



COMPARATIVE STUDY OF DIFFERENT CONTROL TECHNIQUES APPLIED ON 6 DOF ROBOT ARM

Samer Al-Marzoqi¹, Ahmad El-Bayoumy Mansour²

¹Research department faculty of engineering Al-Azhar University

²Department of electrical engineering, Al-Azhar University

ABSTRACT:

Most of the industrial controllers used today for 6 DOF robot arm utilize PID controllers. Using PID controller without tuning method leads to response with unsuitable time-domain specifications (maximum-overshoot, settling-time, rise-time, ...). Ziegler-Nichols method was introduced as the conventional method for tuning the PID controller. By using this method, the enhancement of the response time-domain specifications is not sufficient. Recently, fuzzy logic with and without PID-controller, and genetic algorithm with PID-controller are also introduced to obtain suitable time-domain specifications of the response. This paper will introduce a new comparative study for these four tuning methods applied on 6 DOF robot arm in view of the response time domain specifications and the steady state error. Simulation results are obtained using the Mat lap/Simulink. The results show that the Fuzzy with PID controller responds better than the others in case of change in load and system disturbances.

Keywords: Robot Arm; Forward kinematics; Inverse Kinematics; PID controller; Fuzzy controller; Genetic Algorithm and Fuzzy PID controller.

1. INTRODUCTION

In recent years, industrial and commercial systems take the advantages of robot technology. Concentrated on control of robotic systems, large number of control researches and numerous control applications were presented during the last years [1]. Robot manipulator is one of the interested fields in industrial, educational and medical applications. It works in unpredictable, hazard and inhospitable circumstances which human cannot reach. For example, working in chemical or nuclear reactors is very dangerous, while when a robot is used instead of human it involves no risk to human life. Therefore, modeling and analysis of the robot manipulators and applying control techniques are very important before using them in these circumstances to work with high accuracy. The essential problem is to study the robot manipulator problem from two sides. The first one is the mathematical modeling of the manipulator and the actuators, which includes an analysis for the forward kinematic, the inverse kinematic and the modeling of the dc- motor [2, 3, 4] because it is an important issue in a robot manipulator. The second problem is the control of the robot manipulator. The motivation of the control technique is the usage of the high precision performance of the robot manipulators in complicated and hazardous environments. Various controllers have been designed and applied in the robot manipulator. Designs of various conventional controllers applied on some simple systems have been introduced in [5]. Due to its simplicity of designing and implementation, the conventional PID-controller may be the most widely used in the industrial and commercial applications for the early decades [6]. However, robot manipulator is classified as nonlinear system. Hence, most of the industrial controllers used today for robot arm utilize PID controllers together with tuning method to can obtain response with suitable time-domain specifications. Ziegler-Nichols method was introduced as the conventional method for tuning

the PID controller, but the enhancement of the response time-domain specifications by using this method is not sufficient. Recently, fuzzy logic with PID-controller [21] or without PID-controller [7, 8] and genetic algorithm with PID-controller [9] are also introduced to obtain suitable time-domain specifications of the response. This paper deals with controlling 6 DOF robot arm such that the arm can follow any predefined path. The paper will introduce a new comparative study for these four tuning methods applied on 6 DOF robot arm in view of the response time domain specifications and the steady state error. The results show that the Fuzzy with PID controller is the best method in view of the time domain specifications of the response in case of change in load and system disturbances. The paper is organized as follows: chapter 2 describes the forward kinematic model and the inverse kinematic model is presented in chapter 3. Chapter 4 presents the mathematical model and the transfer function of DC motor. Chapter 5 describes the structure of PID controller. Chapter 6 covers the different tuning control techniques (Ziegler- Nichols Method, Fuzzy Controller without PID-controller, Fuzzy-PID Controller and Genetic-PID controller). Chapter 7 presents the results and chapter 8 introduces the conclusion of the paper.

2. FORWARD KINEMATIC MODEL

The forward kinematic model is to determine the position and orientation of the end-effector, given the values for the joint variables of the robot. The joint variables are the angles between the links in the case of revolute or rotational joints, and the link extension in the case of prismatic or sliding joints while it is possible to carry out all of the analysis using an arbitrary frame attached to each link. In this convention, each homogeneous transformation is represented as a product of six basic transformations. The following equation explains the forward kinematic problem [10, 11].

$A(\theta_1, \theta_2, \dots) = [x, y, z, R_d]$ where θ_1 , and θ_n are the input variables, $[x, y, z]$ are the desired position and R_d is the desired rotation.

$$A(\theta) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R(\theta, v) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -t \\ 0 & 1 \end{bmatrix}$$

Where:

v is the direction vector.

t is the position vector of point on the line.
so the rotations about this line are:

$$\begin{bmatrix} R(\theta, v) & (1-R)t \\ 0 & 1 \end{bmatrix}$$

The axes of the joints in their home position shown in Figure 1 is given by table 1.

