

Impedance Force / Position Control for Planar 3DOF Robot Manipulator by Fuzzy Neural Network Combination

Dr.Saad Z. S. Al-khayt
saeeds70@uomosul.edu.iq

Departement of Mechatronics Engineering, University of Mosul

Received: 21-4-2019

Accepted: 15-7-2019

ABSTRACT

As the demands for more robot's complex tasks were increased, force and torque control had become necessary. When contact forces are present, the performance of the trajectory tracking controller is degraded. Impedance force / position controller is proposed in this paper. The impedance force at the tip is controlled by fuzzy PID controller. PID controller tuned by adaptive linear network is used for trajectory tracking. A combination of fuzzy PID controller and PID controller tuned by neural network is used to generate the required torque at the robot manipulator's joints. The Jacobian matrix is derived for planar 3-DOF to transform the forces into joints' torque. Simulations are presented for robot manipulator with force contact at the tip. The trajectory tracking is improved by using fuzzy PID controller for impedance force of the environment.

Keywords:

Adaptive linear network; Fuzzy PID; Impedance force control; Jacobin matrix; Neural network; Tuned PID controller.

<http://www.uom-arej.org>

Email: alrafidain_engjournal@uomosul.edu.iq

1. INTRODUCTION

Tracking a trajectory is the problem for tasks like spot welding and cleaning a glass with sponge. The controller generates the torques which are due to robot's mechanical structure inertia forces. But when contact force is required, additional torques will be generated at the joints [1]. This contact force should be controlled specially when dealing with fragile (breakable) material such as in the case of removing paint from a glass, surgery, and assembly of parts.

Most industrial applications use Proportional-Integral-Derivative (PID) controller because it is simple and gives accepted performance. The tuning of PID gains is important and not easy task. Simplified assumptions were proposed for this reason [2]. In robot's applications, the case is complex because the mechanical structure's dynamics is nonlinear with coupled terms of joints' angular position and velocity. This leads to the need of intelligent tools for tuning PID

controllers such as neural networks and fuzzy logic [3-8].

Fuzzy logic controller (FLC) is recently used to control systems that have structural and unstructured uncertainties. Force and position control were presented in the works of [9-11]. Neural network (NN) has a strong self-adaptability, learning ability. It was used to compensate the uncertainties of robot's dynamics and environment. Neural network was used to work as a compensator in force tracking impedance control in the works of [12,13].

Fuzzy and neural network force control were implemented for position / force, and impedance force control with uncertainties and unknown environment [14-16]. A limitation was proposed for maximum and minimum values of environment's stiffness and position.

In this paper, tracking the desired trajectory is achieved by PID controller tuned by adaptive linear network. Fuzzy PID controller supplies the required joints' torque due to the

impedance force at the tip of the robot last link. Simulations for planar 3-DOF robot manipulator with force contact at the tip will be presented. In addition to this introduction, this paper contains four other sections. Section 2 presents the theoretical bases of the work. The proposed controller is fully explained in section 3. The obtained results and their corresponding discussions are included in section 4. Finally, section 5 concludes this paper.

2. ROBOT MANIPULATOR KINEMATICS

Kinematics is necessary for performing tasks by robots. The links' transformations are derived in the forwards kinematics. These transformations are required for derivation of Jacobean matrix (J) and dynamic equations.

Inverse kinematics maps Cartesian space coordinates into joint space angles. Joints' angle determine the configuration of the robot mechanical structure. Using the geometrical method, the inverse kinematics for planar 3-DOF (Fig.1) gives the joints' angle as followed [17]:

$$\theta_2 = \cos^{-1} \left(\frac{x^2 + y^2 - a_1^2 - a_2^2}{2a_1 a_2} \right) \tag{1}$$

$$\beta = \text{Atan 2} (y, x) \tag{2}$$

$$\cos \psi = \frac{x^2 + y^2 + a_1^2 - a_2^2}{2a_1 \sqrt{x^2 + y^2}} \tag{3}$$

$$\theta_1 = \beta + \psi \tag{4}$$

$$\theta_3 = \varphi - \theta_1 - \theta_2 \tag{5}$$

where

$\theta_1, \theta_2, \theta_3$ = joints' angular position.

x, y = Cartesian coordinates.

a_1, a_2, a_3 = links' length.

φ = orientation of last link.

Higher derivatives are obtained by numerical differentiation.

