IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 10, Issue 1 Ver. III (Jan – Feb. 2015), PP 37-47
www.iosrjournals.org

DC Motor Position Control Using Fuzzy Proportional-Derivative Controllers With Different Defuzzification Methods

[1] Shravan Kumar Yadav

[1]Shravan Kumar Yadav has done B.Tech. degree program in Electrical & Electronics Engineering(EEE) at Apex Institute of Technology & Management , Bhubaneswar , India.

Abstract: *A fuzzy control system to control the position of a DC motor. The motor was modelled and converted to a subsystem in Simulink. First, a crisp proportional-derivative (PD) controller was designed and tuned using a Simulink block instead of conventional tuning methods such as hand-tuning or Ziegler-Nichols frequency response method. Then a fuzzy proportional-derivative (FPD) controller was designed and system responses of FPDs with different defuzzification methods were investigated. A disturbance signal was also applied to the input of the control system. FPD controller succeeded to reject the disturbance signal without further tuning of the parameters whereby crisp PD controller failed*

The purpose of this project is to control the position of DC Motor by using Fuzzy Logic Controller (FLC) with MATLAB application. The scope includes the simulation and modelling of DC motor, fuzzy controller and conventional PID controller as benchmark to the performance of fuzzy system. The position control is an adaptation of Closed Circuit Television (CCTV) system. Fuzzy Logic control can play important role because knowledge based design rules can be easily implemented in the system with unknown structure and it is going to be popular since the control design strategy is simple and practical. This make FLC an alternative method to the conventional PID control method used in nonlinear industrial system. The results obtained from FLC are compared with PID control for the dynamic response of the closed loop system. Parameters such as peak position in degree, settling time in second and maximum overshoot in percent will be part of the simulation result..

Keywords: *DC motor, Fuzzy logic control, defuzzification, PI controllers, PID controllers*

I. Introduction

Because of their high reliabilities, flexibilities and low costs, DC motors are widely used in industrial applications, robot manipulators and home appliances where speed and position control of motor are required. PID controllers are commonly used for motor control applications because of their simple structures and intuitionally comprehensible control algorithms. Controller parameters are generally tuned using hand-tuning or Ziegler-Nichols frequency response method. Both of these methods have successful results but long time and effort are required to obtain a satisfactory system. Two main problems encountered in motor control are the time-varying nature of motor parameters under operating conditions and existence of noise in system loop.

Analysis and control of complex, nonlinear and/or time-varying systems is a challenging task using conventional methods because of uncertainties. Fuzzy set theory (Zadeh, 1965) which led to a new control method called Fuzzy Control which is able to cope with system uncertainties. One of the most important advantages of fuzzy control is that it can be successfully applied to control nonlinear complex systems using an operator experiences or control engineering knowledge without any mathematical model of the plant (Assilian, 1974), (Kickert, 1976). DC motor control is generally realized by adjusting the terminal voltage applied to the armature but other methods such as adjusting the field resistance, inserting a resistor in series with the armature circuit are also available (Chapman, 2005).

Ziegler-Nichols frequency response method is usually used to adjust the parameters of the PID controllers. However, it is needed to get the system into the oscillation mode to realize the tuning procedure. But it's not always possible to get most of the technological plants into oscillation. The proposed approach uses both fuzzy controllers and response optimization method to obtain the approximate values of the controller parameters. Then the parameters may be slightly varied to obtain the user-defined performance of the real-time control system. Thus, it's an actual problem to design adaptive PID controllers without getting the system into the oscillation mode. In the next section, the mathematical model of a dc motor is used to obtain a transfer function between shaft position and applied armature voltage. This model is then built in MATLAB Simulink, design and tuning of proportional-integral-derivative (PID) controllers are reviewed and a crisp PD control system is designed in Simulink with the proposed design procedure, it's mentioned about the fuzzy logic controller design issues and a fuzzy proportional-derivative controller is designed with the proposed approach. Some of the commonly used defuzzification methods are discussed and system responses with different defuzzification methods are compared. Finally disturbance rejection capabilities of the designed controllers are investigated.

II. Dc Motor Model

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 2.1 shows a separately excited DC motor equivalent model.

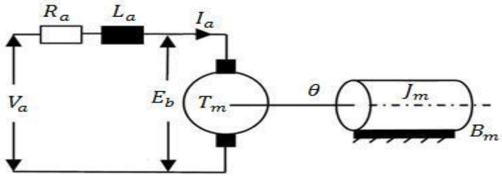


Figure 2.1 DC motor model

$$v_a(t) = R_a.i_a(t) + L_a.\frac{di_a(t)}{dt} + e_b(t) \tag{1}$$

$$e_b(t) = K_b.w(t) \tag{2}$$

$$T_m(t) = K_T.i_a(t) \tag{3}$$

$$T_m(t) = J_m.\frac{dw(t)}{dt} + B_m.w(t) \tag{4}$$

where V_a = armature voltage (V)
R_a = armature resistance (Ω)
L_a = armature inductance (H)
I_a = armature current (A)
E_b = back emf (V)
w = angular speed (rad/s)
T_m = motor torque (Nm)
θ = angular position of rotor shaft (rad)
J_m = rotor inertia (kgm^2)
B_m = viscous friction coefficient (Nms/rad)
K_T = torque constant (Nm/A)
K_b = back emf constant (Vs/rad)

