ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 3, March 2014

Speed control of a Separately Excited DC Motor Using Optimization techniques

S. Tharani, P. Palpandian, N. Gowthaman

Assistant Professor, Department of Electronics and Instrumentation Engineering, Karpagam College of Engineering,

Coimbatore, Tamil Nadu, India.

ABSTRACT: This paper examine the implementation of soft computing techniques for the speed control of separately excited DC motor. The mathematical modeling of separately excited DC motor is design with the specification of DC motor. The most commonly used controller for the speed control of DC motor is Proportional-Integral (PI) controller. The PI controller has some disadvantages such as the high starting overshoot, sensitivity to controller gains and sluggish response due to sudden disturbance. The Fuzzy logic controller and PSO based PID controller is proposed to overcome the disadvantages of the PI controller. With the application of appropriate expert rules, there is no overshoot and the settling time is within the desired value. The manual tuning is eliminated and intelligent tuning takes the centre stage with satisfactory performance.

KEYWORDS: Separately excited DC motor, PID controller, Fuzzy logic controller and PSO-PID.

I. INTRODUCTION

The development of high performance motor drives is very important in industrial as well as other purpose applications such as steel rolling mills, electric trains and robotics. Generally, a high performance motor drive system must have good dynamic speed command tracking and load regulating response to perform task. DC drives, because of their simplicity, ease of application, high reliabilities, flexibilities and favorable cost have long been a backbone of industrial applications, robot manipulators and home appliances where speed and position control of motor are required. DC drives are less complex with a single power conversion from AC to DC. Again the speed torque characteristics of DC motors are much more superior to that of AC motors. A DC motors provide excellent control of speed for acceleration and deceleration. DC drives are normally less expensive for most horsepower ratings. DC motors have a long tradition of use as adjustable speed machines and a wide range of options have evolved for this purpose[2]. In these applications, the motor should be precisely controlled to give the desired performance. The controllers of the speed that are conceived for goal to control the speed of DC motor to execute one variety of tasks, is of several conventional and numeric controller types, the controllers can be: proportional – integral (PI),

proportional – integral – derivative (PID), PI-PD, I-PD. The proportional – integral – derivative (PID) controller operates the majority of the control system in the world. It has been reported that more than 95% of the controllers in the industrial process control applications are of PID type as no other controller match the simplicity, clear functionality, applicability and ease of use offered by the PID controller. PID controller provides robust and reliable performance for most systems if the PID parameters are tuned properly.

The major problems in applying a conventional control algorithm (PI, PD, PID) in a speed controller are the effects of non-linearity in a DC motor. The nonlinear characteristics of a DC motor such as saturation and fiction could degrade the performance of conventional controllers. Generally, an accurate nonlinear model of an actual DC motor is difficult to find and parameter obtained from systems identification may be only approximated values.

In general, the PID controller is structurally suitable and shows an acceptable control performance for many open loop stable processes. However, it has structural limitations in controlling certain types of plants such as unstable, integrating and resonant processes[6]. A Fuzzy logic controller and PSO based PID controller, which corresponds to a PI controller of a plant transfer function changed by the PD feedback, can produce improved control in several situations. Using an internal PD feedback loop can convert an open loop unstable process to an open loop stable process and for resonant or integrating processes can ensure appropriate locations of the open loop stable process poles. Therefore, the PSO based PID controller and Fuzzy logic controller has advantages over the conventional PID controllers. On the other hand, conventional design procedures are based on a plant with fixed parameters although most practical system models have uncertainties. Therefore, design of a satisfactory control scheme requires the consideration of robustness to parameter uncertainties, to stability and performance.

II. MODELING OF DC MOTOR

2.1 Speed control of separately excited DC motor:

The term speed control stand for intentional speed variation carried out manually or automatically DC motors are most suitable for wide range speed control and are there for many adjustable speed drives[1].

