Advances in Dynamical Systems and Applications.
ISSN 0973-5321, Volume 16, Number 2, (2021) pp. 1585-1597

https://www.ripublication.com/adsa.htm

# Comparison of PID & Fuzzy Controller for DC Motor Speed Control using SIMULINK

**Beza Negash Getu**

*Department of Electrical, Electronics and Communications Engineering (EECE), American University of Ras Al Khaimah, Ras Al Khaimah, UAE.*

**ABSTRACT**

Process control is known to be essential in many manufacturing and industrial applications. In this paper, the performance of the conventional PID based controller is compared with the Fuzzy logic controller for controlling the speed of a DC motor. Using MATLAB/SIMULINK simulations, it is demonstrated that fuzzy logic controller achieves more robust controlling of the speed of DC motor with zero overshoot but at higher rise and settling time compared with the conventional PID controller. The time response of the Fuzzy controller can be improved by tuning the membership functions and the fuzzy rules.

**Keywords-** PID, DC motor speed, fuzzy logic controller, membership functions.

## I. INTRODUCTION

Process or system controlling is an essential part in many industrial and manufacturing sectors. Among others, speed controlling of DC motors is important as DC motors are used in many applications like water pumps, fans, robots and some industrial machines. They are easy to use, reliable and cheap to buy but only have one poor attribute, which is the lack of good efficiency. A proportional-integral-derivative (PID) controller is used in order to overcome this problem and make it more efficient to use. The PID control is a closed loop feedback system used to reduce or eliminate the error between the measured or the actual speed and the desired speed in order to achieve an output close to the desired speed. It is a well-developed algorithm that pushes a system or a process towards a target set level. PID controllers are used in most practical control systems ranging from consumer electronics such as cameras to industrial processes that use extensively various motors, robotic applications and chemical processes [1]-[3]. Often a PID controller design involves to start the open-loop response and determine what needs to be improved in the closed response in the presence of the controller in step-by-step basis. In [4]-[5], the Author demonstrated a PID based water level and temperature controlling using MATLAB based simulation. It is shown that suitable values for proportional ($K_p$), integral ($K_i$) and derivative ($K_d$) constant parameters can be adjusted for the PID controller by tuning to maintain the desired water level in the tank or temperature for a heater or furnace system. Due to limitations of PID controller in accurate controlling of non-linear type processes, researchers are proposing a Fuzzy Logic Controller (FLC) as alternative design methodology that can be applied in both linear and non-linear systems [6]-[8]. Fuzzy controller is based on fuzzy concept that was originated by Lotfi Zadeh, a computer scientist at the University of California [9]. As opposed to the modern control theory, FLC design is not based on the mathematical model of the process rather it is based on the skilled human operator for the process without much knowledge of the underlying dynamics in the system. The FLC is a form of a systematic reasoning that can be integrated into automation systems with classical human reasoning based on certain rules applied to the control variables from the knowledge of the system. Fuzzy logic controller based system is a nonlinear mapping between input and output variables and are utilized for inferring complex nonlinear systems. The design of proficient FLC system is governed by several design parameters that include controller architecture, fuzzification method, membership function formulation, rule base acquired from expert knowledge, inference engine and defuzzification method [10].

## II. SYSTEM MODEL

The armature controlled DC motor electric equivalent circuit and the free-body diagram of the rotor is shown in Fig. 1 [4]. A DC motor is one of the the most used actuators in control systems. The input to the motor is a voltage source ($V_a$) applied to the motor's armature, the output is the rotational speed of the shaft $\omega(t) = d\theta(t)/dt$ where $\theta(t)$ is the angular position of the rotor. We further assume a viscous friction model, that is, the friction torque is proportional to shaft angular velocity. The physical parameters of the motor are $J$: moment of inertia of the rotor [kg.m$^2$], $b$: motor viscous friction constant [in N.m.s], $K_b$: electromotive force constant [in V/rad/sec], $K_t$: motor torque constant [in N.m/Amp], $R_a$: electric resistance [in Ω]. $L_a$: electric inductance [in H]. Assuming armature controlled motor, the torque is proportional to the armature current ($T=K_t i_a$) and the back emf is proportional to the angular velocity of the shaft, $e_b=K_b\omega(t)$. Using the above facts and based on Newton's second law and Kirchhoff's voltage law, the DC motor equations relating the motor speed to the physical parameters of the motor can be written as:

$$J\ddot{\theta} + b\dot{\theta} = T = K_t i_a \tag{1}$$

$$L_a \frac{di_a}{dt} + R_a i_a = V_a - K_b \dot{\theta} \tag{2}$$

Where $\dot{\theta} = d\theta / dt = \omega$ is the angular speed of the motor. In the sequel, we use the single dot and double dot notations as an alternative representation of the first derivative and second derivative of the variable.

