

# Real Time Speed Control of a DC Motor Based on its Integer and Non-Integer Models Using PWM Signal

Abdul Wahid Nasir

Electrical & Electronics Eng. Dept.  
NIT Jamshedpur  
Jamshedpur, India  
2014rsee003@nitjsr.ac.in

Idamakanti Kasireddy

Electrical & Electronics Eng. Dept.  
NIT Jamshedpur  
Jamshedpur, India  
2015rsee002@nitjsr.ac.in

Arun Kumar Singh

Electrical & Electronics Eng. Dept.  
NIT Jamshedpur  
Jamshedpur, India  
aksingh.ee@nitjsr.ac.in

**Abstract**—This paper exploits the advantage of non-integer order modeling of a process over integer order, in those cases where the process model is required for control purpose. The present case deals with speed control of a DC motor. Based on the real time open loop response, DC motor is being modeled as integer and non-integer order first order plus delay system. Both these models are then separately used for determining two sets of Proportional-Integral-Derivative (PID) controller parameters through Ziegler Nichols (ZN) closed loop tuning method. In addition to this, a model based control technique i.e. Internal Model Control (IMC) is also implemented using both integer and non-integer model respectively. For carrying out the real time speed control of DC motor, LabVIEW platform has been used. After going through the results, it is observed that the controller performance considerably improves, if non-integer order model is used for controller design rather than integer order model.

**Keywords**—DC motor speed; fractional order system; PID; internal model control; LabVIEW

## I. INTRODUCTION

DC motor is having a prominent role in almost all industrial and robotic applications. Its simplicity, reliability, economic feasibility and the ease with which such types of DC motor speed can be controlled, makes it omnipresent. With the recent advancements in power electronics, various switching control techniques emerged. Pulse Width Modulation (PWM) technique is one of them, which has been used here for the speed control of DC motor by manipulating its duty cycle. Here ZN-PID & IMC control technique is used to meet the control objective. Since both these aforesaid control techniques require model of the process for controller design, therefore integer and non-integer (fractional) order model is determined from open loop step response of the DC motor. Here, the concept of fractional calculus has been introduced in process modelling. The branch of calculus which is evolving due to the generalization of order of conventional integro-differential operator from integer to non-integer, is known as fractional calculus. Due to increasing interest in this among researchers in recent decades [1-3], the solutions of various problems incorporating non-integer order integro-differential operator are now available making its application feasible in various fields of science and technology. These non-integer order

fundamental operators, where ‘ $\alpha$ ’ is the operator order and ‘ $a$ ’ & ‘ $t$ ’ give the limits of the operator is defined as follows:

$${}_a\mathcal{D}_t^\alpha = \begin{cases} d^\alpha/dt^\alpha, & \alpha > 0 \\ 1, & \alpha = 0 \\ \int_a^t (d\tau)^{-\alpha}, & \alpha < 0 \end{cases} \quad (1)$$

Different definitions of this fractional integro-differential are given in [4]. The terms fractional order and non-integer order are used interchangeably. For carrying out all real time experimental work, LabVIEW platform has been used. For interfacing the DC motor dynamics with LabVIEW platform through computer, a LabJack U3-HV data acquisition card (DAC) was employed. For the RPM measurement of the motor, an infrared (IR) proximity sensor was used. In Section II step-wise outline of the work carried out is given. Experimental setup is described in section III. Section IV gives the modelling of a DC motor based on its open loop data. The control schemes implemented for the speed control of DC motor is discussed briefly in section V. Experimental results are presented and discussed in section VI. Based on the various results obtained, conclusion are drawn and scope for future work is given in Section VII.