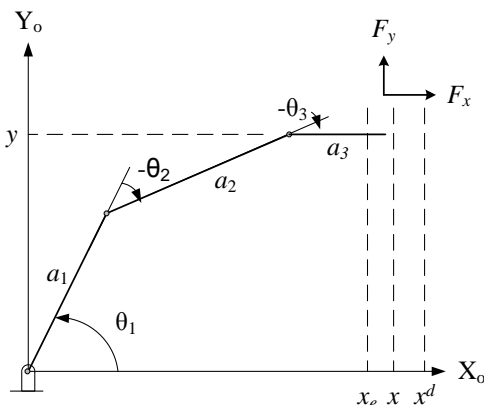


Fig.1 Planar 3-DOF robot manipulator.

Higher-order polynomials are used to describe the $x(t)$ and $y(t)$ Cartesian coordinates. The constraints for the trajectory are zero velocities and accelerations at starting and stop points. These conditions solve the six coefficients in fifth-order polynomial.

$$x(t) = c_{10} + c_{11} \cdot t + c_{12} \cdot t^2 + c_{13} \cdot t^3 + c_{14} \cdot t^4 + c_{15} \cdot t^5 \tag{6}$$

$$y(t) = c_{20} + c_{21} \cdot t + c_{22} \cdot t^2 + c_{23} \cdot t^3 + c_{24} \cdot t^4 + c_{25} \cdot t^5$$

The Jacobean matrix relates the joint rate to Cartesian velocities. At first, the velocity propagation is calculated from the base to the tip. The tip velocity is transformed to the base frame. This gives the Jacobean at the base frame [17]. The force vector (F) is transformed to joints' torque by the following relation:

$$\tau = J^T F \tag{7}$$

where

$$J^T = \begin{bmatrix} c_{123}A - s_{123}B & s_{123}A - c_{123}B \\ a_2 s_{12} - a_3 s_{123} c_3 & a_2 c_{12} - a_3 c_{123} c_3 \\ -a_3 s_{123} & a_3 c_{123} \end{bmatrix}$$

$$A = (a_1 s_{23} + a_2)$$

$$B = (a_1 c_{23} + a_2 + a_3)$$

The simplification $\sin(\theta_1)$ as s_1 will be adopted for all trigonometric function types elsewhere.

2.1 DYNAMIC EQUATIONS FOR 3-DOF ROBOT MANIPULATOR

In order to move a robot manipulator, the links must be accelerated from rest and decelerated to zero velocity at stop. The torques required for such a motion are functions of robot's mechanical structure inertia and desired trajectory. Using the Lagrangian formulation of manipulator dynamics, the dynamic equations are obtained. The details for 3-DOF robot manipulator's dynamic equations are presented in [18].

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) = \tau - \tau_e \tag{8}$$

where M is inertia matrix, C , and G are the column vectors of centrifugal and Coriolis, and Gravity matrices; respectively. τ is vector of joints' total torque. τ_e is vector of external disturbance joints' torque.

3. THE PROPOSED CONTROLLER

The torque at manipulator joints puts the tip at the desired position. The controller determines the torques from the error (e) which is

defined as the difference between the desired and actual angular position.

In self tuning PID controller, the inputs to NN are desired trajectory, actual trajectory, and the error as shown in Fig. 2. The NN output ($O_{NN}(t)$) adjusts the gains of PID controller on-line according to the change of the error signal. This overcomes the limitations of conventional PID controller and ensures stability. The control action of the PID controller after using the NN can be described as [5]:

$$\tau_{\theta} = K_{PNN} e(t) + K_{INN} \int e(t) dt + K_{DNN} \frac{de(t)}{dt} \quad (9)$$

where τ_{θ} is the torque from self learning PID controller, K_{PNN} , K_{INN} , and K_{DNN} are the new gains of PID self tuning controller and equal to: $K_{PNN} = O_{NN}(t) \cdot K_P$, $K_{INN} = O_{NN}(t) \cdot K_I$, $K_{DNN} = O_{NN}(t) \cdot K_D$.

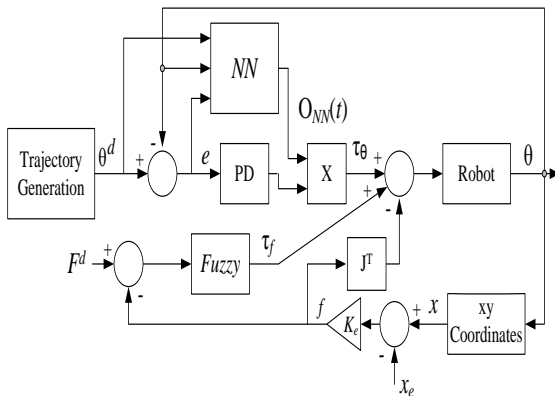


Fig. 2 Proposed fuzzy PID and self tuning PID controller.

The required torque signal to track the desired trajectory is supplied from PID controller tuned by NN. Neural network tunes on-line the gains of PID controller using feedback error learning technique. This produces adaptive controller capable of generating the control signal.

