DC Motor Speed Control: A Case between PID Controller and Fuzzy Logic Controller

Philip A. Adewuyi
Mechatronics Engineering Option, Department of Mechanical and Biomedical Engineering,
Bells University of Technology, Ota, Nigeria
solaadewuyi@gmail.com

Abstract– **Proportional-Integral-Derivative (PID) controllers have gained wide popularity in the control of DC motors. Their performances, though require some degree of manual tuning by the operator, are still satisfactory but a means of auto-tuning is desirable. In this paper, the performance of a select dc motor controlled by a proportional-integral-derivative (PID) controller is investigated. An overshoot is observed with an accompanied large settling time thereby confirming the behavior of a typical PID controller. It is therefore a matter of necessity to tune the PID controller in order to obtain the desired performance. On the other hand, a fuzzy logic based controller applied to the dc motor is investigated. With the application of appropriate expert rules, there is no overshoot and the settling time is within the desired value. With fuzzy logic controller, manual tuning is eliminated and intelligent tuning takes the centre stage with satisfactory performance.**

Keywords– **PID, Fuzzy Logic, Controller, Intelligence and PID Tuning**

I. INTRODUCTION

The importance of PID controllers in process industry cannot be overemphasized because more than half of the industrial controllers in use today utilize PID or modified PID control schemes [1]. PID (proportional-integral-derivative) controller has a simple control structure that is easily understood by the operators which help them (operators) in tuning the PID satisfactorily [2]. Tuning of PID is therefore an important aspect of its implementation.

On the other hand, the use of a fuzzy logic controller is investigated where the fuzzy inference system (FIS) has five membership functions for both input parameters as well as the output parameters for a typical Mamdani-type controller. Mamdani-type FIS based controller is used because of its closeness to human reasoning and language at both sides of the system, that is, input and output [3, 4].

The research tool for this work is Matlab/Simulink version 7.4 where simulations are run and appropriate behaviors regarding each (PID and Fuzzy) controller is displayed. Comparison is drawn between the two controllers in terms of performance which justifies the direction of modern control engineering practice though there are still rooms for improvement.

The other parts of this paper is divided into; The DC motor model, PID controller, tuning of PID, fuzzy logic controller, fuzzy logic controller algorithm, discussions as well as conclusion.

II. THE DC MOTOR MODEL

The dc motor parameters considered for this work is that used in an undergraduate experiment carried out in Carnegie Mellon Control Laboratory of University of Michigan, United States having the following dynamic equations [6]:

$$\frac{d}{dt}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} = \begin{bmatrix}\frac{-b}{J} & \frac{k}{J}\\ \frac{-k}{L} & \frac{-R}{L}\end{bmatrix}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{L}\end{bmatrix}u \qquad (1)$$

$$\dot{\theta} = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} \qquad (2)$$

These equations are in the form of:

$$\ddot{\theta} = A\dot{\theta} + Bu \qquad (3)$$

$$y = \dot{\theta} = C\dot{\theta} + Du \qquad (4)$$

This is converted to a transfer function in order to make the dc motor model similar in terms of transfer function to that of PID in expression.

$$\frac{\theta}{V} = \frac{K}{(Js+b)(Ls+R)+K^2} \qquad (5)$$

J is the moment of inertia

b is the damping ratio

k is the electromotive force constant

R is the electric resistance

L is the electric inductance

u is the source voltage

θ is the position of shaft

i is the armature current

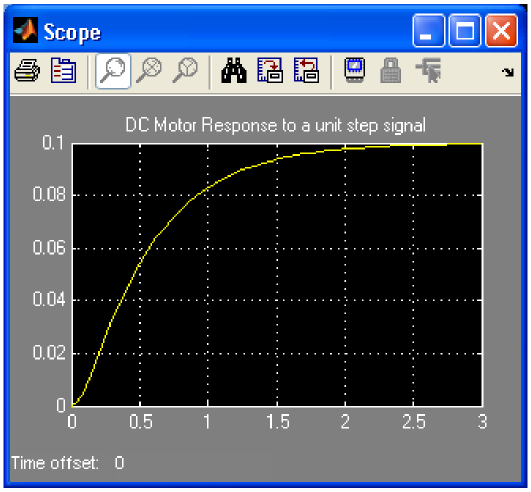

Fig. 1: Uncontrolled DC motor response

Therefore the controlled system/dc motor has a transfer function of the form in equation (5):

The open loop transfer function behaviour of the dc motor to a unit step response is shown below in Fig. 1.