## II. EXPERIMENTAL SETUP

The experimental setup for the DC motor speed control system is shown in Figure 1. It includes one 12 volts / 1-amp DC motor having maximum RPM of 2000. To the shaft of this motor is attached one circular disc, whose one half of the circular area is colored in silver and another half in black. An infrared (IR) proximity sensor having one IR transmitter and one IR receiver is engaged to sense the RPM of motor. The IR transmitter continuously transmits the IR ray on the rotating disc which is reflected back to the IR receiver to give positive output when falls on silver area and zero output when falls on black area, as the black area absorbs the IR ray rather than reflecting it. Thus depending on the rotational speed of the motor a pulsating signal of specific frequency is generated at

the output of IR transmitter. At the interval of every 0.1 second, the output frequency is measured and updated to the control algorithm employed in LabVIEW environment through a LabJack make U3-UV, data acquisition card. Based on current RPM information, the controller generates control signal in the form of PWM signal through the same data acquisition card. This signal is then fed to L298N DC motor driver circuit, through which the speed of motor is regulated to desired point. This is how a real time closed loop speed control system prototyping is implemented.

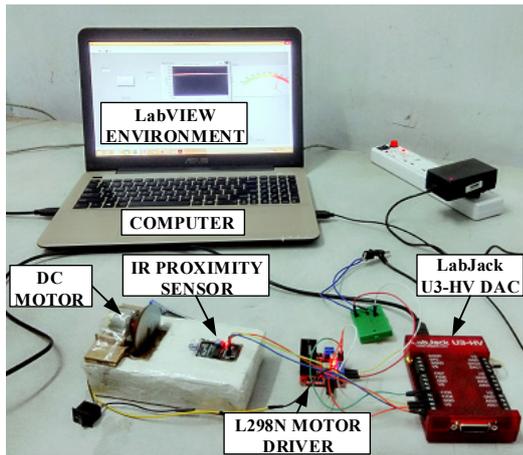


Fig. 1. DC motor speed control system

### III. INTEGER & NON-INTEGERS ORDER MODELLING

For the control system under study i.e. speed control of a DC motor, first of all different open loop step responses are noted to analyze the system behavior. Since PWM is the input manipulating variable and RPM of the motor is output, controlled variable, therefore the duty cycle (in percentage) of PWM signal is varied to obtain the steady state RPM corresponding to each input signal. This input-output data is then plotted as shown in Figure 2.

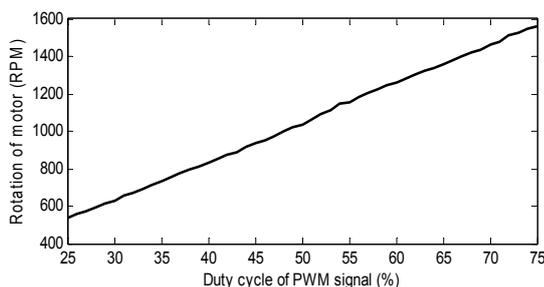


Fig. 2. Input output steady state characteristics

From the nature of the plot, it can be concluded that within the specific input range of duty cycle i.e. 25% to 75%, DC motor speed response almost exhibits linear characteristic. So, the controller designed for any operating condition within this range should work well throughout the RPM range of 550 to 1600. The open loop transfer function relating input i.e.

armature voltage and output i.e. rotational speed is given by (2) [5]:

$$G_{DC\_motor} = \frac{\dot{\theta}(s)}{V(s)} = \frac{K_t}{(Ls + R)(Js + b) + K_e K_t} \quad (2)$$

where

$\dot{\theta}$ :	rotational speed of motor (RPM)
V:	armature voltage (V)
J:	moment of inertia of rotor (Kg.m <sup>2</sup> )
b:	motor viscous friction constant (N.m.s)
$K_e$ :	electromotive force constant (V/rad/sec)
$K_t$ :	motor torque constant (N.m/Amp)
R:	electric resistance (ohm)
L:	electric inductance (H)

In many cases, where  $R \gg L$ , the transfer function given by (2) reduces to first order system given by (3).