A fuzzy PID controller is used to produce additional control signal due to impedance force control. Thus the total torque becomes the summation of the trajectory tracking torque (τ_{θ}) and impedance force torque (τ_f). The inputs to Fuzzy PID are the force error (e_f) and derivative of this error as shown in Fig. 3. During the transient region, this controller works as PD controller. But at steady states it works as PI controller [8]. The membership symbols are described as PB (positive big), PM (positive medium), PS (positive small), PVS (positive very small), ZE (zero), NS (negative small), NM (negative medium), and NB (negative big). Triangular memberships are chosen for PM,

PS, PVS, ZE, NS, and NM fuzzy sets. Trapezoidal memberships are chosen for PB and NB fuzzy sets [8]. The rules are shown in Tables (1 and 2).

The desired force (F^d) and actual force (f) are obtained as following [12]:

$$F^d = k_e (x^d - x_e) \quad (10)$$

$$f = k_e (x - x_e) \quad (11)$$

where x^d and x are the desired and actual x-direction position; respectively. k_e is contact stiffness, x_e is environmental disturbance.

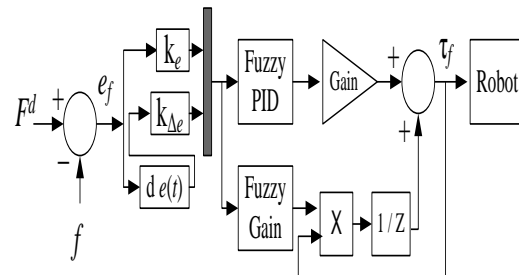


Fig. 3 Fuzzy PID Controller.

Table 1: Rules for Fuzzy PID.

$e \setminus \Delta e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Table 2: Rules for Fuzzy gain.

$e \setminus \Delta e$	NB	NS	ZE	PS	PB
NB	ZE	PVS	ZE	PVS	ZE
NS	PVS	PS	PM	PS	PVS
ZE	PS	PM	PB	PM	PS
PS	ZE	PVS	PS	PVS	ZE
PB	ZE	PVS	PS	PVS	ZE

3.1 ADAPTIVE LINEAR NETWORK

The adaptive linear network has one layer which contains S number of neurons connected to R inputs (Fig. 4). The input is multiplied by weight to produce the output through linear function. The output of this neural network is calculated as [19]:

$$o = \text{purelin} (W \cdot P + b) = W \cdot P + b \quad (12)$$

where

o = network output vector of S rows,
 W = weight matrix of S rows and P columns,
 P = Input vector,
 b = bias vector of S rows.

This neural network is used in many practical applications. A supervised training is achieved by using the least mean square error. The error is defined as the difference between desired output and network output. The performance of the network is improved by the training samples. The least mean square error is used to adjust the weights and biases during learning. It is an approximate steepest descent procedure. In the case of multiple neurons, the weight and bias are written in matrix form as:

$$W(i+1) = W(i) + 2\alpha E(i) P^T(i) \quad (13)$$

$$B(i+1) = B(i) + 2\alpha E(i) \quad (14)$$

where E and B are the error and bias vectors, and α is a learning rate.

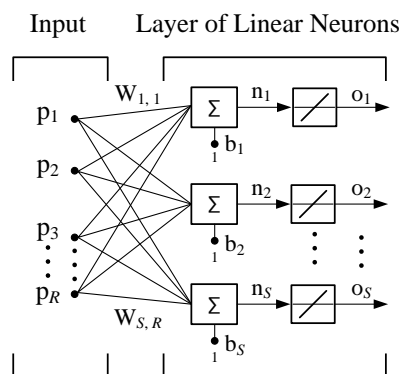


Fig.4 The adaptive linear network architecture.

This NN achieves linear approximation to nonlinear function due to the quadratic error function of least mean square error [5]. Thus the performance is either weak or global minimum according to the training samples.

Adaptive linear network can be combined with delay in the input signal so as to produce adaptive filter. Using many delayed input values supplies the network with more information to minimize the error. The linear network changes to model the nonlinear system when the system moves to a different operating point. If this error is zero, then the network has done its prediction properly. In the proposed control system, multiple neuron adaptive filter is used which contains more than one neuron in the adaptive system.