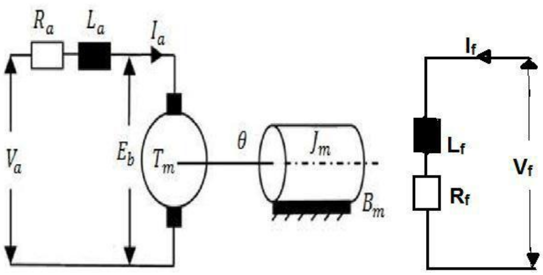


Fig 1: Circuit diagram of Separately excited DC motor

from the figure(1)

Va is the armature voltage. (volt)
Eb is back emf the motor (volt)
Ia is the armature current (ampere)
Ra is the armature resistance (ohm)
La is the armature inductance (Henry)
Tm is the mechanical torque developed (N-m)
Jm is moment of inertia (kg/m^2)
Bm is friction coefficient (N-m/(rad/sec))

angular velocity of a motor is,

$$\omega\ \alpha\ (Va-IaRa)/\varphi$$

$$\omega= (Va-IaRa)/ Ka\varphi\text{--------}\ (1)$$

Where,

Ka=Armature constant= PZ/2πa

From the equation (1) it is clear that, speed of a DC motor can be controlled by three methods[4].
They are:-

1- Variation of resistance in armature circuit.
2- Variation of field flux.
3- Variation of armature terminal voltage.

2.2 Mathematical modeling of DC motor:

The armature voltage equation is given by:

$Va = Eb + IaRa + La\,(dIa/dt)$-----------(2)

Now the torque balance equation will be given by:

$Tm = Jmd\omega / dt + Bm\omega + T_L$-----------(3)

Where,

T_L is load torque in Nm.

Friction in rotor of motor is very small (can be neglected), so Bm= 0

Therefore, new torque balance equation will be given by:

$Tm = Jmd\omega / dt + TL$ --------- (4)

Taking field flux as and Back EMF Constant as K. Equation for back emf of motor will be:

$Eb = K\varphi\omega$ -------------- (5)

$Tm = K\,\varphi\,Ia$ ------------- (6)

Taking Laplace transform of the motor s armature voltage equation we get

$Ia\,(S) = (Va - Eb)/\,(Ra + LaS)$

Now, taking equation (ii) into consideration, we have:

$Ia(s) = (Va - K\,\varphi\omega\,)/\,Ra\,(1 + LaS/Ra)$

And,

$\omega\,(s) = (Tm - TL)\,/JS = (K\varphi\,Ia - TL)\,/JmS$

(Armature Time Constant) $Ta = La/Ra$

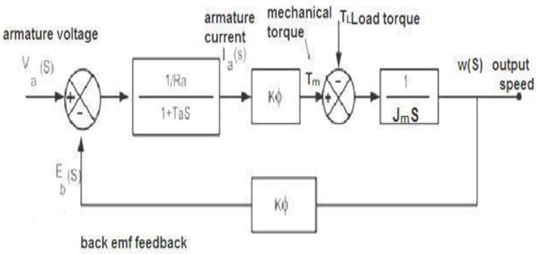


Fig 2: Model of Separately Excited DC motor

After simplifying the above motor model, the overall transfer function will be

$\omega(s) / Va(s) = [K\,\varphi\,/Ra]\,/JmS\,(1+TaS)\,/\,[\,1 + (K^2\,\varphi^2$ $/Ra)\,/JmS(1+TaS)]$

Further simplifying the above transfer function:

$\omega(s)/Va(s) = (1\,/k\,\varphi)\,/\{\,1 + (k^2\,\varphi^2\,/Ra)\,/JmS(1+TaS)\}$ -----------------------(7)

Assuming, $Tem = JmRa/\,(k\,\varphi)^2$ as electromechanical time constant.

Then the above transfer function can be written as:

$\omega(s)\,/Va(s) = (1/k\,\varphi)/\,[STem\,(1+STa) + 1]$------- (8)

Let us assume that during starting of motor, load torque TL = 0 and applying full voltage Va Also assuming negligible armature inductance, the basic
armature voltage equation can be written as:

$Va = K\varphi\omega\,(t) + IaRa$

At the same Torque equation will be:

$Tm = Jmd\omega\,/dt = K\,\varphi\,Ia$ ----- (9)

Putting the value of Ia in above armature equation:

$Va = K\,\varphi\omega\,(t) + (Jmd\omega\,/dt)\,Ra/\,K\varphi$

Dividing on both sides by K,

$Va/K\,\varphi = \omega\,(t) + JmRa\,(d\,\omega\,/dt)/\,(K\,\varphi)^2$ -- (10)

$Va/K\,\varphi$ is the value of motor speed under no load condition. Therefore,

ω (no load) $= \omega\,(t) + JmRa(d\,\omega\,/dt)/(K\varphi\,)^2 = \omega(t) + Tem\,(d\,\omega/dt)$

