ISSN:1687-1006

The Egyptian International Journal of Engineering Sciences and Technology

Vol. 24 (January 2018)) 40–47
http://www.eijest.zu.edu.eg

PID Control Versus Fuzzy Control for Spatial-Link Manipulator

Wael Khalifa[a]*, S. Elnaggar[b], M. Riad Ghazy[b]

[a]*Higher Technological Institute, Tenth of Ramadan city, P. O. Box: 228, Egypt*
[b]*Zagazig University, Zagazig city, 44519, Egypt*

ARTICLE INFO

Article history:
Received 22 November 2017
Received in revised form
17 April 2018
Accepted 20 May 2018
Available online 10 June 2018

Keywords:
PID Control
Fuzzy Logic Control
Modeling
Manipulators
Simulation

ABSTRACT

This study addresses the control problem of spatial link manipulator for three degrees of freedom arm robot. A dynamic model is proposed, and the gravity effect is taken into consideration. In this paper the results of proportional-integral-derivative, PID, are compared with the results of a fuzzy logic controller using MATLAB simulation for controlling the torque of the manipulator. The objective of the controller is to adjust the joints of the manipulator for the desired point. The outputs of the PID controller and gravity controller are combined and used as the commanded torque to the manipulator. The results show that the fuzzy logic control is superior to the PID control as the dynamic performance of the manipulator is considered.

Nomenclature

q_i	Equal θ1, θ2, or θ3 are the joint variables generalized coordinates.
τ_i	The external torques acting at the robot joints (i=1,2,3).
A_{ii}	The effective inertia coefficient at joint i.
A_{ij}	The coupling inertia between joints i and j at joint i
C_{ijj}	The centripetal force dynamic coefficient at joint i due to velocity at joint j.
C_{ijk}	The Carioles force dynamic coefficient at joint i due to velocities at joints j and k.
D_i	The gravity loading at joint i, k.
$C(q,\dot{q})$	The Coriolis /Centripetal Vector.
$D(q)$	The Gravity vector.
τ	The external torques applied at the arm joints (physical input).
T	The transpose column vector.
θ_{1d}	The desired values for θ_1.
θ_{2d}	The desired values for θ_2.
θ_{3d}	The desired values for θ_3.
e	The tracking error defined by $(\theta - \theta_d)$.
d_{2r}	The gravity loading at joints 2.
d_{3r}	The gravity loading at joints 3.
$\hat{\tau}$	The non-physical input.

[a] Corresponding Author. Tel.: +2-010-2666-5629
e-mail address: eng_waelkhalifa2009@yahoo.com

1. Introduction

Manipulator is an actuated machine that is used to move objects in a similar way to a human arm. The manipulator considered for this work consists of three rigid links connected by three revolute joints to provide rotational motion like a human arm (twist, shoulder and elbow). To achieve a desired position, a manipulator is required to accelerate from rest and move on predefined path and finally decelerate to stop. To perform this job, controlling torque is applied on the actuators through the manipulators joints. The problem of manipulator control is to find the time behaviour of the forces and torques delivered by the actuators for executing the desired job. Control of a system usually requires the availability of a mathematical model and some types of controller to apply the model.

There are many control methods used to control the robot manipulator. The most used ones in the industrial processes are the PID control [1][2], fuzzy logic control, [3] optimal control [4][5], adaptive control [6] and robust control. [7] There are many types of controllers that can be used to cause a designed robot arm to move along a desired trajectory. [8]

The PID controller is the most popular type of feedback. It was a primary component for the old governors and it became the typical control for most process control. The proportional-integral-derivative (PID) controller has a simple structure, and its three constant gains are easy to be physically interpreted. The control performances are acceptable in the most of industrial processes. Most robot manipulators used in modern industries are controlled by an independent PID algorithm at each joint. [9]

Recently, PID controllers are found in all regions where control is utilized. Basically, all PID controllers made today depend on microprocessors. [10] This has offered chances to give extra characteristics like automatic tuning, [11] gain scheduling, [12] and continuous adaptation.

