International Journal of Trend in Research and Development, Volume 6(6), ISSN: 2394-9333
www.ijtrd.com

DC Motor Speed Control using Fuzzy Gain Controller with Evolutionary Algorithm

[1]S. Syed Musthafa Gani, [2]S. Boobalan, [3]S. Faizal Mukthar Hussain and [4]S. Ramamoorthi,
[1,3,4]Assistant Professor, [2]Professor &Head, [1,2]Department of EEE, [3,4]Department of CSE,
[1,2,3,4]Mohamed Sathak Engineering College, Kilakarai, Tamilnadu, India

Abstract: The PID controllers are used to achieve the required speed and position control of the DC motors. The soft computing techniques such as particle swarm optimization (PSO) method, Genetic Algorithm (GA) method are used for determining the optimal proportional-integral derivative (PID) controller parameters to simplify the tuning procedures. Recently FUZZY-PID controllers can improve the performance with higher accuracy. But tuning the FPID controller gains (Kp,Ki,Kd,Ku) for desired performance is complex. This Paper presents a particle swarm optimization (PSO) method for optimal tuning of FUZZY-PID controller. The DC motor is modelled in Simulink and the PSO algorithm is implemented in MATLAB. The proposed approach has superior features, including easy implementation, stable convergence characteristic and good computational efficiency. Comparing with PSO PID method, the proposed method has more efficient in improving the step response characteristics such as, reducing the steady-state error; rise time, settling time and maximum overshoot.

Index terms: *DC Motor, Particle Swarm Optimization (PSO), Proportional Integral Derivative Controller (PID), Fuzzy PID (FPID).*

I. INTRODUCTION

DC motors are widely used in industries, robots, and commercial purposes etc. DC motors are characterized by their versatility. By means of various combinations of shunt, series, and separately-excited field windings they can be designed to display a wide variety of volt-ampere or speed-torque characteristics for both dynamic and steady-state operation.The speed of DC motor can be adjusted to a great extent so as to provide easy control and high performance. There are several conventional and numeric controller types intended for controlling the DC motor speed at its Executing various tasks: PID Controller, Fuzzy Logic Controller; or the combination between them: Fuzzy-Swarm, Fuzzy-Neural Networks, Fuzzy-Genetic Algorithm, Fuzzy-Ants Colony.

II. DC MOTOR MODELING

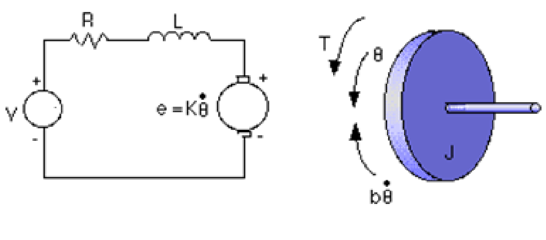

Figure 1: Physical model of DC Motor

$$\frac{\omega(s)}{V(s)} = \frac{K}{(Ls+R)(Js+D)+K^2} \quad (1)$$

where J (kg.m2/s2) is the moment of inertia of the rotor, D isthe damping ratio of the mechanical system, R (ohm) iselectrical resistance, L (H) is electrical inductance, and (Kt=Km=K) K(Nm/A) is the electromotive force constant.

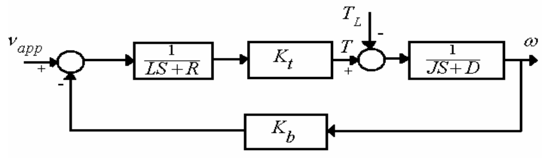

Figure 2: Block diagram of DC Motor

III. FUZZY PID CONTROLLER

Fuzzy logic controller has emerged as one of the most active and useful research areas in fuzzy control theory. The fuzzy logic controllers have been successfully applied in control of various physical processes. The structure of the FLC, composed with the Mamdani's inference engine includes three typical blocks. They are fuzzifier, fuzzy rule base with inference engine and Defuzzifier.