Let us combine the upper equations together

$$v_a(t) = R_a.i_a(t) + L_a.\frac{di_a(t)}{dt} + K_b.w(t) \tag{5}$$

$$K_T.i_a(t) = J_m.\frac{dw(t)}{dt} + B_m.w(t) \tag{6}$$

Laplace transforms of (5) and (6) are:

$$V_a(s) = R_a.I_a(s) + L_a.I_a(s).s + K_b.W(s) \tag{7}$$

$$K_T.I_a(s) = J_m.W(s).s + B_m.W(s) \tag{8}$$

If current is obtained from (8) and substituted in (7) we have

$$V_a(s) = W(s).\frac{1}{K_T}.[L_a.J_m.s^2 + (R_a.J_m + L_a.B_m).s + (R_a.B_m + K_b.K_T)] \tag{9}$$

Then the relation between rotor shaft speed and applied armature voltage is represented by transfer function:

$$\frac{W(s)}{V_a(s)} = \frac{K_T}{L_a.J_m.s^2 + (R_a.J_m + L_a.B_m).s + (R_a.B_m + K_b.K_T)} \tag{10}$$

The relation between position and speed is:

$$\theta(s) = \frac{1}{s}W(s) \tag{11}$$

Then the transfer function between shaft position and armature voltage at no-load is:

$$\frac{\theta(s)}{V_a(s)} = \frac{K_T}{L_a.J_m.s^3 + (R_a.J_m + L_a.B_m).s^2 + (K_T.K_b + R_a.B_m).s} \tag{12}$$

Figure 1.2 shows the DC motor model built in Simulink. Motor model was converted to a 2-in 2-out subsystem. Input ports are armature voltage (Va) and load torque (Tload) and the output ports are angular speed in (w) and position (teta).

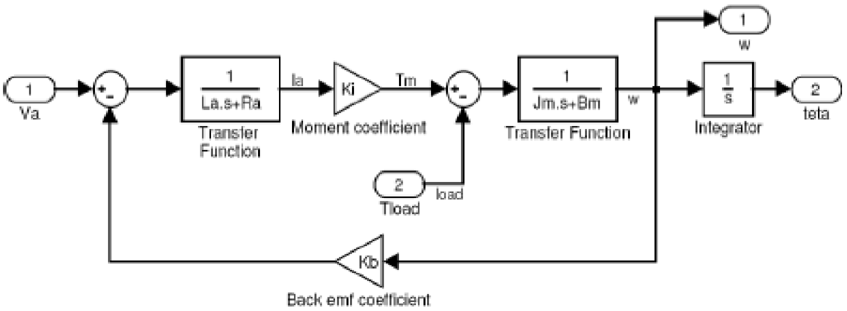


Figure 2.2 Simulink model.

A 3.70 kW, 240V, 1750 rpm DC motor with the below parameters was used:

$R_a = 11.2\ \Omega$
$L_a = 0.1215\ H$
$J_m = 0.02215\ kgm^2$
$B_m = 0.002953\ Nms/rad$
$K_i = 1.28\ Nm/A$
$K_b = 1.28\ Vs/rad$

DC motors are most suitable for wide range speed control and are there for many adjustable speed drives. Intentional speed variation carried out manually or automatically to control the speed of DC motors.

III. Proportional-Integral-Derivative (Pid) Controller

3.1 Basic

PID controllers are widely used in industrial control applications due to their simple structures, comprehensible control algorithms and low costs. Figure 3.1 shows the schematic model of a control system with a PID controller.

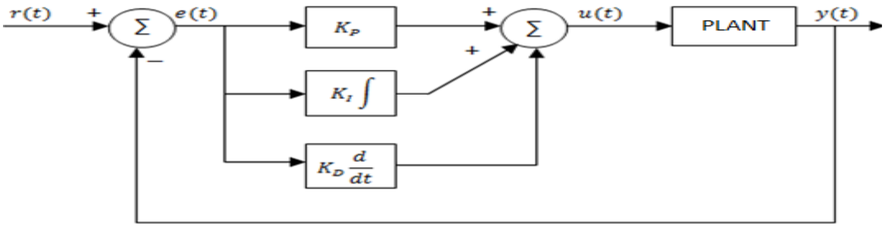


Figure 3.1 PID control system

Control signal u(t) is a linear combination of error e(t) , its integral and derivative

$$u(t) = K_P e(t) + K_I \int e(t)dt + K_D \frac{de(t)}{dt} \qquad (13)$$

$$u(t) = K_P \left(e(t) + \frac{1}{T_I} \int e(t)dt + T_D \frac{de(t)}{dt} \right) \qquad (14)$$