The soft computing methods are used to obtain the fuzzy control as fuzzy logic and fuzzy set theory that were presented in 1965 by Lotfi Zadeh. [13]

He explained that fuzzy logic is not like classical logic in realizing values between false and true by using a set of membership functions constructing the rule base, fuzzification and defuzzification methods. In the view of classical set theory, the meaning of the membership function does not make a difference, but the number may or may not belong to the set (0 or1) that takes on the value. Through the last twenty years, fuzzy logic control ,FLC, has been one of the most methods, active and productive areas of research. It has been effectively utilized in a wide verity of applications like: economic, engineering and other areas including high level of complexity, uncertainty and non-linearity. [14]

Until now, the fuzzy logic controller has been the most successful application in this field. Several applications show that the yield results of the fuzzy logic controllers are better than those using conventional control algorithms. However, fuzzy logic controllers are mainly non-linear and sufficiently successful to give the required non-linear control actions by cautiously modifying their parameters.

In this paper, we used the PID and fuzzy logic control techniques to control the three-link robot manipulator shown in Fig. 1.

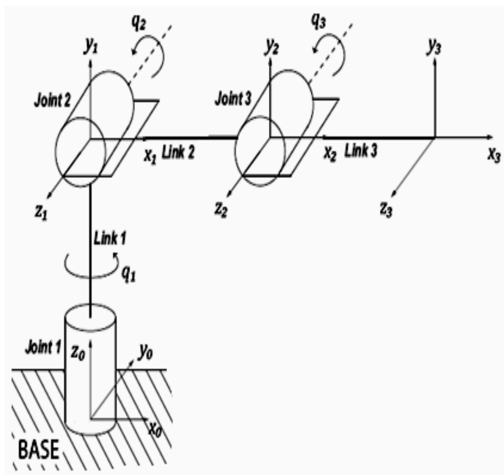

Fig. 1. Three Link Manipulator

2. Dynamic Modeling For Spatial-Link Manipulator

The general equations of motion of the manipulator, shown in Fig.1, based on Lagrangian dynamics can be calculated from

$$\frac{d}{dt}\left(\frac{\partial KE_s}{\partial \dot{q}_i}\right)-\frac{\partial KE_s}{\partial q_i}+\frac{\partial PE_s}{\partial q_i}=\tau_i \tag{1}$$

where, $q_i=\theta_1,\ \theta_2,\ \text{or}\ \theta_3$

The three equations of motion that govern the dynamic behaviour of the manipulator are derived as:

$$A_{11}\ddot{\theta}_1 + C_{112}\dot{\theta}_1\dot{\theta}_2 + C_{113}\dot{\theta}_1\dot{\theta}_3 = \tau_1 \quad (2)$$

$$A_{22}\ddot{\theta}_2 + A_{23}\ddot{\theta}_3 + C_{211}\dot{\theta}_1^2 + C_{233}\dot{\theta}_3^2 + C_{232}\dot{\theta}_3\dot{\theta}_2 + D_2 = \tau_2 \quad (3)$$

$$A_{32}\ddot{\theta}_2 + A_{33}\ddot{\theta}_3 + C_{311}\dot{\theta}_1^2 + C_{322}\dot{\theta}_2^2 + c_{323}\dot{\theta}_2\dot{\theta}_3 + D_3 = \tau_3 \quad (4)$$

The compact matrix-vector notation form can be written as:

$$A(q)\ddot{q} + C(q,\dot{q}) + D(q) = \tau \quad (5)$$

Where, $A(q) = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & A_{23} \\ 0 & A_{32} & A_{33} \end{bmatrix}$,

$$C(q,\dot{q}) = \begin{bmatrix} c_{112}\dot{\theta}_1\dot{\theta}_2 + c_{113}\dot{\theta}_1\dot{\theta}_3 \\ c_{211}\dot{\theta}_1^2 + c_{233}\dot{\theta}_3^2 + c_{232}\dot{\theta}_2\dot{\theta}_3 \\ c_{311}\dot{\theta}_1^2 + c_{322}\dot{\theta}_2^2 + c_{323}\dot{\theta}_2\dot{\theta}_3 \end{bmatrix},$$

$D(q) = [0 \quad D_2 \quad D_3]^T$ and $\tau = [\tau_1 \quad \tau_2 \quad \tau_3]^T$.

finally, the state-space model shown as:

$\dot{y}_1 = y_4$, $\dot{y}_2 = y_5$, $\dot{y}_3 = y_6$,

$$A_{11}\dot{y}_4 = -(C_{112}y_4y_5 + C_{113}y_4y_6) + u_1 \quad (6)$$

$$A_{22}\dot{y}_5 + A_{23}\dot{y}_6 = -(C_{211}y_4^2 + C_{233}y_6^2 + C_{232}y_5y_6) + u_2 \quad (7)$$

$$A_{32}\dot{y}_5 + A_{33}\dot{y}_6 = -(C_{311}y_4^2 + C_{322}y_5^2 + C_{323}y_5y_6) + u_3 \quad (8)$$

Where the state space vector $\boldsymbol{Y}$ is provided as

$Y_1 = \theta_1$, $Y_2 = \theta_2$, $Y_3 = \theta_3$,

$Y_4 = \dot{\theta}_1$, $Y_5 = \dot{\theta}_2$, $Y_6 = \dot{\theta}_3$,

$u_1 = \tau_1$, $u_2 = \tau_2 - D_2$, $u_3 = \tau_3 - D_3$

The compact form of the state space model in matrix-vector form is:

$$\mathbf{M}(t)\dot{\mathbf{y}} = \mathbf{f}(t,y,u) \quad (9)$$

Where,

$$\mathbf{M}(t) = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & A_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & A_{22} & A_{23} \\ 0 & 0 & 0 & 0 & A_{32} & A_{33} \end{array}\right]$$

$$\mathbf{f}(t,y,u) = \begin{bmatrix} y_4 \\ y_5 \\ y_6 \\ -(C_{112}y_4y_5 + C_{113}y_4y_6) + u_1 \\ -(C_{211}y_4^2 + C_{233}y_6^2 + C_{232}y_5y_6) + u_2 \\ -(C_{311}y_4^2 + C_{322}y_5^2 + C_{323}y_5y_6) + u_3 \end{bmatrix},$$

$$\dot{y} = [\dot{y}_1, \dot{y}_2, \dot{y}_3, \dot{y}_4, \dot{y}_5, \dot{y}_6]^T,$$

For the purpose of simulations, the system of equations of motion is converted to first order differential equations using state space technique and then state space model is solved numerically by using Rung Kutta method in MATLAB package using the command ode 45.

3. PID Controller

To control the manipulator, we will use a controller in the computed torque family, which is the “PID-plus-gravity controller”. The control torque signal (τ) of this controller is chosen as PID feedback.

A proportional–integral–derivative controller is a feedback controller. The controller tries to minimize the errors by adjusting the process control inputs. Here, PID controller in parallel form, so the output (τ) of PID controller is the addition of the proportional (P), integral (I), and derivative (D) terms. PID controller is also called three-term control. The calculation of PID controller involves three separate constant parameters. [15]

The independent joint control torques are thus:

$$\tau_1 = k_{1p}(\theta_1 - \theta_{1d}) + k_{1d}(\dot{\theta}_1) + k_{1i}\int e(\theta_1)\,dt \quad (10)$$

$$\tau_2 = k_{2p}(\theta_2 - \theta_{2d}) + k_{2d}(\dot{\theta}_2) + k_{2i}\int e(\theta_2)\,dt + d_{2r} \quad (11)$$

$$\tau_3 = k_{3p}(\theta_3 - \theta_{3d}) + k_{3d}(\dot{\theta}_3) + k_{3i}\int e(\theta_3)dt + d_{3r} \quad (12)$$

The values of the PID gains, K_{iP} , K_{iI} and K_{iD} are used in this study which are obtained by trial and error. These three parameters were tuned to have the best performance. The best values for the parameters are chosen as in table 1.

Table 1. Control Gains

Set	Joint 1	Joint 2	Joint 3
K_{iP}	700	900	400
K_{iI}	160	150	50
K_{iD}	55	50	20

In this way, the system equation (5) can be written as:

$$\ddot{q} = A(q)^{-1}\left[-C(q,\dot{q}) - D(q)\right] + \hat{\tau} \quad (13)$$

with

$$\hat{\tau} = A(q)^{-1}\tau\,, \qquad \tau = A(q)\hat{\tau}$$

So, the system of equations may be decoupled to get new (non-physical) input

$$\hat{\tau} = \begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} = \begin{bmatrix} k_{1p}(\theta_1 - \theta_{1d}) + k_{1d}(\dot{\theta}_1) + k_{1i}\int e(\theta_1)\,dt \\ k_{2p}(\theta_2 - \theta_{2d}) + k_{2d}(\dot{\theta}_2) + k_{2i}\int e(\theta_2)\,dt + d_{2r} \\ k_{3p}(\theta_3 - \theta_{3d}) + k_{3d}(\dot{\theta}_3) + k_{3i}\int e(\theta_3)dt + d_{3r}\end{bmatrix} \quad (14)$$