The PID controller is expressed as,

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \quad (2)$$

Where,

e (t) =r (t) - y (t)(3)

The change rate of error de at the time k will be,
de (t)=e(t) - e(t-1) (4)

A. Fuzzifier

Three types of structure of FLC have been studied:

1) Fuzzy PD control with parallel integral action (fuzzy PD+I)

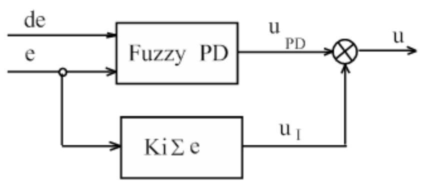

Fig 3: Fuzzy PD controller with deterministic integral control

2) Fuzzy PI control with parallel derivative action (Fuzzy PI+D)

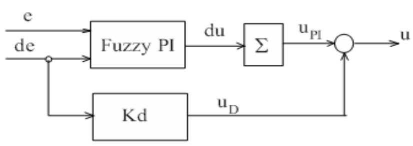

Fig 4: Fuzzy PI controller with parallel derivative control

3) Fuzzy PI control with Parallel fuzzy PD control (fuzzy PI+Fuzzy PD)

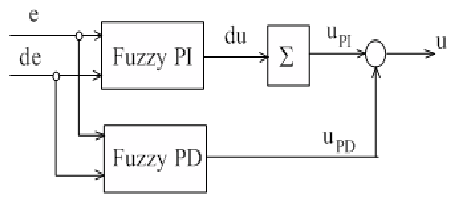

Fig 5: Fuzzy PI control with Fuzzy PD control

The fuzzy PD and fuzzy PI controllers based on Mamdani's fuzzy system are simpler and more applicable when they are used in hybrid structures.

B. Fuzzy Rules

In this stage, the crisp variables are converted into fuzzy variables. They transform the inputs to universe of fuzzy sets. These transformations are closely according to the prescribed membership functions associate with the control variables, the membership functions have been chosen with triangular shapes. The Fuzzy input variables are error e(t) and change in error Δe .The fuzzy output variable is u(t). Each universe of discourse is divided into five fuzzy sets: NB, NS, Z, PS, and PB. Each fuzzy variable is a member of the subsets with a degree of between 0 (non member) and 1.

Table 1: Rule Base for FPID Controller

Δe/e	NB	NS	Z	PS	PB
NB	NB	NB	NS	NS	PB
NS	NB	NS	Z	Z	PB
Z	NB	NS	Z	PS	PB
PS	NS	Z	PS	PB	PB
PB	Z	PS	PB	PB	PB

The fuzzy rules are based on operator's long-time experiences. The variables are processed by an inference engine executes 25 rules. Each rule is expressed as

1) If Δe is NB and e is NS then the output u(t) is NB
2) If Δe is PS and e is PB then the output u(t) is PB.

IV. PARTICLE SWARM OPTIMIZATION (PSO)

PSO, a proposed algorithm by James Kennedy and R.C.Eberhart is one of the modern heuristic algorithms, it has been motivated by the behaviour of organisms, such as fish schooling and bird flocking. Generally, PSO is characterized as a simple concept, easy to implement, and computationally efficient. Unlike the other heuristic techniques, PSO has a flexible and well-balanced mechanism to enhance the global and local exploration abilities. It shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA).

The PSO algorithm starts with random initialization of population and velocity. The search for the optimum solution is continued unless one of the stopping criteria is reached. The stopping criteria are either the maximum iterations are reached or there is no further improvement in the optimal solution.

The PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In every iteration, each particle is updated by following two best values. The first one is the best solution (fitness) it has achieved so far (the fitness value is also stored) this value is called pbest. Another "best" value that is tracked by the particleswarm optimizer is the best value, obtained so far by any particle in population. This best value is a global best called as gbest. The particle takes part of the population as its topological neighbours the best value is a local best and is called best.

