposed controller compared to PID and 2-DOF PID controllers [8]. The controller's initial parameters are set as follows; for PID controller the gains are set as; $k_p = 7.5$, $k_i = 0.05$, and $k_d = 2.2$, for 2-DOF PID controller the gains are set as; $k_p = 10$, $k_i = 1.2$, $k_d = 4.2$, and $\beta = 0.99$, and for the proposed controller SF parameters are set as; are $k_{e\beta} = 0.2$, $k_{ce\beta} = 0.2$, $k_{u\beta} = 185$, $k_{ep} = 0.1$, $k_{cep} = 0.5$, $k_{up} = 0.5$, $k_{ei} = 0.125$, $k_{cei} = 0.125$, $k_{ui} = 80$, $k_{ed} = 0.125$, $k_{ced} = 0.125$, $k_{ud} = 0.5$.

A. Task 1: Tracking the Set-Point

The effect of changing the desired trajectory for different controllers is shown in this task. The first and second trajectories are represented by the following equation:

$$s(k+1) = (1-0.06) * s(k) + 0.06 * K \quad (19)$$

where $s(k+1)$ represents the set point current value, $s(k)$ represents the set point previous value, k represents the number of iterations, and K represents a constant value, its value equals to one for the first trajectory and equals to two for the second trajectory. Figs. 4, and 5 show the system outputs and control signals for the different three cases.

B. Task 2: Effect of External Load

An external load effect is applied in this task. Figs. 6, and 7 show the system outputs and control signals for two trajectories tracking with Eq. (19), an external load disturbance equals to 0.05 is added at $t = 13.8$ Sec. in case 1, and it is added at $t = 13.8$ Sec. and added again at $t = 41.3$ Sec. in case 2.

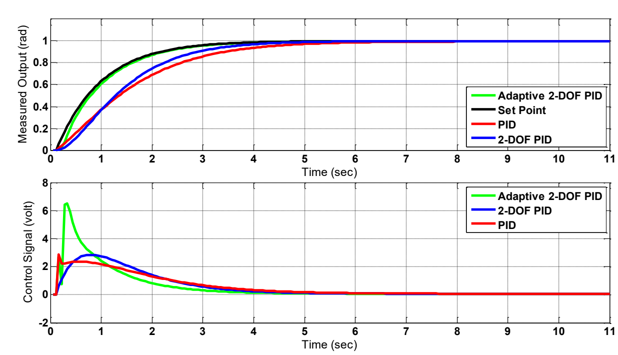

Fig. 4. Performance comparison for variable set point tracking (Task 1-Case 1).

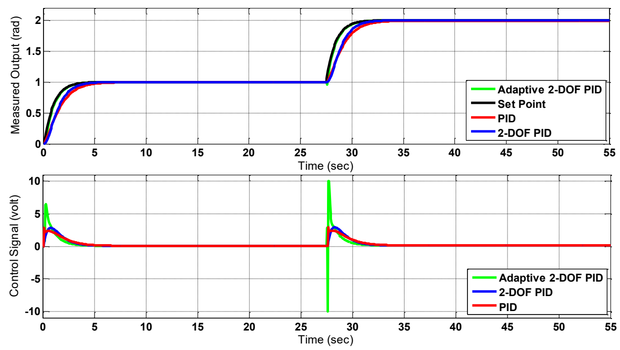

Fig. 5. Performance comparison for variable set point tracking (Task 1- Case 2).

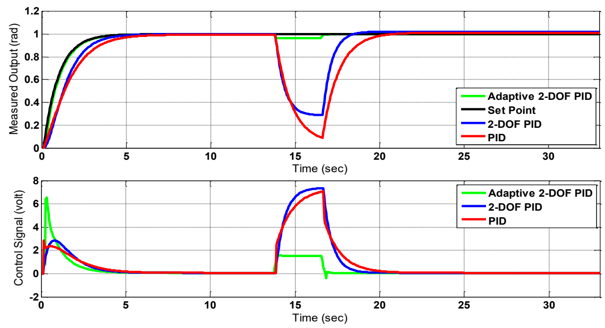

Fig. 6. Performance comparison for variable set point tracking with applying external load (Task 2- Case 1).