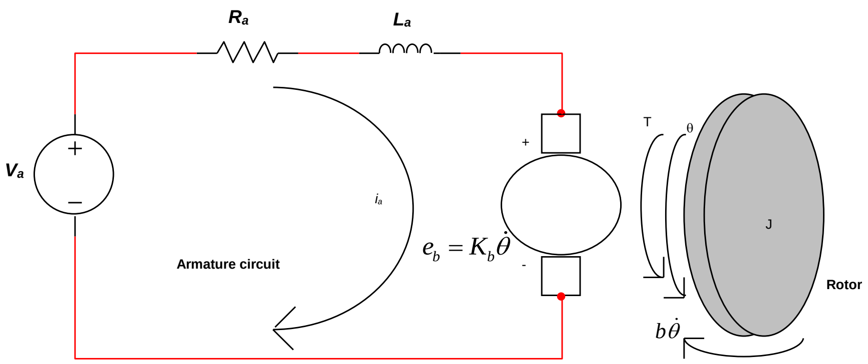


**Fig. 1.** Equivalent circuit of armature controlled DC motor

We can determine the state space model of the system by considering the armature current $i_a(t)= x_1(t)$ and the angular speed as $\omega(t)=x_2(t)$ as the two state variables of the system. The state variables (along with the input functions) used in equations describing the dynamics of a system, provide the future state of the system. Mathematically, the state of the system is described by a set of first-order differential equation in terms of state variables. The state space model takes the following form [11]:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax}+\mathbf{Bu}\\ \mathbf{y} &= \mathbf{Cx}+\mathbf{Du}\end{aligned} \tag{3}$$

Where $\mathbf{x}=[x_1 \quad x_2 \quad \cdots \quad x_n]^T$, $\dot{\mathbf{x}}=[\dot{x}_1 \quad \dot{x}_2 \quad \cdots \quad \dot{x}_n]^T$

and **A**, **B**, **C, D** are $n\times n$ , $n\times m,$ $k\times n$ , $k\times m$ size matrices respectively for a system with $n$ state variables and $k$ outputs; $\underline{u}$ is $m\times 1$ input vector. The matrix **C** is $1\times n$ and **D** is $1\times \underline{m}$ for single output system. The vector $u$ consists of the $m$ input variables to the system and $u$ is $1\times 1$ scalar variable if the system has single input, which is the same as the above DC motor model with $u$ equals to the armature voltage $V_a$.

Rearranging (1) and (2), we can have the two first order differential equations expressed in the form of the armature current and the angular speed of the motor.

$$\frac{di_a}{dt} = -\frac{R_a}{L_a}i_a - \frac{K_b}{L_a}\omega + \frac{V_a}{L_a} \tag{4}$$

$$\frac{d\omega}{dt} = \frac{K_t}{J}i_a - \frac{b}{J}\omega \tag{5}$$

Using (4) & (5) and considering the angular speed of the motor is the output of the system, the state-space model of the system is given as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -R_a/L_a & -K_b/L_a \\ K_t/J & -b/J \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1/L_a \\ 0 \end{bmatrix} u \tag{6}$$

$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7}$$

The state-space model matrices are given as follows:

$$\mathbf{A} = \begin{bmatrix} -R_a/L_a & -K_b/L_a \\ K_t/J & -b/J \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1/L_a \\ 0 \end{bmatrix}; \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}; \mathbf{D} = 0 \tag{8}$$

The transfer function of the system relating the rotational speed output, $\omega(s)$, to the armature voltage input, $V_a(s)$, can be derived by using the general state space to transfer function transformation equation given as [11]:

$$G(s) = \mathbf{C}\left(s\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B} + \mathbf{D} \tag{9}$$

For the armature controlled DC motor,

$$G(s) = \frac{\omega(s)}{V_a(s)} = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} s + R_a/L_a & K_b/L_a \\ -K_t/J & s + b/J \end{bmatrix}^{-1} \begin{bmatrix} 1/L_a \\ 0 \end{bmatrix} \tag{10}$$

Where $[.]^{-1}$ refers of the inverse of the matrix. After some steps, the motor rotational speed output, $\omega(s)$, is related to the armature voltage input, $V_a(s)$, as:

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{K_t / JL_a}{s^2 + s\frac{(JR_a + bL_a)}{JL_a} + \frac{(R_a b + K_t K_b)}{JL_a}} \tag{11}$$

The relation in (11) represents the open-loop input–output relation of the armature controlled DC motor shown in Fig. 1 without the controller.