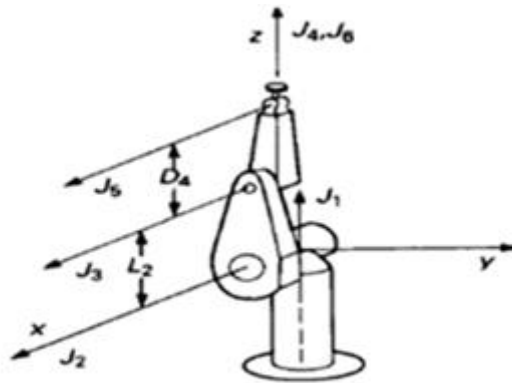


Figure 1: Frame assignment for the 6DOF Robot Arm

Table 1. The axes of the joints in their home position

Joint	V	T
J1	K	0
J2	I	0
J3	I	$L_2 K$
J4	K	$D_3 i$
J5	I	$(L_2 + D_4)K$
J6	K	$D_3 i$

$$\begin{aligned}
 A_1(\theta_1) &= \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_2(\theta_2) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_2 & -\sin\theta_2 & 0 \\ 0 & \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 A_3(\theta_3) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 & L_2 \sin\theta_3 \\ 0 & \sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_4(\theta_4) &= \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & (1-\cos\theta_4)D_3 \\ \sin\theta_4 & \cos\theta_4 & 0 & -\sin\theta_4 D_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 A_5(\theta_5) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_5 & -\sin\theta_5 & (L_2 + D_4)\sin\theta_5 \\ 0 & \sin\theta_5 & \cos\theta_5 & (L_2 + D_4)(1-\cos\theta_5) \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_6(\theta_6) &= \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & (1-\cos\theta_6)D_3 \\ \sin\theta_6 & \cos\theta_6 & 0 & -\sin\theta_6 D_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

So the kinematic matrix is:

$$K(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) = A_1(\theta_1)A_2(\theta_2)A_3(\theta_3)A_4(\theta_4)A_5(\theta_5)A_6(\theta_6) \quad [10]$$

3. Inverse Kinematic Model

The inverse kinematic is to find the joint variables of the robot manipulator for a given position and orientation of the end effectors. The problem of the inverse kinematics (IK) is more difficult than the forward kinematics problem [11,12,19]. It can be mathematically expressed as:

$\theta_k = f_k(x, y, z, \alpha, \gamma, \phi)$, where $k=1 \dots I$, is the joint angles and $(x, y, z, \alpha, \gamma, \phi)$ represents the position and orientation. The inverse kinematics can be solved for robot manipulator according to the following steps:

1. Equate the general transformation matrix to the final transformation matrix of the robot manipulator[11].

$$H_i^0 = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} & x \\ \gamma_{21} & \gamma_{22} & \gamma_{23} & y \\ \gamma_{31} & \gamma_{32} & \gamma_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_1 A_2 \dots A_i \quad (1)$$

2. For the both matrices define:

- a) The elements that contain one joint variable.
 - b) Pairs of elements, which contain only one joint variable.
 - c) Elements, or combinations of elements, contain more than one joint variable.
3. After defining these elements, equate it to the corresponding elements in the other matrix to form equations, and then solve these equations to find the values of joint variables.
 4. Repeat step (3) to identify all elements in the two matrices.
 5. In the case of inaccuracy, solutions look for another one.
 6. If there is more joint variable to be found, multiply equation (1) by the inverse of a matrix for the specified links.
 7. Repeat steps (2) through (6) until solution to all joint variables have been found.
 8. If there is no solution to the joint variable in terms of an element transformation matrix, it means that the arm cannot achieve the specified position and orientation; the position is outside the robot manipulator workspace.

4. DC motor model

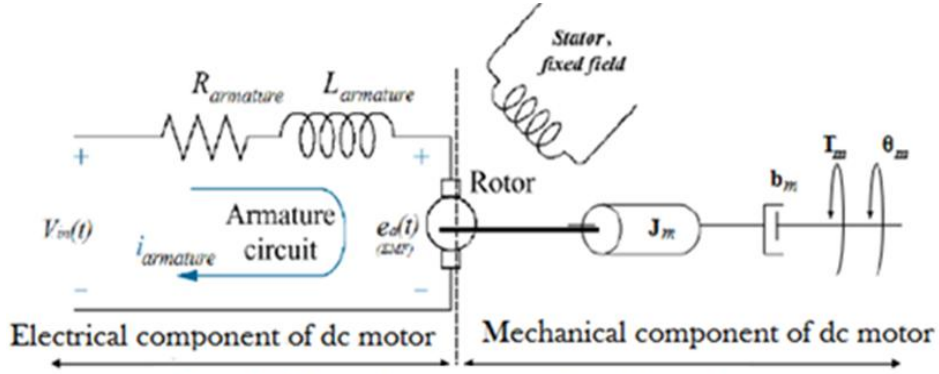


Figure 2 Equivalent representation of dc-motor

In armature control of separately excited DC motors, the voltage applied to the armature of the motor is adjusted without changing the voltage applied to the field. Figure 2 shows a separately excited DC motor equivalent model. The relation between the rotor shaft speed $W(s)$ and the applied armature voltage $V_a(s)$ is represented by the transfer function:

$$\frac{W(s)}{V_a(s)} = \frac{K_T}{[L_a \cdot J_m \cdot s^2 + (R_a \cdot J_m + L_a \cdot B_m)s + [(R_a) \cdot B_m + k_b K_T]]} \quad (2)$$