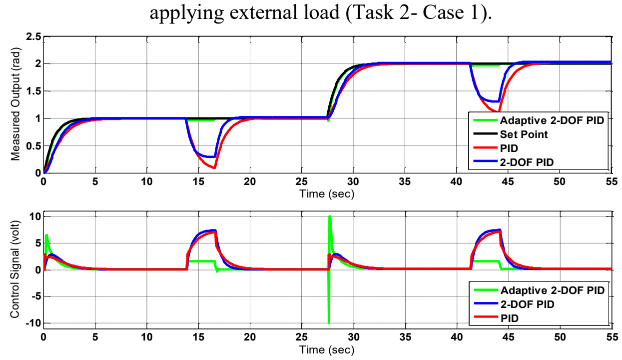

Fig. 7. Performance comparison for variable set point tracking with applying external load (Task 2- Case 2).

There are two performance indices, which indicate the differences between all controllers; the first one is MAE (mean absolute value of error) and the second one is $RMSE$ (root mean square value of error) which are defined, respectively as:

$$MAE = \frac{1}{k_f}\sum_{k=1}^{k_f}\left|e(k)\right| \tag{20}$$

$$RMSE = \sqrt{\frac{1}{k_f}\sum_{k=1}^{k_f}(e(k))^2} \tag{21}$$

where k_f is the number of iterations, $e(k)$ is the error signal.

It can be seen from Tables 5 and 6 which tabulated from the results of the simulations tasks that the adaptive 2-DOF PID controller has better values for both MAE and $RMSE$ than other traditional controllers.

TABLE 5: COMPARISON OF PERFORMANCE INDEX IN SIMULATION WITHOUT LOAD.

	Task 1: Without Load			
Controller Type	Case 1: Variable Set Point		Case 2: Another Variable Set Point	
	MAE	*RMSE*	*MAE*	*RMSE*
PID	**0.0685**	**0.1131**	**0.0313**	**0.0723**
2-DOF PID [8]	**0.0478**	**0.0941**	**0.0192**	**0.0595**
Fuzzy PD [20]	**0.0151**	**0.0707**	**0.0081**	**0.0316**
Adaptive 2-DOF PID	**0.0023**	**0.0050**	**0.0023**	**0.0065**

TABLE 6: COMPARISON OF PERFORMANCE INDEX IN SIMULATION WITH DIFFERENT LOADS.

	Task 2 : With Load			
Controller Type	Case 1: Variable Set Point		Case 2: Another Variable Set Point	
	MAE	*RMSE*	*MAE*	*RMSE*
PID	**0.1085**	**0.2403**	**0.1266**	**0.2604**
2-DOF PID [8]	**0.0852**	**0.1969**	**0.1008**	**0.2122**
Fuzzy PD [20]	**0.0233**	**0.0408**	**0.0285**	**0.0316**
Adaptive 2-DOF PID	**0.0042**	**0.0100**	**0.0054**	**0.0123**

V. PRACTICAL RESULTS

A. *Experimental Set-up*

The proposed adaptive 2-DOF PID controller has been implemented using Arduino Mega 2560 microcontroller Kit to control a robot arm as indicated in Fig. 8. The robot arm consists of 6 DC motors which can drive six links. The different values for links masses and lengths are shown in Table 7.

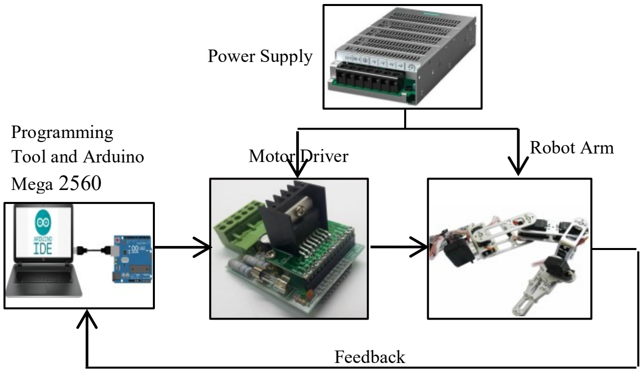

Fig. 8. Main block diagram of the system

TABLE 7 : ROBOT MANIPULATOR PARAMETER VALUES

Parameter	Description	Value	Unit
m_1	Mass of link 1 (on robot arm base)	70	gm.
m_2	Mass of link 2 (the shoulder joint)	70	gm.
m_3	Mass of link 3 (the elbow joint)	70	gm.
m_4	Mass of link 4 (the wrist Y-axis)	40	gm.
m_5	Mass of link 5 (the wrist rotation)	30	gm.
m_6	Mass of link 6 (the gripper)	115	gm.
l_1	Length of link 1 (on robot arm base)	35	mm.
l_2	Length of link 2 (the shoulder joint)	80	mm.
l_3	Length of link 3 (the elbow joint)	80	mm.
l_4	Length of link 4 (the wrist Y-axis)	58	mm.
l_5	Length of link 5 (the wrist rotation)	10	mm.
l_6	Length of link 6 (the gripper)	104	mm.

