IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Tuning the parameters of fractional order $PI^{\lambda}D^{\mu}$ using Particle Swarm Optimization

H S Singh[1*], M K Tiwari[2]

1. Department of Electrical and Electronics Engineering, LDC Institute of Technical Studies, Soraon Prayagraj, India-211025.

2. Department of Electrical and Electronics Engineering, LDC Institute of Technical Studies, Soraon Prayagraj, India-211025.

Abstract

In the process control, PID control methods are most commonly used. Fractional order $PI^{\lambda}D^{\mu}$ control is the generalization and development of integer order PID control. Compared to integer order PID controller, fractional $PI^{\lambda}D^{\mu}$ controller has more advantages such as less rising time, less overshoot and less settling time. As fractional order PID controller has five parameters, so they are more flexible and accurate than integer order PID controller. Fractional order PID controller tuned with analytical method i.e. tuning by Ziegler Nichols Method, Tuning by simplex method gets good result. A number of mathematical calculations are performed on analytical method, which makes calculation of fractional order PID controller parameter complex. Due to roughness of the objective function, we utilize derivation free optimization technique, particle swarm optimization for finding the five parameter of fractional order PID controller. These methods help in finding the actual value of the five parameters of fractional order $PI^{\lambda}D^{\mu}$ controller. Matlab/simulink simulation models are used for showing the various simulation results.

Keywords- *Integer order PID controller; Fractional order* $PI^{\lambda}D^{\mu}$ *controller; Particle swarm optimization; Classical optimization technique; Derivation free optimization technique, matlab/simulink.*

1.Introduction

Proportional-Integral-Derivative (PID) controllers are broadly being used for process control applications in industries. Small settling time for slow industrial processes, simplicity of design and good performance including low percentage overshoot, PID controllers are very suitable. For improvement in appropriate settings of fractional-I and fractional-D actions the performance of fractional PID controllers are very useful. The I- and D-actions being fractional have wider scope of design in a fractional PID controller. The optimal fractional PID controller is better than its integer counterpart. The extensive applications in real industrial processes, this proposed design will be very useful. In the current literature [1] – [9] on control engineering work of fractional PID controller are presented. A frequency domain approach based on the expected intersects frequency and phase margin is mentioned in [2]. In the complex plane, a method based on pole distribution of the characteristic equation was proposed in [5]. In the feedback poles, state space design method is very useful that can be viewed in [6]. The fractional controller can also be designed by cascading a proper fractional unit to an integer-order controller. The fractional-order systems model is complex and the fractional-order controller requires more tuning parameters than that of integer-order controller. Fractional order $PI^{\lambda}D^{\mu}$ controller has 5 degree of freedom while integer order PID controller has 3 degree of freedom, so fractional order $PI^{\lambda}D^{\mu}$ controller has more flexibility than the integer order PID controller. To find five parameter of fractional order $PI^{\lambda}D^{\mu}$ [K_p, K_i, K_d, λ, μ], is an optimal solution problem of five dimensional. Typical optimization technique cannot be used here due to roughness of the objective function. We use derivation free optimization technique and particle swarm optimization for finding the best setting of K_p, K_i, K_d, λ and μ. Proposed optimization technique for fractional order $PI^{\lambda}D^{\mu}$ controller produces better response as compare with the integer order PID controller for real world processes because most of the real world processes are fractional in nature.

2. Methodology

There are various methods used in the tuning of PID controllers. Here we use different PID controller, these are discuss below-

1.Integer order PID Controller

$$K_p + K_i s^{-1} + K_d s \qquad (1)$$

This (1) is the equation of transfer function of integer order PID controller.

Here the order of integration and differentiation is both Unities.