However, the physical torque inputs to the system are:

$$\begin{bmatrix}\tau_{\theta 1}\\ \tau_{\theta 2}\\ \tau_{\theta 3}\end{bmatrix} = A(q)\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix}$$

The error signals

$$e(\theta_1) = \theta_1 - \theta_{1d}$$
$$e(\theta_2) = \theta_2 - \theta_{2d}$$
$$e(\theta_3) = \theta_3 - \theta_{3d}$$

With final positions

$$\begin{bmatrix}\theta_{1d}\\ \theta_{2d}\\ \theta_{3d}\end{bmatrix} = \begin{bmatrix}5\\ 10\\ 15\end{bmatrix}$$

The system has initial positions

$$[\theta_0] = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$$

With a specific end goal to apply all controls of Proportional-Integral-Derivative actions, with the assumption of a state of integration for each angle to facilitate the input process inside the computer as:

$$\begin{aligned} X_1 &= \int e(\theta_1)\,dt \;\Rightarrow\; \dot{X}_1 = (\theta_1 - \theta_{1d}) \\ X_2 &= \int e(\theta_2)\,dt \;\Rightarrow\; \dot{X}_2 = (\theta_2 - \theta_{2d}) \\ X_3 &= \int e(\theta_3)\,dt \;\Rightarrow\; \dot{X}_3 = (\theta_3 - \theta_{3d}) \end{aligned} \quad (15)$$

So, the complete system equations are

$$\left\{\begin{aligned} \dot{X}_1 &= (\theta_1 - \theta_{1d}) \\ \dot{X}_2 &= (\theta_2 - \theta_{2d}) \\ \dot{X}_3 &= (\theta_3 - \theta_{3d}) \\ \begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix} &= A(q)^{-1}\left[-C(q,\dot{q}) - D(q)\right] + \begin{bmatrix} k_{1p}(\theta_1 - \theta_{1d}) + k_{1d}(\dot{\theta}_1) + k_{1i}x_1 \\ k_{2p}(\theta_2 - \theta_{2d}) + k_{2d}(\dot{\theta}_2) + k_{2i}x_2 + d_{2r} \\ k_{3p}(\theta_3 - \theta_{3d}) + k_{3d}(\dot{\theta}_3) + k_{3i}x_3 + d_{3r}\end{bmatrix} \end{aligned}\right\} \quad (16)$$

MATLAB is used to simulate the robot arm model above to demonstrate the effectiveness of the proposed PID plus gravity controller. In the simulation study presented here, only step response (set-point-tracking) is considered. Each of the input torques at the joints is supposed to be a step function. The desired values for the twist, shoulder, and elbow joint angles are, Θ_{1d} = 5 deg., Θ_{2d} = 10 deg. and Θ_{3d} = 15 deg. The parameters of the manipulator are chosen as given in table 2.

Table 2. Manipulator Parameter

$a_1 = 0.5m$	$a_2 = 0.5m$	$a_3 = 1.0m$
$j_1 = 0.2kg\ m^2$	$\rho_2 = 2.0kg/m$	$\rho_3 = 2.0kg/m$
$m_T = 0.55\ kg$	$g = 9.81\ ms^{-2}$	

4. Fuzzy Logic Controller

The fuzzy controller developed for the application is a Mamdani fuzzy controller [16]. Input membership functions chosen are Gaussian. Output membership functions also Gaussian and the number of output variables is one.

In this study, two inputs of fuzzy logic controller are represented by seven membership functions. For designing the FLC, start creating the rule base in FIS Editor in MATLAB by using the controller as a feedback one. Fuzzy logic controller produces the output (control torque) with the two inputs (position and velocity error) as shown in figures (2-4).