$$G_{DC\_motor} = \frac{K_t}{R(Js + b) + K_e K_t} \quad (3)$$

Now the next step is to estimate the DC motor model parameter based on the step response. A step change in duty cycle of PWM signal is made from 50% to 75%, and RPM response data is obtained. Since this response very much matches the characteristics of step response of first order plus delay system. Therefore, it is deduced that the current system can be modelled as first order plus delay system. Making use of the flexible feature [6-7] of fractional order calculus in modelling, the present system is not only modelled as integer but also as non-integer order system as given by (4) and (5) respectively.

$$G_{integer} = \frac{K}{\tau s + 1} e^{-Ls} \quad (4)$$

$$G_{non-integer} = \frac{K}{\tau s^\alpha + 1} e^{-Ls} \quad (5)$$

Here, Genetic Algorithm (GA) is used to estimate the values of parameters for integer model i.e. gain (K), time Constant ( $\tau$ ) and delay (L) to minimize the value of Integral of Squared Error (ISE), the objective function considered [8-9]. Similarly, the unknown parameters of non-integer model are obtained i.e.  $\alpha$ , the order of the filter, along with K,  $\tau$  & L. GA is stochastic search computational tool based on the theory of natural evolution, basically involving the process of selection, crossover and mutation. In 1975, Holland [10], proposed the elementary principles of GA. Based on this novel topic, later some literatures also surfaced [11]. To begin with, a random set of parameters is presumed, which mimics the genes of chromosomes, to be the possible solution of the search problem. An objective function is defined that reflects the fitness of these chromosomes, which when subjected to minimization or maximization under certain constraints yield the optimal result. The objective function considered for estimation of model parameters is Integral of Squared Error (ISE) subject to minimization is given by (6). Different sets of

model parameters are chosen from the search space based on GA at every simulation run, and the step response data of that particular model is obtained. This obtained model data is then compared to already existing system data to generate error which is ultimately used to calculate ISE. This ISE value is again passed to GA as feedback information, and this process continues until optimal minimum is reached. Figure 3 represents the implementation of GA for model parameter estimation.

$$ISE = \int_0^{\infty} \{y_{sp}(t) - y(t)\}^2 dt = \int_0^{\infty} \{e(t)\}^2 dt \quad (6)$$

The discussed GA is implemented using Matlab for current problem of finding model parameters. For carrying out all types of simulation work for model parameter estimation related to non-integer order plants, FOMCON toolbox [12], has been used in Matlab. Table I shows the GA characteristics considered in the present case. Table II gives the integer and non-integer order transfer function model for DC motor speed with their corresponding ISE values and Figure 4 represents the step response for the DC motor speed system along with its integer and non-integer model. In the inset of Figure 4, zoomed responses for specific time duration are given, so that the difference among all the three responses can be identified easily.

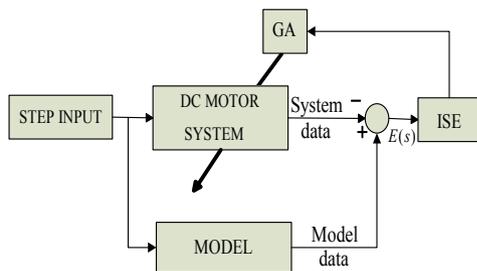


Fig. 3. Model parameters estimation using GA

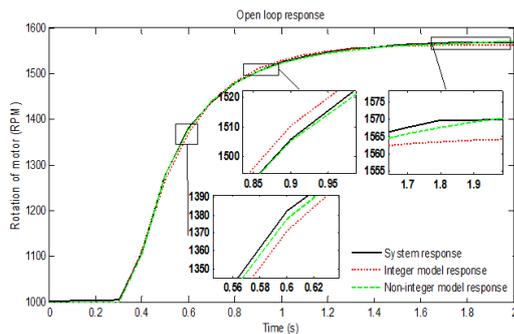


Fig. 4. Open loop step responses

TABLE I. GA CHARACTERISTICS

Population size	50
Fitness Scaling Function	Rank
Crossover Function	0.8
Crossover Fraction	Scattered
Migration Fraction	0.2
Ending Criterion	Function tolerance of 1e-4