4. RESULTS AND DISCUSSIONS

The task is to move the planar 3-DOF robot manipulator along vertical wall. The tip is to be maintained normal to the wall at specified constant x -coordinate position with the last link orientation (φ) equals to 0° . Gravitational and payload force of magnitude 0.1 kg are considered. The step size for numerical integration is 0.001 seconds. Physical parameters for 3-DOF robot manipulator are: masses ($m_1=1$, kg, $m_2=1$ kg, $m_3=0.25$ kg), and lengths ($a_1=1$ m, $a_2=1$ m, $a_3=0.25$ m). The PD gains are: $K_p=[31 \ 0 \ 0; 0 \ 45 \ 0; 0 \ 0 \ 15]$, and $K_D=[60 \ 0 \ 0; 0 \ 80 \ 0; 0 \ 0 \ 25]$. The fuzzy PID parameters: Gain = $[45 \ 0 \ 0; 0 \ 35 \ 0; 0 \ 0 \ 20]$, $k_e = 1$, and $k_{\Delta e} = 0.078$. $R = 30$, $S = 3$, delay = 4, and $\alpha = 0.025$.

The robot starts from rest and moves to the starting point (1.95m, 0.05m) during the first 10 seconds. This will give enough time for NN to tune the PID gains on-line. The environment force is defined as:

$$f = \begin{cases} 0 & x < x_e \\ k_e (x - x_e) & x > x_e \end{cases} \quad (15)$$

where $k_e=1$ kN/m and $x_e=1.94$ m. This force will be transformed into joints' torque by using Jacobean transpose matrix. At position (1.95m, 0.05m) the robot starts to move its third link vertically in horizontal layout so as to maintain normal force against the vertical wall. The stop point is (1.95m, 0.4m).

The simulation results are presented in Figs. 5-13. Fig. 5 shows the time history of angular position at joint 1 using PID tuned by NN without Fuzzy PID controller for impedance force. The error is high because the controller is not supplying all the required torques. The minimum error in radian is 0.2768, 0.0272, and 0.0177 for joints 1, 2, and 3; respectively.

Figs. 6-8 show the angular position tracking using the proposed controller for joints 1, 2, and 3; respectively. The error is maximum at the beginning of simulation. This big error because high torques are required to move the robot's mechanical structure from rest. At starting point, the error becomes small because the NN had acquired enough learning during the first 10 seconds. The two controllers combine their signals so as to track the desired trajectory with applied forces at the tip. The Cartesian coordinates of the desired and actual paths are shown in Fig.9. The deviation is high at the beginning of motion due to the inertial effects of motion initiation. The time history of generated torques due to impedance force

only are depicted in Figs. 10-12. The error is decreasing with time. The errors of angular positions tracking and torques are presented in table 3. The total torques at joints due to robot dynamics and applied forces are presented in Fig. 13. Figure 14 shows the results between applied force and the system response during the last ten seconds of motion. The maximum error is no more than 5.795×10^{-3} Newton.

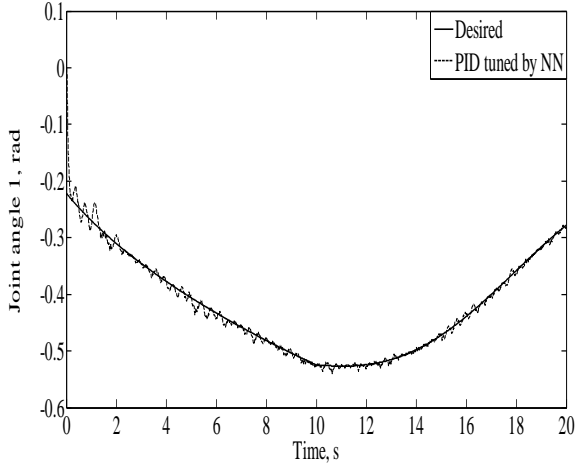


Fig. 5 Link 1 trajectory from PID tuned by NN.

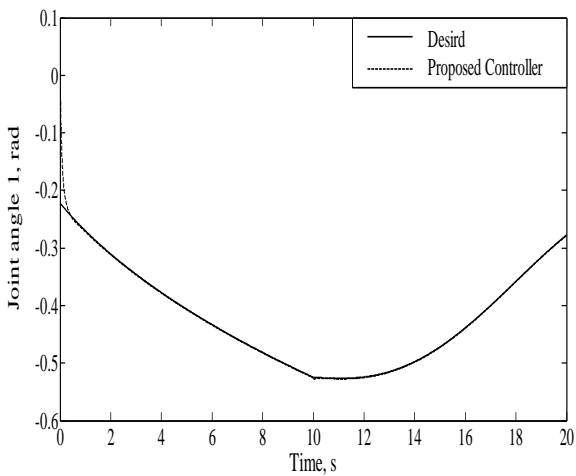


Fig. 6 Link 1 trajectory from proposed controller.