Where,

$K\,\varphi = Km$ (say)

$Tem = JmRa/(K\varphi)^2 = JRa/(Km)^2$

Therefore,

$Jm = Tem\,(Km)^2/Ra$ --------- (11)

From motor torque equation, we have:

$\omega(s) = KmIa(s)/JS - TL/JmS$ ------- (12)

From equation (11) and (12), we have:

$\omega(s) = [(Ra/Km)\,Ia(s) - TL\,Ra/(Km)^2]\,(1/Tem(s)$

Now, Replacing $K\varphi$ by Km in equation (8), we will get:

$\omega(s)/Va(s) = (1/Km)/(1+STem+S^2TaTem)$ ------(13)

The armature time constant Ta is very much less than the electromechanical time constant Tem, (Ta << Tem)

Simplifying,

$1 + STem + S^2TaTem\ 1 + S\,(Ta+Tem) + S^2TaTem = (1 + STem)\,(1 + STa).$

The equation can be written as:

$\omega(s)/Va(s) = (1/Km)/((1 + STem)(1 + STa))$ ----(14)

Tem and Ta are the time constants of the above system transfer function which will determine the response of the system. Hence the dc motor can be
replaced by the transfer function obtained in equation (14) in the DC drive model[3][7].

Specification of the separately excited DC Motor:

Armature resistance (Ra) = 7.56Ω
Armature inductance (La) =0.055 H
Rated voltage =200 V
Mechanical inertia (Jm) = 0.068Kg.m^2
Friction coefficient (Bm) =0.008 N-m/rad/sec
Back emf constant (k) = 3.475V/rad/sec
Rated speed = 550r.p.m

III. REVIEW OF CONTROLLERS

3.1 Conventional 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. 3:

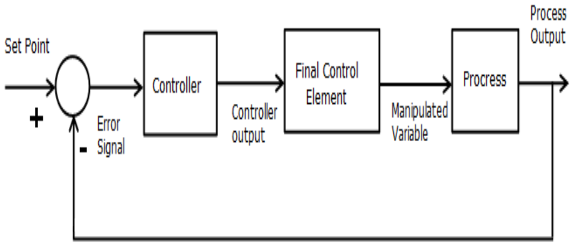


Fig 3: PID Controller

A simple feedback control theory is utilized to represent the overall PID controlled system[5].

This PID controller has the transfer function of the form:

$$K_p + \frac{K_I}{S} + K_D S = \frac{K_D S^2 + K_P S + K_I}{S}$$

It is observed that when the proportional gain alone is chosen arbitrarily, the response of the motor is not satisfactory[8]. 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[9].

3.2 Structure Of Fuzzy Controller

Fuzzy logic controller as in Figure 5.3 consists of four main parts

1. Fuzzification
2. Rule base
3. Inference engine
4. Defuzzification

A.Preprocessing

The inputs are most often hard or crisp measurement from some measuring equipment, rather than linguistic. A pre-processor, the first block in fig, conditions the measurement before they enter the controller. Examples of pre-processing are

a. Quantization in connection with sampling or rounding to integers.
b. Normalization or scaling onto a particular, standard range.
c. Filtering in order to remove noise.

B.Fuzzification

The first block inside the controller is fuzzification, which converts each piece of input data to degrees of membership by a lookup in one or several membership functions. The fuzzification block thus matches the input data with the conditions of the rules to determine how well the condition of each rule matches that particular instance. There is a degree of membership for each linguistic term that applies to that input variable.

C.Rule Base

The rule base consists of rules in the IF-THEN format. The rules are derived from an expert’s experience or from operators.

D.Inference Engine

In order to draw conclusions from the rule base inference engine employs a mechanism that can produce an output from a collection of if-then rules. This is done using the composition rule of inference.

E.Defuzzification

The resulting fuzzy sets area converted to a number that can be sent to the process as a control signal. This conversion of fuzzy values into crisp value is called defuzzification.

F.Post Processing

In case the output is defined in a standard universe this must be scaled to engineering units. The conversion of the output crisp value to the engineering units by applying some scaling factor is called post processing. The post processing block often contains an output gain that can be tuned, and sometimes also an integer.