Block diagram of integral order PID controller is shown in Fig. (1)

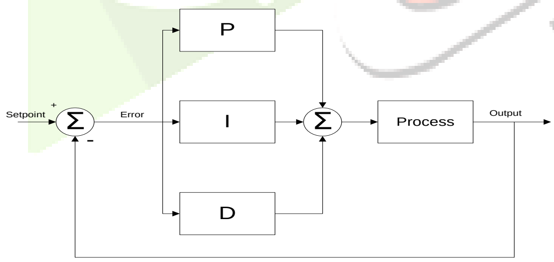

Fig. 1: Block Diagram of integer order PID controller

2.Fractional order $PI^{\lambda}D^{\mu}$ Controller

$$U(t) = K_p\, e(t) + K_i\, D_t^{-\lambda}\, e(t) + K_d\, D_t^{\mu}\, e(t) \qquad (2)$$

This (2) is equation of differential equation of fractional order $PI^{\lambda}D^{\mu}$.

$$G(s) = K_p + K_i\, s^{-\lambda} + K_d\, s^{\mu}$$

This (3) is equation of continuous transfer function of FOPID, is obtained through Laplace Transform which is given as above.

Block diagram of fractional order $PI^{\lambda}D^{\mu}$ controller is shown in the Fig. (2)

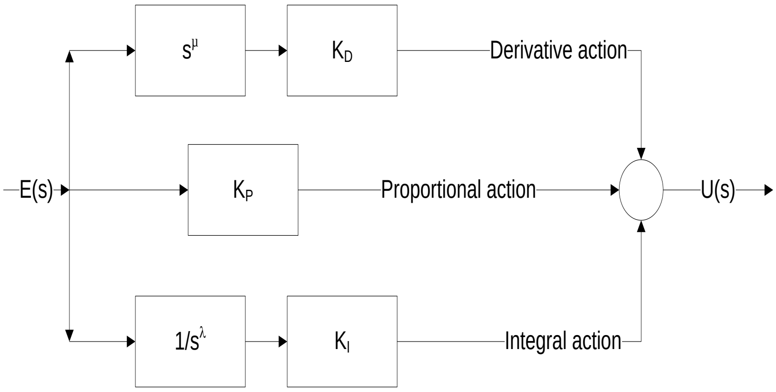

Fig. 2: Block Diagram of fractional order $PI^{\lambda}D^{\mu}$ controller

The order of integration and differentiation are respectively λ and μ (both positive real numbers, not necessarily integers). So we see that integer order PID controller has three parameters, while the fractional order $PI^{\lambda}D^{\mu}$ controller has five. The value of the λ and μ for different controllers, for integer order PID controller λ =1 & μ=1 , for PI controller λ =1 & μ=0 , for PD controller λ =0 & μ=1, for P controller λ =0 & μ= 0. So it is clear from the above that fractional order $PI^{\lambda}D^{\mu}$ controller generalizes the integer order PID controller and expands it from point to plane. Expanding graph of the fractional PID controller shown in the Fig. (3). this expansion adds more flexibility to controller design and we can control our real world processes more accurately.

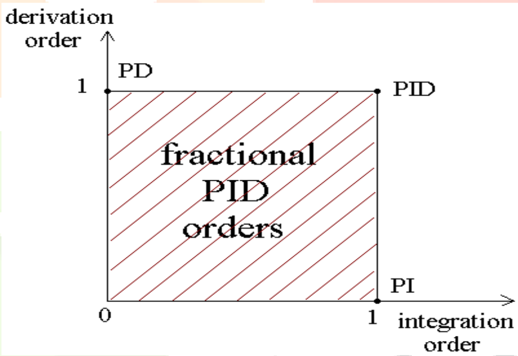

Fig.3: Expanding from Point to Plane

3.Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking, fish schooling or foraging of bacteria. Particle swarm optimization (PSO), is an evolutionary computation technique has become gradually popular to obtain global optimal solution in many areas. Particle swarm optimization has been widely regarded as promising optimization algorithm due to its combination of simplicity (in terms of implementation), low computational cost and good performance.

Starting from the last decade of the 20th century, the trend of biologically inspired computational approaches brought forth many heuristic optimization techniques. Among the most popular ones one can count Evolutionary Algorithm. Particle swarm optimization is a model free optimization method that belong to "blind – search" category. Blind search approach on the other hand, relies on stochastic method and can be performed without any knowledge or assumption about the optimization problem

4.Algorithms for Particle Swarm Optimization

The Particle swarm optimization algorithm attempts to mimic the natural process of group communication of individual knowledge, which occurs when a social swarm elements flock, migrate, forage etc in order to achieve some optimum property such as configuration or location. The 'swarm' is initialized with a population of random solutions. Each particle in the swarm is a different possible set of unknown parameters to be optimized. Representing a point in solution space, each particle adjusts its flying towards a potential area according to its own flying experience and shares social information among particles. The goal is to efficiently search the solution space by swarming the particles towards the best fitting solution encountered in the previous iterations with the intent of encountering better solutions through the course of the process and eventually conversing on a single minimum error solution.