## I. PID CONTROLLER

The PID controller is used to achieve desired set point (in our case rotation speed of the motor) using the proportional, integral and derivative terms as shown in Fig. 2 [11], [12]. PID controllers are probably the most widely used industrial controllers. Even complex industrial control systems may comprise a control network whose main control building block is a PID control module. The proportional gain ($K_p$) provides an overall control action proportional to the error signal, the integral gain ($K_i$) action is to reduce steady-state errors by an integrator and the derivative gain ($K_d$) improves transient response through a differentiator.

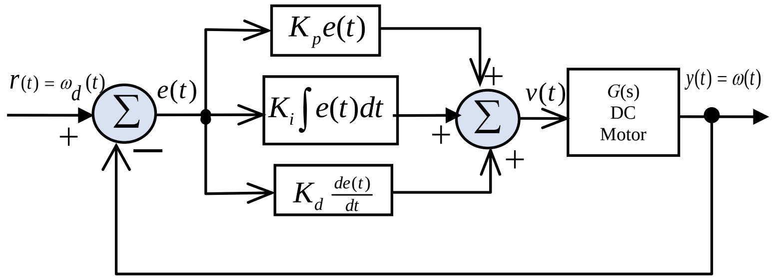


**Fig. 2.** Block diagram of PID controlled DC motor speed control system.

In general design of the PID, the error $e(t)$ is related to the output $v(t)$ as:

$$v(t) = K_p e(t) + K_i \int e(t) + K_d \frac{de(t)}{dt} \tag{12}$$

The error $e(t)$ is the difference between the desired reference input $r(t)$ and the output $y(t)$; in this model $e(t)=\omega_d(t)-\omega(t)$ where $\omega_d(t)$ is the desired rotational speed of the DC motor. Using the Laplace domain, the input-output relation for the controller is given as:

$$C(s) = \frac{V(s)}{e(s)} = K_p + \frac{K_i}{s} + sK_d \tag{13}$$

The transfer function of PID Controller block, $C(s)$ is implemented in SIMULINK as $C(s)=P+I/s+DsN/(s+N)$ where $P=K_p$, $I=K_i$, $D=K_d$ and $N$ is related to the order of the filter coefficients used for the low pass filter in the derivative term. Ideally, $N=\infty$ for pure derivative term that gives $C(s)=P+I/s+Ds$. However, pure derivative is not a good idea as it amplifies measurement noise, so a practical implementation should avoid pure derivatives and use a low pass filter with low order filter coefficients. Therefore, during the PID tuning in SIMULINK, the choice of $N$ affects the simulation results.

## II. FUZZY LOGIC CONTROLLER

Fuzzy logic controller is becoming one of the choices for controlling of many industrial applications. Fuzzy logic is expressed using human language. Based on fuzzy logic, a fuzzy controller converts a linguistic control strategy into a control strategy, and the fuzzy rules are constructed by the experience of a skilled human expert. As shown in Fig. 3, the error, $e(t)$, and the change of the error or the derivative of the error, $de(t)/dt$, are the variable inputs of the fuzzy logic controller. The variable output of the fuzzy logic controller adjusts and calculates the control output $v(t)$ based on the defuzzification process of the fuzzy inference system.

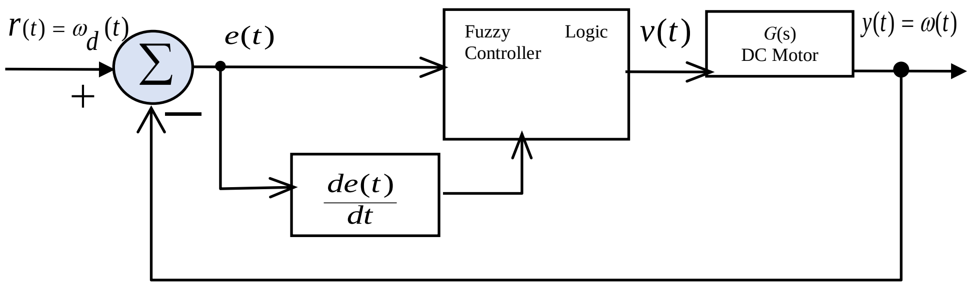


**Fig. 3.** Block diagram of FLC based DC motor speed control system.

In general, there are four principal components of the FLC [6]-[8]. They are:

