

Speed Control of DC Motor Using ACO Based FOPID Controller

Jatin Yadav, Prashant Kumar, Dr. Neeraj Gupta, Ms. Manjeet Kaur

ABSTRACT- Direct current (DC) motors are widely used because of their easily controllable and reliable access. DC motors have many salient features that includes a wide range of torque speed control, reliable operation with greater efficiency and a greater starting torque and many more. A nonlinear model for a DC motor can be represented for considering all uncertainties and non-linearities. In this paper, DC motor is studied which is having a second order system. Many controllers are being used for controlling speed of DC motor. An analysis is carried out on mathematical model of DC motor whose field is excited by an external or separate supply. Tuning of FOPID parameters using Ant Colony Optimization Technique (ACO) is discussed here.

Keywords:- ACO, DC Motor, FOPID Controller, PID Controller, Swarm Intelligence.

I. INTRODUCTION

The speed control as well as the starting torque is high for DC motors. The high transient response and compactness makes DC motor popular in most of the industries. Mostly used motors for industrial purposes are PMBLDCM i.e. permanent magnet brushless DC Motors in which the necessary flux is given by the permanent magnets but not by the field coils and DC motor in which the flux is given by the current through shunt or field coils. Most widely used controller for DC motors is conventional Proportional-Integral-Derivative (PID) controller.

Manuscript received June 17, 2020

Jatin Yadav, Department of Electrical and Electronics Engineering, Amity University, Gurugram, Haryana, India
Mob. 8826266809, (e-mail: j7yadav0@gmail.com)

Prashant Kumar, Department of Electrical and Electronics Engineering, Amity University, Gurugram, Haryana, India

Dr. Neeraj Gupta, Assistant Professor, Department of Electrical and Electronics Engineering, Amity University, Gurugram, Haryana, India

Ms. Manjeet Kaur, Assistant Professor, Department of Electrical and Electronics Engineering, Amity University, Gurugram, Haryana, India

But due to drive inertia, load changes and limitation of the armature current, there is non-linearity in DC motor which causes problems to these controllers. Overshoot and control gains sensitivity are some disadvantages of the PID controller. Ziegler-Nichols method, Cohen-coon rule, Astrom-Hagguland method are some of the approaches or techniques that can be implemented and are impactful in the parameters tuning of PID controller [1]. Smith predictor PID controller, PID dead time controller are some of the variant PID controllers proposed [2]. The PID controllers with λ (order of the integrator) and μ (order of the differentiator) is known as Fractional Order PID controllers which is the recently proposed generalization of the PID controller [3].

One of the popular and mostly used heuristic tuning methods for a PID controller was proposed by John G. Ziegler and Nathaniel B. Nichols called Ziegler-Nichols tuning method [4][5]. Yet these methods have their own limitations such as the instability of the loop due to a little robustness left behind by the loop and also for a dead-time dominant process, the response is quite poor. Evolutionary techniques like PSO (Particle Swarm Optimization) or particle swarm intelligence, combination of genetic algorithm and artificial neural network and fuzzy system are more popular as these approaches are almost able to find an appropriate and ideal solution for designing and optimization process for PID controllers [6]. ACO method is a known modern heuristic optimization technique. This technique is a part of SI (Swarm Intelligence). In Swarm Intelligence, certain behavior and characteristics of various species are taken and study of some of their specific properties help in realizing the tasks e.g. optimization.

II. CONTROLLERS

A. PID Controllers

Apart from the significant development in the advanced control theory, for over 50 years, the most common and popular controller in process industries is the PID i.e. Proportional-Integral-Derivative controller. In 1989, a survey was conducted by the Japanese association called Japan Electric Measuring Instrument Manufacturers Association whose results concluded that approx. 90% of the industries are considering PID type control loops[7][1]. The required setpoint is different from the measured process variable and this difference is calculated as an

'error' by the PID. The control inputs are processed and adjusted by the controller to minimize this error. This standard three-term controller is made up of the following individual terms

- P term (representing the proportional part)
- I term (representing the integral part)
- D term (representing the derivative part)

and thus, mnemonic PID is referring the first letters of these terms.