An analog potentiometer exists in each DC motor, which can transform the mechanical rotation to positional feedback to be compared with the desired position for generating the control signal. Computations and other required data are transmitted from the microcontroller kit to PC to be analyzed.

The controllers' gains are set as follows:

For PID controller the gains are set as; $k_p = 6.0$, $k_i = 0.4$, and $k_d = 0.5$, for 2-DOF PID controller the gains are set as; $k_p = 8.9$, $k_i = 1.5$, $k_d = 4.2$, and $\beta = 0.98$, and for the proposed adaptive 2-DOF PID controller the SFs are set as $k_{e\beta} = 0.2$, $k_{ce\beta} = 0.2$, $k_{u\beta} = 185$, $k_{ep} = 0.1$, $k_{cep} = 0.5$, $k_{up} = 0.5$, $k_{ei} = 0.125$, $k_{cei} = 0.125$, $k_{ui} = 80$, $k_{ed} = 0.125$, $k_{ced} = 0.125$, $k_{ud} = 0.5$.

B. Practical Tasks

The adaptive 2-DOF PID controller is applied for controlling one link robot manipulator to visualize the enhancements of the proposed controller compared to PID and 2-DOF PID controllers [8]. The different experimental results will be shown in the following tasks.

1) Task 1: Tracking the Set-Point

The effect of changing the desired trajectory for different controllers is shown in this task. The two trajectories are represented by Eq. (19), depending on the K value. Fig. 9 and 10 show the system outputs and control signals for different controllers in the two cases.

2) Task 2: Effect of External Load

An external load effect is applied in this task. Figs. 11, 12, and 13 show the system outputs and control signals for tracking one trajectory represented by Eq. (19) with K =1 for three cases with 50 gm., 100 gm., and 150 gm. loads, respectively. To demonstrate the improvements of the adaptive 2-DOF PID controller, it has been compared with other traditional controllers like PID and 2-DOF PID controllers [8].

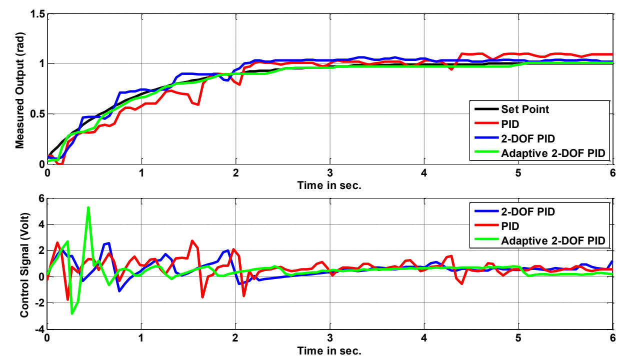

Fig. 9. Changeable set point tracking performance comparison (Task 1-Case 1)

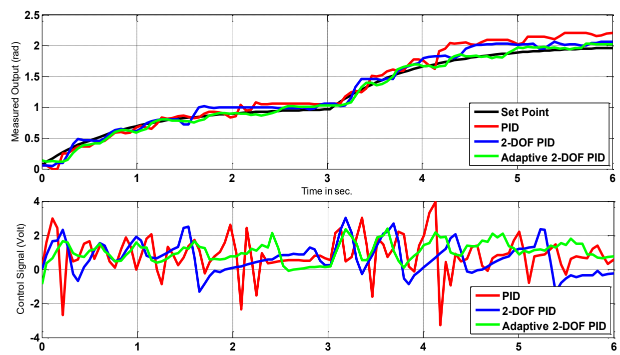

Fig. 10. Changeable set point tracking performance comparison (Task 1-Case 2)

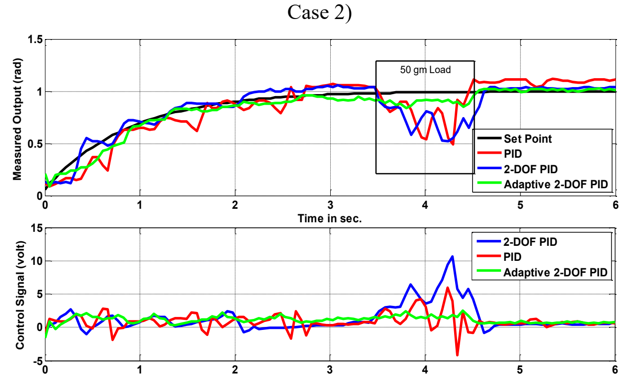

Fig. 11. Changeable set point tracking performance comparison with applying 50 gm. load (Task 2- Case 1)