Since the relation between the position and speed is:

$$\theta(s) = \frac{1}{s} \cdot W(s) \quad (3)$$

Thus the transfer function between the shaft position $\theta(s)$ and the armature voltage at no load is [5]:

$$\frac{\theta(s)}{V_a(s)} = \quad (4)$$

Figure 3. Shows the DC motor model built in Simulink. Input ports are armature voltage (V_a) and load torque (T_{load}) and the output ports are angular speed in (w) and position ($teta$). For DC motor with the following parameters: Moment of inertia $J_m = 0.000052$ Kg.m², Friction coefficient $B_m = 0.01$ N.ms, Back EMF constant $k_b = 0.235$ V/ms⁻¹, Torque constant $K_T = 0.235$ Nm/A, Electric resistance $R_a = 2$ ohm and Electric inductance $L_a = 0.23$ H, the transfer function of the DC motor of the first DOF considered is defined as:

$$\frac{\theta(s)}{V_a(s)} = \frac{39652}{s^2 + 120s^2 + 5380s} \quad (5)$$

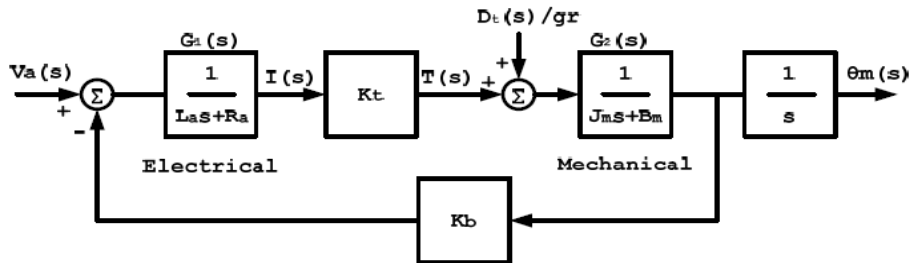


Figure 3: DC Motor Block Diagram Model

5. PID CONTROLLER

PID controllers are widely used in industrial control applications due to their simple structures, comprehensible control algorithms and low costs. Figure 4 shows the schematic model of a control system with a PID controller. The mathematical model of the PID-controller with $e(t)$ as input and $u(t)$ as output is written as:

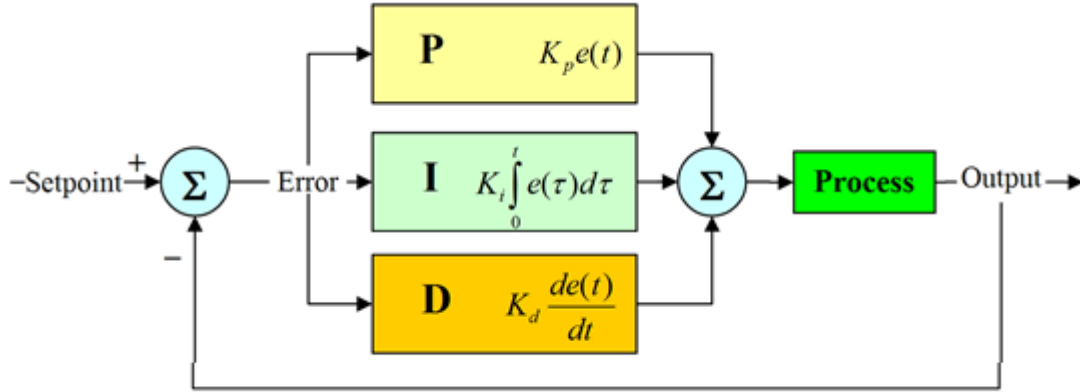


Figure 4: PID controller structure

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (6)$$

or

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int e(t) dt + K_p T_D \frac{de(t)}{dt} \quad (7)$$

Where:

K_p = Proportional gain

K_i = integral gain

K_D = derivative gain

T_i = integral time

T_D = derivative time

If the controller is digital, then the derivative term may be replaced with a backward difference and the integral term may be replaced with a sum. For a small constant sampling time T_s equation (7) can be approximated for the digital controller as [5]:

$$u(n) = K_p \left(e(n) + \frac{1}{T_i} \sum_{j=1}^n e(j) T_s + T_D \frac{e(n) - e(n-1)}{T_s} \right) \quad (8)$$

6. TUNING METHODS

Position of DC motor can be controlled by using one of following different tuning methods:-

6.1 PID Controller (Ziegler- Nichols Method).

6.2 Genetic PID controller.

6.3 Fuzzy Controller.

6.4 Fuzzy-PID Controller.

6.1 Ziegler-Nichols Method

PID controllers are usually tuned using Ziegler- Nichols frequency method. In this method $T_i = \infty$ and $T_d = 0$ is set initially and then using the proportional action only, the value of K_p is increased from 0 to ultimate value K_u slowly until the process start to oscillate. The gain when this occurs is K_u and the period of the oscillation is T_u . The advantage of this method of being very easy to use. Ziegler-Nichols (Z-N) suggested that the value of K_p , T_i and T_d can be found out by using the formula shown in Table3[13, 22, 23].