- Fuzzification
- The rule base
- The inference Engine
- Defuzzification

The error and change of error crisp input variables will be associated to membership functions (triangular, trapezoidal, Gaussian, etcetera) during the fuzzification step. A membership function (MF) is a curve that defines how the value of a fuzzy input variable in a certain region is mapped to a membership value (or degree of membership) between 0 and 1. An input variable from a crisp set will be mapped to a fuzzy set with the specified membership function. The result of the fuzzification step will be an input to the fuzzy inference engine that works based on the rules entered in the controller. The Fuzzy Inference System (FIS) is the key unit of a fuzzy logic system having decision making as its primary work. It uses the “IF…THEN” rules along with connectors “OR” or “AND” for drawing essential decision rules. The IF side is known as the conditions and the THEN side is called the closure. In this work, the Mamdani type of inference system is used. The result obtained by fuzzy logic depends on fuzzy inference rules and fuzzy implication operators. The knowledge base provides the information needed for linguistic control rules and the information for fuzzification and defuzzification. In the defuzzification relationship, an actual control action is obtained from the fuzzy inference engine. The procedure of producing a quantitative crisp output in fuzzy logic given fuzzy sets and the corresponding membership degrees is described as defuzzification. In this FLC simulation, the centroid defuzzification method is used.

## III. SIMULATION RESULTS

### *A. PID Controller*

As shown in Fig. 4, the PID controlled DC motor model was set up in MATLAB SIMULINK. Here, the gain for the feedback loop is assumed unity and in the forward path we have the PID controller, the armature controlled DC motor open loop transfer function $G(s)$ can be determined after substituting the parameters of the motor as derived in (11). The physical and functional parameters of the DC motor used for simulation testing are as follows [4]:

- Armature resistance, $R_a$= 1 Ω
- Armature inductance, $L_a$= 0.5 H
- Moment of inertia, $J$ = 0.01 Kg.m$^2$
- Viscous friction coefficient, $b$ = 0.1 N.m.s
- Torque constant, $K_t$ = 0.1 N.m/A
- Back EMF constant, $K_b$=0.01 V/rad/s

Substituting the parameters, the open-loop transfer function of the motor is given as:

$$G(s) = \frac{20}{s^2 + 12s + 20.2} \tag{14}$$

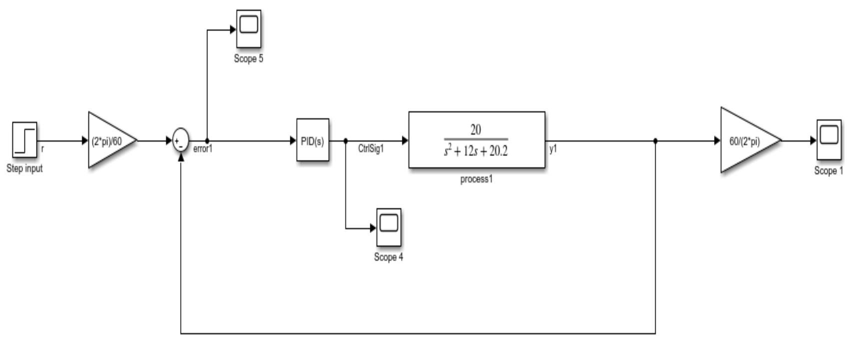


**Fig.4.** PID controlled DC motor SIMULINK model

For the Simulink model shown in Fig.4, a desired motor speed of $\omega_d$=1200 rpm (revolution per minute) was set and PID tuning was performed by adjusting the speed and the robustness in the PID tuner window until we achieved a satisfactory performance. The order of the filter coefficients in the derivative term is selected to be low since pure derivative is not practical as it amplifies measurement noise. Fig. 5 shows the output speed of the motor from scope 1 for $P$=2.84, $I$=7.91, $D$=0.15 and $N$=19. The simulation results show an overshoot: 6.7%, the rise time of 0.27 s and the settling time of 1.2 s. It is to be noted that varying the order of the filter coefficient $N$

creates a variation in the results of the simulations.