TABLE II. MODEL PARAMETERS

Nature of Model	Transfer Function	ISE
Integer Order	$\frac{22.58}{0.235s + 1} e^{-0.347s}$	440
Non-Integer Order	$\frac{23.358}{0.25s^{0.91} + 1} e^{0.364s}$	154

IV. CONTROLLER DESIGN

In spite of advancements in different schemes of controller in the recent past, PID still rules the application domain due to its simple architecture and its ability to control various types of linear and non-linear processes. Most of the tuning of controller parameters for PID requires model of the process. Another very simple control technique based on model of the process is Internal Model Control (IMC). Both these control techniques i.e. PID & IMC are implemented for the present PID & IMC are implemented for the present case of DC motor speed control, and are discussed in nutshell under the following subsection respectively.

A. PID Control

Transfer function representation of PID controller relating error, E(s) and control signal, U(s) is given by (7).

$$\frac{U(s)}{E(s)} = Kp \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (7)$$

where  $K_p$ ,  $T_i$  &  $T_d$  are the controller tuning parameters. The transfer function of PI controller can be obtained by substituting  $T_d=0$ , in (7). Similarly, for obtaining P-type controller transfer function, set  $T_i=\infty$  and  $T_d=0$ . The basic PID controller tuning rules were proposed in [13], and were based on empirical data obtained from extensive experiments. Authors in [14], further modified the tuning rules. Since then various researchers are proposing several other methods for PID controller tuning. At present, Ziegler Nichols closed loop method/cycling method is used. Although it is experimental based control design, but since the model of the process to be controlled is available, therefore, without disturbing the actual process, the controller parameters can be obtained. First of all, ultimate gain,  $K_u$  and ultimate period,  $P_u$  is determined by making  $T_d=0$  and  $T_i=\infty$  and varying proportional gain,  $K_p$  such that the closed loop response becomes oscillatory having constant amplitude. The value of  $K_p$  at which sustained oscillation occurs is the ultimate gain,  $K_u$  and the period of the oscillation is the ultimate period,  $P_u$ . After that the P, PI & PID parameter can be obtained using Table III.

TABLE III. ZN CLOSED LOOP PID TUNING

Controller Type	$K_p$	$T_i$	$T_d$
P	$0.5K_u$	--	--
PI	$0.45K_u$	$P_u/1.2$	--
PID	$0.6K_u$	$P_u/2$	$P_u/8$

B. Open Loop IMC

Internal model control (IMC) is a model based control technique, where a process is “embedded” within the

controller [15]. The open loop IMC structure is given in Figure 5, where  $q(s)$  is the model based controller and  $G(s)$  is the process.

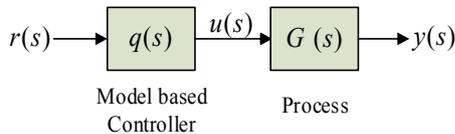


Fig. 5. Open loop internal model control

Consider a first order process  $G(s)$ , given by (8), whose controller transfer function  $q(s)$  is given by (9).

$$G = \frac{K}{\tau s + 1} \tag{8}$$

$$q(s) = f(s)G^{-1}(s) \tag{9}$$

where  $G^{-1}(s)$  is the invertible part of  $G(s)$ , with right hand side zeros and delay being factored out if they are present, to make  $q(s)$  stable.  $f(s)$  is the filter defined by (10).

$$f(s) = \frac{1}{(\lambda s + 1)} \tag{10}$$

The filter,  $f(s)$  is incorporated in the controller transfer function  $q(s)$  to make it proper i.e. physically realizable. The open loop control response  $y(s)$  is,

$$y(s) = q(s)G(s)r(s) = f(s)r(s) \tag{11}$$

As seen from (11), the output response depends on filter time constant  $\lambda$ , hence it is considered as controller parameter. In the present case, IMC technique will also use non-integer model, other than integer order model. Therefore, before implementing non-integer model in real time control in LabVIEW platform, it is firstly converted to the equivalent higher integer order model using method described in [16], because the dynamics of non-integer order system can be well represented with the help of higher integer order model and vice-versa.