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 (14) can be approximated

as:

$$u(n) = K_P \left(e(n) + \frac{1}{T_I} \sum_{j=1}^{n} e(j) T_s + T_D \frac{e(n) - e(n-1)}{T_s} \right) \qquad (15)$$

3.2 Conventional Controller

classical controllers like PI or PID controllers are widely used in process industries because of their simple structure, assure acceptable performance for industrial processes and their tuning is well known among all industrial operators. However, these controllers provide better performance only at particular operating range and they need to be retuned if the operating range is changed. Further, the conventional controller performance is not up to the expected level for nonlinear and dead time processes. In the present industrial scenario, all the processes require automatic control with good performance over a wide operating range with simple design and implementation. Typically two types of conventional controller will be discused in this report namely Proportional-Integral (PI)and Proportional-Integral-Derivative (PID).Both have a signifant functions toward the development of DC servo motor controller

3.3 PI Controller

PI controller is unquestionably the most commonly used control algorithm the process control industry. The main reason is its relatively simple structure, which can be easily understood and implemented in practice, and that many sophisticated control strategies, such as model predictive control, are based on it.

3.4 PID Controller

Conventional PID controllers are characterised with simple structure and simple design procedures. They enable good control performance and are therefore widely applied in industry. However, in a number of cases, such as those when parameter variations take place and/or when disturbances are present, control system based on a fuzzy logic controller (FLC) may be a better choice.

The PID controller is a universal controller which is used particularly in the field of material processing. Practical controllers are usually assembled with one or more operational amplifiers, whereby the PID behaviour is realised by suitable feedbacks.

Several approaches were developed for tuning PID controller such as the Ziegler-Nichols (Z-N) method, the Cohen-Coon (C-C) method, integral of squared time weighted error rule (ISTE), integral of absolute error criteria (IAE), internal-model-control (IMC) based method and gain-phase margin method (Taifour et al, 2012).

It is a generic control loop feedback mechanism (controller) widely used in industrial control systems. A PID controller calculates an "error" value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs.

The PID controller calculation (algorithm) involves three separate constant parameters, and is accordingly sometimes called three-term control: the proportional, the integral and derivative values, denoted P, I, and D.

The influence of the three components can usually be set externally. Each of the three components covers one of the controller's tasks.

i. the P part by the proportional sensitivity Kp .
ii. the I part by the integral action time Tn .
iii. the D part by the derivative action time Tv .

The D part therefore ensures that the controller reacts quickly even in the case of slow changes at its input. The P part takes care of medium amplification and the I part causes the controller to operate accurately without leaving a control difference. deriving the individual controller parameters from the jump reply or rise reply is difficult since the three components overlap.

PID controllers are usually tuned using hand tuning or Ziegler-Nichols methods to obtain the desired performance according to preset criteria. The basic continuous feedback control is PID controller. The PID controller exhibits good performance but is not adaptive enough.

3.5 Tuning PID parameters

PID controllers are usually tuned using hand-tuning or Ziegler-Nichols methods (Jantzen, 2007). Hand-tuning is generally used by experienced control engineers based on the rules shown in Table 1. But these rules are not always valid. For example if an integrator exists in the plant, then increasing K_p results in a more stable control

Table 1 Hand-tuning rules

Operation	Rise time	Overshoot	Stability
K_P ↑	Faster	Increases	Decreases
T_D ↑	Slower	Decreases	Increases
$1/T_I$ ↑	Faster	Increases	Decreases

A simple hand-tuning procedure is as follows:

1. Remove derivative and integral actions by setting T_D =0 and $1/T_I$=0
2. Tune K_P such that it gives the desired response except the final offset value from the set point
3. Increase K_P slightly and adjust T_D to dampen the overshoot
4. Tune $1/T_I$ =0 such that final offset is removed
5. Repeat steps from 3 until K_P as large as possible

The disadvantage of this method is that it should take a long time to find the optimal values. Another method to tune PID parameters is Ziegler-Nichols frequency response method. The procedure is as follows:

1. Increase K_P until system response oscillates with a constant amplitude and record that gain value as K_u (ultimate gain)
2. Calculate the oscillation period and record it as T_u
3. Tune parameters using Table 2

Table 2 Ziegler-Nichols rules

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

Ziegler-Nichols frequency response method gives poor results especially for the systems with a time lag much greater than the dominating time constant (Jantzen, 2007). Damping is generally poor. Rules work better for PID controllers than PI controllers and it is not stated how to calculate the parameters for a PD controller.

Another method proposed by Ziegler and Nichols is the reaction curve or step response method where the unit-step response of the plant is used to adjust parameters. But the plant must not involve any integrators or dominant complex conjugate poles for this method to apply (Ogata, 1997).

3.6 PD controller design

A PD controller was designed to control the DC motor. Control signal of a PD controller is as follows:

$$u(n) = K_P\left(e(n) + T_D\frac{e(n) - e(n-1)}{T_s}\right) \qquad (16)$$

Controller parameters were tuned using Signal Constraint block of Simulink Response Optimization Toolbox instead of conventional methods.