3.3 Fuzzy PI Controller

The idea of fuzzy PI controller is to start with a tuned, conventional PID controller, replace it with an equivalent linear fuzzy controller, make the fuzzy controller nonlinear, and eventually fine-tune the nonlinear fuzzy controller. This is relevant whenever a PI controller is possible or already implemented. When the control problem is to regulate the process output around a setpoint, it is natural to consider error as an input, even to a fuzzy controller, and it follows that the integral of the error and the derivative of the error may be useful inputs as well. In a fuzzified PI controller, however, it is difficult to tell the effect of each gain factor on the rise time, overshoot, and settling time, since it is most often nonlinear and has more tuning gains than a PI controller. It is straight forward to envision a fuzzy PI controller with three input terms: error, integral error, and derivative error. A rule base with three inputs, however, easily becomes rather big and, also rules concerning the integral action are troublesome.

The controller is thus a function of sum of all increments,

$$U_n = \sum_i (CU_i * GCU * T_s) \qquad (15)$$

Its linear approximation is

$$U_n = GCE * GCU * \left[\frac{GE}{GCE} \sum_i^n e_i * T_s + e_n\right] \quad (16)$$

Assume GE as nonzero and comparing with conventional PI controller

$$GCE * GCU = K_p \qquad (17)$$

$$\frac{GE}{GCE} = \frac{1}{T_i} \qquad (18)$$

Where GE, GCE, GU in eqn.(15) to eqn.(18) are gains of error, change in error, integral and output of the controller.

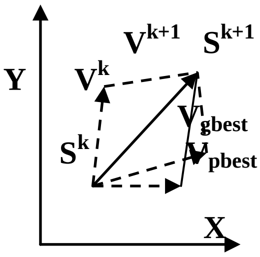
3.4 Particle Swarm Optimization (PSO)

A.Overview

According to the research results for a flock of birds, birds find food by flocking.. The assumption is a basic concept of PSO. PSO is basically developed through simulation of a flock of words in two dimension space. The position of each agent is represented by XY-axis position and the velocity (displacement vector) is expressed by (the velocity of XY axis).Modification of the agent position is realized by using the position and velocity information as shown in figure

Searching procedures by PSO based on the above concept can be described as follows: a flock of agent optimizes a certain objective function. Each agent knows its best value so far (pbest) and its position. More over each agent knows the best value in the group (gbest) among the pbest, namely the best value of the individual agent. Thus the velocity update formula is given as shown in eqn.,

$V_i^{k+1} = W_i V_i^k + C_1 rand*(pbest-S_i^k) + C_2 rand*(gbest-S_i^k)$



Concept of modification of a searching point

Where,

V_i^k - current velocity of agent at iteration,

V_i^{k+1} – modified velocity of the agent

Rand – random number between 0 and 1,

S_i^k-current position of agent at iteration k,

S_i^{k+1} - modified position of agent,

$Pbest_i$- pbest of agent I,

$Gbest_i$- gbest of the group,

W_i-weight function for velocity of agent I,

C_i-weight coefficients for each term

Using the above eqn., a certain velocity that gradually gets close to pbest and gbest can be calculated. The current position can be modified by the following eqn.(6.3),

$S_i^{(k+1)} = S_i^{(k)}+V_i^{(k+1)}S$

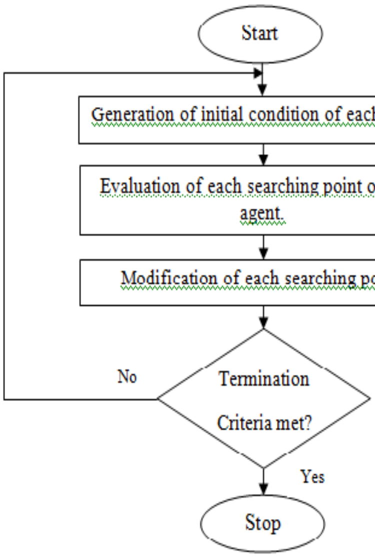


Fig 4: Flowchart for PSO Algorithm

B. Parameters selection of PSO

There are three parameters W, C1, C2 to be considered in the calculation of new velocity. The parameter setting determined hoe it's optimized in the search-space. The inertia weight W is employed to the control the impact of the previous history of velocities on the current velocity. A larger inertia weight W facilitates global exploration while a smaller inertia weight tends to facilitate local exploration to fine-tune the current search area. Suitable selection of the W can provide a balance between global and local exploration abilities thus require less iteration