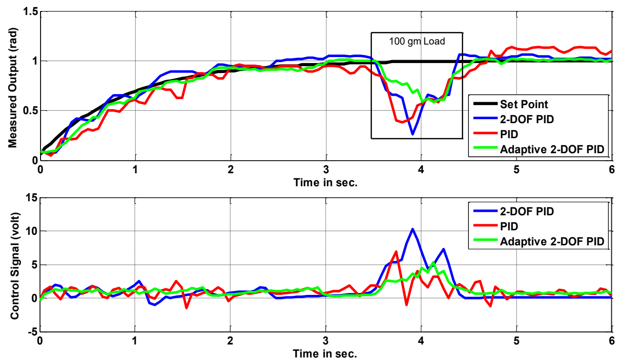

Fig. 12. Changeable set point tracking performance comparison with applying 100 gm. load (Task 2 – Case 2)

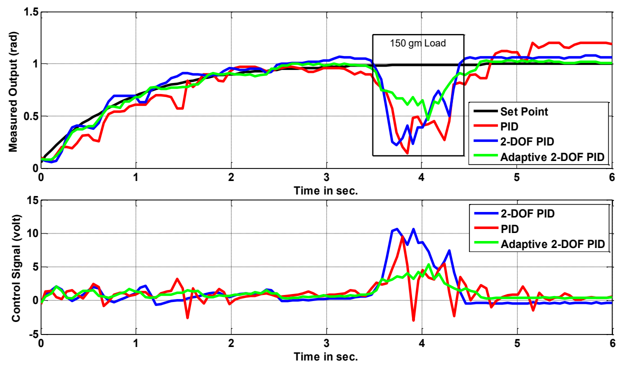

Fig. 13. Changeable set point tracking performance comparison with applying 150 gm. load (Task 2 – Case 3)

TABLE 8: COMPARISON OF PERFORMANCE INDEX WITHOUT LOAD

Controller Type	Task 1: Without Load			
	Case 1: Variable Set Point		Case 2: Another Variable Set Point	
	MAE	*RMSE*	*MAE*	*RMSE*
PID	**0.0731**	**0.0869**	**0.1189**	**0.1461**
2-DOF PID [8]	**0.0525**	**0.0596**	**0.0805**	**0.0938**
Fuzzy PD [20]	**0.0254**	**0.0379**	**0.0556**	**0.0688**
Adaptive 2-DOF PID	**0.0190**	**0.0277**	**0.0491**	**0.0585**

TABLE 9: COMPARISON OF PERFORMANCE INDEX WITH DIFFERENT LOADS

Controller Type	Task 2: With Load					
	Case 1 (50 gm. Load)		Case 2 (100 gm. Load)		Case 3 (150 gm. Load)	
	MAE	*RMSE*	*MAE*	*RMSE*	*MAE*	*RMSE*
PID	**0.1069**	**0.1391**	**0.1227**	**0.1785**	**0.1547**	**0.2346**
2-DOF PID [8]	**0.0889**	**0.1355**	**0.0912**	**0.1580**	**0.1143**	**0.2046**
Fuzzy PD [20]	**0.0510**	**0.0806**	**0.0724**	**0.1317**	**0.0871**	**0.1712**
Adaptive 2-DOF PID	**0.0488**	**0.0634**	**0.0642**	**0.1122**	**0.0677**	**0.1235**

It can be seen from Tables 8 and 9 which tabulated from the results of the experimental tasks that the adaptive 2-DOF PID controller has better values for both *MAE* and *RMSE* than other traditional controllers.

VI. CONCLUSIONS

In this paper, an embedded adaptive 2-DOF PID controller using the supervisory logic controller is applied for the position control of a robot manipulator through simulation, designing, and implementation. The simulation results depict the differences between the proposed controller and the regular ones and proved the effectiveness of tracking trajectories. The experimental results on the robot manipulator are done by using Arduino kit to demonstrate the robustness of the employed adaptive 2-DOF PID controller in dealing with a real time system, also approved the ability of the proposed controller to regulate the output against unknown external disturbances. In general, it can be said that the proposed adaptive 2-DOF PID controller is applicable, effective, and it has flexibility, adaptability, and accurate tracking performance with robust characteristics against uncertainties when compared to other controllers. In the future work, the stability analysis of the proposed controller will be studied.