Signal Constraint is a block where response signals can be graphically constrained and model parameters should be automatically optimized to obtain the performance requirements (Mathworks, 2008).

Performance criteria were specified as:
Rise time $t_r \leq 1$ s
Settling time $t_s \leq 2$ s
Maximum overshoo $M_p \leq 10\%$
Steady state error(e) $\leq 1\%$
The objective in control system design is to find a control signal that satisfies the performance requirements (Veremey).

3.6 Simulink implementation

Figure 3.2 shows the PD control system designed in MATLAB Simulink where controller coefficients were adjusted using the Signal Constraint block. Integral coefficient of PID controller was set to zero.

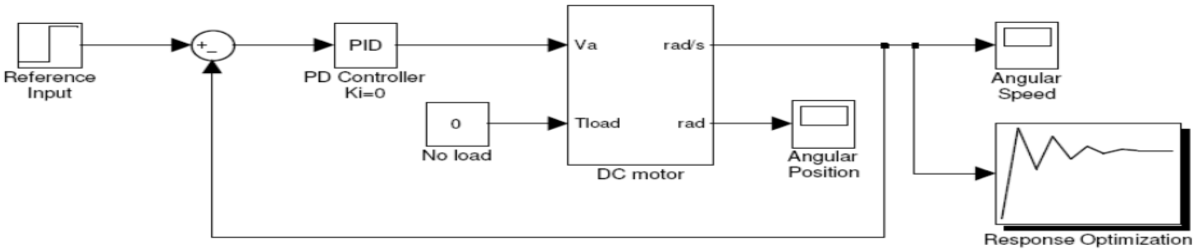


Figure 3.2 Crisp PD control system

Overshoot is not desired especially in position control systems. It can be seen from Figure 5 that Signal Constraint block adjusted the parameters such that a very small overshoot occurs. Table 3 shows the values of the performance criteria obtained with the adjusted controller parameter.

Table 2 Performance specifications for crisp PD control system

t_r	0.64s
t_s	0.88s
M_p	0.2%
e	None

IV. Fuzzy Logic Controller

A fuzzy logic controller has four main components as shown in Figure 5.1, fuzzification interface, inference mechanism, rule base and defuzzification interface. FLCs are complex, nonlinear controllers. Therefore it's difficult to predict how the rise time, settling time or steady state error is affected when controller parameters or control rules are changed. On the contrary, PID controllers are simple, linear controllers which consist of linear combinations of three signals.

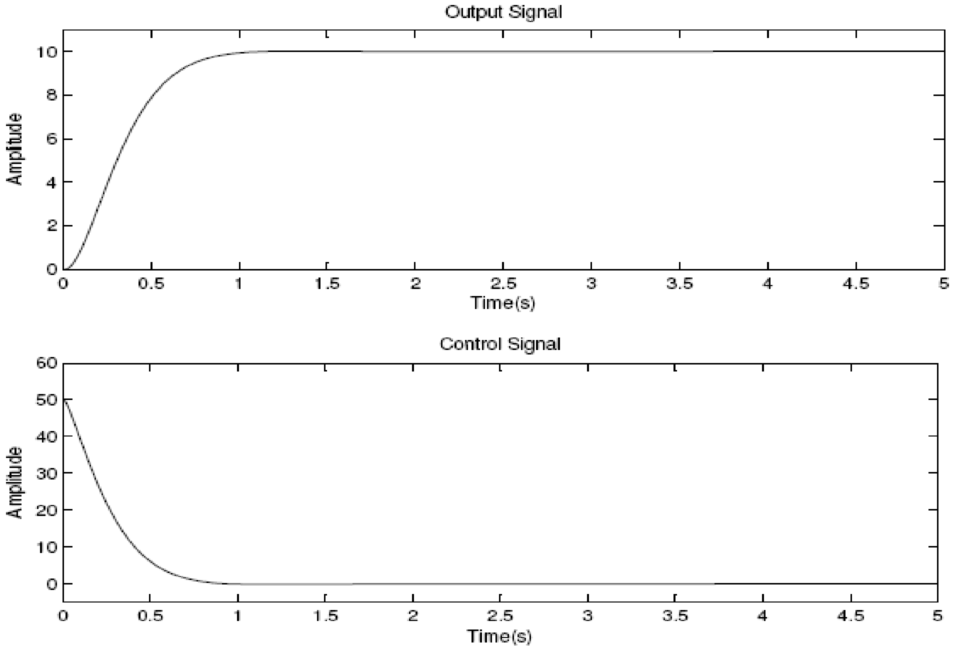


Figure 4.1 Output and control signals for crisp PD control system

Implementation of an FLC requires the choice of four key factors (Mamdani, 1977): number of fuzzy sets that constitute linguistic variables, mapping of the measurements onto the support sets, control protocol that determines the controller behaviour and shape of membership functions. Thus, FLCs can be tuned not just by adjusting controller parameters but also by changing control rules, membership functions etc.