It could be observed that the motor's response to a unit step input signal, that is, an equivalent of 1V supply voltage is 0.1 rad/sec. This is one-tenth of the desired response. Also, the settling time is 3s of which a reduction is sought. Steady state error could also be improved in due course.

III. PID CONTROLLER

To provide an improvement to the performance of the dc motor, a PID controller is introduced and applied. This PID controller for the control of the dc motor is set up in Matlab/Simulink environment as shown in Fig. 2:

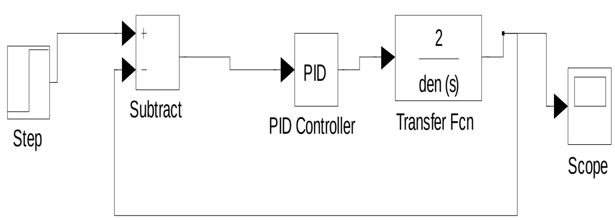

Fig. 2: PID controlled system

A simple feedback control theory is utilized to represent the overall PID controlled system.

This PID controller has the transfer function of the form:

$$K_p + \frac{K_I}{s} + K_D S = \frac{K_D S^2 + K_P S + K_I}{s} \qquad (6)$$

It is observed that when the proportional gain alone is chosen arbitrarily, the response of the motor is not satisfactory. The same problem is experienced when the integral gain and the derivative gain alone are concentrated on.

Therefore, in order to have the desired motor response, the PID controller has to be tuned. Tuning of PID controller using a trial and error method wastes time and if not properly tuned the dc motor could be damaged. To save us a lot of efforts, a tuning guide proposed by Ziegler-Nichols is adopted with the aim of; shortening the rise time, eliminate/reduce the overshoot, quickening the settling time of the system to a steady state, and reducing to a tolerable value the steady-state error which is the difference between the steady-state output and the desired output [5].

When the PID controller is properly tuned according to Ziegler-Nichols tuning rule applied to a unit step input system, and with proportional gain, K_p = 250, integral gain = 100, and derivative gain = 20, the following response or plot is obtained:

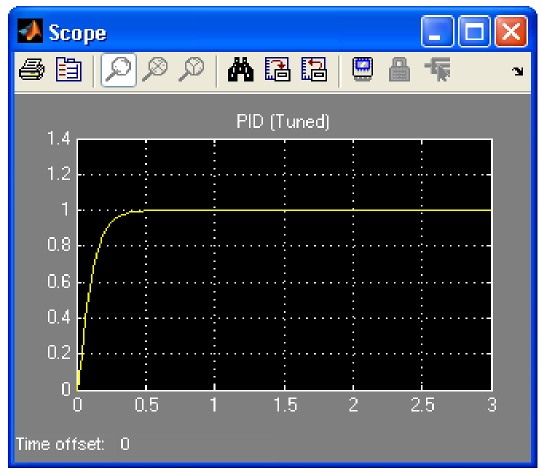

Fig. 3: Response of a Tune PID controlled system

IV. FUZZY LOGIC CONTROLLER

Fuzzy logic can be described as a nonlinear mapping of an input data vector into a scalar output that is the vector output case decomposes into a collection of independent multi-input/single-output systems [7]. As a matter of fact, fuzzy logic is used to mimic to the best of its ability the human mind. Approximate behavior is its hallmark rather than exactness [8]. These flexibilities of fuzzy logic and its simplicity make it desirable to be investigated especially in areas of dc motor control where proportional-integral-derivative controller is most popular.

Fuzzy Logic Controller Algorithm

To successfully build a fuzzy logic controller system, the following steps have to be taken:

i. The input parameters are supplied to the fuzzy controller

ii. Fuzzification, which is a process of taking the crisp (traditional) inputs and determine the degree to which they belong to each of the appropriate fuzzy sets.

iii. These fuzzified sets are then applied to the antecedents of fuzzy rules. For this multiple antecedents, the fuzzy operator (AND) is used to obtain a single member that represents the result of the antecedent evaluation such that:

$$\mu A \cap B(x) = \min\,[\mu A(x), \mu B(x) \qquad (7)$$

where $A = \{x \in U, \text{and x meets some conditions}\}$

The membership function for A is denoted by:

$$\mu_A(x)/\mu_B(x) = \begin{cases} 1 \; if \; x \in A \\ 0 \; if \; x \notin A \end{cases}, \qquad (8)$$

for a continuous membership function of zero-one.

