

# Enhancement of the Dynamic Performance of a DC Motor using Fuzzy Logic Algorithm

Islam Ahmed El Tohamy  
Energy Power and Telecom Solutions  
Ain Shams University  
Cairo, Egypt

Dr. Adel Ahmed  
Electrical Power and Machines  
Ain Shams University  
Cairo, Egypt

Prof. Dr. Diaan El Koshairy  
Electrical Power and Machines  
Ain Shams University  
Cairo, Egypt

**Abstract**— Among the most effective controllers that are world wide spread are the Proportional integral and derivative PID Controllers. It is well known that tuning the parameters of the conventional PID changes the plant response such that plant could be faster and more robust. In conventional PID controllers the parameters  $K_p$ ,  $K_i$ ,  $K_d$  are kept constant during the plant operation and they are adjusted only once at the beginning of the plant operation, This paper presents online tuning technique using Fuzzy logic algorithm for continuous updating of the parameters of the PID for achieving effective speed control of the DC motor. This technique proved to be enhancing the overall performance of the plant by decreasing the overshoot, reducing settling time and rise time. Additionally this Fuzzy logic PID algorithm is found to be very effective in bringing the DC motor system to more stable operation. The MATLAB/SIMULINK is used for modeling of the DC motor system along with the Fuzzy logic PID controller and conventional PID controller. The speed responses of the DC motor for both controllers are presented and compared.

**Keywords**—Fuzzy Logic Controller; PID controller; DC Motor; MATLAB / SIMULINK; stability margin; Nyquist plot

## I. INTRODUCTION

Seeking of robust, effective and stable drive system is the most critical issue that faces control engineers nowadays in the field of automation technology [1]. DC Motors are used extensively in the automation technology and industrial application due to their outstanding performance and simplicity of their control commands, they are the first choice in applications that require accurate speed regulation as well as precise positioning [2].

PID controllers are widely used due to their simple implementation, easy to tune in addition to reasonable stable output response. About 85% of dynamic control systems are PID based systems [3]. However PID controllers are reliable and stable, in some applications they can't satisfy the system needs when maximum accuracy and effectiveness are needed, this may be due to aggressive changes in the load profile that the PID controller should track and noises that the system would face. This is obvious when undesirable overshoot, time lag and oscillations appear clearly in the output responses of a PID controlled system [4]. In DC motor control, tuning parameters of the PID controllers became more challenging

task for control engineers, this is due to the nonlinearity relationship between the motor speed and the input voltage along with the damping coefficient that is dependent on the motor parameters [5]. Therefore it is important to correctly tune the parameters of the PID controllers to get an optimal output response for effective DC motor control.

In conventional PID controllers the three parameters  $K_p$ ,  $K_d$  and  $K_i$  are constant during the plant operation and tuned only once at the beginning of operation using several methods like Ziegler-Nichols method sometimes using trial and error but as mentioned optimal response was difficult to be achieved [6]. Therefore with the aid of the evolution in the field of artificial intelligence, computer programming and microprocessors, new tuning algorithms are proposed using neural networks, fuzzy logic, genetic algorithms, ....etc., the idea is to optimize these three parameters as possible to get better performance than these conventional controller methods [4,7]. In this paper, Fuzzy logic algorithm for continuous tuning of the PID controllers parameters for DC motor speed control is presented.

Fuzzy logic concept was firstly introduced by Dr. Lotfi Zadeh in 1965 [8]. Fuzzy models or sets are mathematical means of representing input/output information, they may have crossed ranges and the final output is based on the fuzzy rules. Fuzzy rules are decisions like what people take based on imprecise and non-numerical information hence the term fuzzy [9]. In the Fuzzy logic based PID controller for DC motor speed control, two closed control loops are used, one is for conventional PID to control the output speed by controlling the input voltage and the other is for the Fuzzy logic controller to continuously update the parameters of PID controller aiming to reach the desired performance, hence these PID controller parameters are continuously changing based on the outputs of the Fuzzy logic controller [10]. DC motor mathematical model is required for judging whether the output response of Fuzzy logic PID is superior to the conventional PID controller or not by comparing rise time, settling time, overshoot and stability margin of both controllers responses.

## II. OPEN LOOP DC MOTOR MODEL

In armature voltage controlled DC motor, the field current  $i_f$  is held constant, and the armature current is controlled through

the armature voltage  $u_a$ . In this case, the motor torque increases linearly with the armature current. Therefore ;

$$T_M = K_f \cdot i_a = K_f \cdot i_a \quad (1)$$

,Where  $K_f$  is the motor constant.

DC motor system consists mainly of two systems one is the electrical system and the other is the mechanical system , both are coupled by the back emf constant and the torque constant , they are assumed to be the same ( $K_f$ ) , Fig. (1) shows this clearly.