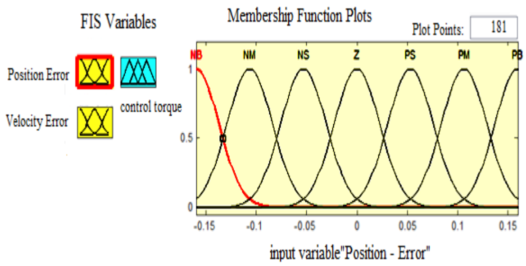

Fig. 2. Membership function for position error

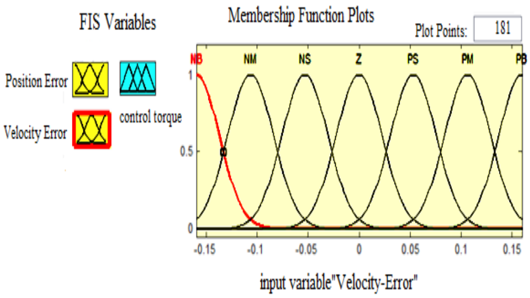

Fig. 3. Membership function for velocity error

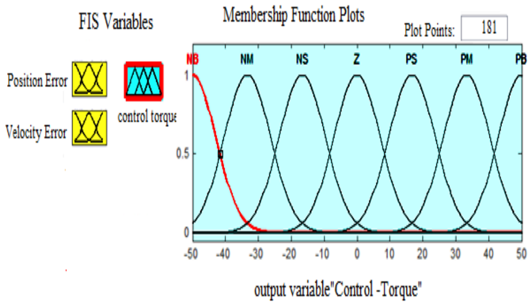

Fig. 4. Membership function for output control torque

4.1. Fuzzy Rules

People take decisions depending on the rules. Although, people may not know about it, all the decisions they make are all based on statements like " if-then" statements. Generally, the input to an "if-then" rule is the actual value of the input variable (Position error, Velocity error) also the output is a whole fuzzy set (control torque). Later this set will be defuzzified, assigning one value to the output. Interpreting "if-then" rules in five-part process as:

Fuzzify inputs

The outputs e_θ and $\dot{e}_\theta$ which are considered as input to fuzzy logic control ,FLC, are always crisp numerical value. These values must be assigned from human practical experience, but here, we get these values from the simulation results for the same manipulator with PID controller.

Apply fuzzy operator

Fuzzy logic operators is used to change the antecedent of rules which have more than one part to one number that represents the result of the antecedent for the activated rule. In our case the logic operators are “AND” which support the minimum method.

Apply implication method

Each rule is weighted in respect to others by a weight value varying between 0 and 1 which gives an impact on the single output number of the antecedent of the rule, therefore, it affects the implication method. The implication technique is described as the forming of the consequent (a fuzzy set) depending on the antecedent (single number). Implication takes place for each rule and we use the "AND" technique to truncate the output fuzzy set.

Aggregate all outputs

It is just a matter of conversation that all the fuzzy sets that represent the output of each rule and merging them into a single fuzzy set. Listing all the truncated output functions that returned by the implication process for each rule leads to the input of the aggregation process. However, the output is one fuzzy set for each output variable.

Defuzzify

The input of the defuzzification process is a fuzzy set and the output is a single number (i.e.) it is a change from the aggregated fuzzy set to a crisp output. Five defuzzification methods can be used such as: centroid, bisector, largest of maximum, smallest of maximum and middle of maximum. The most common one is the centroid calculation which returns the centre of the area under the curve.

Table (3) shows the output control actions corresponding to the input linguistic variables. The Fuzzy logic control torque is derived using a fuzzy logic MATLAB toolbox by processing the steps explained above.

Here NB means Negative Big, NM means Negative Medium, NS means Negative Small, Z means Zero and PS means Positive Small, PM means Positive Medium and PB means Positive Big.

5. Simulation Results

Simulation runs have been run to test the performance of the PID and FL controllers as applied to the considered manipulator. The response of the three linked rigid manipulator parameters (Θ1, Θ2 and Θ3) for both controllers are shown in figures (5-7).

The settling and the overshoot time were calculated from both controllers. By simulation result, the output response of both the PID and Fuzzy logic were compared as shown in table 4. From the analysis and comparison of both controllers, the FLC performs better compared to PID controller in terms of percentage overshoot (0% for Θ_1, 4.39% for Θ_2 and 3.33% for Θ_3) while the PID overshoot is (2.9% for Θ_1, 14.6% for Θ_2 and 6.40% for Θ_3). The results showed that, when using FLC with the spatial link manipulator, the overshoot of the angle Θ_1 is improved from 5.12° when using PID to 5° when using FLC by (2.34% improvement) and the angle Θ_2, the overshoot is decreased from 11.75° when using PID to 10.48° when using FLC by (10.80% improvement). At the same time the overshoot of the angle Θ_3 is decreased from 15.96° when using PID to 15.5° when using FLC by (2.88% improvement). When comparing the settling time for both controllers, the FLC performs better with 0.25S, 0.27S and 0.20S respectively.