C. Steps for PSO

Step 1:Initialization

a)Position

1. $x_0^i = x_{min} + rand*(x_{max} - x_{min})$
2. b)Velocity
3. $v_0^i = 0.1*randn(dim, n)$
4. where
5. x_{max} & x_{min} - boundary of the position
6. dim - dimension of the design plane
7. n - no of particle
8. Step 2:Updation of Velocity
9. $v_{k+1}^i = w*v_k^i + c_1*rand().*(p_i - x_k^i) + c_2*rand().*(p_k^g - x_k^i)$
10. w - inertia factor (in between 0.4 & 1.4)
11. c_1 - self confidence of the particle
12. c_2 - swarm confidence (in between 1 and 2)
13. p_i , p_k^g - the personal best and global best
14. Step 3:Updation of Position
15. $x_{k+1}^i = x_k^i + v_{k+1}^i$

For the optimizing the I-PD controller parameters, Position of each particle represents a set of Kp, τi, τd values. So the updation of particle's position causes the particle to move in the potential areas of the problem plane that will give minimum settling time.

D. Fitness function

The general equation of PID controller is

$$U(t) = K_p * e(t) + \frac{1}{\tau_i} * \int e(t)dt + \tau d \frac{de(t)}{dt}$$

The variable e(t) represents the tracking error which is the difference between the desired input value and the actual output. This error signal will be sent to the PID controller and the controller computes both the derivative and the integral of this error signal. The signal U(t) from the controller is now equal to the proportional gain (K_p) times the magnitude of the error plus the integral gain (K_i) times the integral of the error plus the derivative gain (K_d) times the derivative of the error

In PID controller design methods, the most common performance criteria are integrated absolute error (IAE), the integrated of time weight square error(ITAE) and integrated of square error (ISE) that can be evaluated analytically in the frequency domain These three integral performance criteria in the frequency domain have their own advantage and disadvantage. For example, disadvantage of the IAE and ISE criteria is that its minimization can result in a response with relatively small overshoot but a long settling time because the ISE performance criterion weights all errors equally independent of time. Although the ITSE performance criterion can overcome the disadvantage of the ISE criterion, the derivation processes of the analytical formula are complex and time consuming

The IAE,ISE and ITSE performance criterion formulas are as follows:

$$IAE = \int_0^T |e(t)|dt = \int_0^T |r(t) - y(t)|\, dt$$

$$ISE = \int_0^T e(t)^2 dt = \int_0^T [r(t) - y(t)]^2 dt$$

$$ITSE = \int_0^T te(t)^2 dt = \int_0^T t[r(t) - y(t)]^2 dt$$

Here in this tuning time domain criteria is used for evaluating the PID controller. These performance criteria in the time domain include the overshoot, rise time, settling time , and steady state error.

For our case of design, we had to tune all the three parameters of PID such that it gives the best output results or in other words we have to optimize all the parameters of the PID for best results. Here we define a three dimensional search space in which all the three dimensions represent three different parameters of the PID. Each particular point in the search space represent a particular combination of [KP KI KD] for which a particular response is obtained The performance of the point or the combination of PID parameters is determined by a fitness function or the cost function. This fitness function consists of several component functions which are the performance index of the design. The point in the search space is the best point for which the fitness function attains an optimal value. For the case of our design, we have taken four component functions to define fitness function. The fitness function is a function of steady state error, peak overshoot, rise time and settling time. However the contribution of these component functions towards the original fittness function is determined by a scale factor that depends upon the choice of the designer. For this design the best point is the point where the fitness function has the minimal value.

The chosen fitness function is:-

$$F = (1 - \exp(-\beta))(M_p + E_{ss})) + (\exp(-\beta))(T_s - T_r)$$

Where F:- Fitness function
MP :- Peak Overshoot
TS :- Settling Time
Tr :- Rise Time   β:-Scaling Factor, In this case the scaling factor chosen is β=1

IV. RESULTS AND DISCUSSIONS

Using the mathematical model equation the simulation model of separately excited DC motor system was developed and with the specification parameter of the separately excited DC motor we obtain the open loop response .Sundaresan and Krishnaswamy proposed a method to obtain the parameters of first order plus dead time

(FOPDT)process transfer functions from the open loop response. The Zeigler-Nichols open loop method is used for tuning the controller P,PI and PID controller parameters[14]. The simulated responses of different PID structures are obtained with values of Kp, Ki and Kd parameters shown in the table 1 . T he closed loop response of PI and PID are shown in the fig .the maximum peak overshoot and settling time are very large.