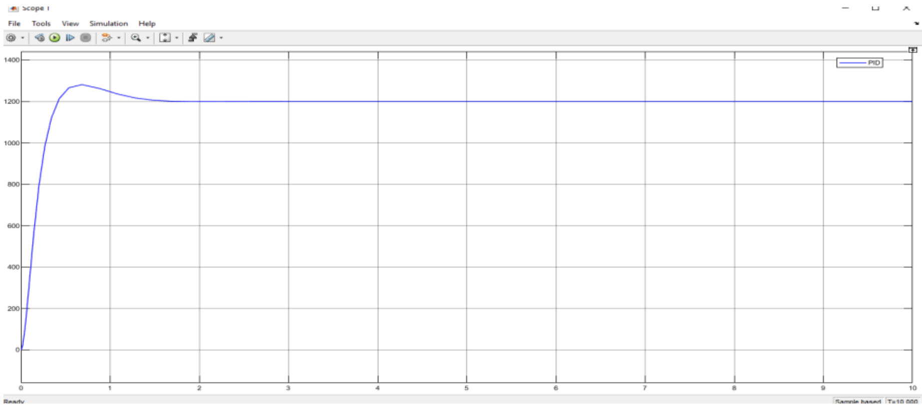


**Fig.5.** Speed response of the PID controlled DC motor

### *B. Fuzzy Logic Controller*

In order to compare the PID controller and the fuzzy logic controller a Simulink model consisting the parallel connection of the two systems is constructed as shown in Fig. 6 to observe their outputs by giving the same reference input.

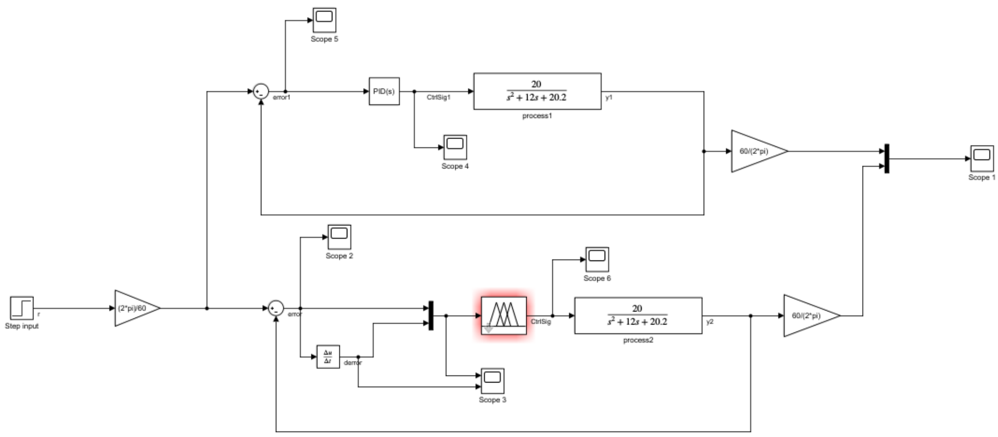


**Fig. 6.** Simulink Model of PID and Fuzzy logic Controller for the DC Motor

As shown in Fig. 6, the inputs to the Fuzzy controller are the “error”, the derivative error “derror” and the output is the control signal “CtrlSig” that drives the plant in this case the DC motor modelled by its transfer function. The range of the error, the derivative error and the control signal are specified by observing their corresponding signals at the various stages as shown in the Fig. 6. The output due to triangular and the Gaussian membership functions were tested. The linguistic variables used for the membership functions are “Low”, “Medium” and High” for both the “error”, “derror” and the “CtrlSig”. The standard deviation of the Gaussian membership function used for the simulation is 10. The ranges specified for the “error”, “derror” and “CtrlSig” are [-10 130], [-180 10] and [0 255], respectively. Fig 7. Shows the membership functions for the triangular type and Fig.8 shows the membership functions for the Gaussian type.