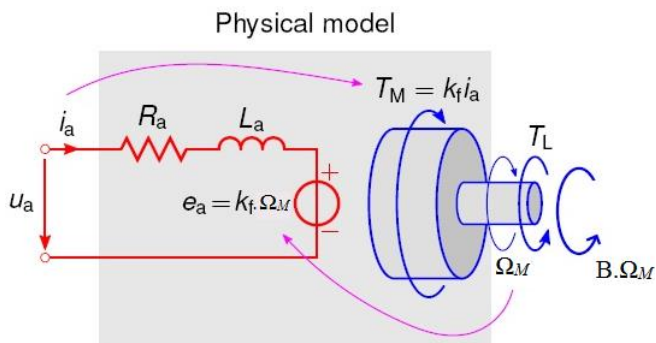


Fig.(1) Physical and mathematical model of DC motor

**System electrical equations :**

$$L_a \frac{di_a}{dt} = U_a - e_a - R_a \cdot i_a \quad (2)$$

Where  $e_a$  is the back e.m.f ,

$$e_a = K_f \cdot \Omega_M \quad (3)$$

Substituting and using Laplace transformation :

$$I_a(s) = \frac{1}{s} * \frac{1}{L_a} [U_a(s) - E_a(s) - R_a \cdot I_a(s)] \quad (4)$$

**System Mechanical equations :**

$$J \frac{d\Omega_M}{dt} = T_e - T_L - B \cdot \Omega_M \quad (5)$$

Similarly ;

$$\Omega_M(s) = \frac{1}{s} * \frac{1}{J} [T_e(s) - T_L(s) - B \cdot \Omega_M] \quad (6)$$

Hence the full block diagram of the DC motor can be developed as follows in Fig. (2).

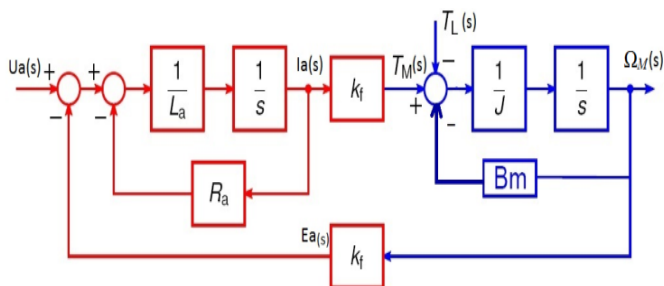


Fig.(2) block diagram of DC motor

This block diagram can be reduced as follows in Fig.(3)

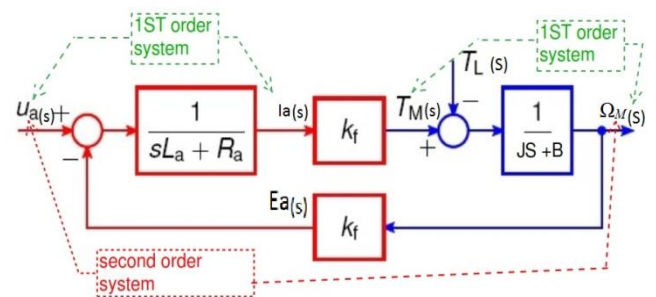


Fig.(3) Reduced block diagram of DC motor with constant field current

Hence the transfer function relating the angular speed and the input voltage is as follows

$$\therefore \frac{\Omega_M}{U_a} = \frac{K_f}{(L_a s + R_a)(J s + B) + K_f^2} \quad (7)$$

The data of the DC motor studied is shown in Appendix I.

**III. PID CONTROLLER STRUCTURE**

*A. Conventional PID Controller*

PID controller structure is shown in Fig.(4) in the time domain , it's transfer function is as per equation (8) in the (S) domain.

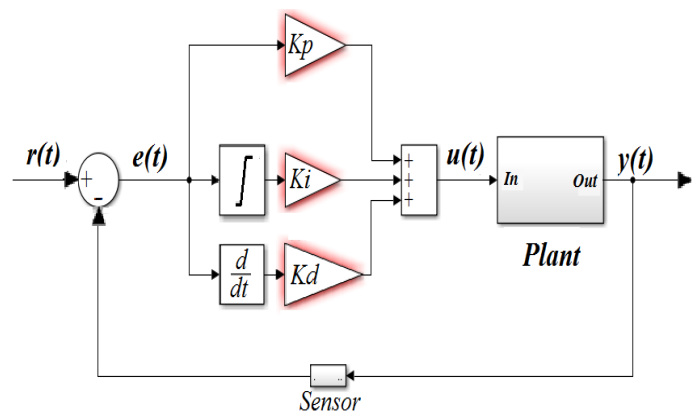


Fig.(4) PID controller structure.

$$G(s) = K_p + \frac{K_i}{s} + K_d * s \quad (8)$$

*B. Fuzzy logic PID Controller*

The parameters of the conventional PID controller could greatly improve the output response of the plant if correctly tuned . Nevertheless the desired dynamic response can't be fully reached, this is because the PID parameters are kept constant during the whole plant operation [7]. Ziegler and Nichols method is a popular method that mainly focuses on tuning the PID controller and improve the dynamic performance of the system response, but sometimes using this method the plant will still suffer from large overshoot and large settling time for the step response. This might not be acceptable for some processes [11].

The idea here in the Fuzzy logic controller is to constantly update the preset parameter values of the PID controller depending on the inputs, Then the fuzzy algorithm will develop a relationship between the input and output

linguistically using Fuzzy sets and the final values of the outputs will be determined using fuzzy rules Hence online tuning could be achieved [10,11].

The architecture of the fuzzy PID is shown in Fig.(5) ,it consists of two parts. The first is the control loop of normal PID, the second part is the fuzzy controller loop that constantly inject the rehabilitation factors to the PID controller.

