International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES)
Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585
Volume 4, Issue 6, June-2018

Tuning of Fractional-Order PID Controller for DC Motor Speed Control Using Genetic Algorithm and Simulated Annealing Algorithm

Sudhanshu Goyal[1] ,ShivNarayan[2]
[1]*Electrical Department & PEC (Deemed to be University) Chandigarh, Sudhanshu.g.eee@gmail.com*
[2] *Electrical Department & PEC (Deemed to be University) Chandigarh, shivnarayan@pec.ac.in*

Abstract— ***The aim of the paper is to investigate the efficiency of fractional order control system over conventional integer order controller. The Paper presents a comprehensive study of the optimal design of fractional and integer order controllers for a DC motor speed control system. In this paper, genetic algorithm and simulated annealing has been used to tune the fractional and integer order and PID controllers. The algorithm aims at optimizing the gains of the FOPID and Integer order PID controller, and the objective function here is to reduce the value of integral of the absolute error (IAE). After results obtained simulations shows that, the time domain performance of fractional order PID Controller is better than Integer order PID Controller.***

Keywords— fractional-order controller; genetic algorithm; DC motor speed control; simulated annealing; PID

INTRODUCTION

Proportional-Integral-Derivative (PID) controllers have been in practice in process industries for decades because of their easiness, robustness and easy execution in practical applications. Furthermore, the PID controllers can provide brilliant control performance regardless of the wide-ranging characteristics of the plant. The origination of PID control was in 1910, while the straightforward Ziegler-Nichols (Z-N) tuning rules in 1942[1]. The PID controller includes three control terms, which controls the output of controller to apply accurate and optimal control. These terms have three gain, the proportional gain, the integral gain and derivative gain. Simply these terms can be taken in time as: the proportional (P) is proportional to the error between the set point and the present value, the integral (I) accounts on the error and integrates them over time, and the derivative (D) estimates the future tendencies of errors, based on the current rate of change. Using these gains controller calculates error signal. An illustration where fractional order controller can be applied are the DFIG Based Wind energy system [2], Electric V2G Based micro-grid stabilization devices [3] etc.

Lately, many dynamic systems have been modelled by using fractional order calculus because it provides much more realistic picture for a dynamic system. For example, PID Controllers are used in majority in process industries because they are simple in modelling. However, results have shown that the extra degree of freedom i.e. two order of integrator and differentiator gives us possibilities to enhance further the plant performance [4], [5].

In this Paper, Optimization for Integer order PID and Fractional order PID controllers has been performed by using genetic algorithm and simulated annealing algorithm. The optimization has been done for minimizing the value of IAE (Integral of Absolute Error).According to the results obtained in the paper, Fractional order PID Controller shows better time response results then Integer order PID Controller.

FRACTIONAL CALCULUS

Fractional calculus is the generalized term of integer calculus. Here operator ${}_aD_t^\alpha$ is considered where a and t signify the limits of the operation and α signifies the fractional order for integrator or differentiator. This is given such as in Equation (1) –

$$ {}_aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \Re(\alpha) > 0 \\ 1 & \Re(\alpha) = 0 \\ \int_a^t (d\tau)^{-\alpha} & \Re(\alpha) < 0 \end{cases} \quad \textbf{(1)} $$

There are many definitions given for fractional order differential and integral. The most popular are listed below.

1. Riemann-Liouville definition
2. M.Caputo definition
3. Grunwald-Letnikov definition

A. *Definitions of the fractional operator*

1) Riemann-Liouville definition

$$aD_t^{\alpha} = \frac{1}{\Gamma(m-\alpha)}\left(\frac{d}{dt}\right)^m \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-m+1}}\, d\tau \quad \textbf{(2)}$$

where m−1<α<m, m∈N, α∈R+ and $\Gamma(\cdot)$ is Euler's gamma function

2) Caputo definition

$$0D_t^{\alpha} = \frac{1}{\Gamma(m-\alpha)}\int_0^t \frac{f^m(\tau)}{(t-\tau)^{\alpha-m+1}}\, d\tau \quad \textbf{(3)}$$

where m−1<α<m, m∈N.