The PID controller can be represented as:

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt} + K_i \int_0^t e(t)$$

PID controller experiences various limitations such as, noise in derivative, feedback controller, etc. Fig. 1 depicts the block diagram of PID controller. The steady-state error as well as the rise time can be decreased by the proportional controller i.e. (K_p) while the transient response is degraded by the integral controller (K_i) even though the steady state error is eliminated with its help. Reduction in overshoot, improved transient response and a stable system can be achieved with the help of derivative control (K_d) [8].

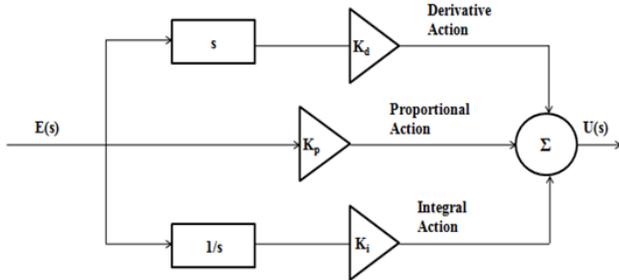


Fig. 1: PID Controller

B. FOPID Controller

Fractional calculus based conventional PID controller is further expanded into FOPID ($PI^\lambda D^\mu$).

In case of, conventional PID controller, the transfer function would be:-

$$G_{PID} = \frac{u(s)}{e(s)} = K_c \left[1 + \frac{1}{\tau_I s} + \tau_D s \right]$$

Similarly, for a FOPID, it would be

$$G_{FOPID} = \frac{u(s)}{e(s)} = K_c \left[1 + \frac{1}{\tau_I s^\lambda} + \tau_D s^\mu \right]$$

Here, the arbitrary real numbers, λ and μ can attain any value

K_c = amplification gain

τ_i = integration constant

τ_d = differentiation constant

The control system dynamics can be better adjusted by a $PI^\lambda D^\mu$ controller as it has more flexibility as compared to conventional PID controller. The fractional order system has a difficult analog realization apart from its simplicity. Intuitively, on comparison with conventional PID controller, $PI^\lambda D^\mu$ is having greater degree of freedom and thus a better performance is expected from $PI^\lambda D^\mu$ with appropriate control parameters [9].

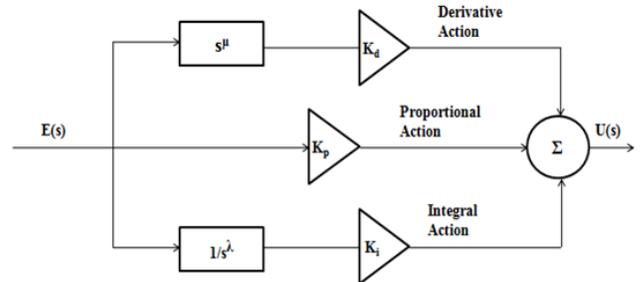


Fig. 2: FOPID Controller

III. MATHEMATICAL MODELLING OF DC MOTOR

The two different control modes of using a dc motor in control systems are:-

- Armature-Control mode (with its field current fixed)
- Field-Control mode (with its armature current fixed)

Here, the Armature-Control mode (with fixed field current) is considered because of its ability to maintain constant torque levels as well as the field current throughout the application. Fig. 3 illustrates the separately excited DC motor.