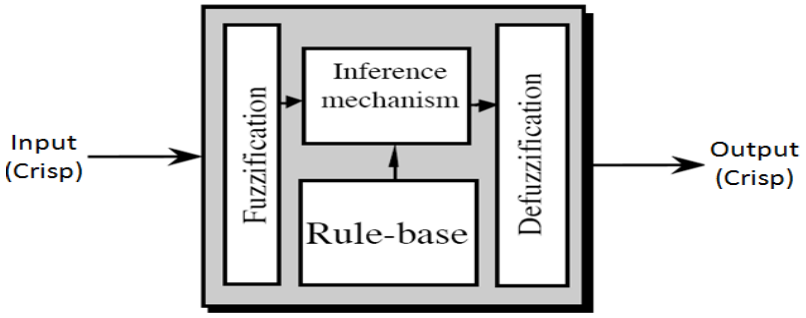


Figure 4.2 Fuzzy logic controller

4.1 Fuzzy Logic Control

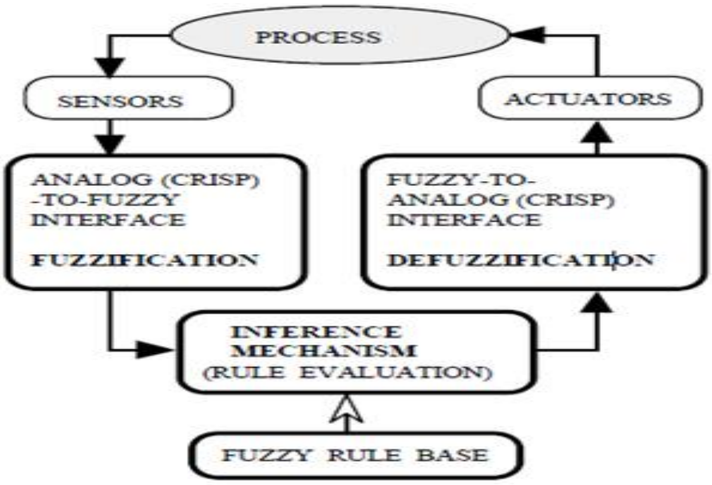


Figure 4.3 Fuzzy logic control

- All machines can process crisp or classical data such as either '0' or '1'.
- The crisp input and output must be converted to linguistic variables with fuzzy components.
- For converting the crisp value to fuzzy data is done by first step –Fuzzification .
- In the second step, to begin the fuzzy inference process, one need combine the Membership Functions with the control rules to derive the control output, and arrange those outputs into a table called the lookup table.
- Furthermore, that control output should be converted from the linguistic variable back to the crisp variable and output to the control operator. This process is called defuzzification or step 3.

4.2 Structure of Fuzzy Logic Controller (FLC)

A typical FLC consists of three basic components, namely input signal fuzzification, a fuzzy engine and output signal defuzzification. The fuzzification block tranforms the continous input signal into linquistic fuzzy variable. The fuzzy engine handles rule inference where human experience can easily be injected through linguistic rules. The defuzzification block transforms the fuzzy control actions to continuous (crisp) signals which can be applied to the physical plant. The knowledge base includes fuzzy sets, which are defined on the interval of the inputs and outputs of the FLC, and rule base, which is constructed from fuzzy implication.
The error and error change for both position and time are scaled using appropriate scaling factors.
These scaled input data then converted into linguistic variables which may be viewed as labels of fuzzy sets.

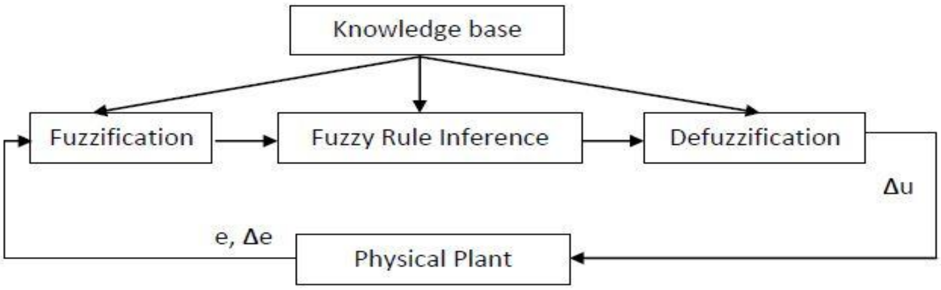


Figure 4.4 Typical configuration of a fuzzy logic controller

V. Defuzzification And The Look Up Table

The conclusion or control output derived from the combination of input, output membership functions and fuzzy rules is still a vague or fuzzy element, and this process in called fuzzy inference. To make that conclusion or fuzzy output available to real applications, a defuzzification process is needed. The defuzzification process is meant to convert the fuzzy output back to the crisp or classical output to the control objective. Remember, the fuzzy conclusion or output is still a linguistic variable, and this linguistic variable needs to be converted to the crisp variable via the defuzzification process.

