Optimal Tuning of Fractional Order PID Controller for First Order Time Delay System (FOTD) Using PSO Algorithm

M.Theja[1], N.Ramesh Raju[2],
[1]Department of EEE,SIETK, Puttur
[2]Department of EEE, SIETK, Puttur

Abstract: this paper an innovative design method for determining fractional order PID ($PI^{\lambda}D^{\mu}$) controller parameters of a First Order Time Delay System (FOTD) system using particle swarm optimization algorithm is presented. This paper presents how to employ the particle swarm optimization to seek efficiently the optimal parameters of $PI^{\lambda}D^{\mu}$ controller. The study is made for this controller against parameter variation of FOTD system. System of the controller depicted here is based on user- specified peak overshoot and settling time and has been formulated as a single objective optimization issue. Finally, better simulation results and control presentation of the PSO based Fractional-Order PID (FOPID) will be showed in these controllers in gathering with those of the integer-order PID controllers. This work has been simulated in MATLAB environment with FOMCON (Fractional Order Modeling and Control) tool box.

Keywords: Particle Swarm Optimization (PSO), Fractional Order $PI^{\lambda}D^{\mu}$ Controller, First Order Time Delay System (FOTDS).

I Introduction

In the past decades, modern control theories have made great advances. Control techniques including optimal control, H1/H2 control, fuzzy control, neural network control, predictive control, and so on, have been developed. Nevertheless, the Proportional-Integral-Derivative (PID) control technique has still been widely utilized in many industrial applications such as process control, motor drives, flight control, etc. Now a day, more than 90% control loop in industry are PID control. This is mainly due to the fact that PID controller possesses robust performance to meet the global change of industry process, simple structure to be easily understood by engineers and easiness to design and implement.

The tuning method of PID controllers have been studied intensively in the past such as the well-known Ziegler-Nichols tuning rule[2] for the first

order plus time-delay transfer functions, which models a wide class of processes possessing an S-shape reaction curve in step responses. Another research line for the design of PID controllers is to determine the stabilizing parameter set of PID controller, this set was first shown in [3] as convex polygons for delay- free systems by an extension of Hermite-Biehler theorem presented by Pontryagin [4]. Then, the approach was applied to first–order plus time-delay systems and the convex polygon property was extended to this case [5]. By using the Nyquist stability criterion, the same results as those in [6] were obtained which gave an alternative simple derivation. Some other methods with the alike Smith structure like GPC (Generalized Predictive Control) have been successfully used in PID controller design for delayed SISO systems.

Recently, there are increasing interest to enhance the performance of PID controller by using the concept of fractional calculus, where the orders of derivative and integrals are non-integer. Fractional calculus in non-local makes it able to emphasize mathematically the long-memory. Fractional Order PID controller (FOPID or, $PI^{\lambda}D^{\mu}$ where and are the integrating and derivative orders and they are non-integers) proposed by [9] is a generalization of the PID controller using fractional calculus. A FOPID controller is characterize by five parameters, i.e., the proportional gain, the integrating gain, the derivative gain, the integrating order and the derivative order. Over the last years, FOPID controllers find many applications in irrigation canal control[10], temperature tracking[11], motion control of DC motor [12], [13], boost converter control [14].

The above research results show that FOPID controller has better performance and robustness than conventional PID controller. In the literature, many approaches have been proposed to design FOPID controller. These approaches can be classified into two classes: analytic methods and heuristic methods. In the analytic context, the parameters of FOPID controller are tuned by minimizing a nonlinear objective function depending on the specifications imposed by the designers. In [15], tuning of FOPID controller is re casted as a Quantitative Feedback Theory (QFT) loop shaping problem, where the optimization objective is the high frequency gain of the nominal loop subjected restrictions given by the specifications. [16] proposed a new analytic method to design FOPID controller by expanding the control loop signal and reference model input and output over a piecewise orthogonal functions. As far as the heuristic methods, rule–based methods and evolutionary algorithm based methods were explored by several authors.

In [17], a complex of tuning cods was stated based on a first order plus time model of the process by minimizing the integrated absolute error with a border for the maximum sensitivity.