Let x and v denote a particle coordinates (positions)and its corresponding flight speed (velocity)in a search space, respectively. Therefore, the i-th particle is represented as $x_i=(x_{i1},x_{i2},\ldots,x_{id})$ in the d-dimensional space.The best previous position of the i-th particle is recorded and represented as $pbest_i=(pbest_{i1},pbest_{i2},\ldots,pbest_{id})$.the index of the best particle among the all particles in the group is represented by the gbest. The rate of velocity for the particle **i** can be represented as $\mathbf{v_i=(v_{i1},v_{i2}\ldots v_{id})}$.The modified velocity and positions of the each particle can be Calculated using the current velocity and the distance from the following equations,

$$V_{id}^{(t+1)}=\omega.v_{id}^{(t)}+c1*rand()*(pbest_{id}-x_{id}^{(t)})+c2*rand()*(gbest_{d}-x_{id}^{(t)}) \quad (5)$$

$$X_{id}^{(t+1)}=x_{id}^{(t)}+v_{id}^{(t+1)},\ i=1,2,3...n.,d=1,2...m \quad (6)$$

where c1 and c2 are cognition and social parameters respectively, rand1() and rand2() are constant numbers in the range of [0,1], w is the inertia weight. Vi represents the velocity of the ith particle and Xi is its position, pbesti and gbest are local best and global best positions respectively. The velocity of particle in equation depends on its previous velocity, its own thinking and social psychological adaptation of the population.

The inertia weight $\omega$is set according to the following equation,

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{\min})}{iter_{max}} \times iter(7)$$

The advantages of PSO over the other optimization techniques are Lower sensitivity to the nature of the objective function Derivative free property unlike genetic algorithm Easy implementation and Fast convergence.

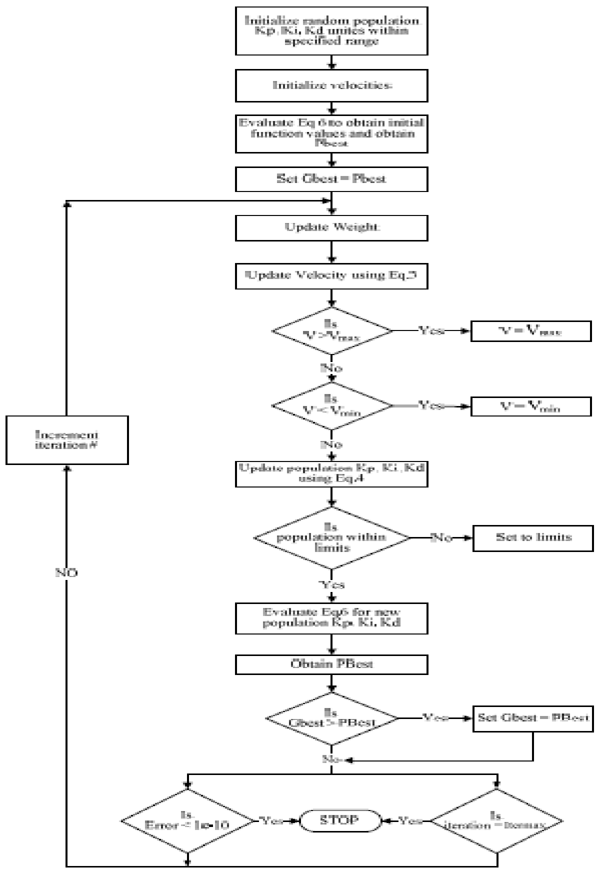

Figure 6: PSO flow chart

The FPID controller parameters (Kp,Ki,Kd,Ku) are adjusted to keep the DC motor at an optimum level of performance under varying operating conditions. It requires a performance index which is the function of system parameters. The commonly used performance indexes are, integral square error (ISE), integral of the absolute magnitude of error (IAE) ,integral time square error (ITSE), integral of time multiply by absolute error (ITAE).ITAE performance index which produces smaller overshoots and oscillations.