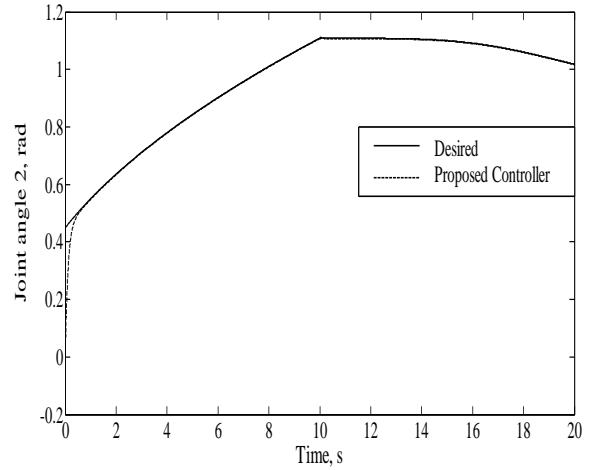


Fig. 7 Link 2 trajectory from proposed controller.

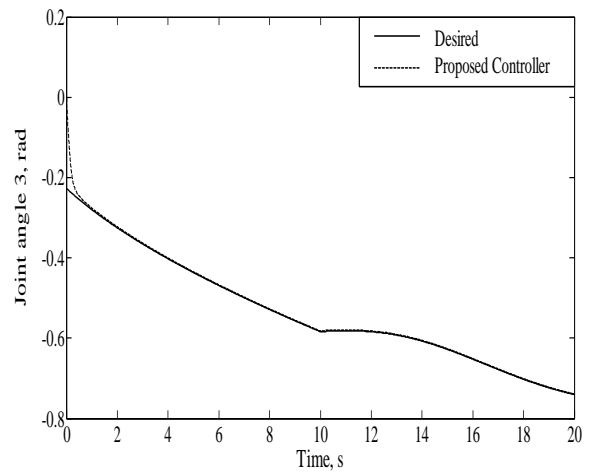


Fig. 8 Link 3 trajectory from proposed controller.

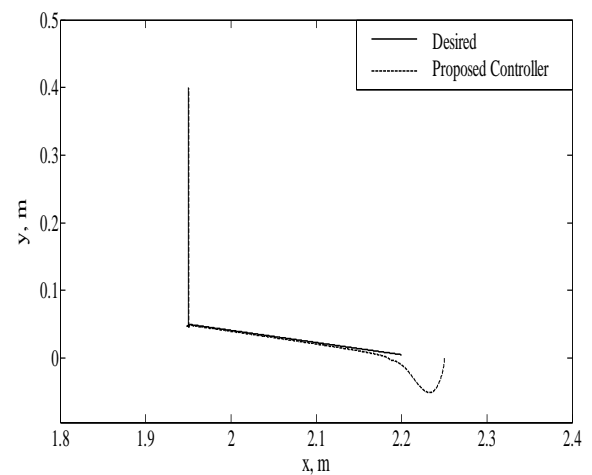


Fig. 9 Cartesian coordinates path from proposed controller.

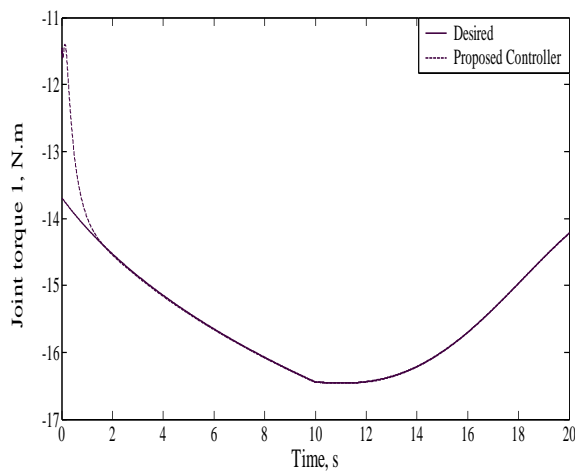


Fig. 10 Torque at joint 1 from proposed controller.

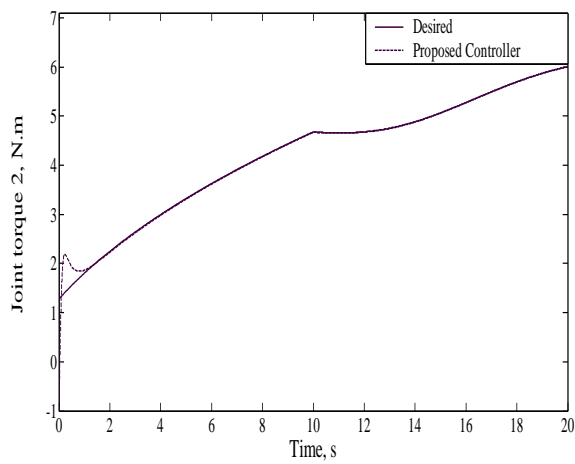


Fig. 11 Torque at joint 2 from proposed controller.