In Particle swarm optimization each particle moves in a search space with a velocity according to its own previous best solution and its group previous best solution. The modified velocity and position of each particle can be calculated using the current velocity and distance as shown in the following formulas.

$$v_{id}(t+1) = wv_{id}(t) + c_1\, \emptyset_1\big(P_{id}(t) - x_{id}(t)\big) + c_2\, \emptyset_2\left(P_{gd}(t) - x_{gd}(t)\right) \ldots\ldots\ldots\ldots\ldots.(4)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad \ldots\ldots\ldots(5)$$

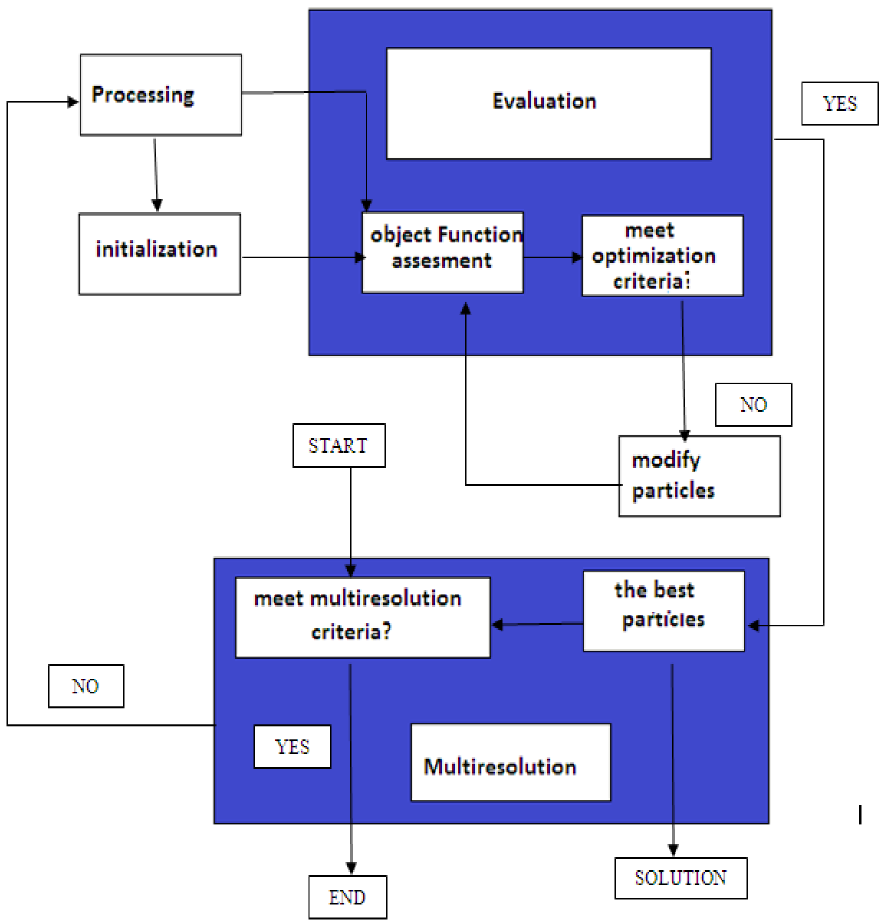

FIG 4. Block diagram of particle swarm optimization (PSO) algorithm

Where v and x denote velocity and position of each particle respectively, i, d denote number of particles and component respectively, t denotes steps of iterative computation; , c_1, c_2 denotes control parameters of the system, the value of c_1, and c_2 generally taken 2, and value of w (inertia factor) commonly ranges between 0.4 to 0.9; ϕ_1, ϕ_2 denote two uniform random numbers in the region [0,1], P_{id} denotes the previous best position of the particle, P_{gd} denotes the previous best position of the group. We define controller performance criteria with ISE (integral squire error) and then define objective function f which is inverse of performance criteria and minimize the 'f' using the particle swarm optimization. Our goal is to find out optimum set {K_p, K_i, K_d, λ, μ} for which f=0. The solution space is five dimensional, the five dimensions being K_p, K_i, K_d, λ and μ. So each particle has five dimensional positions and velocity vectors. The personal and global bests are also five dimensional. The limit on the position vector particles (i. e. the controller