5.1 Defuzzification methods

Defuzzification interface uses the implied fuzzy sets or the overall implied fuzzy set to obtain a crisp output

value. There are many defuzzification methods but the most common methods are as follows:
1) Center of gravity (COG)
2) Bisector of area (BOA)
3) Smallest of maximum (SOM)
4) Mean of maximum (MOM)
5) Largest of maximum (LOM)

For discrete sets COG is called center of gravity for singletons (COGS) where the crisp control value u_{COGS} is the abscissa of the center of gravity of the fuzzy set u_{COGS}. is calculated as follows:

$$u_{COGS} = \frac{\sum_i \mu_c(x_i)x_i}{\sum_i \mu_c(x_i)}$$

Where x_i is a point in the universe of the conclusion (i=1,2,3…) and μ_c is the membership value of the resulting conclusion set. For continuous sets summations are replaced by integrals.

The bisector of area (BOA) defuzzification method calculates the abscissa of the vertical line that divides the area of the resulting membership function into two equal areas. For discrete sets, μ_{BOA} is the abscissa x_j that minimizes.

$$\left|\sum_{i=1}^{j} \mu_c(x_i) - \sum_{i=j+1}^{i_{max}} \mu_c(x_i)\right|, \quad i < j < i_{max}$$

Here imax is the index of the largest abscissa ximax . BOA is a computationally complex method.

Another approach to obtain the crisp value is to choose the point with the highest membership. There may be several points in the overall implied fuzzy set which have maximum membership value. Therefore it's a common practice to calculate the mean value of these points. This method is called mean of maximum (MOM) and the crisp value is calculated as follows:

$$u_{MOM} = \frac{\sum_{i \in I} x_i}{|I|}, \quad I = \{i \mid \mu_c(x_i) = \mu_{max}\}$$

Here I is the (crisp) set of indices i where reaches its maximum μ_{max} , and |I| is its cardinality (the number of members).

One can also choose the leftmost point among the points which have maximum membership to the overall implied fuzzy set. This method is called smallest of maximum (SOM) or the leftmost maximum (LM) defuzzification method. Crisp value is calculated as follows:

$$u_{SOM} = x_{\min(I)}$$

Another possibility is to choose the rightmost point among the points which have maximum membership to the overall implied fuzzy set. This method is called largest of maximum (LOM) or the rightmost maximum (RM) defuzzification method where crisp value is
calculated as

$$u_{LOM} = x_{\max(I)}$$

VI. Advantage Of Using Fuzzy Logic Controller

The advantages provided by a FLC are listed below:

- It provides a hint of human intelligence to the controller.
- It is cost effective.
- No mathematical modeling of the system is required.
- It is simple to design.
- Linguistic variables are used instead of numerical ones.
- Non-linearity of the system can be handled easily.
- System response is fast.
- Reliability of the system is increased.
- High degree of precision is achieved.

VII. Simulink Implementation

Inputs of FPD are "error" and "change of error" where the output is "control". Input and output variables of FPD consist of seven fuzzy sets namely NB (negative big), NM (negative medium), NS (negative small), Z (zero), PS (positive small), PM (positive medium) and PB (positive big) as shown in Figure 8.1 and 8.2

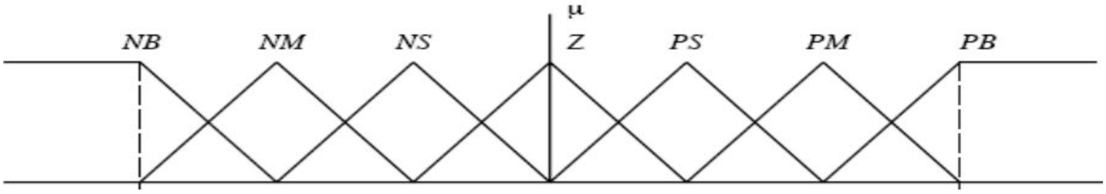


Figure 7.1 Fuzzy input variables "error" and "change of error"

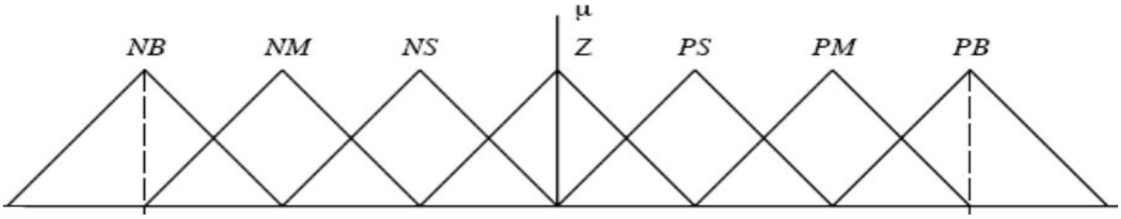


Figure 7.2 Fuzzy output variable "output"