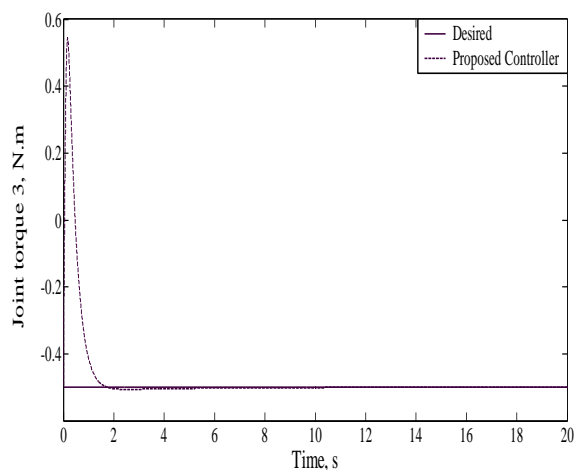


Fig. 12 Torque at joint 3 from proposed controller.

Table 3: Errors of angular positions and torques at joints

Results	Location	Joint		
		1	2	3
Angular Position (rad)	Start	0.00039	0.00057	-0.00096
	Stop	0.00039	0.00042	-0.0008
Torques (N.m)	Start	0.0033	0.0086	0.0003
	Stop	0.0037	0.0065	0.0002

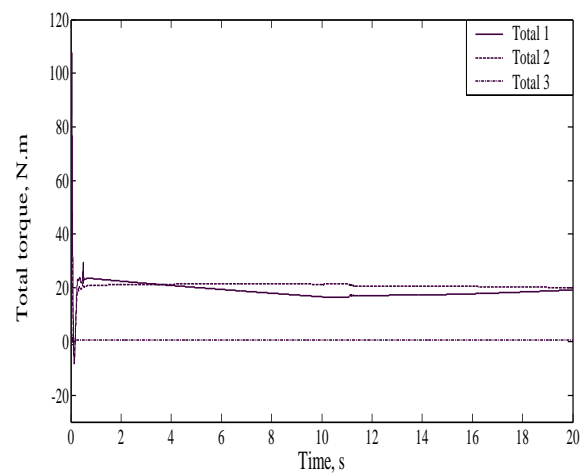


Fig. 13 Total torques at joints from proposed controller.

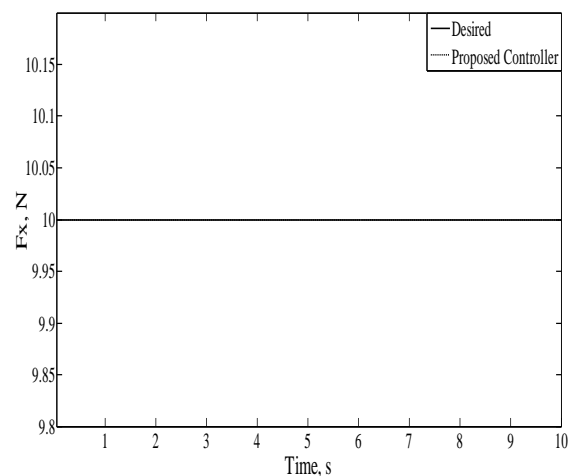


Fig. 14 Results between applied force and the system response.

Circular motion is considered as a second case to show the error in two directions. The circular path's radius is 0.5 meter and the radial impedance force's amplitude is 10 Newtons. Start point on the circular path is (2m ,0m) rotating in counter clockwise direction. The Cartesian coordinates of the desired and actual paths are shown in Fig.15. The time history of generated forces due to applied radial impedance force only are depicted in Figs. 16-17. The errors of Cartesian positions and forces are presented in table 4. The error in the vertical direction is bigger than that in the horizontal direction due to the presence of gravitational effects.

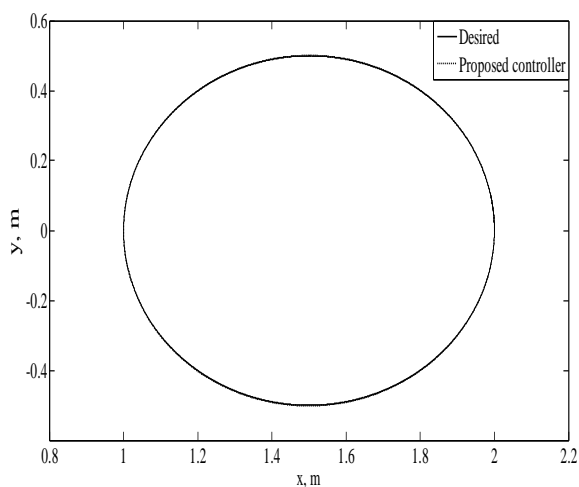


Fig. 15 Cartesian coordinates of circular path from proposed controller.