Table 3. Rules for joint angle fuzzy controller

position error ⇒ Velocity error ⇓	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NB	NM	NM	NS	Z
NM	NB	NB	NM	NM	NS	Z	PS
NS	NB	NM	NM	NS	Z	PS	PM
Z	NM	NM	NS	Z	PS	PM	PM
PS	NM	NS	Z	PS	PM	PM	PB
PM	NS	Z	PS	PM	PM	PB	PB
PB	Z	PS	PM	PM	NB	NB	PB

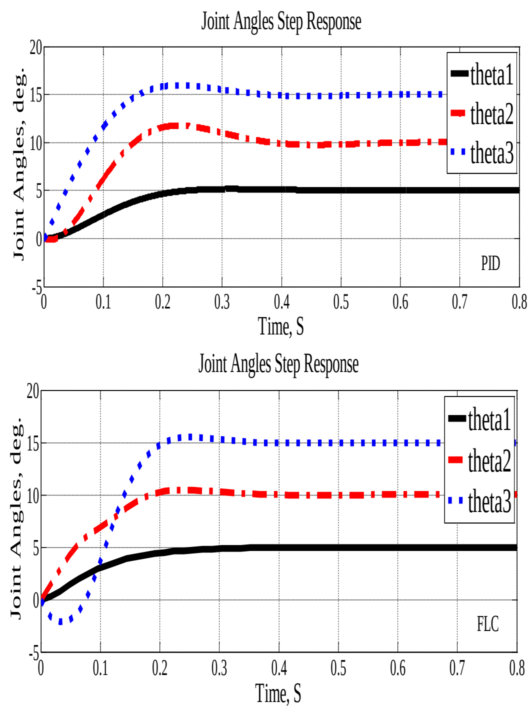

Fig. 5. Joint angles response using (PID & FLC)

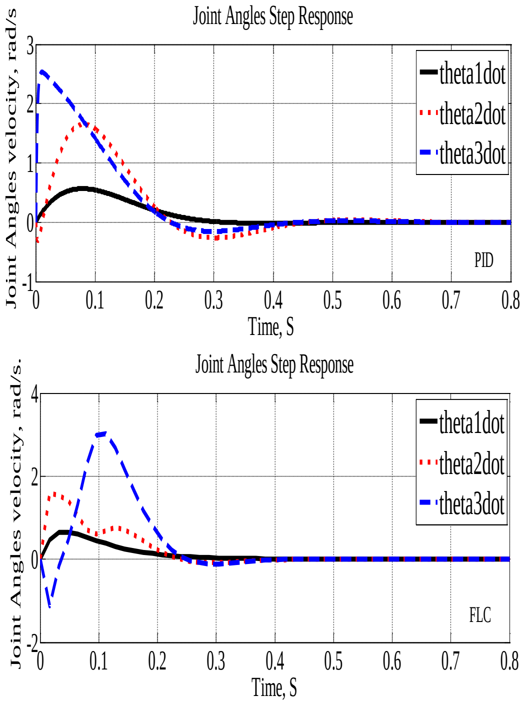

Fig. 6. Joint angles velocity using (PID &FLC)

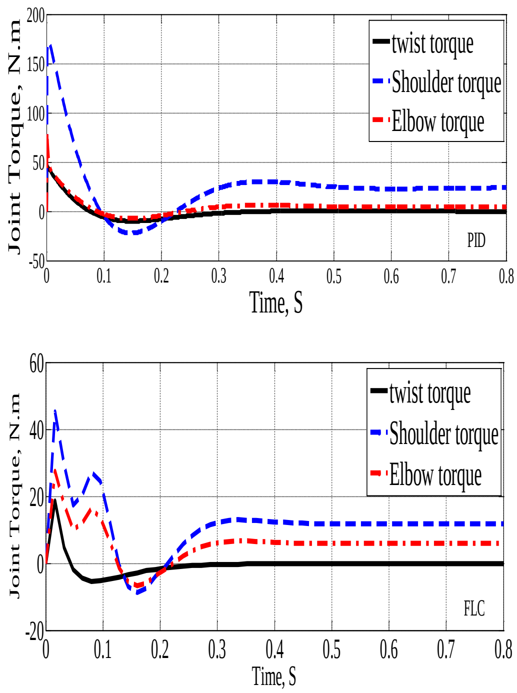

Fig. 7. Joint torque using (PID & FLC)

Moreover, the torque needed to drive the controllers is less in FLC than the PID as shown in Fig.7. This has the advantage of requiring smaller driving motors and less power consumption. The maximum angular velocity of the links during motion is also less for FLC controller than the PID as shown in Fig.(6), which means less inertia forces and stresses on the frame and the joints of the manipulator.