Table 1 Tuning Parameter

Controller	Kp	Ki	Kd
PI Tuning (Ziegler Nichols)	11.889	149.5	-
PID Tuning (Ziegler Nichols)	15.85	332.98	0.1886

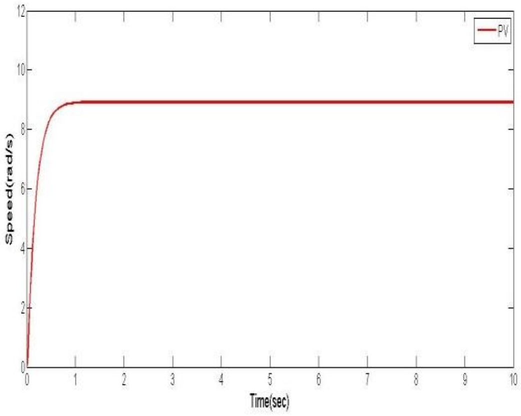


Fig 5: Open loop response of process

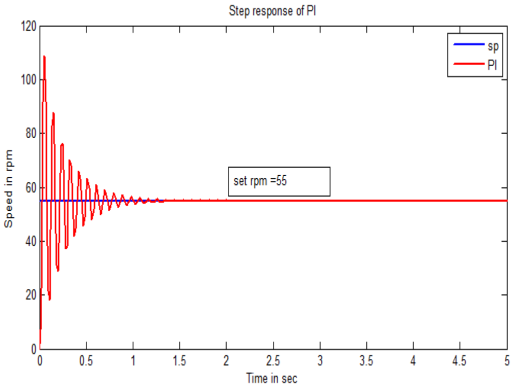


Fig 6:Speed response of DC motor using PI controller

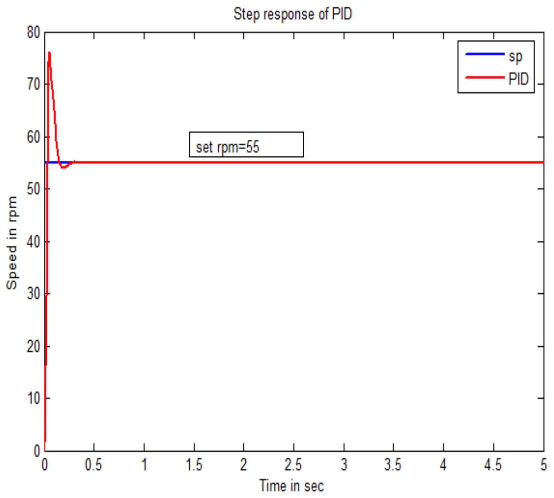


Fig 7:Speed response of DC motor using PID controller

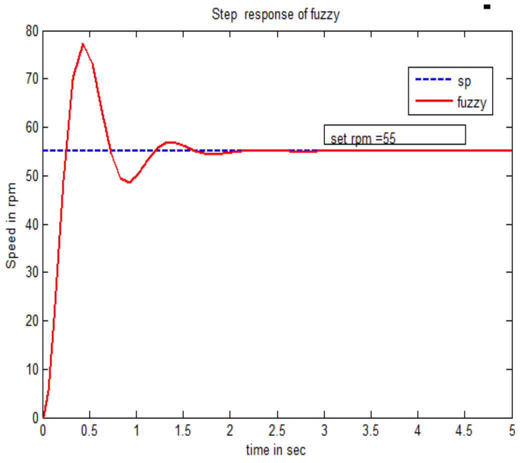


Fig 8:Speed response of DC motor using Fuzzy logic controller

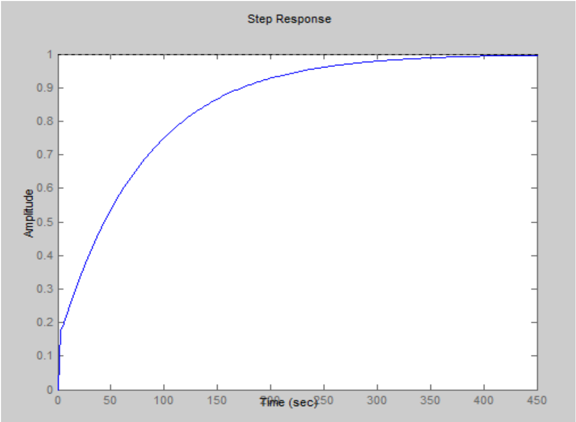


Fig 9:Speed response of DC motor using PSO-PID

Table 2-Comparison of various controllers

Approach	Settling time (Sec)	Peak Overshoot (%)	ISE	IAE
PI Tuning (Ziegler Nichols)	2.12	53.8	251.8	10.38
PID Tuning (Ziegler Nichols)	0.55	21.1	70.81	2.504
Fuzzy controller	1.7	15	0.0020	0.0116
PSO Algorithm approach	2.0	0	0.0076	0.0023

From the table 2, the settling time, percentage of peak overshoot and performance indices for PI,PID,Fuzzy controller and PSO-PID are shown. The comparison shows that conventional controller results in response with maximum overshoot and PSO-PID results in quicker response with no overshoot than conventional controller.

V. CONCLUSION

The mathematical modeling of a DC motor was studied and the PID controller was tuned in order to maintain the speed of the DC motor. When there is no control over the process, it generates an inverse response together with an overshoot and considerable delay time. But when the PID control is implemented to the process, the problems of overshoot and delay time are controlled in the ongoing process and are removed considerably but then it was showing instability in terms of rise time and settling time. To overcome this instability in rise time and in settling time different types of controllers such as FUZZY, PSO based PID controller has been used. The PSO based control scheme helps to remove those delay times and the inverted response shown in graphs. Rise time and settling time are also reduced. The controller simulation has be done using MATLAB/SIMULINK and the results are compared.

REFERENCES