3) Grünwald-Letnikov definition

$$aD_t^{\alpha} = \lim_{h\to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} (-1)^j \binom{\alpha}{j} f(t-jh) \quad \textbf{(4)}$$

$$where\ \binom{\alpha}{j} = \frac{\alpha(\alpha-1)(\alpha-2)\dots(\alpha-j-1)}{j!}$$

FRACTIONAL ORDER CONTROLLERS

There are four Types of Fractional Order Controllers used. These are

1. TID (Tilted Integral Derivative) controller
2. CRONE controller (Controle Robuste d'Ordre Non Entier)
3. $PI^{\lambda}D^{\mu}$ Controller
4. Fractional Order lead-lag compensator

1. $PI^{\lambda}D^{\mu}$ Controller

A fractional order PID controller is an addition to the classical/integer-order PID controller. It has five parameters for tuning, two additional parameters when compared to traditional integer PID controller. The fractional-order PID controllers are more robust against the changes in the parameters of a system. FOPID controller C(s) is represented with Equation (5) -

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad \textbf{(5)}$$

Where, K_p, K_i, K_d are proportional, integral and derivative gains, λ and μ are the value of order of integrator and differentiator. The orthodox integer-order PID controller is a particular case like where λ and μ are equal to one is the PID controller; similarly, PI and PD controllers are distinct cases of $PI^{\lambda}$and PD^{μ} controllers. Considering the PID plane, integer-order PID has four points that are fixed in the plane where as in fractional-order it is possible to have entire plane as seen in Figure 1

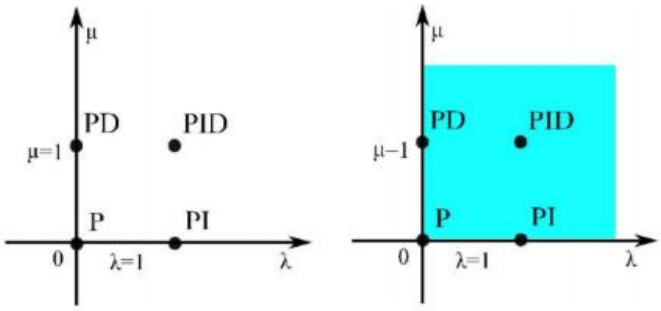

Fig. 1 Control Space of PID Vs FOPID structure

To get the correct modelling of dynamic and process plants order of system increases to capture the dynamic behaviour of the plant [6]. Tuning of PID controllers for the higher-order processes is challenging due to its order. One of the technique is to reduce higher-order model into a lower-order fractional model.

MODELLING OF DC MOTOR

In control systems, the DC motors speed control is done by using in two different control methods: -

1. Armature control method where field current is fixed.
2. Field control method where armature current is fixed.

Here armature control mode with fixed field current considered i.e. also known as armature-control [7]. In this system

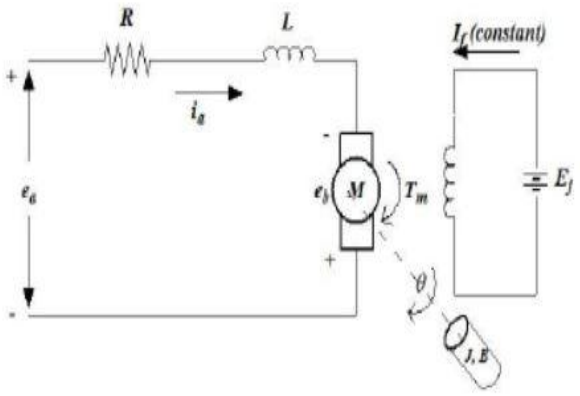

Fig. 2 Separately excited constant field current DC motor

R = Resistance of armature (Ω).

L =Inductance of armature winding (H).

e_b = Back emf (V)

I_a = Armature current (A).

I_f = Field current (A).

e_a = Applied armature voltage (V)

Ω = Angular speed of motor shaft (rad/sec.)

J = Equivalent moment of inertia of motor and load referred to motor shaft (kg-m^2)

B = Equivalent friction coefficient of motor and load referred to motor shaft (Nm*s/rad)

T_m = Torque developed by motor (Nm)

The KVL equation for the armature circuit is-

$$e_a - i_a R - L_a \frac{di_a}{dt} = e_b \quad \textbf{(6)}$$

And The torque equation is

$$T_e = T_L + T_F + T_W + J\frac{dw}{dt} \quad \textbf{(7)}$$

T_L = Load Torque

T_F = Friction Torque = $B * w(s)$

T_W = Windage Torque (Negligible)

$$T_e = T_L + B * w(s) + J\frac{dw}{dt} \quad \textbf{(8)}$$

The electromagnetic torque Tm motor develops is proportional to the multiplication of armature current (i_a) and air gap flux, i.e.

$$T_m = K_1 K_f i_f i_a \quad \textbf{(9)}$$

Where K_1 is constant.