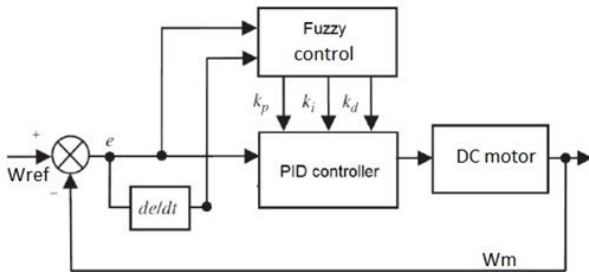


Fig.(5) Architecture of fuzzy PID controller

$K_{p,o}$  ,  $K_{i,o}$  and  $K_{d,o}$  will be considered to be the preset/initial values of the PID where the updated tuned values will be  $K_{p,u}$ ,  $K_{i,u}$  and  $K_{d,u}$  and they are equal to the multiplication of the Preset/initial values by rehabilitation factors  $K_{p,f}$ ,  $K_{i,f}$  and  $K_{d,f}$  that are originated from the fuzzy controller.

$$K_{p,u} = K_{p,o} \cdot K_{p,f} \quad (9)$$

$$K_{i,u} = K_{i,o} \cdot K_{i,f} \quad (10)$$

$$K_{d,u} = K_{d,o} \cdot K_{d,f} \quad (11)$$

#### IV. SIMULATION

##### A. Assigning of the fuzzy sets and membership functions

Using the MATLAB/SIMUINK as the tool of simulation , Two inputs are defined to the fuzzy tool box of MATLAB that are the error (e) and it's rate of change (de/dt) where the output will be the rehabilitation factors  $K_{p,f}$ ,  $K_{i,f}$  and  $K_{d,f}$  . Fuzzification of the input and the output is done by transforming inputs and outputs into fuzzy sets with definite ranges that consist of five semantic values for inputs and seven semantic values for the output. Ranges for the fuzzy sets of the inputs  $[e , de/dt] = [-1 1]$  and for the outputs  $[K_{p,f} , K_{i,f} , K_{d,f}] = [0 2]$ . The semantic values and their legend is as per Table I. The MATLAB fuzzy tool box relation between input and output in addition to fuzzy sets and membership functions are shown from Fig.(6) through Fig.(8).

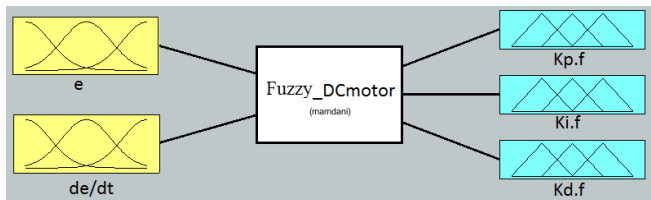


Fig.(6) Number and relation of the fuzzy input /output fuzzy sets in MATLAB Fuzzy-tool box.

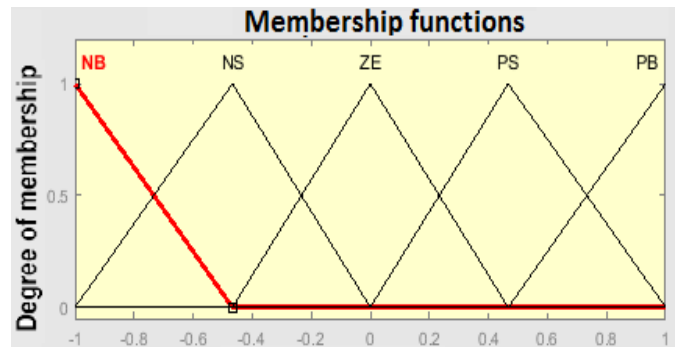


Fig (7). Membership of inputs fuzzy sets (e , de/dt)

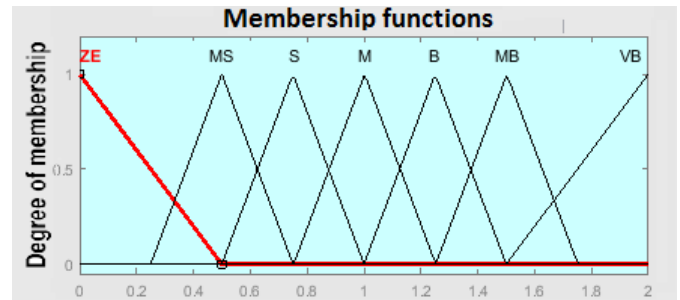


Fig (8). Membership of output fuzzy sets ( $K_{p,f}$ , $K_{i,f}$  , $K_{d,f}$ .)

TABLE I. SEMANTIC VALUES LEGEND

Output		Input	
Term	Legend	Term	Legend
ZE	zero	NB	Negative Big
MS	Medium small	NS	Negative small
S	Small	ZE	Zero
M	Medium	PS	Positive small
B	Big	PB	Positive big
MB	More big		
VB	Very big		

##### B. Establishment of Fuzzy rules:

Fuzzy rules are established to link the three output parameters to the variation of the inputs , in other words to select the appropriate values for the three gains ( $K_p$ ,  $K_i$  and  $K_d$ ) from their fuzzy set depending on particular set of inputs in order to reach the optimum response required.

Five membership functions are built for inputs and seven for the outputs thus they will give total of 35 rules, but for simplicity we consider only five membership functions for both thus producing only 25 rules. Appendix II Shows the fuzzy rules developed in the fuzzy-tool box of MATLAB.