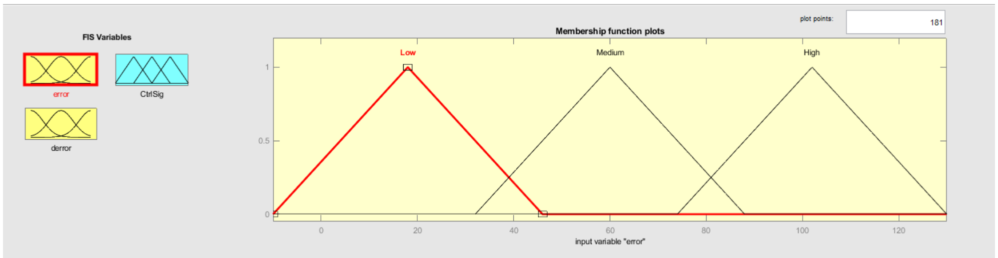


**Fig.7a.** Triangular type membership function for "error".

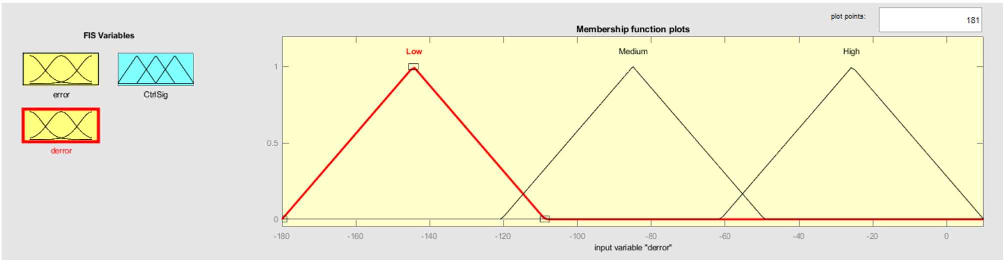


**Fig.7b.** Triangular type membership function for "derror".

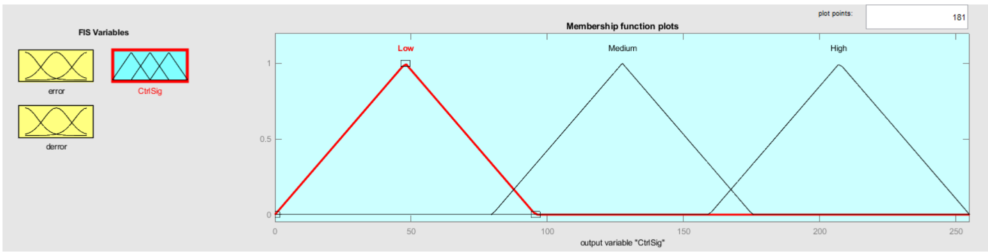


**Fig.7c.** Triangular type membership function for "CtrlSig".

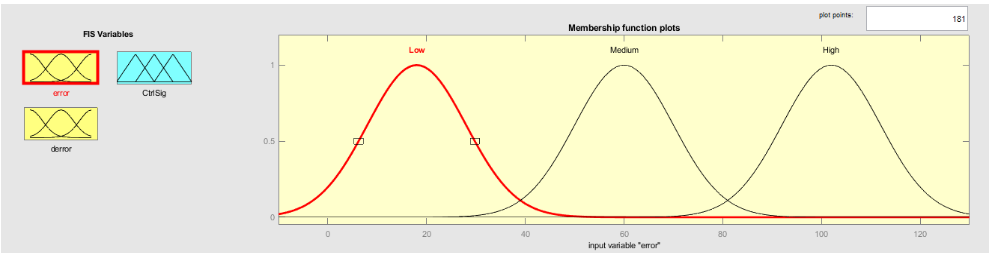


**Fig.8a.** Gaussian type membership function for "error".

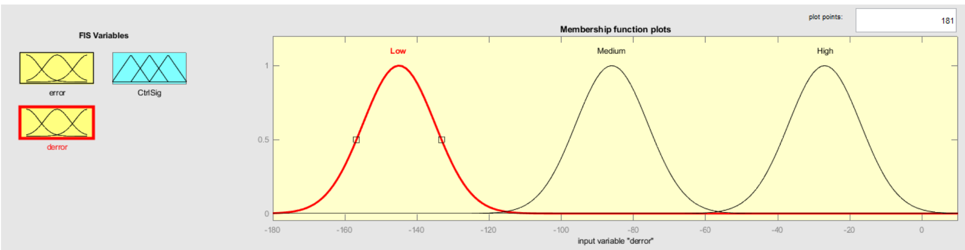


**Fig.8b.** Gaussian type membership function for "derror".

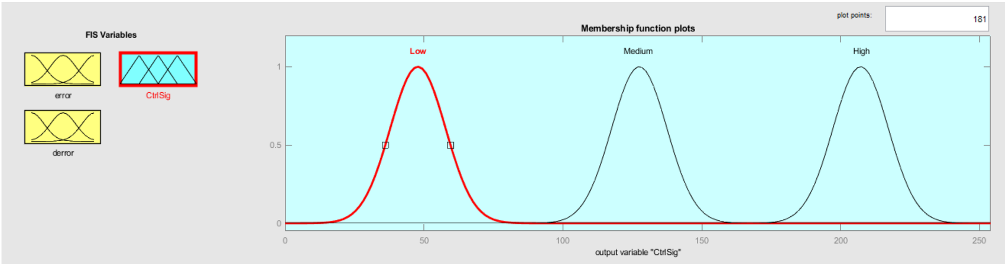


**Fig.8c.** Gaussian type membership function for “CtrlSig”.

The design of fuzzy logic controller is based on the rules that are defined in the fuzzy rule editor. The rules are constructed using Table 1 and they are shown in the Fuzzy rule editor as shown in Fig 9.