In first case of armature controlled dc motor here the field current is kept fixed, so

$$T_m = K_t i_a \quad \textbf{(10)}$$

Where K_t is known as the motor torque constant.

The motor back emf is proportional to the speed is given by

$$e_b = K_a \phi w(s) \quad \textbf{(11)}$$

$$e_b = K_b w(s) \quad \textbf{(12)}$$

Where K_b is the back emf constant.

Using these equations, final transfer function is

$$G(s) = \frac{\omega(s)}{e_b(s)} = \frac{K_t}{(R+sL)(Js+B)+K_t K_b} \quad \textbf{(13)}$$

TABLE I
DC MACHINE PARAMETERS

Parameters	Value
Power(P)	5hp
Armature Voltage(E_a)	200v
Field Voltage (V_f)	300v
Armature Resistance(R_a)	1.75 Ω
Armature Current(I_a)	2.9a
Field Resistance (R_f)	200 Ω
Armature Inductance(L_a)	2.8 Mh
Moment Of Inertia(J)	0.00303(Kg-M2)
Friction Coefficient Of Motor (B)	0.005(Nm/Rad)
Motor Constant(K_t)	0.0924
Back Emf Constant (K_b)	0.0924
Rated Speed (N)	550rpm

$$G(s) = \frac{0.0924}{8.4896*10^{-7}s^2+5.32*10^{-3}s+0.01729} \quad \textbf{(14)}$$

In the 80's digital processers replaced the old analog control system and during this new PID controller's method emerge using the new range of controller technology. Previously, parameters were optimized using traditional "trial and error" methods. PID controllers have been commonly used for control systems. In the worst condition, the bad tuning can result in a poor performance of the plant. The aim of the tuning of PID Controller is to satisfy the performance specifications that can be time domain or frequency domain of the plant, such as the rising time (tr), the maximum overshot (Mp), the settling time (ts) and the steady state error(ess), required gain or phase margin. However, it is hard to obtain the needed standards of these requirements at once.

B. *Optimal Tuning based on Genetic Algorithm*

1) Genetic Algorithm Based Optimization: Genetic Algorithm (GA) is a random global search technique that impersonators the procedure of natural selection means survival of the fittest. In starting, Algorithm has no information of the right solution and depends totally on answers using its mutation and crossover operators to get the best solution.

GA comprises of three essential operators: selection, crossover and mutation. First, we have to consider population which is a subset of solution. Things to consider while deciding population is first it should be diverse and optimal number of population should be selected. From the population selected child solutions are reproduced using crossover and mutation. Child solution may be similar or different from parent solutions. The Crossover is similar to biological crossover. In this using parent, one or more offspring are produced. Crossover coefficient tells us how different child solutions are from parent solutions. In GA crossover is applied in high probability. Mutation means the minor tweak in the parent to get new solution. It is important keep diversity in population used in optimization and it is kept with low probability. To get the random search mutation kept high.

For Genetic Algorithm the optimization process has been done in MATLAB by the help of Optimization tool. Population size is set to 100 and fitness function IAE (Integral of Absolute Value) is minimized by selecting fittest individual value. From that reproduction of fittest value and mutation, diverse values are produced and the process is again repeated.