Ziegler-Nichols like tuning rules for FOPID controller was given in [18]. Evolutionary algorithms including Genetic Algorithm (GA), Particle Swarm Optimization (PSO) and Electromagnetism like algorithm (EM) are also used to design FOPID controller. Genetic algorithms were adopted by Cao and Meng and Xue [19] to design FOPID controller by recasting the problem to an optimization problem. In [20], particle swarm optimization was used to design FOPID controller for an AVR system. An Improved Electromagnetism- like Algorithm with Genetic Algorithm (IEMGA) technique was proposed in [21] for FOPID controller design through minimizing the Integrated-Square-Error (ISE). Lately, Differential Evolution (DE) as a plain and impressive plot for global optimization over steady spaces is becoming increasingly popular. In the present work, scheme of FOPID controller using PSO algorithm styles has been investigated for time delay systems.

II First Order Time Delay System (FOTD)

The empirical method of identifying the system is the most modern method. Empirical models use data gathered from experiments to define the mathematical model of a system. A step change in the input to a process produces a response, which is called process reaction curve .A variety of empirical modelling methods exists. One method for developing models uses system identification methods. System identification methods use the input and output data to create difference equation which are used for representations that model the data.

In general terms, the time constant (τ), describes how fast the process variable (PV) moves in response to a change in the controlled output (CO). The time constant must be positive and it must have units of time. Most often it has units of minutes or seconds. Step test data implies that the process is in manual mode (open loop) and initially at steady state.

The transfer function models are required only for the simulation studies of the controller design. Here we are controlling the level (H) of the tank by manipulating the flow rate (Q). The most commonly used model to describe the dynamics of the industrial level process is general First Order plus Time Delay Process (FOPTD). And the FOPTD model structure is given in equation (1)

$$G(s) = \frac{K}{\tau s + 1} e^{-\theta ds} \quad (1)$$

Ɵd – Time delay

Kp – Process gain

τ - Time constant

Here the process of interest is approximated by a First Order plus Time Delay Process. The dead time approximation

III Heart of Fractional Calculus and Fractional Linear Systems

The fractional calculus is a 300 years old mathematical discipline, but there are still various mathematical praises that may lead to various conclusions. In fact a unique praise for fractional integration exists. However infinity of formulae could be derived for fractional differentiation. This prevent establishment of systematic theory for fractional linear systems. In other to eschew this obscurity, which is usually relevant to primary status, generally limitation of them to null value is made.

(a) Fractional Integration

The fractional integral (Riemann-Liouville integral) of a function f (t) is defined by23:

$$\left({}_aI_t^{\alpha} f\right)(t) \triangleq \frac{1}{\Gamma(\alpha)}\int_a^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}}d\tau \quad (2)$$

where t>α and α is the real positive integration order, is the Euler Gamma function:

$$\Gamma(\alpha) = \int_0^{\infty} e^{-t} x^{\alpha-1} dx \quad (3)$$

(b) Fractional Differentiation

The Riemann-Liouville definition24 is given by:

$${}_a^{RL}D_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)}\left(\frac{d^m}{dt}\right) \int_a^t \frac{f(\tau)}{(t-\tau)^{1-(m-\alpha)}}d\tau \quad (4)$$

where m is the integer satisfying m-1<α<m. It is main to regard that these two compliments are specific occasions of a limitless probability of successive differ integration.

IV the Integer and Fractional Order PID Controllers

The integer order PID controller has the following transfer function:

$$K_p + T_i s^{-1} + T_d s \quad (5)$$

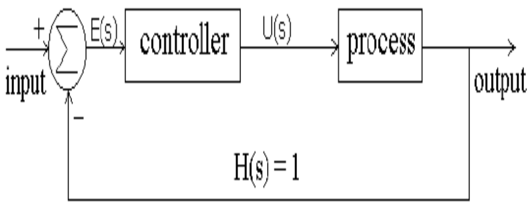

Fig. 1: Generic Closed Loop System

The real objects or processes that we want to control are generally fractional (for example, the voltage-current relation of a semi-infinite lossy RC line). However, for many of them the fractionality is very low. In general, the integer-order approximation of the fractional systems can cause significant differences between mathematical model and real system. The main reason for using integer-order models was the absence of solution methods for fractional-order differential equations. PID controllers belong to dominating industrial controllers and therefore are objects of steady effort for improvements of their quality and robustness. One of the possibilities to improve PID controllers is to use fractional-order controllers with non-integer derivation and integration parts. A fractional PID controller therefore has the transfer function:

$$K_P + T_i S^{-\lambda} + T_d S^{\mu} \quad (6)$$

The orders of integration and differentiation are respectively λ and μ (both positive real numbers, not necessarily integers). Taking λ =1 and μ =1, we will have an integer order PID controller. So we see that the integer order PID controller has three parameters, while the fractional order PID controller has five.