V. RESULTS AND DISCUSSIONS

The PID controller parameters were tuned using Ziegler Nichol's closed loop technique. As discussed in subsection IV.A, ultimate gain  $K_u$  and ultimate period,  $P_u$  is determined. Since two types of model are available for the process. i.e. integer and non-integer order model, therefore for each nature of model,  $K_u$  and  $P_u$  are derived respectively as shown in Table IV.

TABLE IV. KU AND PU VALUES

Parameter	IO model	FO model
$K_u$	0.2125	0.2800
$P_u(\text{sec})$	1.8338	1.3035

Based on these values of  $K_u$  and  $P_u$ , the controller parameters for P-type, PI-type and PID-type control are determined using ZN formula as given in Table V for both the

integer & non-integer model, which have been tabulated as follows in Table V.

TABLE V. P, PI AND PID CONTROLLER PARAMETERS

Controller Type	$K_p$	$T_i(s)$	$T_d(s)$	
P	IO model	0.106	--	
	FO model	0.140	--	
PI	IO model	0.095	1.528	
	FO model	0.126	1.086	
PID	IO model	0.127	0.917	0.229
	FO model	0.168	0.652	0.163

The P, PI & PID obtained from integer order model is tested against the step change in RPM. Similarly, the controller performance is also evaluated for controllers obtained from non-integer order model. All the real time closed loop controlled response data is collected and plotted to have the comparative study. Figures 6-8, represent the servo response for P, PI & PID control respectively. Table VI gives the various performance indices for the different controllers under test.

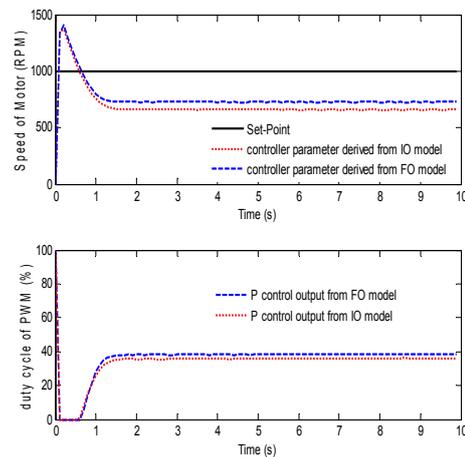


Fig. 6. Servo Response of P controlled system

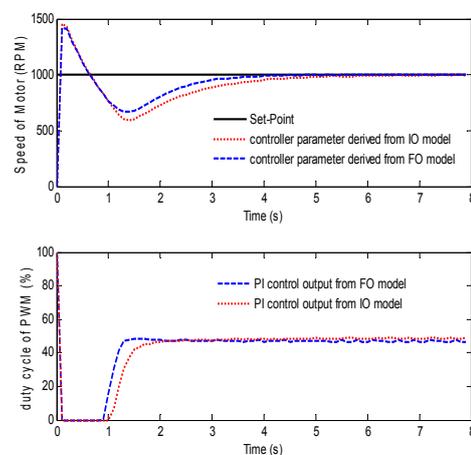


Fig. 7. Servo Response of PI controlled system

After analyzing the closed loop step responses in Figures 6–8 and the data indicating various performance indices in Table VI, it can be easily deduced that performance for the class of controllers obtained using non-integer order model is superior in every aspect than those obtained from integer order model for all P, PI & PID control schemes. There is one insignificant exception in P-type controller whose maximum overshoot is slightly more for the controller design using non-integer order model rather than integer order model. Other performance index parameters such as rise time ( $t_r$ ) and peak time ( $t_p$ ) have not been included in Table VI, as these parameters are almost identical for both cases, i.e. for controller obtained from integer and non-integer model. Even, Figures 6-8, the closed loop step response, conveys the same message. Moving to other control technique employed for real time speed control of DC motor, i.e. IMC, where the filter time constant,  $\lambda$  is controller parameter. The value of  $\lambda$  determines the nature of control performance. The aggressiveness of controller is inversely