REFERENCES

[1] B. Xiao, S. Yin, “Exponential tracking control of robotic manipulators with uncertain dynamics and kinematics”, IEEE Transactions on Industrial Informatics, vol. 15, no. 2, pp. 689–698, 2019.

[2] E. Akdogan, M. A. Adli, “The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherabot,” Mechatronics, vol. 21, no. 3, pp. 509–522, 2011.

[3] X. Wang, B. Hou, “Trajectory tracking control of a 2-DOF manipulator using computed torque control combined with an implicit lyapunov function method,” Journal of Mechanical Science and Technology, vol. 32, no. 6, pp. 2803–2816, 2018.

[4] A. Mehiri, R. Fareh, “Comparison study on advanced control of Two 2-DOF Manipulator robots”, International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2017.

[5] A. Savran, G. Kahraman, “A fuzzy model based adaptive PID controller design for nonlinear and uncertain processes”, ISA Trans, vol. 53, pp. 280-288, 2014.

[6] S. Preitl, R.E. Precup, Z. Preitl, “Aspects Concerning the Tuning of 2-DOF Fuzzy Controllers”, Automatic Control and Robotics vol. 9, no. 1, pp. 1-18, 2010.

[7] N.G. Adar, R. Kozan, “Comparison between Real Time PID and 2-DOF PID Controller for 6-DOF Robot Arm”, Acta Phys. Pol. A 130(1), pp. 269-271, 2016.

[8] A. D. Shakibjoo, M. D. Shakibjoo, "2-DOF PID with reset controller for 4-DOF robot arm manipulator", International Conference on Advanced Robotics and Intelligent Systems (ARIS), pp. 1-6, 2015.

[9] Y. K. Choi , M. J. Chung, Z. Bien, “An adaptive control scheme for robot manipulators”, Int. J. Control, vol.44, no.4, pp. 1185-119, 1986.

[10] A.I. Al-Odienat, A.A. Al-Lawama, “The Advantages of PID Fuzzy Controllers Over the Conventional Types”, American Journal of Applied Sciences 5 (6), pp. 653-658, 2008.

[11] W. Yang, W. Zhang, D. Xu, W. Yan, "Fuzzy Model Predictive Control for 2-DOF Robotic Arms", Assembly Automation, Vol. 38 No. 5, pp. 568-575, 2018.

[12] M. Bi, "Control of Robot Arm Motion Using Trapezoid Fuzzy Two-Degree-of-Freedom PID Algorithm", Symmetry 12, no. 4:665, 2020.

[13] S. Yadegar, A. C. Soh ,"Design Stable Robust Intelligent Nonlinear Controller for 6 DOF Serial Links Robot Manipulator", International Journal of Intelligent Systems and Applications (IJISA), MECS , pp.19-38, July 2014.

[14] F.L. Lewis, C.T. Abdallah, D.M. Dawson, "Control of Robot Manipulators", Maxwell Macmillan Publishing Co, Oxford, UK, 1993.

[15] W.S. Aboud, "Adaptive Control of Robot Manipulators with Velocity Estimation and Bounded Torque", Al-Nahrain Journal for Engineering Sciences 20 : 853-863, 2017.

[16] B. Siciliano, O. Khatib, "Springer handbook of robotics", Springer Science & Business Media, 2008.

[17] M. Kumar, V.V. Patel, "Tuning of Two Degree of Freedom PID Controller for Second Order Processes", International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 5, January 2015.

[18] A.I. Al-Odienat, A.A. Al-Lawama, "The Advantages of PID Fuzzy Controllers Over the Conventional Types", American Journal of Applied Sciences 5 (6), pp. 653-658, 2008.

[19] R.H. Mohammed, F. Bendary, K. Elserafi, "Trajectory Tracking Control for Robot Manipulator using Fractional Order-Fuzzy-PID Controller", International Journal of Computer Applications (0975 – 8887), Volume 134 – No.15, January 2016.

[20] A. M. El-Nagar, A. Abdrabou, M. El-Bardini, E. A. Elsheikh, "Embedded Fuzzy PD Controller for Robot Manipulator", International Conference on Electronic Engineering (ICEEM), pp. 1-6, 2021.