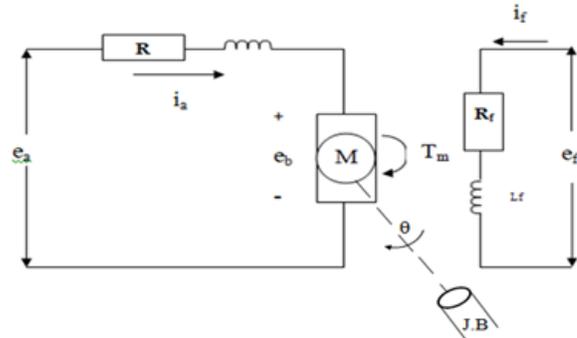


Fig. 3: Separately Excited DC Motor with constant Field Current

R = Armature resistance

L = inductance of the Armature winding (H)

e_a = Applied Armature Voltage (V)

e_b = back e.m.f. (V)

i_a = Armature current (A)

i_f = field current (A)

The developed Torque by the motor is T_m (Nm) and the motor shaft's angular displacement is θ (rad. / sec.). When referring to motor shaft, the motor's equivalent moment of inertia is J ($kg\cdot m^2$) while the coefficient of the motor's equivalent friction is denoted by B ($Nm\cdot s / rad.$) [10].

Generally, for the applications having linear range of magnetization curve, the DC motors are preferably used. The flux is directly proportional to the field current, i.e.

$$\Phi = K_f i_f$$

where, K_f is a constant of proportionality.

The torque developed is also directly proportional to the air gap flux and the armature current product,

$$T_m = K_T i_a$$

where, K_T is known as motor torque constant which is a constant of proportionality. The back e.m.f. and the speed are also directly proportional to each other.

Therefore,

$$e_b = K_b \frac{d\theta}{dt}$$

For an armature circuit, the differential equation would be

$$L \frac{di_a}{dt} + Ri_a + e_b + e_a$$

and thus, the equation for the torque would be would be

$$J \frac{e^2 \theta}{dt^2} + B \frac{d\theta}{dt} - T_m = K_t i_a$$

With the initial conditions as zero, we will take the laplace transform of the equations and thus,

$$\begin{aligned} E_b(s) &= K_b s \theta(s) \\ (Ls + R)I_a(s) &= E_a(s) - E_b(s) \\ (Js^2 + Bs)\theta(s) &= T_m(s) = K_T I_a(s) \end{aligned}$$

Thus the Final Transfer Function would be,

$$\frac{\theta(s)}{E_a(s)} = \frac{K_T}{s [(R + sL)(Js + B) + K_T K_b]}$$

or

$$G(s) = \frac{\omega(s)}{E_a(s)} = \frac{K_T}{(R + sL)(Js + B) + K_T K_b}$$

IV. SWARM INTELLIGENCE

In 1989, Beni and Wang were the first one to include this expression “swarm intelligence” in the cellular robotic systems context. This artificial intelligence technique is basically the collective study of the behaviors of self-organized and decentralized systems. A group of simple agents interacting with each other locally as well as with the environment make up a typical SI (swarm intelligence) system. Usually the behavior of the individual agents is not dedicated by any centralized control structure. But such agents interacting locally are often leading to the global behavior emergence. Bee swarming, animal herding, ant colonies and bacteria molding are some of the examples of systems found in nature.

Depending on the evolutionary computation, the artificial intelligence is succeeded by the computational intelligence (CI). This technique i.e. CI is one of the famous optimization technique. The learning elements i.e. evolution and adaption is combined by CI to create some sensible and intelligent programs. The complementary view is often given by CI research but the statistical methods are not rejected. Multimodal optimization techniques, parameter control methods and the fitness function design are the areas where the fundamental application of the computational intelligence is found. The computational intelligence is imperative because, when compared with the methods of traditional optimization, the optima are more quickly found for more complicated and difficult optimization problems [11]. The further classification of SI is as follows:-

- PSO (Particle Swarm Intelligence)
- ACO (Ant Colony Optimization)
- Bees Search
- Cuckoo Search

V. ACO – ANT COLONY OPTIMIZATION

The real ants, without using any visual cues find the shortest route between their source and their destination e.g. food. Pheromone is a chemical that is deposited by these ants along their path during their movement from source to the destination and then the chemical deposited is used by the ants to follow the path (the path which is rich in pheromone). This behavior of real ants is used to relate the meta heuristic algorithm i.e. ACO as the definition of the methods used to solve wide range of problems. ACO is a part of SI or ant algorithms. Swarm intelligence deals with the algorithmic approaches inspired by many insects’ behaviors such as ant colonies. There are artificial ants that solve the optimization problems in consideration and the information is exchanged via a communication method similar to the one used by the real ants [12].