Table 3: Ziegler-Nichols rule

Controller	K_P	T_I	T_D
P	$0.5 K_u$		
PI	$0.45 K_u$	$T_u / 1.2$	
PID	$0.6 K_u$	$T_u / 2$	$T_u / 8$

6.2 Tuning of PID controller using genetic algorithm approach

Genetic Algorithms (GAs) are a stochastic global search method that mimics the process of natural evolution. It is one of the methods used for optimization. The computational systems keep on improving and this makes GA useful for optimization. The genetic algorithm starts with no knowledge of the correct solution and depends entirely on responses from its environment and evolution operators such as reproduction, crossover and mutation to arrive at the best solution. GA starts with an initial chromosome & check for the fitness value. The fittest chromosomes are taken as parents further they are reproduced, crossed over & mutated. The offspring is checked for the value of the fitness. The implementation of the tuning procedure through genetic algorithms starts with the definition of the chromosome representation. The gains of K_P , K_I and K_D are real numbers and characterize the individual to be evaluated. [14, 18] GA can be applied to the tuning of PID position controller gains to ensure optimal control performance at nominal operating conditions. The block diagram for the entire system is given in Figure 5 and also the genetic algorithm parameters chosen for the tuning purpose are shown Table 4.

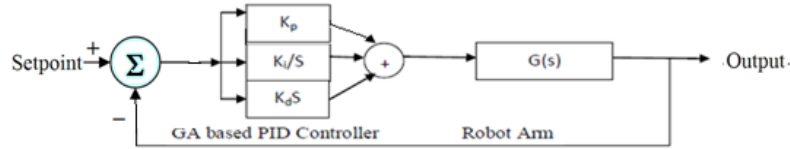


Figure 5: Block Diagram of the Entire System

Fitness Function (based on model reference approach):

Minimize J

Where:

$$J = \frac{\sum [(y_m - y_s)^2]}{M_{ax}t} \tag{9}$$

y_s system response.

y_m model response.

$M_{ax}t = 500$ Sample

$$y_m = \begin{cases} \frac{t}{[(N)_m - 1]} & \text{for } 0 \leq t \leq N_m \\ 1 & \text{for } t \geq N_m \end{cases} \tag{10}$$

$N_m = 21, 26, 31$ and 36 Sample

Note .Each 100 sample equal 1sec (sampling time =0.01sec).

Table 4: Parameters of GA

GA property	Value/Method
Population Size	80
Maximum number of Generations	100
Performance Index/Fitness function	Mean Square Error
Selection Method	Normalized Geometric Selection
Probability of Selection	0.05
Crossover Method	Scattering
Mutation Method	Uniform Mutation
Mutation Probability	0.01

6.3 Fuzzy Logic Controller:

Fuzzy logic is expressed by means of the human language. Based on fuzzy logic, a fuzzy controller converts a linguistic control strategy into an automatic control strategy, and fuzzy rules are constructed by expert experience or knowledge database. First, set the error $e(t)$ and the error variation $\Delta e(t)$ of the angular velocity to be the input variables of the fuzzy logic controller. The control voltage $u(t)$ is the output variable of the fuzzy logic controller. The linguistic variables are defined as {NB, NM, ZR, PM, PB}, where NB means negative big, NS means negative medium, ZR means zero, PM means positive medium and PB means positive big [15, 16] and [17, 24, 25]. The fuzzy rules are summarized in Table 5. The fuzzy inference mechanism in this study follows as:

Table 5: Fuzzy Rules

$\Delta e/e$	NB	NM	ZR	PM	PB
NB	NB	NB	NM	NM	ZR
NM	NB	NM	NM	ZR	PM
ZR	NM	NM	ZR	PM	PM
PM	NM	ZR	PM	PM	PB
PB	ZR	PM	PM	PB	PB

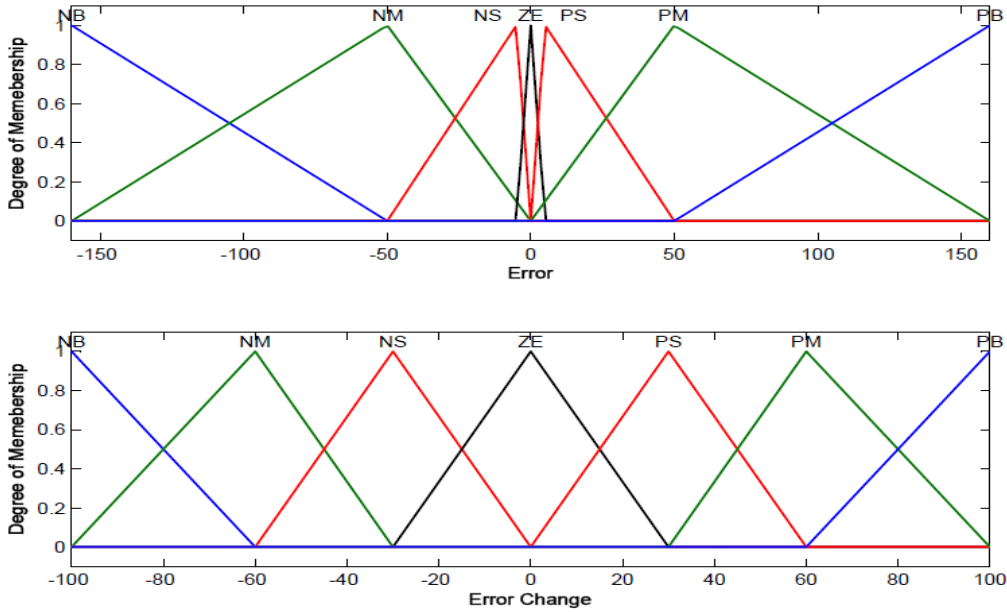


Figure 6: Membership function for inputs e and Δe

Here max-min type decomposition is used and the final output for system is calculated by using center of area gravity method.

$$\mu_B(u(t)) = \text{MAX} [\mu_{A1j}(e(t)),]$$

Where:

$\mu_{A1j}(e(t))$ = membership function of $e(t)$.