The fractional order PID controller generalizes the integer order PID controller and expands it from point to plane. This expansion adds more flexibility to controller design and we can control our real world processes more accurately. We will design both integer order and fractional order PID controllers using the particle swarm optimization (PSO) algorithm and display the advantages the fractional order controllers provide us over the integer order controllers.

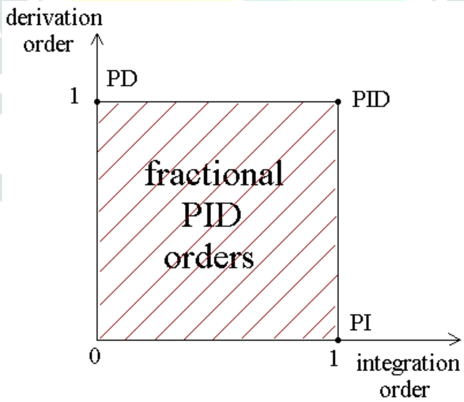

Fig. 2. Expanding from Point to Plane

V Standard PSO Algorithm

Particle swarm optimization method was introduced by Kennedy and Eberhart in1995. It is evolutionary optimization technique and stochastic method, developed by observing the social movement of swarms such as fish schooling and bird flocking. This method is robust in solving problems featuring nonlinearity, non-differentiability, multiple optima and high dimensionality. It has stable convergence characteristics with good computational efficiency and easily implementable. Unlike other evolutionary methods where the evolutionary operators manipulate the particle, each particle in PSO flies in the search space with velocity which is dynamically adjusted according to its own flying experience and flying experience of its companions'.

At the beginning PSO algorithm introduces 'N' number of particles randomly. The objective function value is obtained for each particle. Then based on the flying velocity of the particle and its group the new population of particles are generated for next generation in seeking still better solution. The best value obtained by the particle so far is called pbest and the best value obtained among all the particles is called gbest. Each particle in the group updates their velocity based on the pbest and gbest as given in equation (1) and (2).

Let us assume jth particle is represented as xj = (xj,1, xj,2, ……xj,n) in n dimensional space. The previous best position of the jth particle is recorded as pbestj = (pbestj,1, pbestj,2, ……pbestj,n). The best particle among the group is represented by gbestg. The velocity of the particle j is represented as vj = (vj,1, vj,2 …… vj,n). The calculation of modified velocity and position of each particle using velocity and distance through pbestj,g to gbestg is done as shown in the following formulas:

$$V_{j,n}(t+1) = w.v_{j,n}(t) + c1*rand()*(pbest_{j,n}-x_{j,n}(t)) + c2*rand()*(gbest_g-x_{j,n}(t)) \quad (6)$$

$$x_{j,n}(t+1) = x_{j,n}(t)+v_{j,n}(t+1) \quad (7)$$

j= 1, 2 ,……………,N n=1, 2 ,……………,M

Where,

Nnumber of particles in a group

M number of members in a particle

t generation number

vj,n(t) velocity of particle j at generation t

w inertia weight factor

c1, c2 acceleration constant

rand() Random number between 0 and 1

xj,n(t) current position of particle j at generation t

pbestj pbest of particle j

gbestg gbest of the group

The updated particles are the population for next generation and continue the above procedure up to the specified number of generations. The better solution is obtained at each subsequent generation.

VI. MATLAB SIMULATION

The simulation result of proceeding plant is obtained by using MATLAB model. The model is constructed as shown in figure 3. The calculated value is

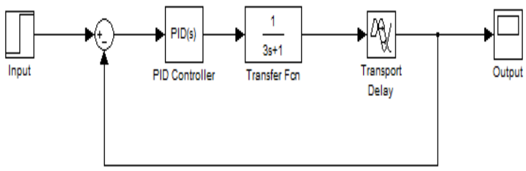

Fig 3. MATLAB/ Simulink Model

Case1: Consider the following stable process Where K = 0.4, Tc = 0.2, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd .

Case2: Consider the following stable process Where K = 0.4, Tc = 0.4, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd .

Case3: Consider the following stable process Where K = 0.4, Tc = 0.6, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd .

Case4: Consider the following stable process Where K = 0.4, Tc = 0.6, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd.

Case5: Consider the following stable process Where K = 0.4, Tc = 0.8, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd.

Case5: Consider the following stable process Where K = 0.4, Tc = 1, and Td=0 Simulation results using MATLAB for different PID tuning techniques are summarized in Table 1&2. Used for kp, ki and kd.

TABLE 1

Simulation PID Parameters

PID Parameters										
	System parameters			Controller Parameters			Time response parameters			
S.No	K	Tc	Td	Kp	Ki	Kd	Overshoot	ISE	ettlingtim	Obj Fnc
1	0.4	0.2	0	3.8975	4.9953	3.9817	5.17	59.09	3.99	71.27
2	0.4	0.4	0	3.9563	3.9563	0.6904	3.25	8.08	0.00	20.18
3	0.4	0.6	0	3.9942	3.9942	3.2462	2.00	30.13	3.99	70.62
4	0.4	0.8	0	4.9158	4.9158	2.0897	1.83	20.84	1.48	60.82
5	0.4	1	0	3.8975	5.6656	3.2702	2.77	20.44	3.89	70.48
Average Values of PID Parameters(AVPID)				3.9942	5.9158	4.0897				

TABLE 2

Simulation FOPID Parameters

Fractional Order pid (Fopid) Parameters												
	System parameters			Controller Parameters					Time response parameters			
S.No	K	Tc	Td	Kp	Ki	Kd	Lamda	Mu	Overshoot	ISE	ettlingtim	Obj Fnc
1	0.4	0.2	0	11.9527	10.8575	11.4445	1.29	0.0001	1.27	12.64	1.31	16.62
2	0.4	0.4	0	11.8836	10.6603	11.5061	1.2187	0.0019	0.16	9.51	0.57	11.24
3	0.4	0.6	0	11.926	10.2917	10.6743	1.0884	0.4891	0.81	16.02	1.80	21.46
4	0.4	0.8	0	11.9599	11.845	10.1945	1.049	0.492	0.59	17.84	1.44	22.18
5	0.4	1	0	11.8212	10.6039	10.2683	1.0988	0.4476	0.53	15.43	1.30	19.34
Average Values of PID Parameters(AVFOTD)				11.9599	10.8575	18.5061	1.29	0.4891	1.2652	17.8419	1.4373	22.1774

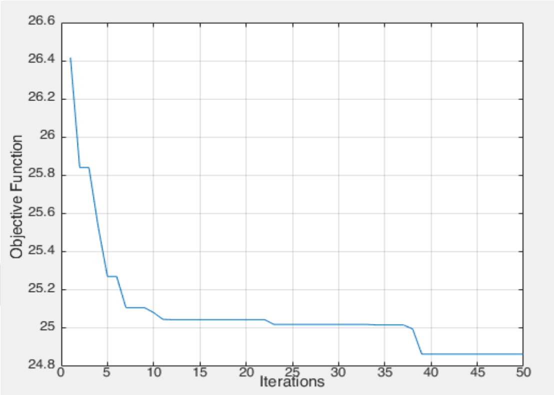

Figure 4 PSO objective Function vs Iterations

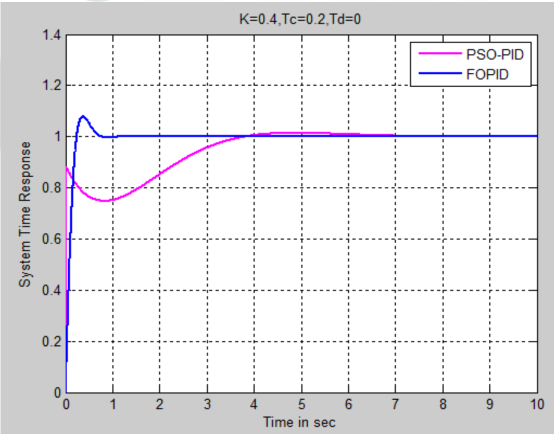

Figure 5 Case1 results (Where K = 0.4, Tc = 0.2, and Td=0)