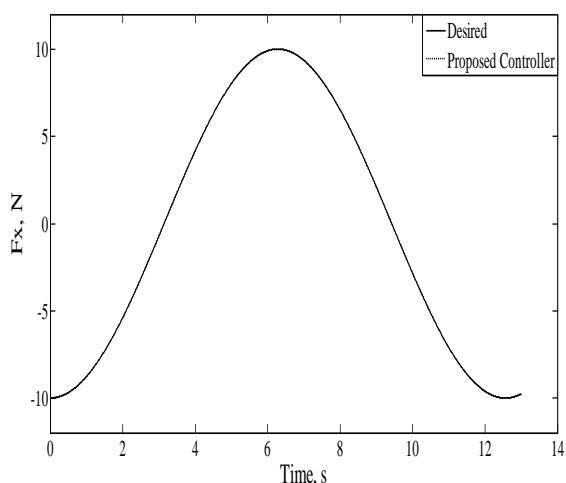


Fig. 16 Force in the horizontal x-direction from proposed controller.

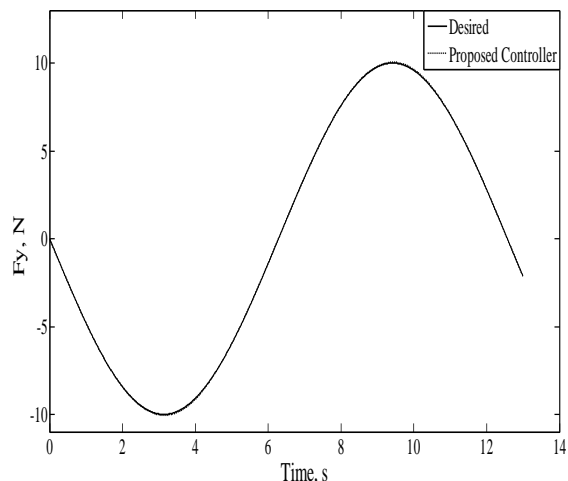


Fig. 17 Force in the vertical y-direction from proposed controller.

Table 4: Errors of Cartesian positions and forces

Results	Location	Direction *10 ⁻³	
		x	y
Position (m)	Start	1.067	1.371
	Stop	0.901	1.157
Force (N)	Start	7.394	8.261
	Stop	6.303	9.866

5. CONCLUSION

The kinematics and dynamics of planar 3-DOF present a nonlinear system with high coupling terms. Adaptive linear network tuning for PID gains is used because it has small size and fast learning. This controller improves tracking performance. The combination of PID tuning by NN and fuzzy PID controllers is proposed in this work. This combination provides robust tracking control with payload and impedance force applied at the tip of the last link. Simulation results for 3-DOF robot manipulator have demonstrated the effectiveness of this combination. Fuzzy PID controller for impedance force control improves the trajectory tracking. The error in radian is reduced from 0.2768, 0.0272, and 0.0177 to 0.00039, 0.00057, and -0.00096 for joints 1, 2, and 3; respectively. In the circular motion results, the error is increased due to the additional force component in the vertical direction. Future research will seek

advantages by applying the proposed controller to nonplanar robot manipulator.

ACKNOWLEDGEMENTS

The author thanks the Ministry of Higher Education and Scientific Research in Iraq and college of engineering in Mosul University for their encouragement to researchers and faculty members.