The ITAE is given as,

$$ITAE = \int_0^\infty t\,|e(t)|\,dt \quad (8)$$

In this approach an **ITAE** criterion is used for evaluating the PID controller. A set of good control parameters P, I and D can yield a good step response that will result in performance criteria minimization in the time domain. Therefore, the performance criterion is defined as follows,

$$\text{Ji}= min(\sum_{m=1}^{M}(t * absolute(error))) \quad (9)$$

V. SIMULATION MODEL OF DC MOTOR WITH PSO BASED FUZZY PID CONTROLLER

To verify the efficiency of proposed algorithm a DC motor transfer function is considered. The PSO algorithm is used to search an optimal parameter set containing Kp, Ki Kd and Ku. The optimum values generated by the algorithm are stored in work space.

Table 2: Parameters of the motor

Parameters	Values
D	0.0026 Nm-Sec
Kt=Km=K	0.2516 Nm/Amp
J	0.00473 Kg m^2/s^2
R	2 Ohm
L	0.0052 mH

The DC motor Transfer function is given as,

$$\frac{0.126}{0.00473s^2 + 0.00261s}$$

Table 3: PSO Parameters

No. of Particles	20
No. of Swarms	4 (Kp,Ki,Kd,Ku)
No. of Iterations	200
c1, c2	2,2
Wmax,Wmin	0.9,0.4
Max.particle velocity	0.5

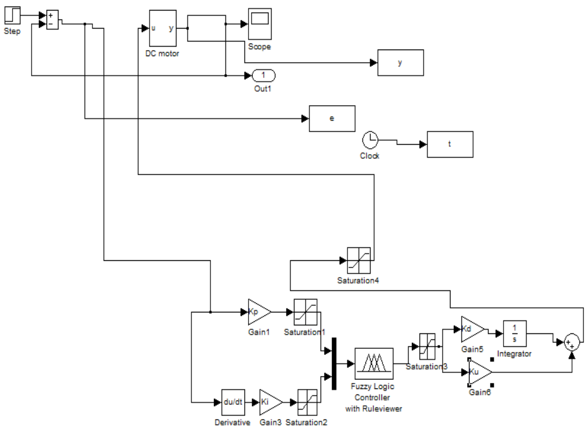

Figure 7: Simulink model of PSO based Fuzzy PID Controller for DC motor

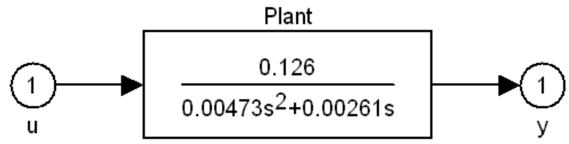

Figure 8: Plant model of DC motor

At first the PID controller gains are optimized by using PSO algorithm. In this method the system performance consists of some overshoots and takes more settling time.

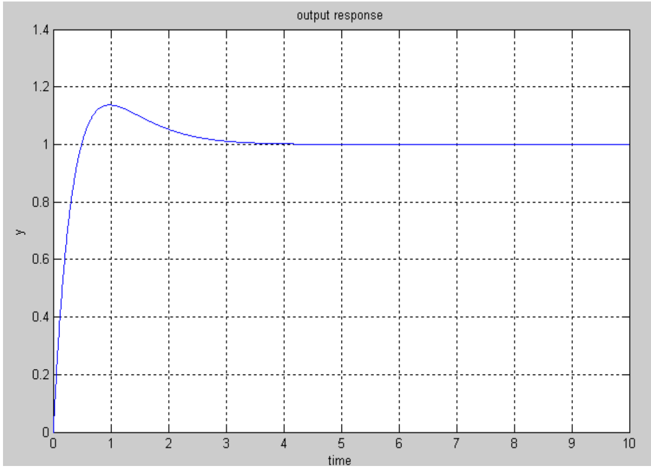

Figure 9 : Step response of the DC motor with PSO based PID controller

Then the Fuzzy PID (FPID) controller is implemented to the system. The gains of the fuzzy PID controllers are optimized by the PSO algorithm. It will give the good performance compared to the PSO PID Controller. It reduces the overshoot and gives quick settling time.