proportional to  $\lambda$  value, whereas robustness is directly proportional to value of  $\lambda$ . Hence the value of  $\lambda$  is obtained through extensive simulations and is found to be  $\lambda=0.1$ . Finally, IMC open loop control is implemented in LabVIEW environment using the same value of  $\lambda$ . Here only open loop control has been implemented in order to just validate the fact that, since non-integer model captures the real time system dynamics better than integer order model, therefore the model based control technique incorporating non-integer model gives improved results rather than integer model. Figure 9 shows the open loop control responses for IMC obtained from both integer and non-integer order model for the step change made at  $t=0.4$  second, from initial value of 0 to 600 RPM. Table VII gives the transfer functions of different IMC along with performance indices data. Figure 9 and Table VII data clearly depict the supremacy of IMC based on non-integer order model over integer order model.

TABLE VI. PERFORMANCE INDICES FOR P, PI & PID CONTROLLED SYSTEM

Performance Indices	P-type controller design using		PI-type controller design using		PID-type controller design using	
	Integer model	Non-integer model	Integer model	Non-integer model	Integer model	Non-integer Model
Settling time, $\pm 5\%$ (s)	1.26	1.13	3.94	2.96	3.48	2.56
Settling time, $\pm 2\%$ (s)	1.43	1.28	5.02	3.72	3.91	2.86
Overshoot (%)	38	41	45	41	41	37.5
Undershoot (%)	--	--	40	33	36.5	33.4
Offset	340	274	Nil	Nil	Nil	Nil
ISE	1.16e+7	8.12e+6	3.24e+6	2.52e+6	3.05e+6	2.35e+6
IAE	3.29e+4	2.72e+4	9.49e+3	7.19e+3	8.71e+3	6.44e+3

VI. CONCLUSION

In this paper, two different control techniques, PID and IMC have been implemented for the real time speed control of a DC motor. The aim of the present paper is to highlight the fact that non-integer order model is more efficient in capturing the dynamics of real time systems as compared to integer order model, and also the controllers based on non-integer models yield better performance rather than those based on the integer model. The conclusion is well validated by the results obtained from experimental setup. This work can be extended to control any system, wherever a model is required for controller design