ACO is a probabilistic one that is used to solve various computational problems which find appropriate and suitable paths through graphs. The multi-agent methods i.e. artificial ‘ants’ are inspired by the real ant’s behavior. The predominant paradigm being used is the biological ant’s communication particularly based on pheromone. These artificial ants are combined with local search algorithms to make a suitable method for various optimizing tasks that involves graphs for example, routing of vehicles or the internet routing.

This ACO, for example can be considered as a class of optimizing algorithms modeled on the action and behavior of ant colony. The simulating agents i.e. artificial ‘ants’ find the optimal and appropriate solutions of the problems moving through a parameter space with all kind of feasible solutions. In the real world, the real ants direct each other to the resources while they explore their environment by laying down the chemical pheromone. Similarly, the artificial ‘ants’ have the records of their positions and their solutions quality so that more and more ants are able to locate the better solutions during the later simulation iterations. The main benefit of using ACO is the confirmation of convergence, adaptive to various changes like distance, speed, position, and providing rapid and appropriate solutions. Although, ACO offer splendid benefits but suffer from uncertainty in convergence time and difficulty in theoretical analysis [13][14].

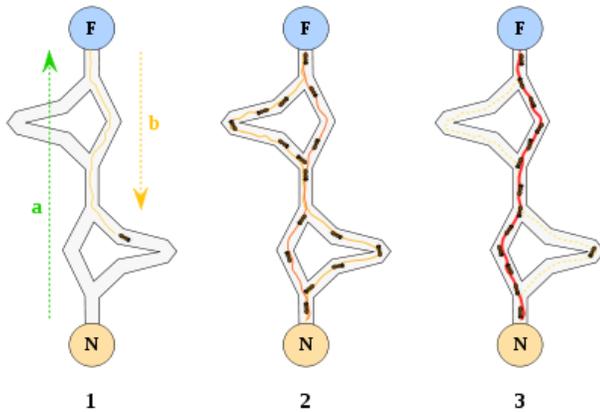


Fig. 4: ACO Technique

VI. RESULTS AND DISCUSSION

In these results, we have used ACO technique to compare the FOPID controller to a conventional PID Controller.

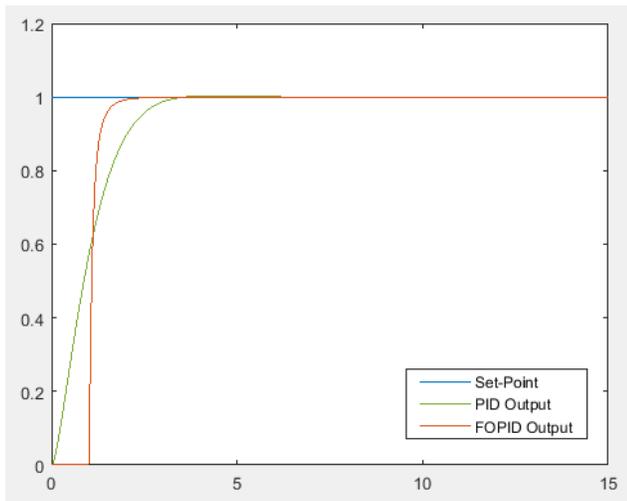


Fig. 5: Comparison of Conventional PID controller with FOPID controller

The above figure show the dynamic responses of both the controllers i.e. Fractional as well as the conventional PID controller. From this graph, it can be clearly observed that in case of a FOPID controller system, the rise time is shorter in comparison to that of a traditional PID controller.