REFERENCES

- [1] M. Homayounzade and M. Keshmiri, "Adaptive position/force control of robot manipulators with force estimation", 2nd RSI/ISM International Conference on Robotics and Mechatronics, 2014, pp.736-741.
- [2] J.-C. Shen and H.-K. Chiang, "PID tuning rules for second order systems", IEEE 5th Asian Control Conference, vol.1, pp.472-477, 2004.
- [3] Q.-S. Lin, Y.-F. Yao, and J.-X. Wang, "Simulation and application of neural network PID auto-tuning controller in servo-system", IEEE 2nd International Workshop on Database Technology and Applications, 2010, pp.1-4.
- [4] K. Zhang and X. An, "Design of multivariable self-tuning PID controllers via quasi-diagonal recurrent wavelet neural network", 2nd International Conference on Intelligent Human Machine Systems and Cybernetics, vol.2, 2010, pp.95-99.
- [5] S. Z. S. Al-khayy "Tuning PID controller by neural network for robot manipulator trajectory tracking," Al-Khwarizmi Engineering Journal, vol.8, no.1, pp.19-28, 2013.
- [6] M. M. F. Algreer and Y. R. M. Kuraz, "Design fuzzy self-tuning of PID controller for Chopper-Fed DC motor drive" ,Al-Rafidian Engineering, vol.16, no.2, pp.54-66, 2008.
- [7] M. Ataei and S. E. Shafiei, "Sliding mode PID-controller design for robot manipulators by using fuzzy tuning approach", Proceedings of the 27th Chinese Control Conference, Kunming, Yunnan, China, 2008, pp.170-174.
- [8] S. Z. S. Al-khayy, "Comparison between Fuzzy Logic Based Controllers for Robot Manipulator Trajectory Tracking, The First National Conference for Engineering Sciences FNCES'12 / November 7-8, 2012
- [9] F. Wang, Z. Luo, H. Liu, and L. Wang, "Impedance model based fuzzy force control for robot manipulator contacting with a constrained surface with uncertain errors", IEEE International Conference on Robotics and Biomimetics, 2010, pp. 1555-1558.
- [10] P. N. Koustoumpardis and N. A. Aspagathos, "Model reference fuzzy learning force control for robotized sewing," 19th Mediterranean Conference on Control and Automation, 2011, pp.1460-1465.
- [11] M. P. Plius, M. Yilmaz, U. Seven, and K. Erbatur, "Fuzzycontroller scheduling for robotic manipulator force control," 12th IEEE International Workshop on Advanced Motion Control, 2012, pp. 1-8.
- [12] H.-R. Wang, L. Yang, and L.-X. Wei, "Intelligent position / force control for uncertain robot using neural network compensation" ,Journal of Machine Learning and Cybernetics, IEEE Proceedings of 2005 International Conference, vol.2, 2005, pp.1175-1179.
- [13] S. Jung and T. C. Hsia, "Reference Compensation Technique of Neural Force Tracking Impedance Control for Robot Manipulators", 8th World Congress on Intelligent Control and Automation, July 6-9 2010, Jinan, China, 2010, pp. 650-655.
- [14] H.-R. Wang, L. Yang, and L.-X. Wei, "Fuzzy-neuro position / force control of robot manipulators with uncertainties", Journal of Soft Computing – Afusion of Foundations, Methodologies and Applications, vol.11, no.4, 2007, pp.311-315.
- [15] N. Kharmandar and A. A. Khayyat, "Force impedance control of a robot manipulator using a neuro-fuzzy controller", IEEE International Conference on Mechatronics Science, Electric Engineering and Computer, 2011, pp.559-563.
- [16] W.-C. Wang, C.-H. Lee, "Fuzzy neural network-based adaptive impedance force control design of robot manipulator under unknown environment", IEEE International Conference on Fuzzy Systems, 2014, pp. 1442-1448.
- [17] J. J. Craig, Introduction to Robotics: Mechanics and Control, Prentice-Hall, Inc., 2005.
- [18] A. A. Atef and M. Y. Sa'adah, "Soft motion trajectory for planar redundant manipulator", In 2006 9th International Conference on Control, Automation, Robotics and Vision, pp. 1-6. IEEE, 2006.
- [19] H. Lehuy and G. Sybille, MATLAB Help ver.7, Math works, Inc., 2004.

السيطرة على القوة المعوقة و الموقع لذراع روبوت مستوي ثلاثي درجات الحرية بجمع المنطق المضرب و الشبكة العصبية

د.سعد زغلول الخياط

saeeds70@umosul.edu.iq

جامعة الموصل – قسم هندسة الميكاترونس

المستخلص

نظراً لزيادة الطلب على مهام الروبوت المعقدة، أصبح التحكم في القوة والعزم ضرورياً. عندما تسلط قوة على طرف ذراع الروبوت، فإن أداء المسيطر سوف يتدهور مما يؤدي الى زيادة مقدار الخطأ المسار. خلال هذا البحث، تم اقتراح مسيطر على القوة المعوقة و موقع نهاية الذراع الأخير في الروبوت. المسيطر التناسبي-التكاملي- التفاضلي المضرب يتحكم بالقوة المعوقة. السيطرة على تتبع المسار تتم بواسطة المسيطر التناسبي-التكاملي- التفاضلي الذي تقوم بتوليف مكاسبه شبكة خطية متكيفة. مجموع العزوم من المسيطران يكون العزم المطلوب في مفاصل الروبوت. لتحويل القوى الى عزوم في المفاصل، تم اشتقاق مصفوفة جاكوبي لروبوت مستوي ثلاثي درجات الحرية. نتائج المحاكاة العددية أظهرت ان تتبع المسار باستخدام المسيطر التناسبي-التكاملي- التفاضلي المضرب للتحكم بالقوة المعوقة تحسن بشكل جيد في الاختبارات.