##### A. Matlab Tuning method for the conventional PID

In order to start the simulation the preset/initial three parameters  $K_{p,o}$ ,  $K_{i,o}$  and  $K_{d,o}$  are needed to be determined in order to be inserted in both conventional and fuzzy PID controllers ,MATLAB/SIMULINK includes a PID block that could tune the PID parameters based on reference speed tracking and disturbance rejection. Fig.(9) shows the tuned response by MATLAB for the presented DC motor system along with the tuned values that will be considered as the preset values.

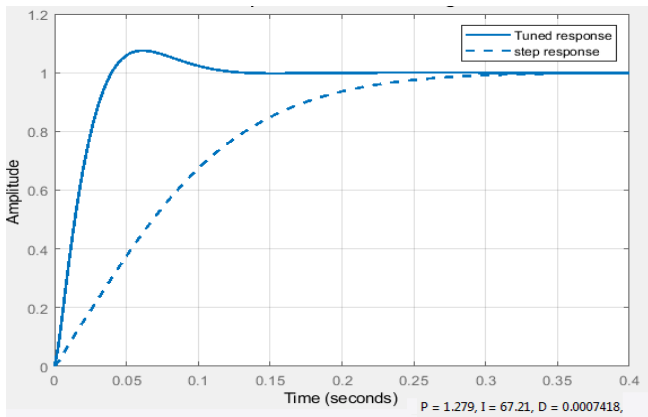


Fig. (9): Tuned response by MATLAB

**B. Simulink Model and simulation assumptions:**

The Simulink model for the fuzzy logic PID controller for dc motor speed control (FLCP) is shown in Fig.(10). The required speed is assumed to be the rated value of the Dc motor 125.7 rad/sec, The motor is assumed to start and run at half its rated load that is 4.18 N-m then it will be subjected to a load disturbance at  $t = 0.6$  sec at which the load changed to be 35 N-M. Simulation starts by inserting the initial/preset values acquired from MATLAB tuning method in both conventional and fuzzy PID controllers then let the system shown in Fig.(10) to run with the conventional PID Dc motor system simultaneously. The simulation times are 1.5 seconds for the load disturbance and 3 seconds for speed tracking. It can be seen in Fig. (10) that the input to the fuzzy controller is the error and it's rate of change however the outputs are the three rehabilitation factors ( $K_{p,f}$ ,  $K_{i,f}$  and  $K_{d,f}$ ), they are multiplied by the preset values of the PID controllers  $K_{p,o}$ ,  $K_{i,o}$  and  $K_{d,o}$  then the resulting updated values  $K_{p,u}$ ,  $K_{i,u}$  and  $K_{d,u}$  are injected directly to the PID controller this whole system is the Fuzzy Logic PID controller (FLCP).

**V. SIMULATION RESULTS**

In Fig. (11), Fig. (12) and Fig. (13), the speed responses of both controllers are shown. They are a clear prove that the speed response of the fuzzy logic PID controller is much better in performance than the conventional PID controller in terms of settling time, rise time and over shoot.

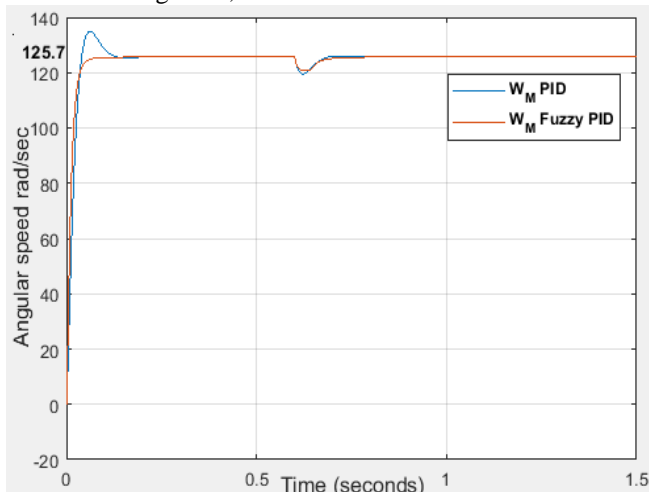


Fig (11). Response of fuzzy logic PID controller with the conventional PID controller.

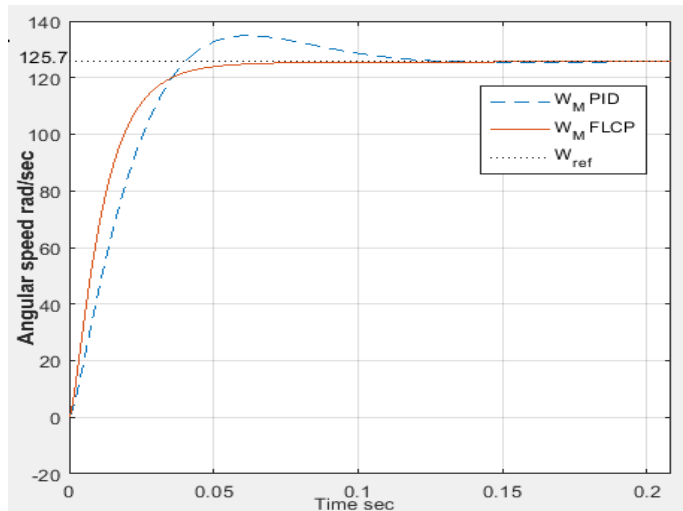


Fig (12). Speed Responses during motor starting

In Fig. (14) the change of rehabilitation factors generated from the Fuzzy logic controller is shown, they abruptly change at starting and at the load disturbance and then stabilize at steady state values determined by the fuzzy controller.