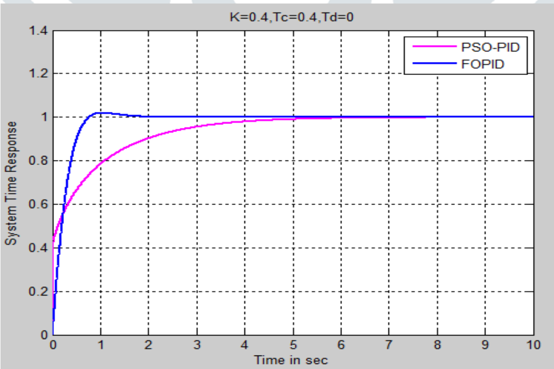

Figure 6 Case2 results (Where K = 0.4, Tc = 0.4, and Td=0)

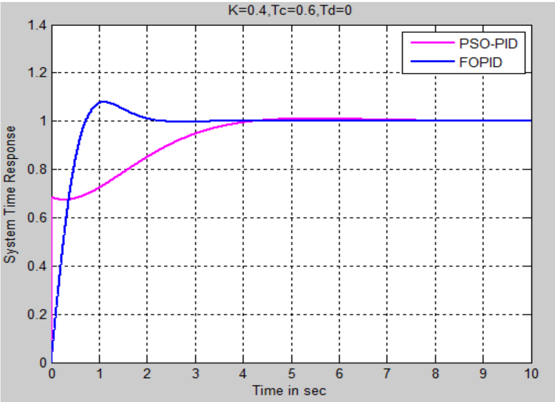

Figure 7 Case3 results (Where K = 0.4, Tc = 0.6, and Td=0)

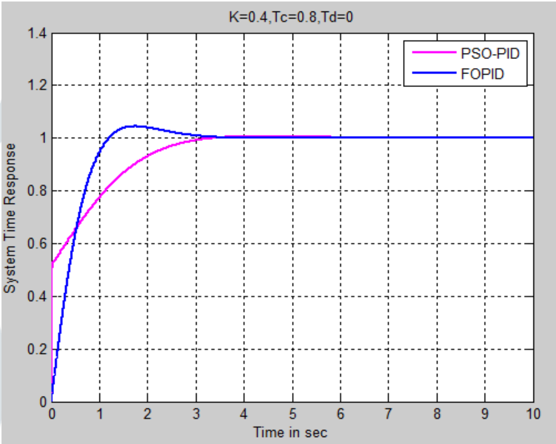

Figure 5 Case4 results (Where K = 0.4, Tc = 0.8, and Td=0)

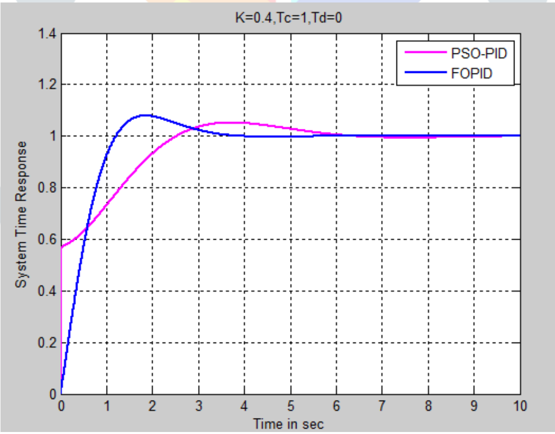

Figure 5 Case5 results (Where K = 0.4, Tc = 1, and Td=0)

The convergence of the objective function for 100 generation is presented in Fig 4. It shows that the objective function value using PSO- $FOPI^{\lambda}D^{\mu}$ is less than the value using PSO- $PI^{\lambda}D^{\mu}$. The optimum FOPID controller parameters specifications are shown in Table 1. The optimum PID controller parameters specifications are shown in Table 2.

VII Conclusion

In this paper FOPID controller parameters are tuned by Particle Swarm Optimization (PSO). In continuation to the previous work, the system response study with PID controller is also tuned using Particle Swarm Optimization. The results show that the time response of FOTD system is better than PSO tuned fractional order PID controller.. This work has been simulated in MATLAB/SIMULINK environment in combination with FOMCON tool box. The obtained results show that when fractional order PID controller is tuned with PSO algorithms,

REFERENCES

1. Ziegler JG, Nichols NB. Optimum setting for automatic controller. Transactions ASME 64. 1942.
2. Ho M-T, Data A, Bhattacharyya SP. Design of P, PI and PID controller for interval plants. Proc. American control conference; 1988; Philadelphia, PA, USA.