$\mu_{A2j}(\Delta e(t))$ = membership function of $\Delta e(t)$.

$\mu_Bj(u(t))$ = membership function of $u(t)$.

j is an index of every membership function of fuzzy set, m is the number of rules and is the inference result. Fuzzy output $u(t)$ shown in Figure 7. Can be calculated by the center of gravity defuzzification as:

$$u(t) = \frac{\sum_{i=1}^m \mu_B(u(t)) \cdot u_i}{\sum_{i=1}^m \mu_B(u_i(t))} \tag{11}$$

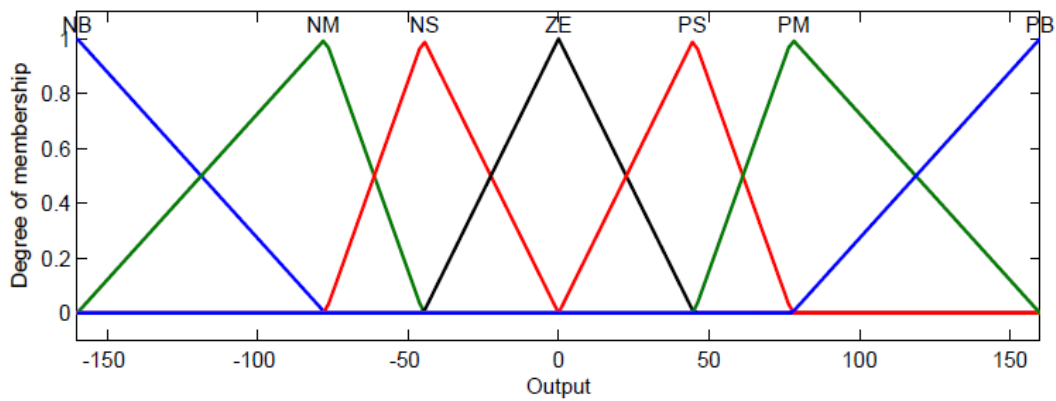


Figure 7: Membership function for output u

NB stands for negative big with numerical equivalent of 0, NM stands for negative medium with numerical equivalent value of 0.25, ZE stands for zero with numerical equivalent value of 0.5, PM stands for positive medium with a numerical equivalent of 0.75, and PB stands for positive big with numerical equivalent of 1.

6.4 FUZZY PID CONTROL

Fuzzy-PID (FPID) provides good performance as compared to standalone Fuzzy. Simple rule base are used for Fuzzy controller while FPID uses different rule base for proportional, integral and derivative gains to make response faster [12]. Fuzzy-PID controllers are arranged into two sorts: the direct action fuzzy control and the fuzzy supervisory control. The direct action sort replaces the PID control with a feedback control loop to compute the action through fuzzy reasoning where the control actions are resolved directly by means of a fuzzy inference. These sorts of fuzzy controllers are also called PID-like controllers [19, 21]. On the other hand, the fuzzy PID type attempts to give nonlinear action for the controller output utilizing fuzzy reasoning where the PID gains are tuned based on a fuzzy inference system rather than the traditional methodologies. The design process of the fuzzy controller is described as follows:

- Define the input and output variables of FLC. In this work, there are two inputs of FLC, the error $e(t)$ and its rate of change of error $\frac{de}{dt}$ and three outputs, and K_d are respectively as shown in Figure 8.

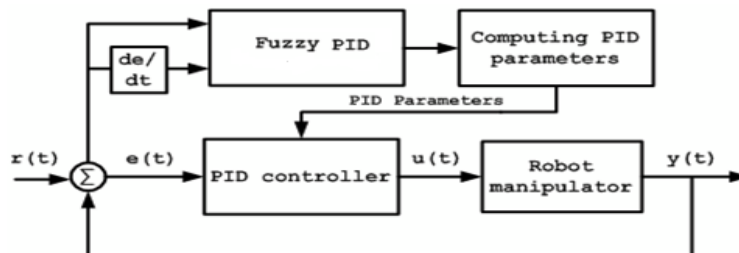


Figure 8: Fuzzy PID control structure

- Fuzzify the input and output variables by defining the fuzzy sets and membership functions. and the output of FLC has the following membership function as shown in Figure 9 for the three outputs K_p , and K_d .
- Outline the inference mechanism rule to get the input-output relation. This work utilized Mamdani (max-min) inference mechanism as shown in Figure 10.
- Defuzzify the output variable. Here, the center of gravity (COG) method, the most frequently utilized method, is utilized. The control activity is [20]:

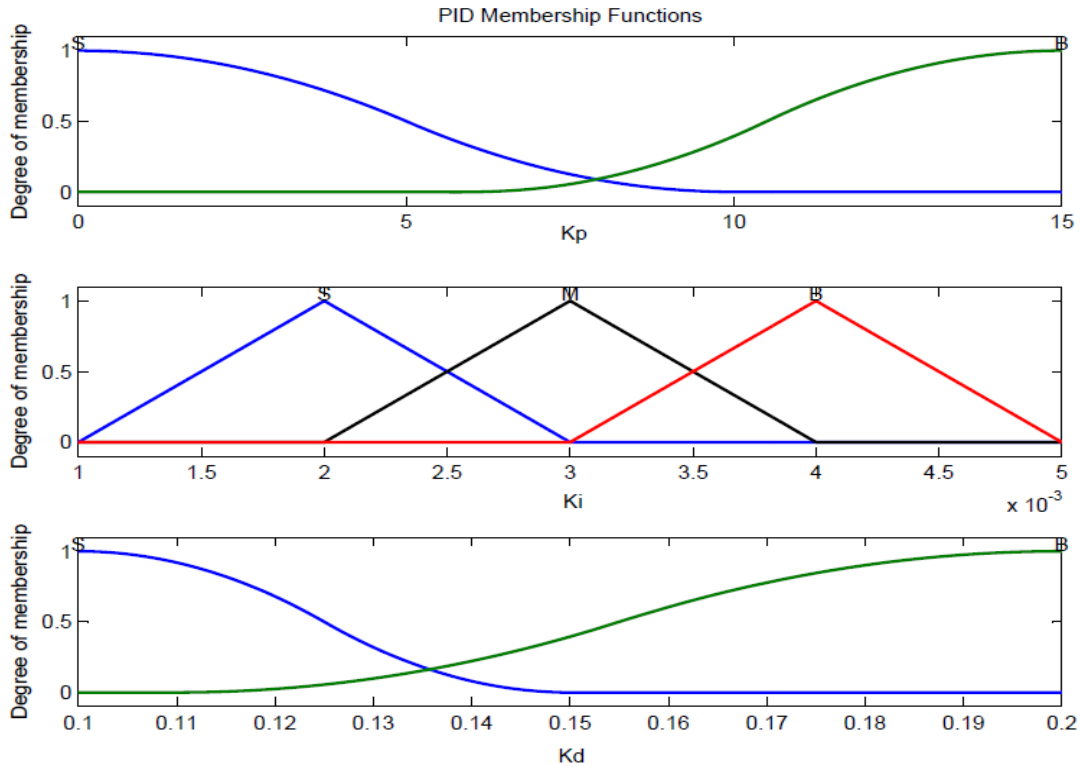


Figure 9: PID parameters membership functions (K_p , and K_d)

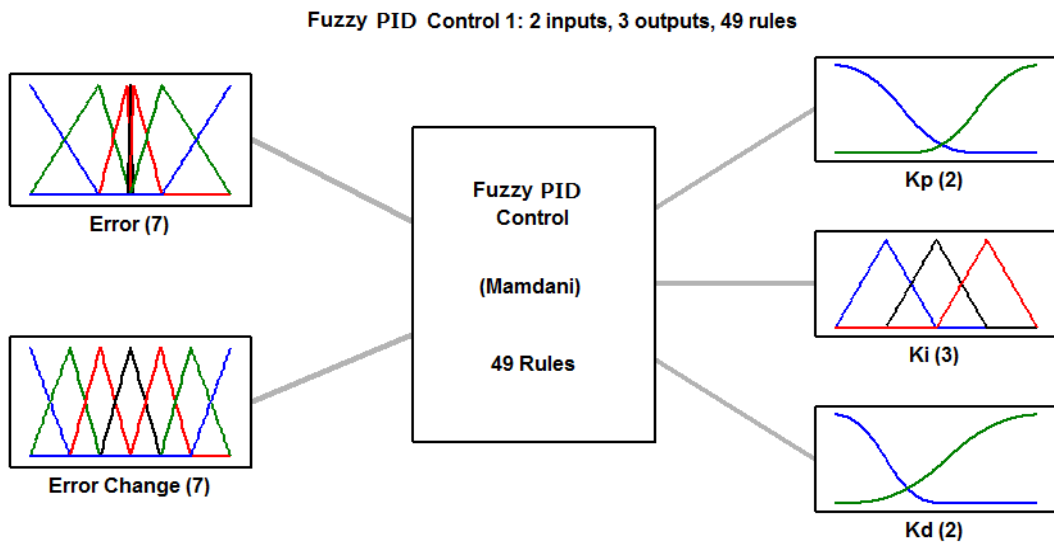


Figure 10: Fuzzy inference block

7. Result AND Discussion

Simulation result in Figure 11 and Figure 12 shows the output response of the proposed controllers using the step as input signal. The tow figures show the performance of using the different control approach. In addition, they show the effectiveness of the Fuzzy PID controller for rejection disturbance. If the load torque with $-0.7 N.m$ is applied on the desired angle. The obtained result shows the effect of the disturbance on the output response after 0.5 second and the efficacy of the Fuzzy PID controller for tuning PID parameters and eliminating the disturbance.