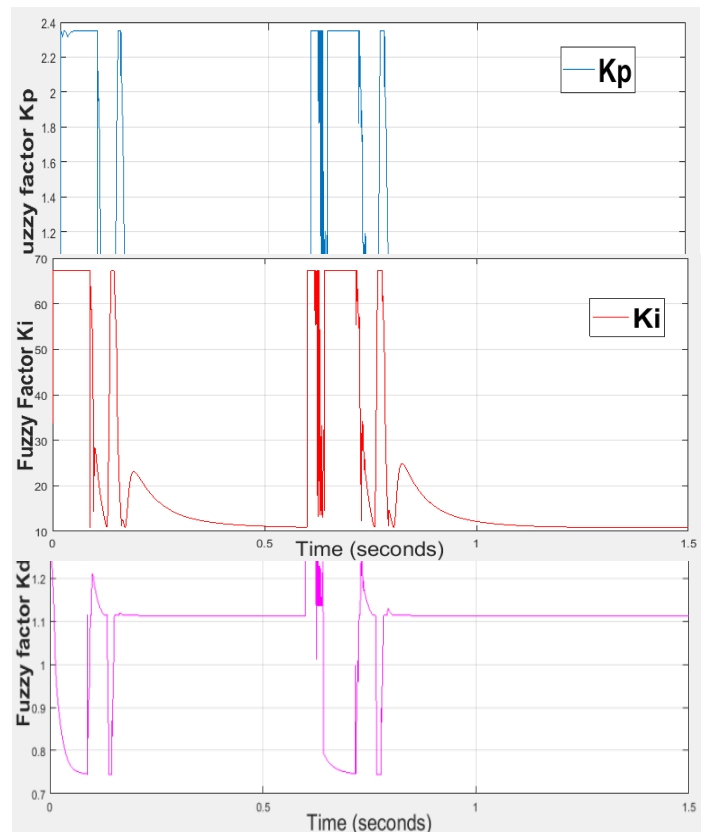


Fig (14). Rehabilitation factors by Fuzzy logic controller



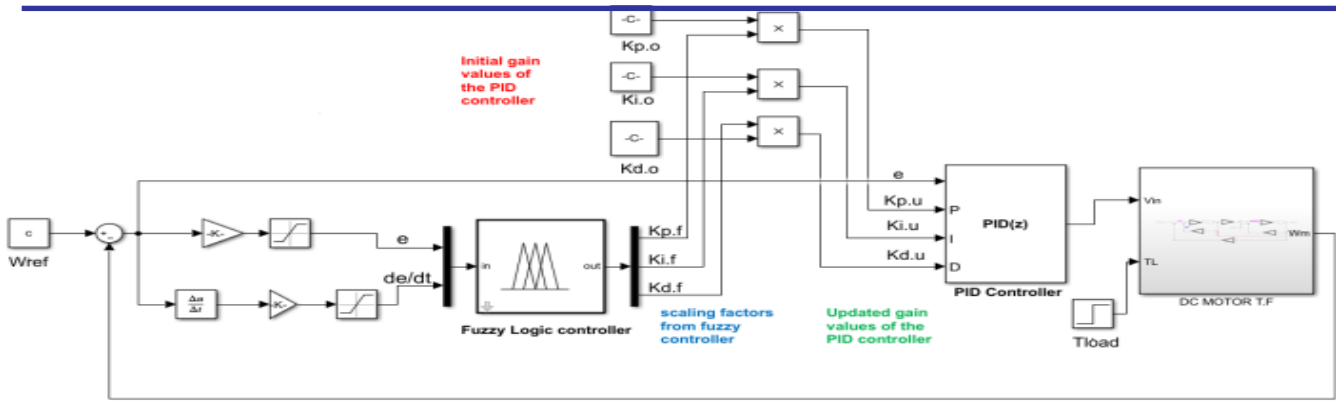


Fig (10). Simulink model of the Fuzzy Logic Controller

Fig. (15) is the Surface view from MATLAB tool box for the three outputs  $K_{p.f}$ ,  $K_{i.f}$  and  $K_{d.f}$ , they are in fact three dimensional curves that graphically show the effect of variation of the inputs ( $e$ ,  $de/dt$ ) to the outputs parameters.

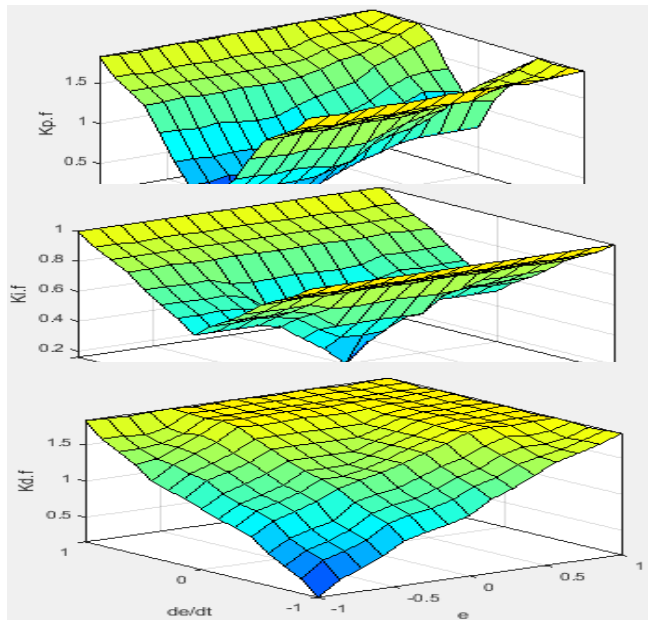


Fig (15). Surface view from Fuzzy Tool box.