3. Pontryagin LS. One the zeros of some elementary transcendental function. American Mathematical Society Translation. 2. 1955.
4. Silva GJ. Data A, Bhattacharyya SP. New results on the synthesis of PID controllers. IEEE Trans Automat Contr. 2002; 47(2):c241–52.
5. Marteli G. Comment on new results on the synthesis of PID controller. IEEE Trans Automat Contr. 2005; 50(9):1468–9.
6. Ou L-L, Zhang WD, Gu D. Sets of stabilizing PID controllers for second-order integrating process with time- delay. IEEE Proc Contr Appl. 2006; 153(5):607–14.
7. Tan KK, Lee TH, Leu FM. Optimal Smith predictor design based on a GPC approach. Ind Eng Chem Res. 2002; 41:1242–48.
8. Tan KK, Hung SN, Lee TH. Development of a GPC-based PID controller design for unstable systems with dead time. ISA Trans. 2000; 39(1):55–70.
9. Podlubny I. Fractional-order systems and controllers. IEEE Trans Automat Contr. 1999; 44(1):208–14.
10. Domingues J, Valerio D, Da Costa JS. Rule-based fractional control of an irrigation canal. Proceedings 35th Annual Conference of IEEE Industrial Electronics IECON 09; 2009 Nov 3–5; Portugal. IEEE; 2009. p. 1712–7.
11. Ahn HS, Bhambhani V, Chen RQ. Fractional-order integral and derivative controller for temperature profile tracking. Sadhana, Indian Academy of Sciences. 2009; 34:833–50.
12. Petras I. Fractional-order feedback control of a Dc motor. J Electr Eng. 2009; 60:117–28.
13. Xue DY, Zhao CN, Chen YQ. Fractional order PID control of a DC motor with elastic shaft: A case study. American Control Conference, 2006; 2006 Jun 14–16; Minneapolis, MN.
14. Tehrani K, Amirahmadi A, Rafiei S, Griva G, Barrandon L, Hamzaoui M, Rasoanrivo L, Sargos F. Design of fractional-order PID controller for boost converter based on multi-objective optimization. 4th International Power Electronic and Motion Control Conference; 2010 Sep 6–8; Ohrid. IEEE; 2010. p. 179–85.
15. Cervera J, Banos A, Monje CA, Vinagre B. Tuning of fractional PID controllers by using QFT. 32nd Annual Conference on IEEE Industrial Electronics, IECON 2006; 2006 Nov 6–10; Paris. IEEE; 2006. p. 5402–7.
16. Bouafoura MK, Braiek NB. Controllers design for integer and fractional plants using piecewise orthogonal functions. Comm Nonlinear Sci Numer Simulat. 2010; 15(5):1267–78.
17. Padula F, Visili A. Tuning rules for optimal PID and fractional-order PID controllers. Journal of Process Control. 2011; 21:69–81.
18. Valerio D, Da Costa JS. Tuning of fractional PID controller with Ziegler-Nichols-type laws. Signal Processing. 2006; 86:2771–84.
19. Xue D, Meng L. Design of optimal fractional-order PID controller using multi-objective GA optimization. Control and Decision Conference (CCDC); 2009 Jun 17–19; Guilin. IEEE; 2009. p. 3849–53.
20. Sadati M, Zamani M, Karimi-Ghartemani N, Parniani M. Design of a fractional order PID controller for on AVR system using particle swarm optimization. Control Engineering Practice. 2009; 17(12):1380–7.
21. Chang FK, Lee CCh. Fractional-order PID controller optimization via improved electromagnetism–like algorithm. Expert Syst Appl. 2010; 37:8871–8.

M.Theja is currently pursuing her M.Tech (E.E.E) Control System domain Siddhartha Institute of Engineering & Technology, Puttur, Chittoor, Andhra Pradesh. She completed her graduation in Sri Venkateswara Engineering College for Womens,Tirupathi,Andhra Pradesh. Her areas of interests includes Control Systems, Renewable energy sources and Power electronic convertors.

Mr. N. Ramesh Raju was born in 1971. He received his M.Tech. from IIT Kharagpur in 1994. B.Tech. from Kaktiya University in 1993. At present he is working as Professor and Head EEE department at Siddharth Institute of Engineering & Technology, Puttur. He has 16 years of teaching experience and 4 years of industrial experience. Presently he is doing research on applications of Soft computing techniques in Control System.