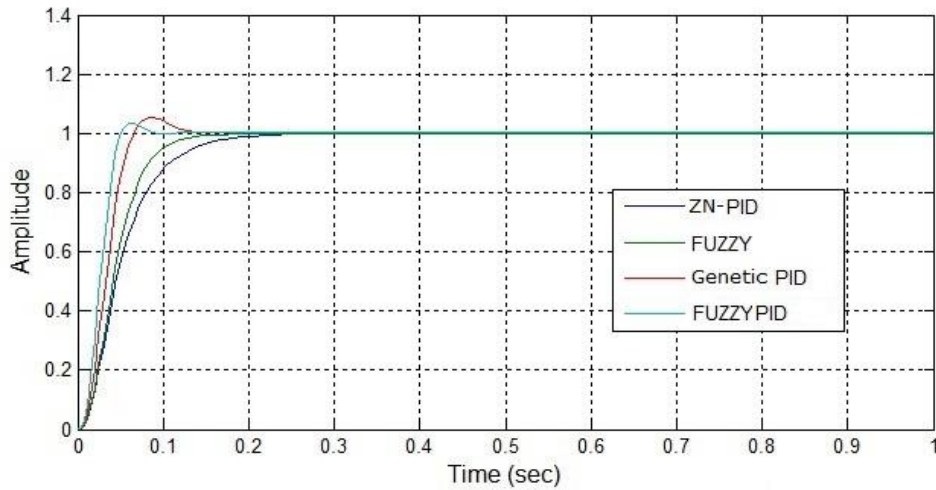


Figure 11: Output responses for step input using proposed controllers without disturbance

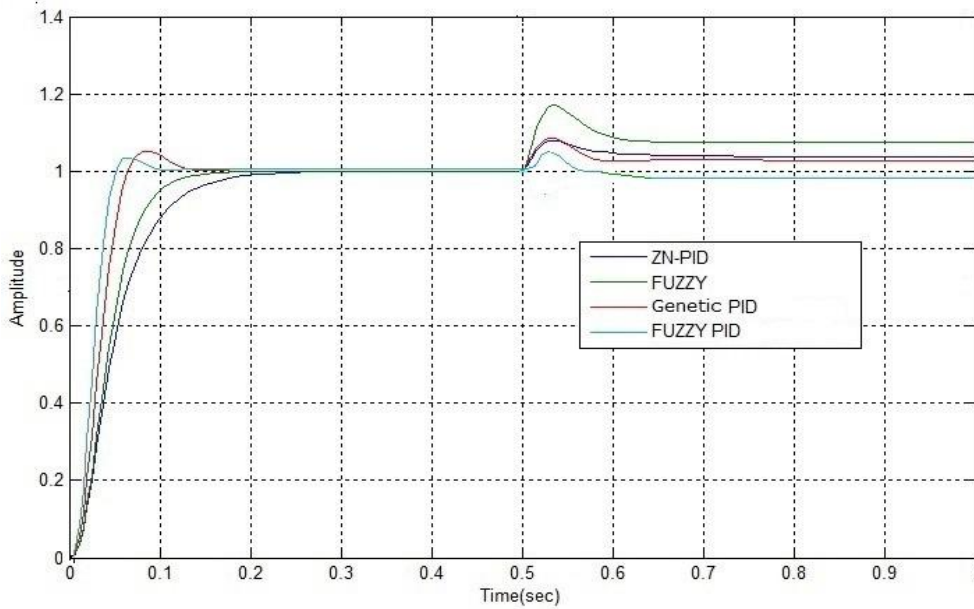


Figure 12: Output responses for step input using proposed controllers with disturbance

Clearly, the fuzzy PID control achieved better performance than other tuning methods in terms of time response. The above figures show the effect of small disturbance after 0.5 second and effectiveness of the fuzzy PID control in eliminating the presence of disturbances. Performance of proposed controllers is summarized in Table 9.

Table 9: Time-Domain Specifications Parameter Result

Steady state error (SSE)	Time-Domain Specifications Parameter Characteristic			Controller type
	Settling time $t_s(s)$ sec	Rising Time $t_r(s)$ sec	Overshoot (OS%)	
0.009	0.25	0.0905	-	PID controller(Ziegler-Nichols)
0.007	0.16	0.068	-	Fuzzy logic control
0.005	0.13	0.041	0.05	Genetic PID control
0.001	0.08	0.028	0.02	Fuzzy PID control

8. CONCLUSION

Robot manipulators have become increasingly important in the field of flexible automation. There are so many Controllers to control the actuator accurately and robustly. In this paper, DC linear servo motor is used as actuator. Applying a control technique is important to guarantee high efficiency and lower error for the motion of the robot. This paper has introduced a new comparative study for four controlling methods (Ziegler-Nichols method, fuzzy logic PID-controller with and without PID-controller, and genetic algorithm with PID-controller) applied on 6 DOF robot arm in view of the response time domain specifications and the steady state error. The 6 DOF robot arm is considered such that the arm can follow any predefined path. Simulation results are obtained using the Mat lap/Simulink. The results show that the Fuzzy PID Controller can achieve better accuracy and has less or no deviation from the trajectory compared to the other controllers.