parameters) are set as practical assumptions, allow K_p to vary between 1 and 1000, K_i and K_d between 1 and 500, λ and μ between 0 and 2. Initializations of five variables are also done in the above mentioned ranges. After running the particle swarm optimization algorithms, the position vector of best particle i.e. optimized values of the five controller parameters are obtained. Block diagram of particle swarm optimization (PSO) algorithm shown in figure (4) and flowchart diagram of particle swarm optimization algorithms is shown in figure (5).

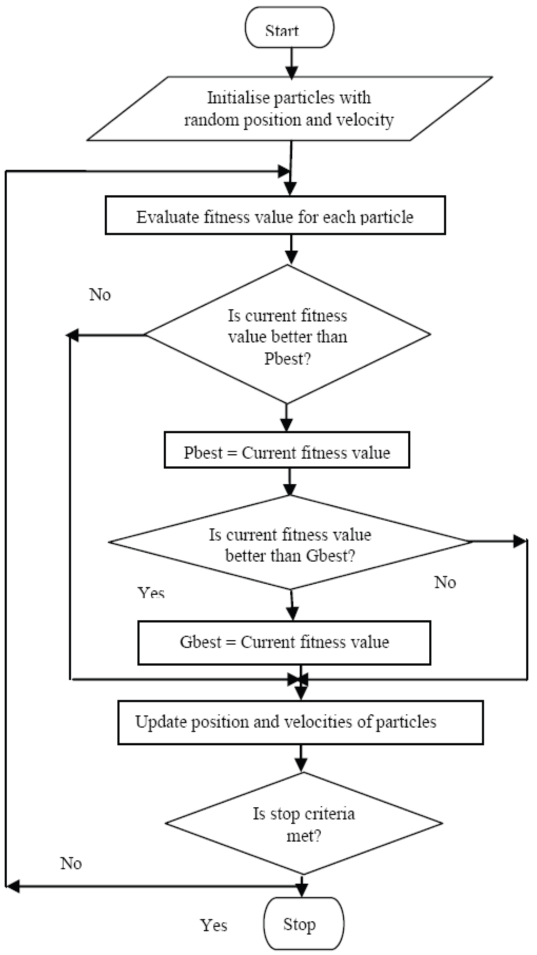

Fig. 5. Flow chart diagram of PSO algorithm

3. Result and Discussion

Different tuning methods of fractional order PID controller on some plants, their simulation results are discussed here. These simulation results are simulated using mat lab/simulink software simulation model.

Results of these different tuning methods are shown on the basis of peak overshoot, settling time and rise time.

3.1 Tuning of Some FOPDT systems-

FOPDT systems provide simple characterization of the process and give valuable information about dynamics of many applications in process control industry.

3.1.1. System 1 taken as for the tuning

Example-

$$G(s) = \frac{1}{1+s}\, e^{-0.1}$$

Which is a POPDT model with T=1 and L=0.1, since 0.1≤ T≤50 and L ≤0.5. Apply Ziegler – Nichols second set of tuning rule and from table five parameters of fractional PID controller are obtained as Kp= 1.2570, Ki=1.3105, Kd=-0.2588, λ= 1.1230, μ= 0.1537. So equation of controller becomes

$$G_c(s) = 1.2570 + 1.3105\frac{1}{s^{1.1230}} - 0.2588 s^{0.1537} \dots\dots\dots\dots\dots\dots\dots(3.2)$$

Step response by Z- N second set of tuning rule for system-1 is shown in the fig. 6.