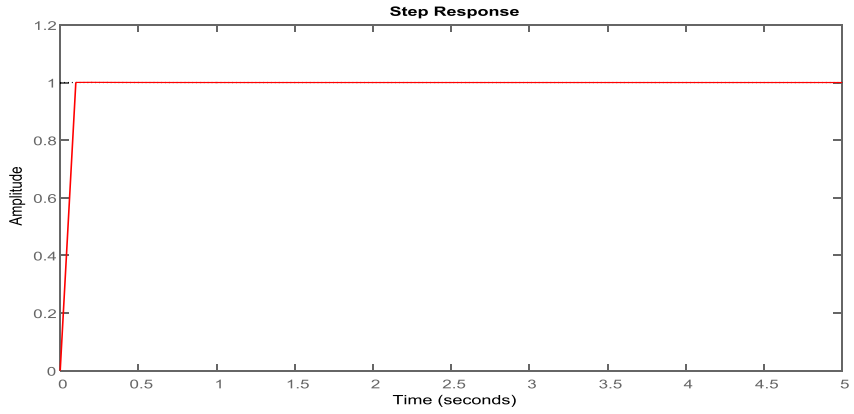

Fig. 3 Step Response of PID Controller Using Genetic Algorithm

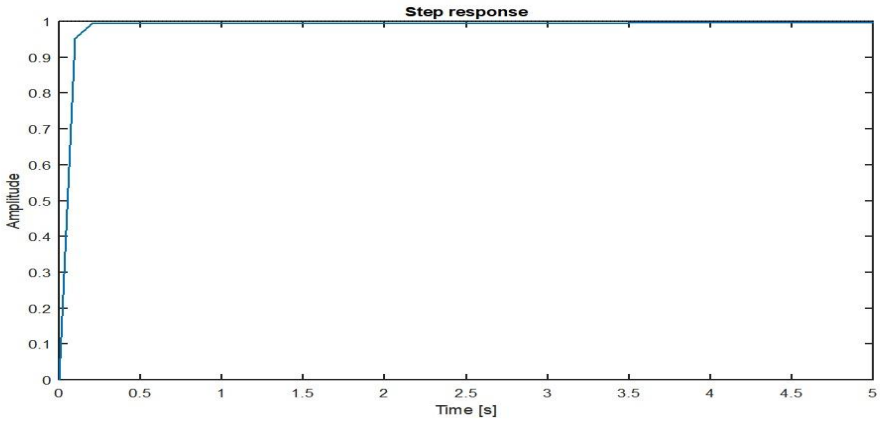

Fig. 4 Step Response of FOPID Controller Using Genetic Algorithm

TABLE II

COMPARISON OF TIME DOMAIN PARAMETERS BETWEEN PID AND FOPID USING GENETIC ALGORITHM

Parameters	GA Based PID Controller	GA Based FOPID Controller
Rise Time(sec)	0.0674	0.0659
Settling Time(sec)	0.1825	0.0941
Peak Overshoot(Percentage)	4.3008	0
Peak Time	0.1343	0.1000

2) Simulated Annealing Based Optimization **:**The SA algorithm takes on the phenomenon of the annealing of solids and the problem of solving finds optimal solution to optimization problems. The name and motivation is from annealing which is the technique in metallurgy of heating a solid and then con-trolled cooling it down until it crystallizes and reduces defects. The atoms of the solid signify possible answers of the optimization problem and the small random displacement in the atoms that affects the energy of atom. After each iteration algorithm randomly changes the current solution to get the new solution in the periphery of the old solutions [8], [9]. Here "Temperature" is the most important parameter. Temperature is the function on which iterations are done. Because the name of the algorithm is based on method of heating and cooling. Basically temperature is started at 1.0 and reduced at the end of each iteration by multiplying with a fixed value called α. The value of α is typically kept between 0.8 to 0.99

For simulated annealing algorithm also MATLAB Optimization tool is used. Here we have used Boltzmann Annealing function with exponential temperature update rate.