VII. CONCLUSION

In this paper, FOPID controller is studied to optimize the speed control of DC motor. The requirements for speed control process of a DC motor with the help of FOPID and with the use of optimization technique is discussed in this paper. The parameters of FOPID and conventional PID controller are optimally tuned by ACO. Both of the controllers are compared in simulation. The results of FOPID controller can reduce settling time, overshoot and steady state error. Advantages of using FOPID over PID, Ziegler-Nichols tuning method and Ant Colony

Optimization technique is briefly discussed here. Also, conventional PID controller is slower than FOPID controller while stopping the motor. With this we have shown that FOPID has more flexibility and capability.

REFERENCES

- [1] Ahuja, Ashu, and Sanjeev Kumar Aggarwal. "Design of fractional order PID Controller for DC motor using evolutionary optimization techniques." (WSEAS TRANSACTIONS on SYSTEMS and CONTROL).
- [2] Singhal, Rinku, Subhransu Padhee, and Gagandeep Kaur. "Design of Fractional Order PID Controller for Speed Control of DC Motor." (International Journal of Scientific and Research Publications) 2, no. 6 (2012).
- [3] Idir, Abdelhakim, Madjid Kidouche, Yassin Bensafia, Khatir Khettab, and Sid Ahmed Tadjer. "Speed Control of DC Motor Using PID and FOPID Controllers Based on Differential Evolution and PSO." (International Journal of Intelligent Engineering & Systems) 2018.
- [4] Pandey, Narayan Dutt, and Dr. Pratibha Tiwari. "Efficient Fractional Feedback Speed Controller in DC Motor Using FOPID Fed with PSO." (International Journal of Science and Research) 2015.
- [5] Tajbakhsh, Hadi, and Saeed Balochian. "Robust Fractional Order PID Control of a DC Motor with Parameter Uncertainty Structure." (International Journal of Innovative Science, Engineering and Technology) 1, no. 6 (2014).
- [6] Kumar, Vimal, and Dr. A.S. Jhunjhare. "Fractional Order PID Controller for Speed Control of DC Motor using Genetic Algorithm." (International Journal for Scientific Research and Development).
- [7] Mohammed, Ibrahim K., and Abdulla I. Abdulla. "Fractional Order PID Controller Design for Speed Control DC Motor based on Artificial Bee Colony Optimization." (International Journal of Computer Applications.) 2018.
- [8] Puangdownreong, Deacha. "Fractional Order PID Controller Design for DC Motor Speed Control System via Flower Pollination algorithm." (ECTI Transactions On Electrical Eng., Electronics, and Communications) 17 (2019).
- [9] Sondhi, Swati, and Yogesh V. Hote. "Fractional Order Pid Controller for load frequency control." Energy Conversion and management 85, 2014.
- [10] Almatheel, Yaseer Ali, and Ahmed Abdelrehman. "Speed Control of DC Motor Using Fuzzy Logic Controller." (International Conference On Communication, Control, Computing and Electronics Engineering (ICCCCEE)) 2017.
- [11] Cordon, Oscar, Francisco Herrera, and Thomas Stützle. "A review on the Ant Colony Optimization metaheuristic: Basis, models and new trends." (Mathware and Soft Computing) 2002.
- [12] Dorigo, Marco, Mauro Biratari, and Thomas Stützle. "Ant Colony Optimization." IEEE Computational Intelligence Magazine (IEEE Computational Intelligence Magazine), 2006.

- [13]Ning, Jiaxu, Changsheng Zhang, Peng Sun, and Yunfei Feng. "Comparative Study of Ant Colony Algorithms for Multi-Objective Optimization." (MDPI Journals) 2018.
- [14]Ibrahim, H.E.A., and Ahmed A. Hakim Mahmoud. "DC Motor Control using PID Controller Based on Improved Ant Colony Algorithm." (International Review of Automatic Control (I.R.E.A.CO.)) 7 (2014.).