Figure 7.3 shows the fuzzy PD control system designed in Simulink

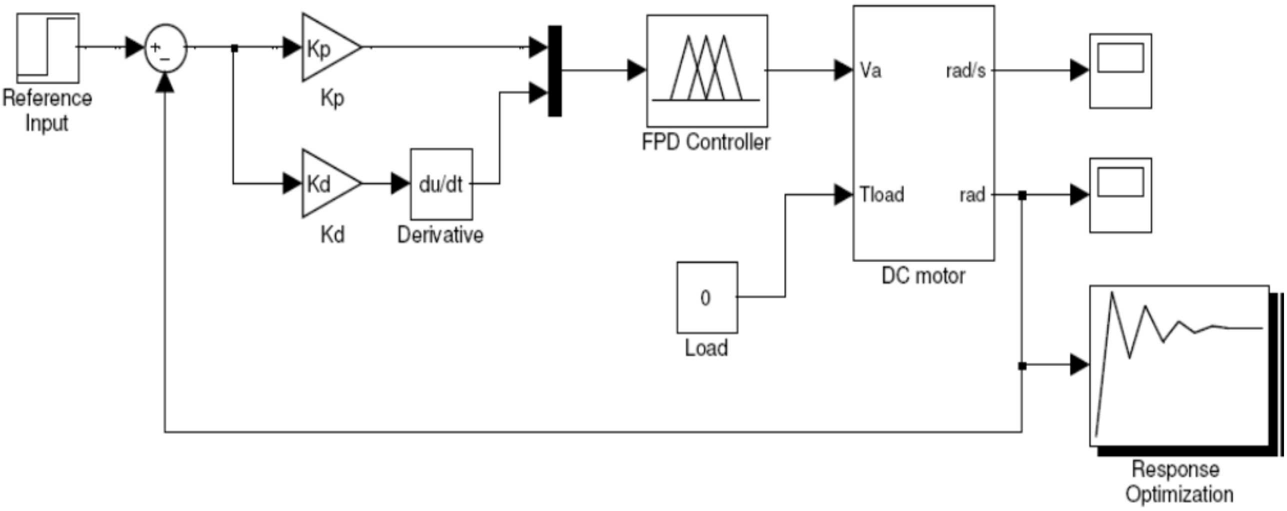
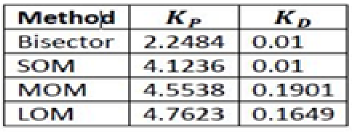


Figure 7.3 Fuzzy PD control system

Different defuzzification methods were used to obtain the control signal. Table 4 shows the tuned values of the controller parameters for different defuzzification methods

Table 3 Controller parameters for different defuzzification methods



Method	K_P	K_D
Bisector	2.2484	0.01
SOM	4.1236	0.01
MOM	4.5538	0.1901
LOM	4.7623	0.1649

Disturbance rejection is important in controller design. The controller must be able to dampen out the effects of disturbance signals existing in the system loop. Therefore a disturbance signal (Gaussian type noise with zero mean and 0.05 variance) was applied to the input of the control system.

VIII. Result

Figure 8.4 and 8.5 shows the system responses and control signals for the fuzzy control systems with different defuzzification methods.