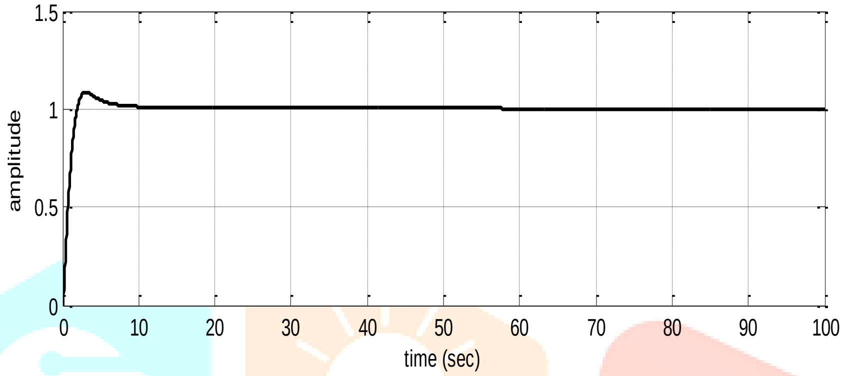

Fig.6: Step response by Z- N second set of tuning rule for system-1

Second tuning is Simplex method for function minimization and five parameters of fractional order PID controller recorded as K_p=1.2048, K_i=1.4244, K_d=-0.4320, λ= 1.0510, μ= 0.1678, simulation result are shows in the Fig 7.

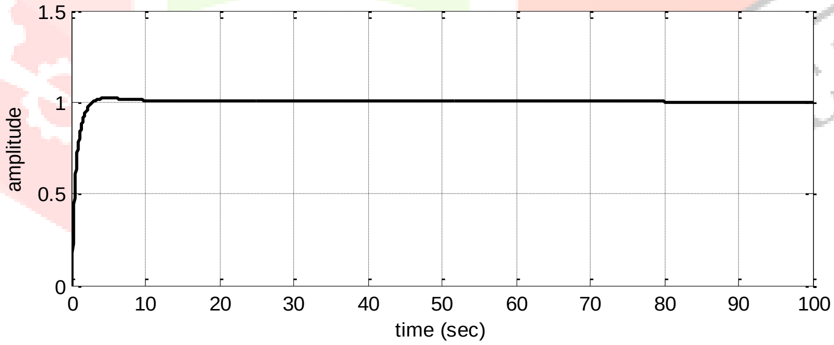

Fig.7: Step response by simplex minimization tuning rule for system 1

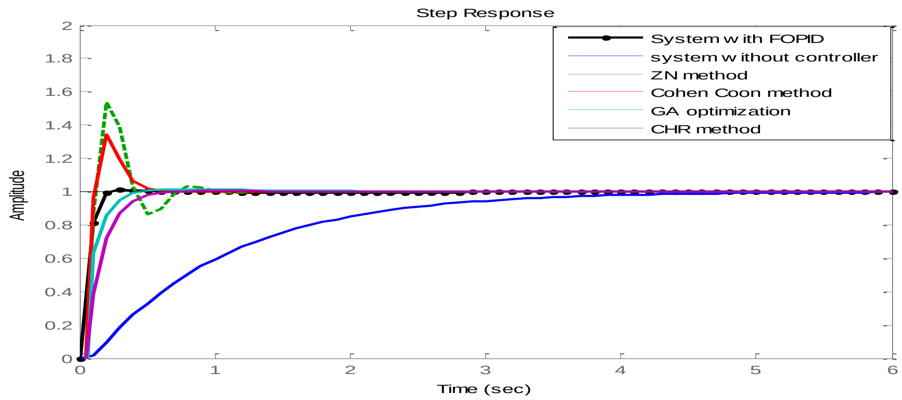

Fig.8: Comparison of Step responses by various methods for system 1

Comparison of various method are compared with PSO as shown in the figure above with the help of

Coding for plant 1 as shown in Fig.8.

Comparison of different simulation methods for system1 shown in the Table1;

Table 1

Tuning methods	Max. Peak overshoot (%)	Settling Time (sec)	Rise time (sec)
Z – N	09	08	02
Simplex minimization	2.1	5.5	02
PSO	0	0.3	0.2

This is clear from the comparison table1; the performance of the controller gives better, when Zeigler Nichols second set of tuning rule is compared with simplex minimization method and simplex minimization compared with particle swarm optimization technique.