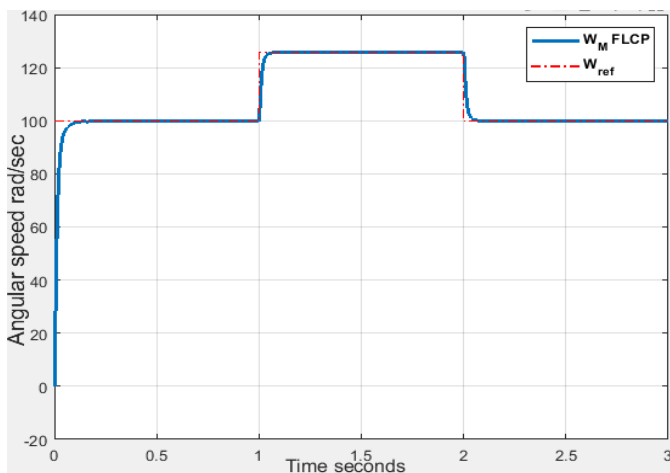


Fig (13). Speed tracking by Fuzzy Logic Controller PID.

## VI. STABILITY ENHANCEMENT

In order to compare between the two controllers in terms of stability, First the parameters of the fuzzy logic PID controller are assumed to be any value that produced from the fuzzy logic tool box output  $(K_{p,u}, K_{i,u}, K_{d,u})_{Fuzzy}$  because as per Fig.(14) they are continuously changing specially at starting of the DC motor and at any load disturbance [12].

Second MATLAB will be used to draw the Bode plot & Nyquist for the open loop transfer functions  $[C(S)*P(S)]$  for DC motor system with both controllers added [13].

Fig.(16) shows the closed loop system diagram with a controller added.

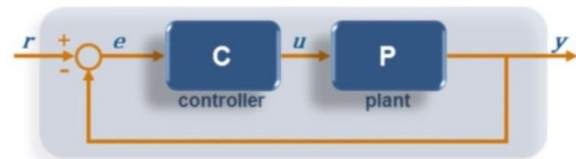


Fig (16) closed loop system

Where (C) will be the PID transfer function for both controllers that is equal to equation (8) after adding their parameters that are  $(K_{p,u}, K_{i,u}, K_{d,u})_{fuzzy}$  that were acquired from the workspace of the Fuzzy logic PID controller and  $(K_{p,o}, K_{i,o}, K_{d,o})$  that were acquired from the MATLAB tuning method for the conventional PID controller as mentioned earlier in section III.

(P) will be the DC motor transfer function that is equal to Equation (7) after adding the DC motor parameters in Appendix I.

The open loop Transfer functions for the DC motor system with both controllers are as per below equations (11) and (12).

$$[C(s)P(s)]_{conv.} = \frac{12.403 (s + 1668) (s + 54.29)}{s (s + 470.3) (s + 29.72)} \quad (11)$$

$$[C(s)P(s)]_{fuzzy} = \frac{18.392 (s + 563.9) (s + 17.35)}{s (s + 470.3) (s + 29.72)} \quad (12)$$

Bode plot for both controllers and DC motor transfer functions are shown in Figure 17.

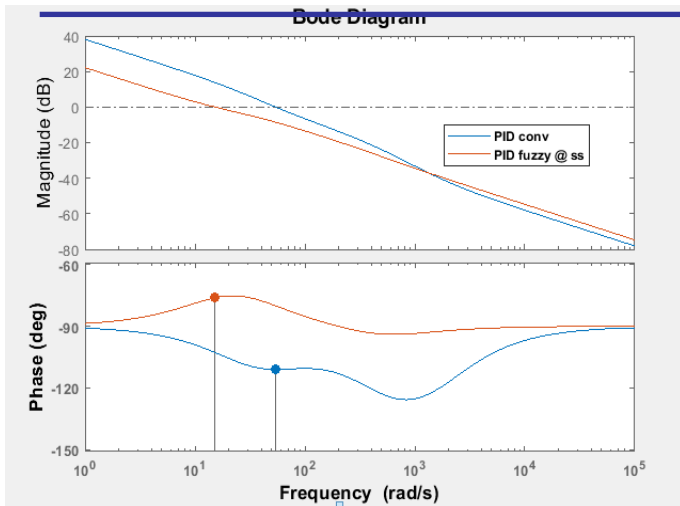


Fig.(17) Bode Plot of both open loop transfer functions

From the Bode Plot in Fig (17) both controllers led the system to have infinite gain margin ( $GM_{conv.PID}$  ,  $GM_{fuzzy}=\infty$ .) because their phase plot with the Dc motor system never pass the 180° however, the phase margin (PM) increased in case of fuzzy logic PID controller where ( $PM_{fuzzy}= 104$  at 15.2 rad/sec) and ( $PM_{ConvPID} = 69$  at 54.2 rad/sec). It is well known that by increasing the phase margin the system becomes more stable and impregnable to noises and disturbances therefore the Fuzzy logic PID controller enhances the stability of the DC motor system better than conventional PID controller. This is clear from the location of the (zeros) of  $[1+C(s)P(S)]$  which is the denominator of the below closed loop transfer function.