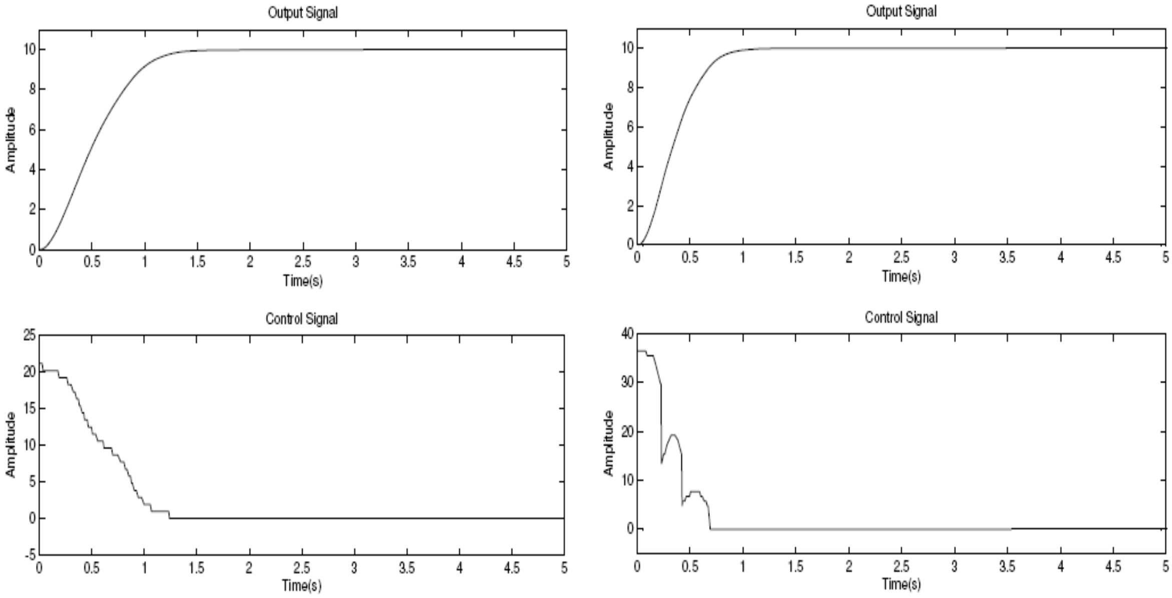


Figure 8.1 Output of Bisector and SOM

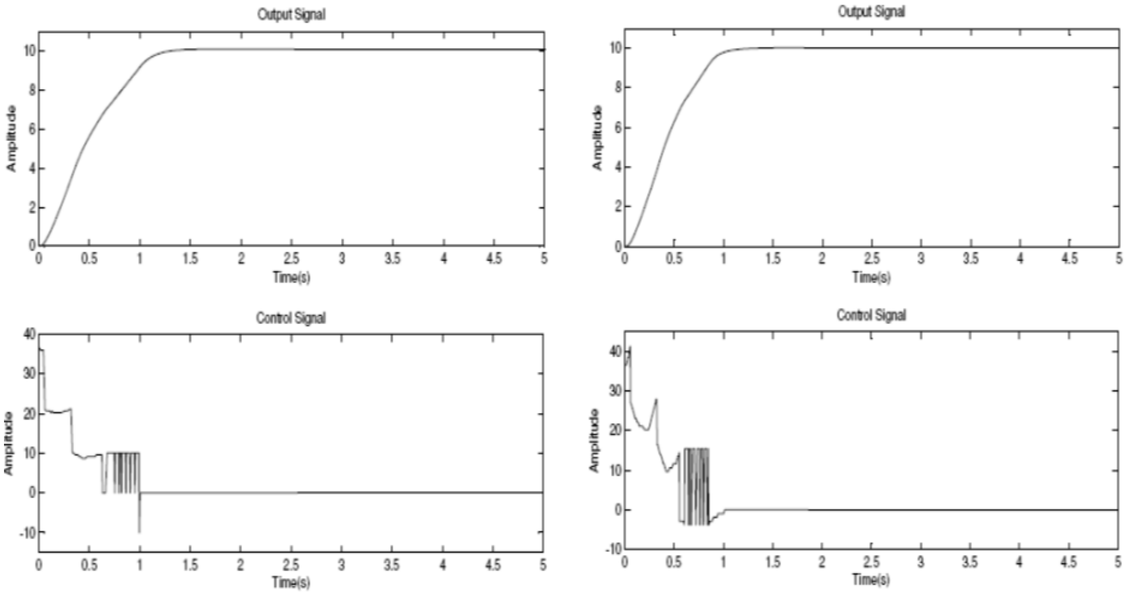


Figure 8.2 Ouput of MOM and LOM

IX. Conclusion And Future Scope

From the study of the fuzzy logic and its industrial applications , fuzzy logic has emerged as one of the most active and fruitful areas for research in the applications of fuzzy set theory, especially in the realm of industrial processes, which do not lead themselves to control by conventional methods because of lack of quantitative data regarding the input-output relations. Fuzzy control is based on fuzzy logic –a logical system which is much closer in spirit to human thinking and natural language than traditional logical systems. The fuzzy logic controller (FLC) based on fuzzy logic provides a means of converting a linguistic control strategy.

In spite of the easy implementation of traditional control "PI", its response is not so good for non-linear systems. The improvement is remarkable when controls with Fuzzy logic are used, obtaining a better dynamic response from the system.

The basic concepts of fuzzy logic and design of the system based on this logic has been studied . The Fuzzy Logic Controller was designed so as to achieve desirable results. This controller can be implemented in different practical applications of motors, the feasibility of the controller in the corresponding applications can be studied and changes can be made according to the requirement. Different strategies like Genetic Algorithm can also be applied for tuning the controller. Also, instead of just fuzzy controller.

Reference

[1]. Warwick K. "Control System",Prentice-Hall, 1989.

[2]. Y.F.Li and C.C. Lau, "Developement of fuzzy algorithm for servo sys tems," IEEE Control System, April 1989, pp.65-72.

[3]. S.M. Smith and D.J. Comer, "Automated Calibration of a fuzzy Logic Controller Using a Cell State Space Algorithm," IEEE Control System. August 1991,pp.18-28

[4]. Lee C. "Fuzzy Logic in Control Systems",Fuzzy logic controller, part II IEEE Trans. On Systems, Man. And cybernetics 1990.

[5]. Jantzen, J., Foundations of Fuzzy Control, WS: John Wiley & Sons, Ltd., 2007.

[6]. Kickert, W. J. M. and van Nauta Lemke, H. R., Application of a Fuzzy Controller in Warm Water Plant. Automatica, 12(4), 301-308, 197

Shravan Kumar Yadav S/O Dr. Sheo Bhajan Ram Yadav was born in (Jharkhand) in India, on April, 1992.

He has done B.Tech. Degree program (4th year) in Electrical & Electronics Engineering (EEE) at Apex Institute of Technology & Management, Bhubaneswar, India. He has published more than two research papers and successfully completed his four weeks industrial training from 09 June 2013 to 10 July 2013 at NALCO, Angul, India (A Government of India Enterprise), India, PH-+ 91-9040316409.
(E-mail: callshravanjsr@gmail.com).