3.1.2 System 2 taken for example is

$$G(s) = \frac{1}{1+1.5s}\, e$$

When value of five set parameter of fractional order PID controller is calculated by Ziegler – Nichols second set of tuning rules, when $0.1 \leq T \leq 50$ and $L \leq 0.5$. The value are K_p= 1.4098, K_i=1.6486, K_d= -0.2138, λ= 1.1016, μ= 0.1856. And the controller equation becomes as

$$G_c(s) = 1.4098 \quad \text{.................(3.4)}$$

Step response by Z- N second set of tuning rule for system 2 are shown in the Fig.9

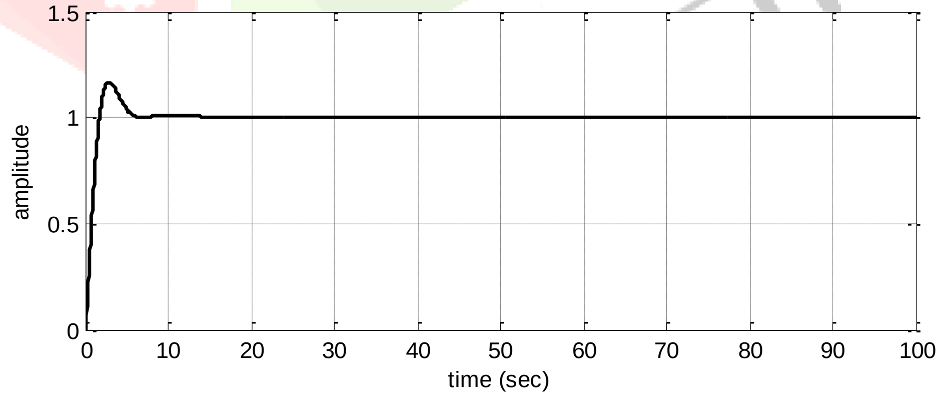

Fig 9. Step response by Z- N second set of tuning rule for system 2

Second tuning is Simplex method for function minimization and five parameters of fractional order PID controller recorded as K_p=1.2048, K_i=1.4244, K_d=-0.4320, λ= 1.0510, μ= 0.1678 Second tuning is done by Simplex method for function minimization and five parameters of fractional order PID controller obtained as K_p=1.2048, K_i=1.4244, K_d=-0.4320, λ= 1.0510, μ= 0.1678 ,simulation result are shown in Fig.10

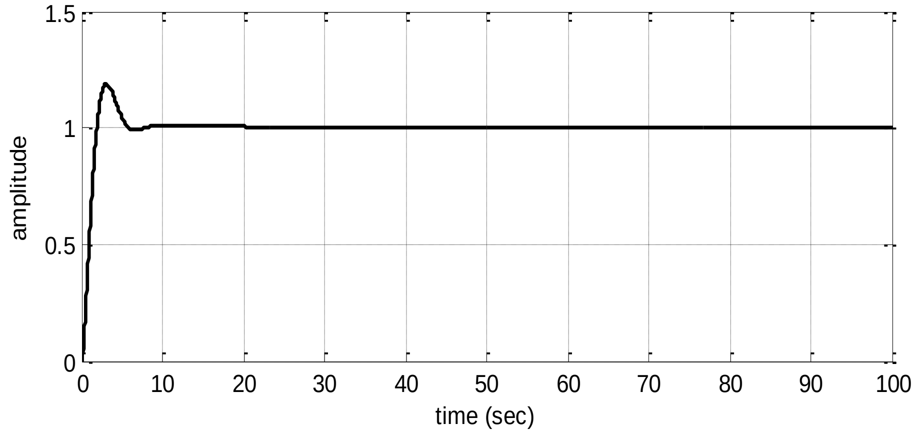

Fig10: Step response by simplex minimization tuning rule for system 2

Then the comparison of different tuning methods is compared with PSO as shown in the figure bellow with the help of coding for plant 2 as shown in figure (11).