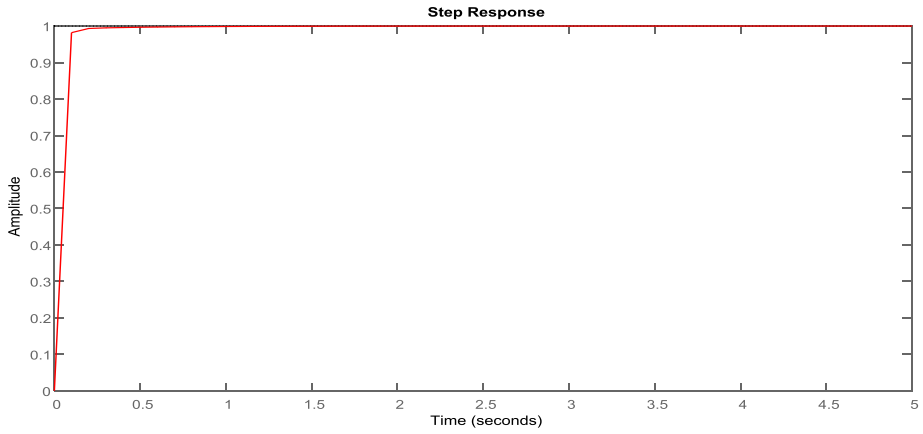

Fig. 5 . Step Response of PID Controller Using Simulated Annealing Algorithm

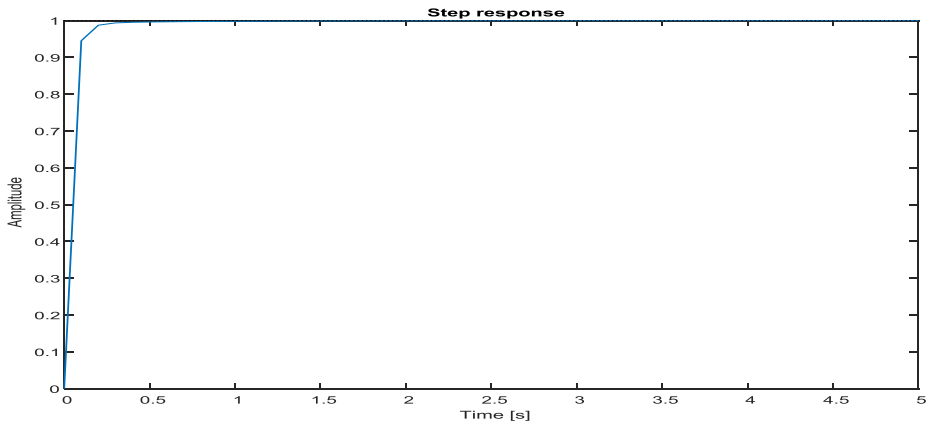

Fig. 6 Step Response of FOPID Controller Using Simulated Annealing Algorithm

TABLE III

COMPARISON OF TIME DOMAIN PARAMETERS

Parameters	SA Based PID Controller	SA Based FOPID Controller
Rise Time (sec)	0.0600	0.0526
Settling Time(sec)	0.1597	0.0886
Peak Overshoot (Percentage)	2.6416	0
Peak Time (sec)	0.1294	0.1000

CONCLUSIONS

In this paper, we have shown that how Fractional order PID controller provides us better time domain performance. We have used genetic algorithm (GA) and simulated annealing (SA) algorithm to minimized IAE (Integrated absolute error). Simulation results were very promising and results are shown in this paper. Paper have shown that simulated annealing outperforms the genetic algorithm for dc motor speed control. It is be-cause Fractional order PID controller can fulfil more performance requirement because it has two more parameters, which provide us better performance.

In future, we can consider outspreading the work in the field of system based on fuzzy dynamic systems and complex order can be used to increase the performance and robustness of system because it can provide better values so design performance can be achieved more accurately then integer order fractional order PID controller.

REFERENCES

1. K. Ogata, Modern Control Engineering, vol. 17. 2009.

2. N. Ullah, M. A. Ali, A. Ibeas, and J. Herrera, “Adaptive Fractional Order Terminal Sliding Mode Control of Doubly Fed Induction Generator Based Wind Energy System,” IEEE Ac-cess, vol. 3536, no. c, 2017.

3. Xiangwu Yan, Zhengyang Duan, and Zheng Lv, “Control method of fractional-order PI con-troller applied in electric vehicle to grid based micro-grid stabilization device,” in 2014 IEEE Conference and Expo Transportation Electrification Asia-Pacific (ITEC Asia-Pacific), 2014, pp. 1–6.

4. I. Podlubny, “Fractional-order systems and PIλDμ-controllers,” IEEE Trans. Automat. Contr., vol. 44, no. 1, pp. 208–214, 1999.

5. R. Andriambololona, R. Hanitriarivo, T. Ranaivoson, and R. Raboanary, “Two definitions of fractional derivatives of powers functions,” Pure Appl. Math. J., vol. 2, no. 1, pp. 10–19, 2013.

6. J. Viola and L. Angel, “Design of a fractional PI controller to control a flow and level sys-tem,” in 2013 25th Chinese Control and Decision Conference (CCDC), 2013, pp. 1211–1216.

7. F. Busetti, “Simulated annealing overview,” World Wide Web URL www. geocities. com/ …, no. 1, pp. 1–10, 2003.

8. B. Suman, “Simulated annealing-based multiobjective algorithms and their application for system reliability,” Eng. Optim., vol. 35, no. 4, pp. 391–416, 2003.

9. B. L. Theraja and A. K. Theraja, A textbook of electrical technology. S. Chand & Company Ltd, 2005.