$$CL(S) = \frac{C(s)P(s)}{1 + C(s)P(s)} \quad (13)$$

If there is a zero in right hand side of the (S) plane for  $[1+C(s)P(S)]$  then the closed loop system is unstable in other words Nyquist plot could be drawn for  $[1+C(s)P(S)]$  for both controllers and then check the phase margin or distance between their plots and the (-1) unstable point , higher distance means the system is better in terms of stability.

Fig (18) shows this Nyquist plot and proved that the closed loop system is asymptotically more stable in case of fuzzy logic PID controller at steady state as the phase margin is higher by using it than by using the conventional PID controller ,also the plot in case of fuzzy logic PID controller is far away from the (-1) unstable point than the plot in case of using the conventional PID.

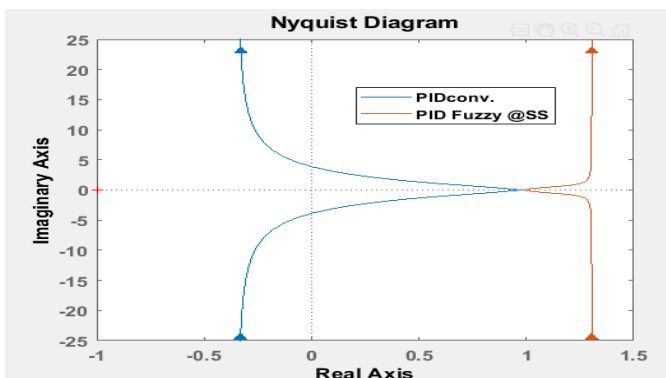


Fig (18) Nyquist plot for closed loop T.F

## VII. TABULATED RESULTS

In the TABLE II a comparison between both controllers in terms of overshoot, rise time ,settling time and stability margin is presented.

TABLE II. COMPARISON IN STEADY STATE AND TRANSIENT PERFORMANCE OF BOTH CONTROLLERS

Controller Type	System characteristics			
	Rise time (ms)	Percentage overshoot (%)	Settling time (ms)	Open loop Stability margin
PID controller	28.031	6.989	0.103	69 at 54.2 rad/sec
Fuzzy logic PID controller	23.903	0.314	0.052	104 at 15.2 rad/sec

## VIII. CONCLUSION

In this Paper, the fuzzy logic algorithm is used to perform online tuning for the parameters of a PID controller for a closed loop speed control of a DC motor, the new controller called the Fuzzy logic PID controller (FLCP) , it continuously update the parameters of the PID during the starting of the DC motor and any load disturbance in order maintain the desired operation based on the fuzzy rules and according to the inputs that are the error and it's rate of change. MATLAB/SIMULINK is used to simulate the whole system then a comparison in speed response between the new controller and the conventional PID controller is done in terms of overshoot, rise time, settling time and stability margin, the comparison showed that the fuzzy logic PID controller is faster and more robust because it's speed response has lower settling time, rise time and much lower overshoot compared to conventional PID. The speed response obtained by using the new controller is tested for reference speed that showed the robustness of the system. Finally by testing the stability of the system using both controllers, the fuzzy logic PID controller led the system to more stable operating point better than using the conventional PID because it increased the phase margin at steady state, in other words the system became more impregnable to small noises and disturbances. The Fuzzy logic PID controller could replace the conventional ones in applications that require much more accuracy in speed tracking as well as precise positioning.

## IX. APPENDIXES

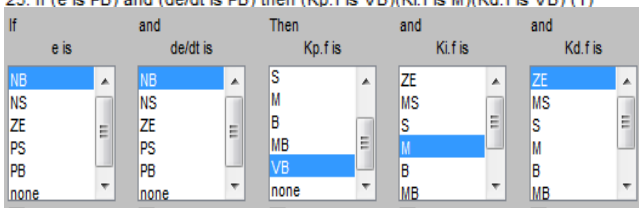
### A. Appendix (I): Parameters of the studied DC motor

#### DC MOTOR DATA

Parameter	Nomenclature	Value/unit
Moment of inertia	J	0.05 kg.m <sup>2</sup>
Friction coefficient	B	0.00 N.m.s/ rad
Machine constant	K <sub>f</sub>	0.836 N.m/A <sup>2</sup>
Armature resistance	R <sub>a</sub>	0.5 Ω
Armature inductance	L <sub>a</sub>	0.001 H
Rated voltage	U <sub>a</sub>	110 volt
Rated current	I <sub>a</sub>	10 A
Rated speed	Ω <sub>M</sub>	125.7 rad/sec