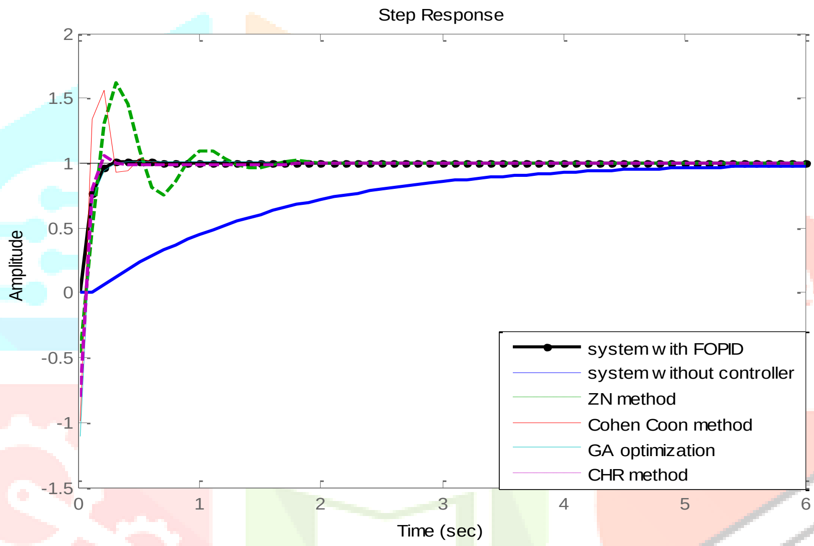

Fig.11 Comparison of Step responses by various methods for system2

Comparison of different simulation methods for system 2 is shown in Table 2

Table 2

Tuning methods	Max. Peak overshoot (%)	Settling Time (sec)	Rise Time (sec)
Z – N	17	06	2.1
Simplex minimization	18.5	5.5	2.1
PSO	0	0.4	0.3

The comparison table 2 shows that performance of the controller tuned by particle swarm optimization technique is better than the Z-N and Simplex minimization methods.

4. Conclusion & Future Scope of Work

In this paper few tuning methods such as Ziegler Nichols method, Tuning by simplex minimization method for function minimization and particle swarm optimization method are proposed for tuning of fractional order PID controller to better characterize the real dynamical system. As fractional order PID controller has five parameters, these methods help in finding the actual value of these parameters. Fractional order PID controller tuned with analytical method i.e. tuning by Ziegler Nichols Method and Tuning by simplex method both produces good results. A number of mathematical calculations are performed on analytical method, which makes calculation of fractional order PID controller parameter complex. Due to roughness of the objective function, we utilized derivation free optimization technique and particle swarm optimization (PSO) for finding the five parameters of fractional order PID controller.

For two plants, simulation results studied and compared by using mat lab/simulink software where it is found that particle swarm optimization technique gives best result than the other tuning methods. Results of these proposed optimization techniques will be tested on real run time industries application.

5.REFERENCES

[1] I. Podlubny, I. Petras and B. M. Vinagre, P. O' Leary, L. Dorcak, "Analogue realizations of fractional-order controllers," Nonlinear Dynamics, vol 29, pp. 281- 296, 2002.

[2] B. M. Vinagre, I. Podlubny, L. Dorcak and V. Feliu, "On fractional PID controllers: A frequency domain approach," Proc. Of IFAC Workshop on Digital Control – Past, Present and Future of PID Control, pp. 53-58, 2000.

[3] Schlegel Milos and Cech Martin, "The fractionalorder PID controller outperforms the classical one," 7th International Scientific-Technical Conference –PROCESS CONTROL 2006, June 13-16, 2006, Kouty nad Desnou, Czech Republic.

[4] Igor Podlubny, Ivo Petras, Blas M. Vinagre, YangQuan Chen, Paul O' Leary and Lubomir Dorcak, "Realization of fractional order controllers," Acta Montanistica Slovaca, vol 8, 2003.

[5] I. Petras, "The fractional order controllers: Methods for their synthesis and application," Journal of Electrical Enginnering, vol 50, no. 9-10, pp. 284-288, 1999.

[6] L. Dorcak, I. Petras, I. Kostial and J. Terpak, "Statespace controller design for the fractional-order regulated system," Proc. Of the International Carpathian Control Conference, pp. 15-20, 2001.

[7] I. Podlubny, "Fractional-order systems and $PI^{\lambda}D^{\delta}$ controllers," IEEE Trans. On Automatic Control, vol.44, no. 1, pp. 208-213, 1999.

[8] Ivo Petras, Lubomir Dorcak and Imrich Kostial, "Control quality enhancement by fractional order controllers," Acta Montanistica Slovaca, vol 3, no. 2, pp. 143-148, 1998.

[9] Ivo Petras and Blas M. Vinagre, "Practical application of digital fractional-order controller to temperature control," Acta Montanistica Slovaca, vol 7, no. 2, pp. 131-137, 2002