REFERENCES

- [1] Jong-Hwan Kim, Hyun Myung, "Robot Intelligence Technology and Applications," 4th Edition, 2016.
- [2] Y. Yang, W. Wang, D.-J. Yu and G. Ding, "A Fuzzy Parameters Adaptive PID Controller Design of Digital Positional Servo System," *IEEE Proceeding of the First International Conference on Machine Learning and Cybernetics.*, Vol. 1, pp. 310-314, Nov. 2015
- [3] Asan M. K., Saravanakumar. G., Valamarthi. K., Devaraj. D., Radhakrishnan. T. K., Real-Coded Genetic Algorithm for System Identification and Tuning of a Modified Model Reference Adaptive Controller for a Hybrid Tank System, *Applied Mathematical Modelling*, 37, pp. 3829-3847, 2015.
- [4] Wong, C. C., Lin, B. C., Lee, S. A. and Tsai, C. H., "GA-Based Fuzzy System Design in FPGA for an Omni-Directional Mobile Robot," *Journal of Intelligent & Robotic Systems*, Vol. 44, pp. 327_347 (2015).
- [5] K. Ogata "Modern Control Engineering, 4th Edition", Dorling Kindersley Pvt. Ltd., India (2009).
- [6] Y. Yang, W. Wang, D.-J. Yu and G. Ding, "A Fuzzy Parameters Adaptive PID Controller Design of Digital Positional Servo System," *IEEE Proceeding of the First International Conference on Machine Learning and Cybernetics.* Vol. 1, pp. 310-314, Nov. 2002.
- [7] M.W. Spong, S. Hutchinson and M. Vidyasagar, *Robot Modeling and Control*, 1st Edition, Jon Wiley & Sons, Inc, 2005.
- [8] J.J. Crage, *Introduction to Robotics Mechanics and Control*, 3rd Edition, Prentice Hall, 2005.
- [9] K. H. Kalil, *Nonlinear Systems*, 3rd Edition, Prentice-Hall, 2002.
- [10] Indranil Pan, Saptarshi Das and Amitava Gupta, "Tuning of an optimal fuzzy PID controller with stochastic algorithms for networked control systems with random time delay", *ISA Transactions*, Volume 50, Issue 1, pp. 28-36, January 2011.
- [11] Baogang Hu, George K. I. Mann and Raymond G. Gosine, "New methodology for analytical and optimal design of fuzzy PID controllers", *IEEE Transactions on Fuzzy Systems*, Volume 7, Issue 5, pp. 521-539, October 1999.
- [12] Zhi-Wei Woo, Hung-Yuan Chung and Jin-Jye Lin, "A PID type fuzzy controller with self-tuning scaling factors", *Fuzzy Sets and systems*, Volume 115, Issue 2, pp. 321-326, October 2000.
- [13] Q.Wang, P Spronck and R Tracht, "An Overview Of Genetic Algorithms Applied To Control Engineering Problems", *Proceedings of the Second International Conference on Machine Learning And Cybernetics*, 2003
- [14] P. I. Corke, "A simple and systematic approach to assigning Denavit-Hartenberg parameters," *IEEE Transactions on Robotics*, vol. 23, pp. 590-594, June 2007.

- [15] R. P. Paul and H. Zhang, "Computationally efficient kinematics for manipulators with spherical wrists," *Int. J. Robot. Res.*, vol. 5, pp. 32-44, 2012
- [16] Jolly Shah, S.S.Rattan, B.C.Nakra, Kinematic Analysis of 2-DOF Planer Robot Using Artificial Neural Network, World Academy of Science, Engineering and Technology, Vol. 81, pp. 282-285, 2011
- [17] S. A. Mazhari and S. Kumar, "PUMA 560 Optimal Trajectory Control using Genetic Algorithm, Simulated Annealing and Generalized Pattern Search Techniques," *International Journal of Electrical, Computer and Systems Engineering*, Vol. 2, No. 1, pp. 71-80, March 2008.
- [18] Atul Kumar Dewangan, Sashai Shukla, Vinod Yadu "Speed Control of a Separately Excited DC Motor Using Fuzzy Logic Control Based on Matlab Simulation Program" *International Journal of Scientific & Technology Research* Volume 1, Issue 2, ISSN 2277-8616 pp. 52 – 54 , March 2012.
- [19] L. Wang, M. Tian and Y. Gao, "Fuzzy Self-adapting PID Control of PMSM Servo System," *IEEE International Electric Machines & Drives Conference*. Vol. 1, pp. 860–863, May 2007.
- [20] A new method for position control of a 2-DOF robot arm using neuro – fuzzy controller, *Indian Journal of Science and Technology*, Vol. 51, pp. 2253-2257, 2012
- [21] M. D. Youns, S. M. Attya and A. I. Abdulla, "Position Control of Robot Arm Using Genetic Algorithm Based PID Controller", *Al-Rafidain Engineering*, Mosul, Iraq, Vol.21, No. 6, pp. 19-30, December 2013.
- [22] S. Yadegar an Azura binti Che 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.
- [23] Ch. R. Kumar, K. R. Sudha, D. V. Pushpalatha and Ch. V. N. Raja , "Fuzzy C-Means Controller for a PUMA-560 Robot Manipulator", *IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions*, IIT Kanpur, India, pp. 57-62 , July 2013.
- [24] Manafeddin Namazov, "DC motor position control using fuzzy proportional-derivative controllers with different Defuzzification methods" *An Official Journal of Turkish Fuzzy Systems Association* Vol.1, No.1, pp. 36-54, July 2010.
- [25] Y. Li, K. Ang, G. Chong. PID Control System Analysis and Design: Problems, Remedies and Future Directions. *IEEE Control Systems Magazine* Vol.1, No.1, pp. 32–41, February 2006.
- [26] J. Han, "From PID to active disturbance rejection control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 900–906, March 2009.
- [27] Mendel, J. M., Fuzzy logic systems for engineering: A tutorial. *Proceedings of the IEEE*, vol. 83, no. 3, pp. 345–377, March 1995.
- [28] A. Costa, A. De Gloria, F. Giudici, and N. Olivieri, "Fuzzy logic microcontroller," *IEEE Micro*, vol. 17, no. 1, pp. 66-74, November 1997.