1. W. Lord and J. Hwang, " DC motor model parameters ". IEEE Trans. on Industrial electronics and Control Instrumentation, vol. IECI-23, No.3, August, 1977.
2. R. Schulz, "A frequency for determining the parameters of high performance DC motors".IEEE Trans. on Industrial Electronics, vol. IE-30, No.1, Februray, 1983.
3. A. Rajkumar and R. Somanahtham, "Determination of DC motor transients using a microcomputer ". IEEE Trans, on Instrumentation and Measurement,vol. 37, No.4, Dec. 1988.
4. Abu zaharin and Mohd. Nasir Taib, "A Study On The DC Motor Speed Control By Using Back-EMF Voltage", IEEE Conf. on Sensor, pp. 359-364, 2003.
5. Ahmed Rubaai, " Design and Implementation of Parallel Fuzzy PID controller for high performance Brushless Motor Drives an integrated environment for rapid control prototyping". IEEE Trans. On industry application, vol.44, No.4, july/August, 2008.
6. P. Melba Maryand N.S. Marimuthu , "Design of self-tuning fuzzy logic controller for the control of an unknown industrial process", IET Control Theory Appl., Vol. 3, Issue. 4, pp.428–436, 2009.
7. Speed Control of Separately excited D.C Motor Using computational method. Malkeet saini ,Neeraj Sharma International Journal of Engineering Research & Technology, (IJERT) Vol. 1 Issue 7, September - 2012 ISSN: 2278-0181
8. Ogata K., Modern Control Engineering , 2nd ed., Prentice Hall, NJ, 1990.
9. Kuo B.C., Automatic Control Systems, 6th ed., Prentice Hall, NJ, 1991.
10. The hierarchical expert tuning of PID controllers Using tools of soft computing., F. Karray, W. Gueaieb, IEEE Journal.
11. Kiam Heong Ang Chong, G. and Yun Li,"PID Control System Analysis, Design, and Technology", IEEE Transactions on control systems technology, vol.13, No. 4, 2005.
12. Yun Li, Kiam Heong Ang and Gregory C.Y.Chong, "PID Control System Analysis and Design - Problems, Remedies and Future Directions', IEEE Control systems magazine, pp. 32-41, 2006.
13. E. Govindakumar and S. Manoharan "Enhancement of the PID controller performance for horizontal tank process " IJSSST vol.14, issue 5, 2013
14. K. R. Sundaresan and R. R. Krishnaswamy, "Estimation of time delay, time constant parameters in time, frequency and laplace domains " Can.J.Chem.Eng,Vol.56,257,1978