iv. Results in step iii are applied to the membership function of the fuzzified set.

v. Defuzzification is carried out using centroid technique to obtain the crisp output. This is expressed mathematically as:

$$COG = \frac{\int_a^b \mu_A(x) x dx}{\int_a^b \mu_A(x) dx} \qquad (9)$$

This algorithm is implemented in matlab with a five member fuzzy inference system used for the input parameters, that is, error and change in error and also for the output. A Mamdani-type fuzzy inference approach is utilized. The set up is as shown hereunder:

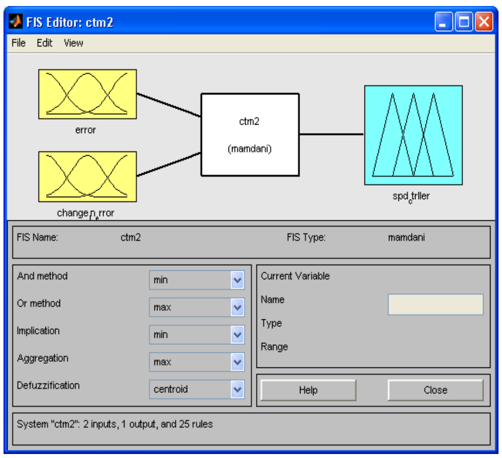

Fig. 4: Mamdani-type fuzzy logic set-up

The membership function is displayed in Fig. 5.

The operational rule for the controller is set up based on expert knowledge and shown in Table 1:

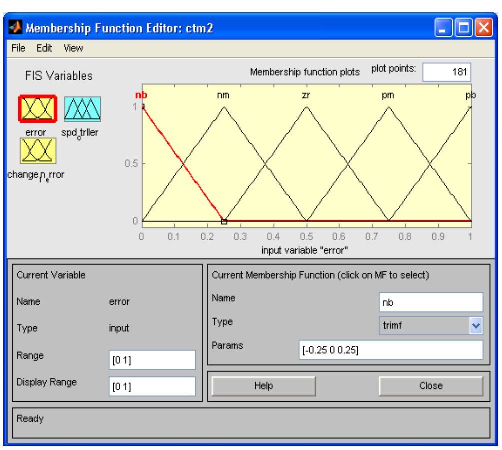

Fig. 5: Membership function of the input to the fuzzy logic controller

Table 1: Fuzzy Rules

CHANGE IN ERROR (Δe)	ERROR (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

*NB stands for negative big with numerical equivalent of 0, NM stands for negative medium with numerical equivalent value of 0.25, ZR 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.

The graphical view of these rules when simulation is run is shown in the Fig. 6:

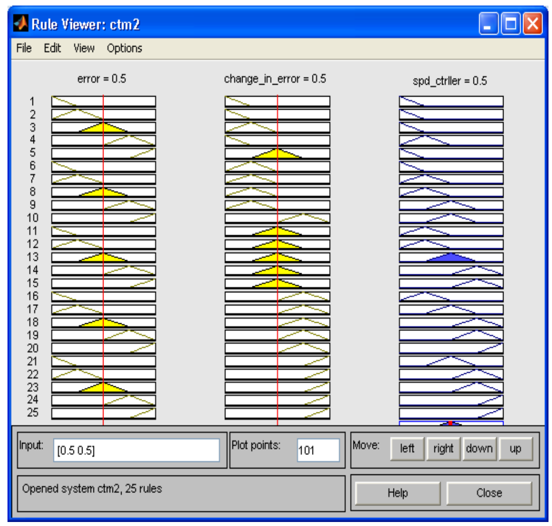

Fig. 6: Rule viewer for the fuzzy logic controller

The simulation is run and the three-dimensional behavior of the controller is shown in Fig. 7:

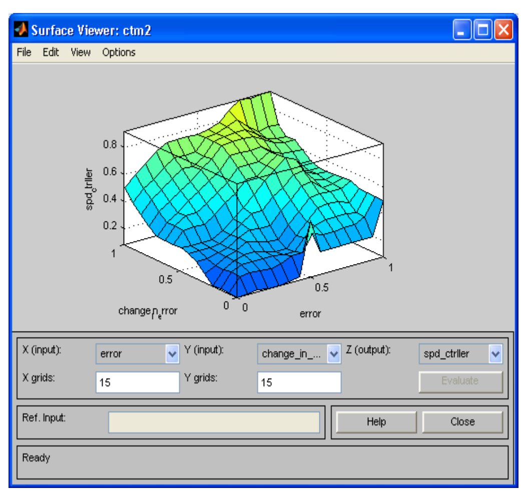

Fig. 7: 3-Dimensional view of the fuzzy logic controller response

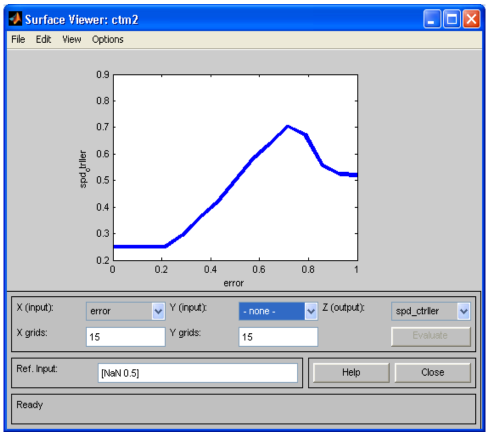

Fig. 8: Fuzzy logic response to error in 2-dimensional view

The 2-D relationship between the error and output speed is also displayed in Fig. 8:

The 2-D relationship between the change in error and output speed is also displayed in Fig. 9:

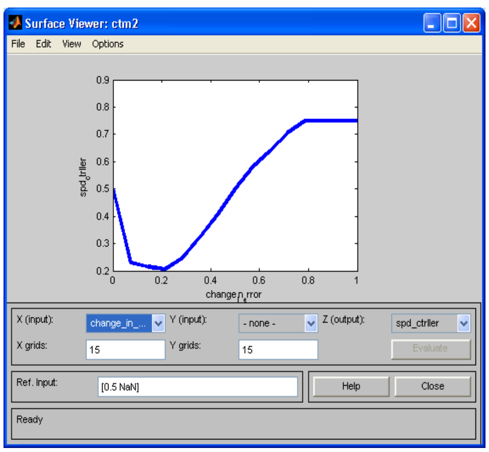

Fig. 9: Fuzzy logic response to change in error in 2-dimensional view

When the fuzzy logic controller is applied to the dc motor, the set up in matlab/simulink environment is displayed in Fig. 10:

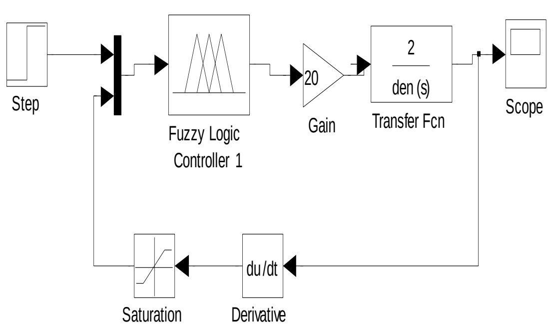

Fig. 10: Fuzzy logic controlled system

A unit step signal is applied to a closed-loop fuzzy logic motor controlled system. This signal corresponds to a supply voltage of 1V and the system is run. The output, depicting the response of the dc motor is displayed via a scope after simulation as we have it hereunder in Fig. 11.

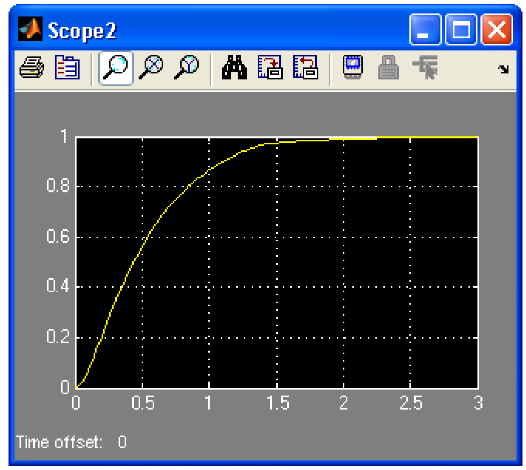

Fig. 11: Response of fuzzy logic controlled system

V. COMPARISON

To compare the responses of the proportional-integral-derivative controller (PID) and that of fuzzy logic controller, the following matlab/simulink arrangement is utilized:

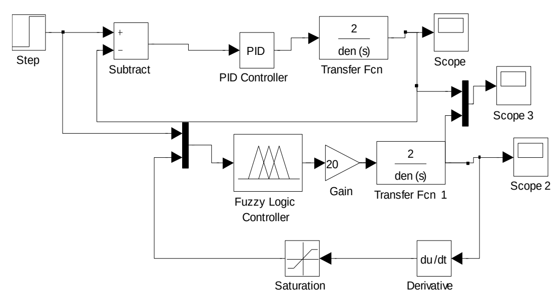

Fig. 12: Combination of PID and Fuzzy logic controlled system

A unit step input signal is applied and the combined responses are as outlined in Fig. 3 is observed:

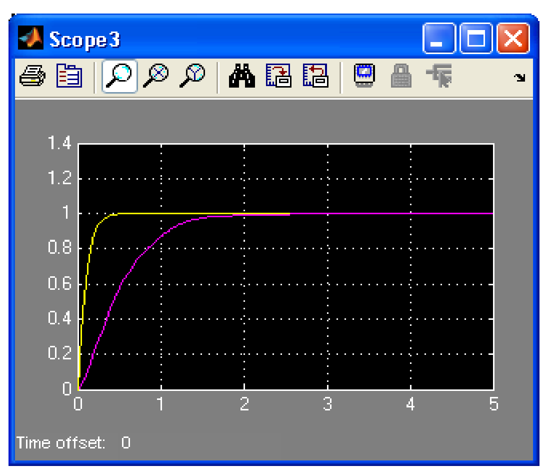

Fig. 13: PID and Fuzzy logic controller responses

A) Discussions

The PID controller when properly tuned responds faster to the input parameter. The overshoot is removed with derivative gain set to 250. Reduction of steady state error is achieved by setting the integral gain to 100. The integral gain is set at 20 as zero steady state error is aimed. The settling time is 0.5s which is much more improved compared to that of the uncontrolled system of figure 1. The speed response has also improved from 0.1 rad/sec for the uncontrolled system to the desired value of 1 rad/sec of the controlled system. This is a ten times improvement in speed response.

The fuzzy logic (Mamdani-type) based controller has a sluggish response to the input signal. It has no overshoot though the settling time is 1.8s. It requires no tuning and human manipulations are eliminated or otherwise reduced.

VI. CONCLUSION

The use of PID controller as a means of control of a dc motor is no doubt ahead in terms of system robustness and predictability especially for a well tuned PID controller. Another consideration of a way of controlling a dc motor using fuzzy logic (Mamdani-type) controller shows an appreciable performance though not optimal. Fuzzy logic based controller requires no tuning but has a sluggish response to the input signal and cannot readily predict stability as well as robustness of the dc motor. So, there are still more rooms for improvement in this regard.

REFERENCES

[1] K. Ogata (2009) "Modern Control Engineering, 4th Edition", Dorling Kindersley Pvt. Ltd., India.

[2] K. J. Astrom and T. Hagglund (1995) "PID Controllers: Theory, Design, and Tuning", Instrument Society of America, USA, pp. 134-151.

[3] S.R. Khuntia, K.B. Mohanty, S. Panda and C. Ardil (2010) "A Comparative Study of P-I, I-P, Fuzzy and Neuro-Fuzzy Controllers for Speed Control of DC Motor Drive", International Journal of Electrical and Computer Engineering, Vol. 5, Iss. 5, pp. 287-291.

[4] S. Panich (2010) "Development of Fuzzy Controller for Water Level in Stream Boiler Tank", Journal of Computer Science, Vol. 6, Iss. 11, pp. 1233-1236.

[5] D. Xue, Y. Q. Chen, and D. P. Atherton (2007) "Linear Feedback Control: Analysis and Design with Matlab", Society for Industrial and Applied Mathematics, USA, pp. 183-235.

[6] PID Design Method for the DC Motor, www.library.cmu.edu/ctms/examples/motor/pid2.htm [accessed on October 9, 2012].

[7] J. M. Mendel (1995) "Fuzzy Logic Systems for Engineering: A Tutorial", Proceedings of the IEEE, Vol. 83, No. 3, pp. 345-377.

[8] L. A. Zadeh (1994) "Fuzzy Logic, Neural Networks, and Soft Computing", Communication of the ACM, Vol. 3. No. 3, pp. 77-84.