Table 4 Comparison between two controllers

Time Domain	PID Control			Fuzzy Logic Control		
	Θ_1	Θ_2	Θ_3	Θ_1	Θ_2	Θ_3
Settling time (sec)	0.39	0.48	0.46	0.25	0.27	0.20
Overshoot (%)	2.9	14.6	6.40	0	4.39	3.33
Steady state error	0	0	0	0	0	0

6. Conclusion

In this study PID and Fuzzy Controllers are successfully designed and implemented for control of three rigid link manipulator. Based on the results above it is concluded that the best performance is obtained by using the fuzzy control technique as it gives smaller overshoot, zero steady state error and smaller settling time than obtained using PID controller which gives high overshoot and settling time with zero steady state error.

References

[1] K. Astrom and T. Hagglund, "PID controllers: Theory, design and tuning". 2nd ed., The Instrumentation, Systems and Automation Society (ISA), (1995).

[2] Ang, K.H., Chong, G.C.Y., and Li, Y., "PID control system analysis, design, and technology". IEEE Trans Control Systems Tech, 13(4), pp.559-576, (2005).

[3] Ishikawa, T., "A study on fuzzy control of an arm robot". B. E. thesis, Dept. Contr. Eng., Tokyo Inst. Technol., (1988).

[4] Bryson, A. E., "Optimal Control–1950 to 1985". IEEE Control Systems. 16 (3), pp. 26–33, doi:10.1109/37.506395, (1996).

[5] Haggag, S.; Desokey, F.; Ramadan, M., "A cosmological inflationary model using optimal control". Gravitation and Cosmology. Pleiades Publishing. 23 (3): pp.236–239, ISSN 1995-0721, (2017). doi:10.1134/S0202289317030069.

[6] Astrom, Karl, "adaptive control". Dover. pp. 25–26, (2008).
[7] Morari, Manfred; Zafiriou, Evanghelos, "Robust Process Control. Prentice Hall". ISBN 0-13-782153-0, (1989).
[8] Sukvichai, K., "The Application Of Nonlinear Model Reference PID Controller For a Planar Robot Arm", pp. 637–640, (2008).
[9] M.W.Spong and M.Vidyasagar, "Robot Dynamics and Control", Canada. John Wiley Sons Inc, (1989).
[10]M.K. Refal, "Microprocessor based digital controllers for d.c motor speed control", IEEE Transactions on control systems, Vol, 10 Dec. 86.
[11]Koivo H N, Tanttu J T., "Tuning of PID controllers: survey of SISO and MIMO techniques[C]//Proceedings of the IFAC Intelligent Tuning and Adaptive Control Symposium", pp. 75-80, (1991).
[12]Apkarian, P., and Adams, R., "Advanced gain-scheduling techniques for uncertain systems", IEEE Transactions on Control Systems Technology, 6(1), pp. 21-32, (1998).
[13]Zadeh, L. A., "Fuzzy Sets", Information and Control, Vol. 8, pp. 335-353, 1965vol. 148, No. I, pp. 1-8, (2001).
[14]Nil, M., Yuzgec, U., Sonmez, M., and Cakir, B., "Fuzzy neural network based intelligent controller for 3-DOF robot manipulators", Proceedings of 5th International Symposium on Intelligent Manufacturing Systems, pp.884-895, (2006).
[15]Abdullah I. Al-Odienat, Ayman A. Al-Lawama, "The Advantages of PID Fuzzy Controllers Over the Conventional Types", American Journal of Applied Sciences 5 (6), ISSN 1546-9239, pp. 653-658, (2008).
[16]Mamdani, EH., "Applications of fuzzy logic to approximate reasoning using linguistic systems", J IEEE Trans. Comput., pp. 1182-1191, (1977).