**Table 1.** Rules used for the Fuzzy rule editor.

| **error/derror** | **Low** | **Medium** | **High** |
|---|---|---|---|
| **Low** | Low | Medium | Medium |
| **Medium** | Low | Medium | High |
| **High** | Medium | Medium | High |

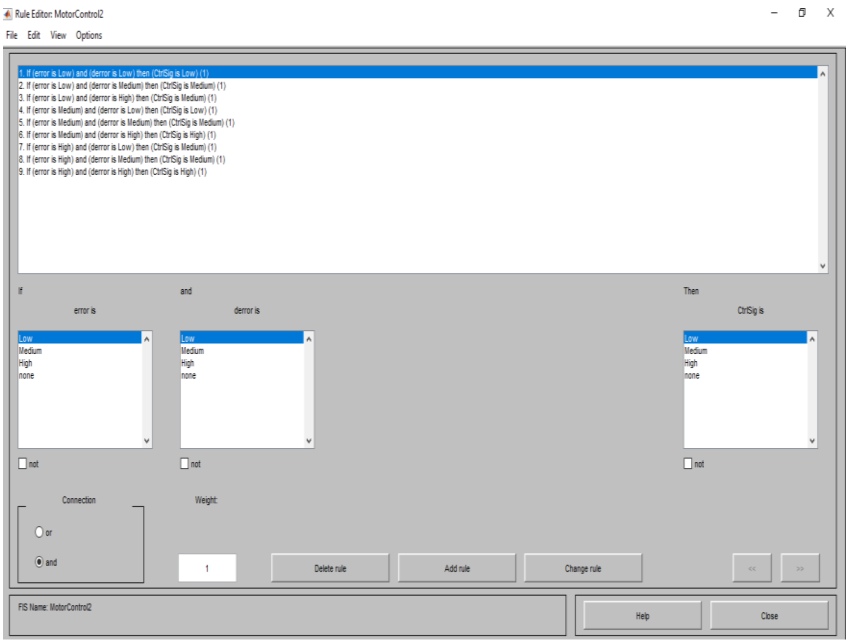


**Fig.9.** Rules entered in the Fuzzy Rule Editor.

The parallel connected system shown in Fig. 6 was simulated after entering all the required inputs for the Fuzzy logic controller. Figure 10 and 11 show the results of the

FLC for the triangular and Gaussian membership functions respectively. The result of the PID is shown in both plots for comparison purposes. As seen in the Figures, both the PID and FLC achieve satisfactory controlling of the desired reference input $\omega_d$=1200 rpm. However, it is observed that there is zero overshoot for the FLC as compared to nonzero overshoot for the PID controller. Furthermore, we have less rise time and settling time for the PID compared with the FLC. There is no difference in the achievement of the triangular versus the Gaussian membership functions having the same result in both cases. Table 2 shows the results of the different output response parameters.