B. Appendix (II): 25 Fuzzy rules inserted in MATLAB

1. If (e is NB) and (de/dt is NB) then (Kp.f is VB)(Ki.f is M)(Kd.f is ZE) (1)
2. If (e is NS) and (de/dt is NB) then (Kp.f is VB)(Ki.f is M)(Kd.f is S) (1)
3. If (e is ZE) and (de/dt is NB) then (Kp.f is VB)(Ki.f is M)(Kd.f is M) (1)
4. If (e is PS) and (de/dt is NB) then (Kp.f is VB)(Ki.f is M)(Kd.f is MB) (1)
5. If (e is PB) and (de/dt is NB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)
6. If (e is NB) and (de/dt is NS) then (Kp.f is B)(Ki.f is S)(Kd.f is S) (1)
7. If (e is NS) and (de/dt is NS) then (Kp.f is B)(Ki.f is S)(Kd.f is B) (1)
8. If (e is ZE) and (de/dt is NS) then (Kp.f is B)(Ki.f is S)(Kd.f is MB) (1)
9. If (e is PS) and (de/dt is NS) then (Kp.f is MB)(Ki.f is S)(Kd.f is VB) (1)
10. If (e is PB) and (de/dt is NS) then (Kp.f is VB)(Ki.f is S)(Kd.f is VB) (1)
11. If (e is NB) and (de/dt is ZE) then (Kp.f is ZE)(Ki.f is MS)(Kd.f is M) (1)
12. If (e is NS) and (de/dt is ZE) then (Kp.f is ZE)(Ki.f is MS)(Kd.f is MB) (1)
13. If (e is ZE) and (de/dt is ZE) then (Kp.f is MS)(Ki.f is ZE)(Kd.f is MB) (1)
14. If (e is PS) and (de/dt is ZE) then (Kp.f is S)(Ki.f is MS)(Kd.f is VB) (1)
15. If (e is PB) and (de/dt is ZE) then (Kp.f is S)(Ki.f is MS)(Kd.f is VB) (1)
16. If (e is NB) and (de/dt is PS) then (Kp.f is B)(Ki.f is S)(Kd.f is B) (1)
17. If (e is NS) and (de/dt is PS) then (Kp.f is B)(Ki.f is S)(Kd.f is VB) (1)
18. If (e is ZE) and (de/dt is PS) then (Kp.f is B)(Ki.f is S)(Kd.f is VB) (1)
19. If (e is PS) and (de/dt is PS) then (Kp.f is MB)(Ki.f is S)(Kd.f is VB) (1)
20. If (e is PB) and (de/dt is PS) then (Kp.f is VB)(Ki.f is S)(Kd.f is VB) (1)
21. If (e is NB) and (de/dt is PB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)
22. If (e is NS) and (de/dt is PB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)
23. If (e is ZE) and (de/dt is PB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)
24. If (e is PS) and (de/dt is PB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)
25. If (e is PB) and (de/dt is PB) then (Kp.f is VB)(Ki.f is M)(Kd.f is VB) (1)



X. REFERENCES

- [1] Ray, L.R.; Stengel, R.F. (1991). "Stochastic Robustness of Linear-Time-Invariant Control Systems"
- [2] P. C. Sen, "Electric Motor Drives and Control: Past, Present and Future", IEEE Transaction on Industrial Electronics, Vol. IE37, No. 6, 1990, pp. 562-575
- [3] Robert A. Paz, "The Design of the PID Controller", Klipsch School of Electrical and Computer Engineering, 2001.
- [4] M. Kushwah and A. Patra, "Tuning PID controller for speed control of dc motor using soft computing techniques-a review," Advance in Electronic and Electric Engineering, vol. 4, pp. 141-148, 2014.
- [5] Shuang Cong, Guodong Li, Xianyong Feng, "Parameter identification of Non linear DC Motor model using compound evolution Algorithms, Proceedings of the World Congress on Engineering 2010 Vol I WCE 2010, June 30 - July 2, 2010, London, U.K.
- [6] More recent survey that covers the Ziegler-Nichols and Kappa-Tau tuning rules: "Automatic Tuning of PID Controllers," Karl J. Åström and Tore Hägglund, Chapter 52, The Control Handbook, IEEE/CRC Press, 1995, William S. Levine ed.
- [7] A. Kiam Heong, G. Chong, and L. Yun, "PID control system analysis, design, and technology," IEEE Transactions on Control Systems Technology, vol. 13, pp. 559-576, 2005
- [8] Kartik Sharma and Dheeraj Kumar Palwalia, "A modified PID control with adaptive fuzzy controller applied to DC motor" IEEE international conference on Information, Communication, Instrumentation and control ICICIC', (2017)
- [9] Von Altrock, Constantin (1995). Fuzzy logic and NeuroFuzzy applications explained.
- [10] Zhao, Z.-Y., Tomizuka, M., and Isaka, S. (2001) Fuzzy gain scheduling of PID controllers.
- [11] Visioli, A. (2000) Tuning of PID controllers with Fuzzy Logic.
- [12] Birdwell, J. D., & Yongmei Wang. (n.d.). Lyapunov stability analysis of systems using the fuzzy-PID controller. Proceedings of 1994 American Control Conference - ACC '94.
- [13] T. H. Chiew, Z. Jamaludin, A. Y. Bani Hashim, K. J. Leo, L. Abdullah and N. A. Rafan "Analysis of Tracking Performance in Machine Tools for Disturbance Forces Compensation using Sliding Mode Control and PID Controller" International Journal of Mechanical & Mechatronics Engineering IJMME-IJENS Vol:12 No:06