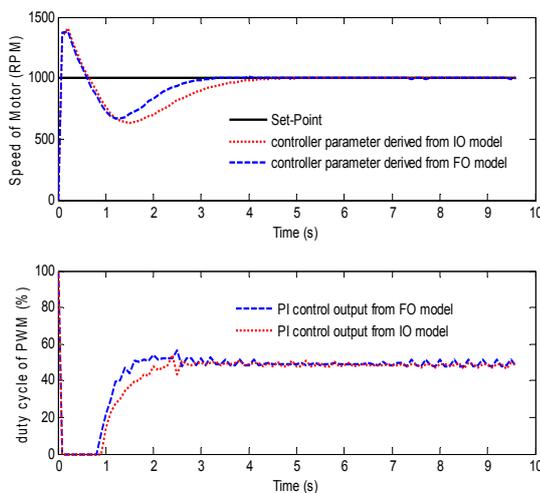


Fig. 8. Servo Response of PID controlled system

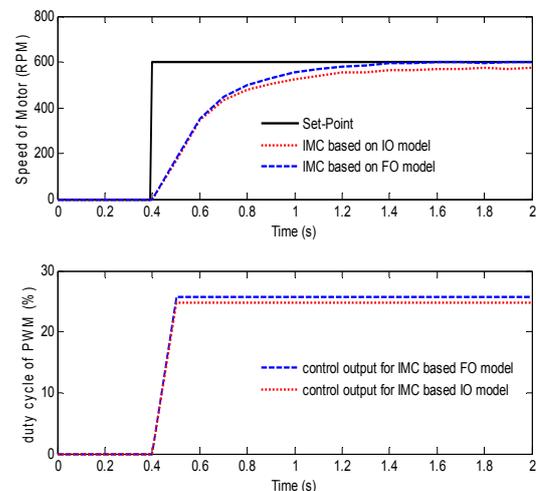


Fig. 9. IMC open loop control responses

TABLE VII. IMC PERFORMANCE INDICES

IMC based on	Controller TF, $q(s)$	offset	ISE	IAE
integer order model	$\frac{0.235s + 1}{22.58(0.1s + 1)}$	30	3.22e+5	1.51e+3
non-integer order model	$\frac{0.25s^{0.91} + 1}{23.358(0.1s + 1)}$	negligible	2.79e+5	1.12e+3

## REFERENCES

- [1] A. S. Elwakil, "Fractional order circuits and systems magazine: An Emerging interdisciplinary research area", IEEE Circuits and Systems, Vol. 10, No. 4, pp. 40-50, 2010
- [2] M. O. Efe, "Fractional order systems in industrial automation – A Survey", IEEE Transaction of Industrial Informatics, Vol. 7, No. 4, pp. 582-591, 2011
- [3] S. Kumar, "A new fractional modelling arising in engineering sciences and its analytical approximate solution", Alexandria Engineering Journal, Vol. 52, No. 4, pp. 813-819, 2013
- [4] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, V. Feliu, Fractional order systems and controls: Fundamental and applications, Springer-Verlag, London, 2010
- [5] "DC Motor Speed: System Modeling", [http://ctms.engin.umich.edu/C\\_TMS/index.php?example=MotorSpeed&section=SystemModeling.html](http://ctms.engin.umich.edu/C_TMS/index.php?example=MotorSpeed&section=SystemModeling.html)
- [6] A. Vishwesh, P. S. V. Natraj, "Fractional order modeling of neutron transport in a nuclear reactor", Applied Mathematical Modelling, Vol. 37, No. 23, pp. 9747-9767, 2013
- [7] T. Djamah, R. Mansouri, S. Dfenounce, M. Bettayeb, "Optimal low order model identification of fractional dynamic systems", Applied Mathematics and Computation, Vol. 206, No. 2, pp. 543-554, 2008
- [8] M. Lankarany, A. Rezaade, "Parameter Estimation Optimization Based on Genetic Algorithm Applied to DC Motor", 2007 International Conference on Electrical Engineering, Lahore, pp. 1-6, April 11-12, 2007
- [9] S. Udomsuk, K. L. Areerak, K. N. Areerak, A. Srikaew, "Parameters identification of separately excited DC motor using adaptive tabu search technique", International Conference on Advances in Energy Engineering, Beijing, pp. 48-51, June 19-20, 2010
- [10] J. H. Holland, Adaption in Natural & Artificial Systems. Cambridge MA: MIT Press, 1975
- [11] D. E. Goldberg, Genetic Algorithms in search Optimization and Machine Learning. Boston, MA: Addison-Wesley, 1989
- [12] A. Tepljakov, E. Pettenkov, J. Belikov, "FOMCON: a MATLAB toolbox for fractional-order system identification and control", International Journal of Microelectronics and Computer Science, Vol. 2, No. 2, pp. 51-62, 2011
- [13] J. G. Ziegler, N. B. Nichols, "Optimum settings for automatic controllers", Transactions of ASME, Vol. 64, pp. 759-768, 1942
- [14] G. H. Cohen, G. A. Coon, "Theoretical considerations in retarded control", Transaction of ASME, Vol. 75, pp- 827, 1953
- [15] B. Wayne Bequette, Process Control: Modeling, Design and Simulation, 2nd Ed., Prentice Hall, 2002
- [16] D. Xue, C. Zhao, Y. Chen, "A modified approximation method of fractional order system", International Conference on Mechatronics and Automation, Luoyang, Henan, pp. 1043-1048, June 25-28, 2006