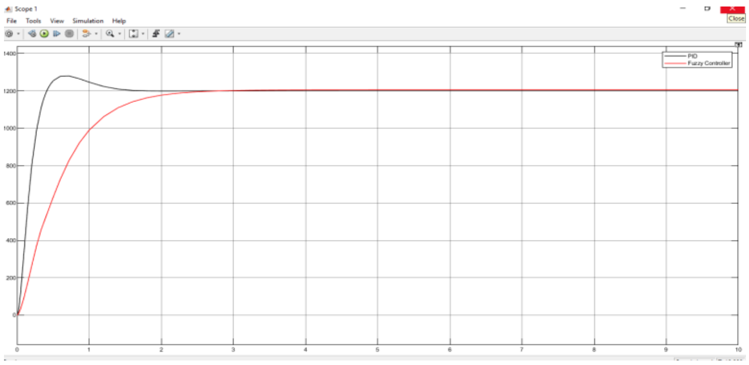

**Fig.10.** Output speed response of the PID and FLC (triangular membership for the FLC).

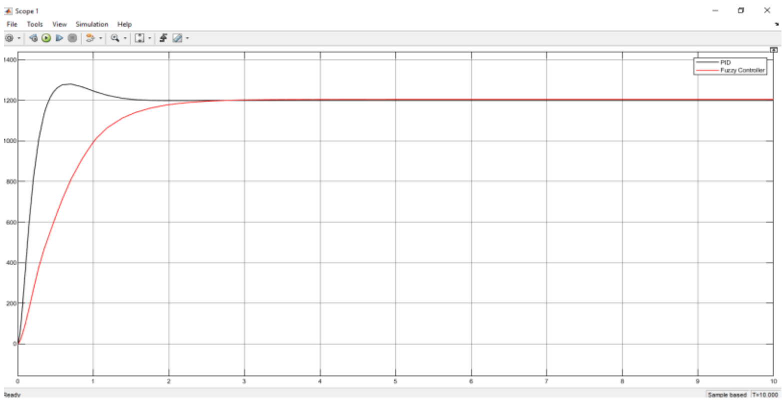

**Fig.11.** Output speed response of the PID and FLC (Gaussian membership for the FLC).

**Table 2.** Output speed response performance parameters.

| Output Response Parameters | PID Controller | FLC with triangular membership | FLC with Gaussian membership |
|---|---|---|---|
| **Overshoot (%)** | 6.6562 | 0 | 0 |
| **Rise time (s)** | 0.2682 | 1.1839 | 1.1526 |
| **Settling time (s)** | 1.1966 | 2.0919 | 2.0591 |

## IV. CONCLUSIONS

This paper presents a PID and Fuzzy logic based speed control system for the armature controlled DC motor. The FLC is developed with three membership functions for each variable. Triangular and Gaussian membership functions are tested with nine rules entered in the Fuzzy logic editor. From the results of the MATLAB/SIMULINK simulations, both the PID and the FLC techniques track the desired speed with the PID having some overshoot at the beginning and the FLC showing zero overshoot. Therefore, FLC method could be more useful than PLC in some applications that are very sensitive to overshoot and oscillation transient behaviors. FLC is also advantageous for those plants that are difficult to get accurate mathematical model and depend on expert human knowledge for the process. On the other hand, the PID controlled system shows better response with respect to rise and setting time compared with the FLC. However, the FLC time response result can also be improved by increasing the number, tuning the range of the membership functions and their corresponding set of rules.

## ACKNOWELEDGMENT

The author appreciates the support and the incentive provided by the American University of Ras Al Khaimah (AURAK) through the initiation and continuous encouragement of the School of Graduate Studies and Research to be active and stay involved in research.

## REFERENCES

[1] B. W. Bequette, Process Control: Modelling, Design and Simulation, Prentice Hall, 2003.

[2] Carl Knopse, Guest Editor, “PID Control,“ *IEEE Control System Magazine*, February 2006.

[3] S. Bennett, “Development of the PID controller,” *IEEE Control System. Magazine*, Vol. 13, 1993, pp. 58–65.

[4] Beza Negash Getu, “Water Level Controlling System Using Pid Controller,” *International Journal of Applied Engineering Research*, Vol. 11, No. 23, pp. 11223-11227, 2016.

[5] Beza Negash Getu, “Tuning the Parameters of the PID Controller Using MATLAB,” *Journal of Engineering and Applied Sciences,* Volume 14, Issue 8, pp. 2538-2545, 2019.

[6] Hina Shahid, Sadia Murawwat, Intesar Ahmed, Sana Naseer, Rukhsar Fiaz, Ayesha Afzaal, Shumaila Rafi, “Design of a Fuzzy Logic Based Controller for Fluid Level Application,” *World Journal of Engineering and Technology*, 2016, 4, 469-476.

[7] Neeraj Srivastava, Deoraj Kumar Tanti, Md Akram Ahmad "Matlab simulation of temperature control of heat exchanger using different controllers," *Automation, Control and Intelligent Systems,* 2(1): pp.1-5, 2014.

[8] P. Singhala1, D. N. Shah, B. Patel, "Temperature Control using Fuzzy Logic, " *International Journal of Instrumentation and Control Systems (IJICS)*, Vol.4, No.1, pp. 1-10, 2014.

[9] L. A. Zadeh, "Fuzzy sets," *Information and Control*, Vol. 8, pp. 338-353, 1965.

[10] Arpit Jain1, Abhinav Sharma, "Membership Function Formulation Methods for Fuzzy Logic Systems: a Comprehensive Review," *Journal of Crtitical Reviews* Vol 7, Issue 19, pp.8717-8733, 2020,

[11] Dorf R. C., Bishop R. H., *Modern Control Systems*, 12th Edition, Pearson Education, 2011.

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