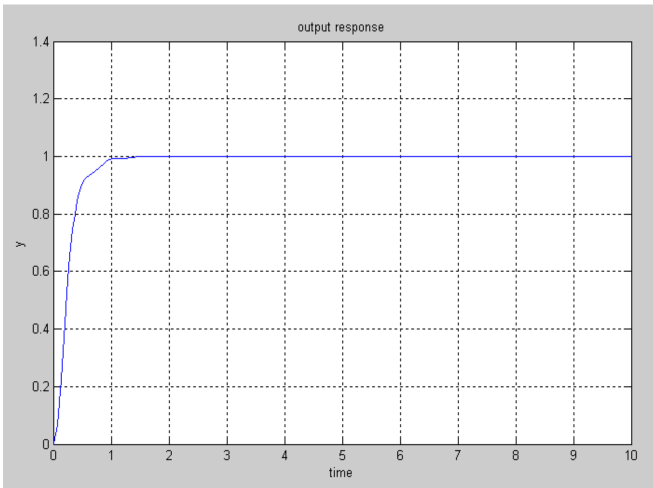

Figure 9 : Step response of theDC motorwith PSO based Fuzzy PID (FPID) controller

Table 4: Controller gain Results

Controller Gains	Kp	Ki	Kd	Ku
PSO PID	7.892	8.046	1.957	_
PSO FPID	1.904	0.306	5.523	0.667

Table 5: Comparison Results

Characteristics	PSO PID	PSO FPID
Rise time	0.4	0.25
Settling Time	5.98	2.0554
Peak Overshoot	1.137	Nil

CONCLUSION

The conventional controllers used for the DC motor have large settling time, overshoot and oscillations. Hence, when evolutionary algorithms are applied to control system problems, their typical characteristics show a faster and smoother response. An artificial intelligent technique has been proposed for DC motor speed control. The Simulation results shows that the proposed PSO-FPID controller provides a satisfactory performance with no overshoot and transient oscillations with zero steady state error. The work can be extended in future by including non-linear parameters to the system modelled and thereby comparison can be made with respect to linear system. Also the other evolutionary computing techniques like Hybrid Particle Swarm optimization (HPSO), GA-PSO, Hybrid GA, Fuzzy PSO, Distributed Evolution etc.,and can be implemented to improve the performance characteristics.

References

[1] Sharifian M.B.B , Rahnavard and Delavari .H "Velocity control of DC motor based intelligent method and optimal integral state feedback controller" International journal of computer theory and engineering Vol 1 .No 1.2009

[2] J. Kennedy and R.C Eberhart, "Particle swarm optimization," in Proc.IEEE Int. Conf. Neural Networks, vol. IV, Perth, Australia, 1995, pp.1942–1948.

[3] Z.-L. Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," IEEE Trans. Energy Conversion, vol. 19, pp. 384-391, June 2004.

[4] Asim Ali khan ,Nushkamrapal "Fuzzy PID controller: Design ,tuning and comparison between conventional PID controller" 1-4244-0457-6/06 IEEE

[5] Jan Jantzen: "Tuning of fuzzy PID controllers" Denmark. Tech. Report no 98- H871(fpid), 30 Sep 1998.PP:1-22.

[6] Hyung-soo Hwang, Jeong-Nae Choi, Won-HyokLee: "A Tuning Algorithm for The PID controllerUtilizing Fuzzy Theory", IEEE Proceedings 0-7803-5529-6/99/ Q1999 IEEE.PP:2210-2215.

[7] Shi, Y.H. and Eberhart, R.C. (2001). Fuzzy Adaptive Particle Swarm Optimization. Proc. Congresson Evolutionary Computation. Seoul,Korea.461