ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 11, May 2016

Design of Fractional Order PID Controller Based on Hybrid Bacterial For aging - Particle Swarm Optimization

Abdelelah Kidher Mahmood, Buraq Mahmood Abawi
Assistant Professor , PG. Dip. Student
College of Engineering University of Mosul

***ABSTRACT - Fractional order PID controller become more applicable in recent years although the difficulties in the design and realization. But the fractional calculus earlier provide special tools for computations and in the implementation of FOPID controller. In this work FOPID controller has been utilized using special tools which can be added to MATLAB/Simulation like FOMCON and NINTGER and the parameters of it tuned through hybridization between two intelligent optimization methods. The hybrid combination of bacteria foraging and particle swarm optimization has been performed and implemented on FOPID controller for a servomotor position dynamics that minimizing ISE directly on line. Simulation results show that the FOPID controller performs better time domain performance with less value of ISE than the conventional PID controller, besides the system with FOPID controller more flexible and robustness than with conventional PID controller*.**

***Index Terms*— fractional calculus, fractional order controllers, particle swarm optimization, Bacteria Foraging.**

I. INTRODUCTION

The PID controllers have remained, by so far; the most commonly used in practical and all industrial feedback control applications. The main reason is due its relatively simple structure, which can be simply understood and implemented in practice. They are thus, more acceptable than other controllers in practical applications unless evidence shows that they are insufficient to fulfill some of the specifications. Many techniques have been suggested for their parameters tuning. Although all the existing techniques for the PID controller parameter tuning perform well and acceptable performance, a continuous and an intensive research work is still underway towards system quality control enhancement and performance improvements.

On the other hand, in recent years, it is remarkable to note the increasing number of studies related with the application of fractional controllers in many areas of science and engineering. Really this is due to a better understanding of the fractional calculus and realization. In the range of automatic control, the fractional order controllers where are the generalization to conventional integer order controllers that would lead to more precise and robust control performances. So the researchers proposed a generalization of the PID controller as $PI^{\lambda}D^{\mu}$ controller which is known as fractional order PID (FOPID) controller, where the integral and derivative orders are usually fractional. In FOPID besides Kp, Ki, Kd there are two more parameters λ and called the fractional integral and derivative orders respectively. In case If $\lambda = 1$ and $\mu = 1$, then it becomes integer PID. If λ and μ are in fractions then it becomes fractional order PID.Tuning five parameters where are Kp, Ki ,Kd, λ and μfor $PI^{\lambda}D^{\mu}$is a great task , therefore an optimization technique is required to deduce that parameters such that an optimal objective function reached [1][2]. The improvement in time domain performance and the robustness of the system must be increased. More published work on this topics and especially in earlier years, were in 2012, researchers (ShivajiKarad) and (Dr.S. Chatterji) and (PrasheelSuryawanshi) used conventional PID and FOPID controller, where they have been applied on bioreactor using simulation's compare of the results they found that the fractional order FOPID controller better than the conventional PID controller[1].In 2012, (D.S. Karanjkar) and (S.Chatterji) And (P.R. Venkateswaran) searched toward the study of FOPID controller, and how to adjust parameters for this type of controllers using (Ninteger toolbox with Matlab / Simulink) and utilizing the different methods of connection (series , parallel) and comparison of the results given .The application of fractional calculus in design a FOPID controller offers performance exceeds the design of the conventional controller given in [3].In 2012 (Mazin Z. Othman) and (Emad A. Al-Sabawi) designed fractional order FOPID controller where tuning parameters of controller has been utilized by a Genetic Algorithm(GA) optimization technique. Where the tuning processed one in according to principle of model reference adaptive control .From their work they explained the FOPID controller gives more freedom in design that increase accuracy in tracking dynamic system[4].In 2013 the researchers (AnguluriRajasekhar) and (Shantana Das) and (Ajith Abraham) designed FOPID controller for speed regulator in a DC motor drive, where they determine parameter of controller using Artificial Bee Colony (ABC) as another tool of the optimization. They proved that the FOPID controller more effective and flexible than PID [5] .In 2014, the researchers (Abdelelah K. Mahmood) and (Bassam F. Mohammed) designed and realized the fractional order FOPID controller via using one of the Intelligent Optimization Method which is the Particle Swarm Optimization (POS) by minimizing the objective function which represented as the minimum integral of the absolute value of the time error signal (ITAE). The controller has been converted to digital form using a special approximation method as a continuous fraction expansion using MATLAB, and using C language on PIC microcontroller for DC motor as a position control. They shown similarity in the performance of closed loop system for both continuous and discrete [6].

II. FRACTIONAL ORDER SYSTEM

Fractional calculus is have a fundamental operator $_aD^r_t$, where a andt are the limits of the operation and r ∈ R. The continuous integro-differential operator is defined as:

$$aD_t^r = \begin{cases} \frac{d^r}{dt^r} & : r > 0 \\ 1 & : r = 0 \\ \int_a^t (d\tau)^{-r} & : r < 0 \end{cases} \quad)1 (\ldots$$

The three equivalent definitions most frequently used for the general fractional differ-integral are the Grünwald-Letnikov (GL) definition, the Riemann-Liouville (RL) and the Caputo definition. The GL definition is given by:

$$aD_t^r f(t) = \lim_{h \to 0} h^{-r} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} (-1)^j \binom{r}{j} f(t - jh) \ldots . (2)$$

where [.] means the integer part

The RL definition is given as :

$$aD_t^r f(t) = \frac{1}{\Gamma(n-r)} \frac{d^r}{dt^r} \int_a^t \frac{f(\tau)}{(t-\tau)^{r-n+1}} d\tau \ldots \ldots (3)$$

The Caputo definition can be written as:

$$aD_t^r f(t) = \frac{1}{\Gamma(r-n)} \int_a^t \frac{f(\tau)^{(n)}}{(t-\tau)^{r-n+1}} d\tau .. (4)$$

The initial conditions for the fractional order differential equations with the Caputo derivatives are in the same form as for the integer-order differential equations. In the above definition, Γ (m) is the factorial function, defined for positive real (m), by the following expression**:**

$$\Gamma(m) = \int_0^\infty e^{-u} u^{m-1} du \qquad \ldots \ldots (5)$$

$$\Gamma(m+1) = m! \ldots \ldots \ldots \ldots \ldots (6)$$

Laplace transform of non integer order derivatives is necessary for an optimal study. Fortunately, not very big differences can be found with respect to the classical case, confirming the utility of this mathematical tool even for fractional systems. Inverse Laplace transformation is also useful for time domain representation of systems for which only the frequency response is known. The most general formula is the following:

$$\mathcal{L}\left\{\frac{d^m f(t)}{dt^m}\right\} = S^m \mathcal{L}\{f(t)\} - \sum_{k=0}^{n-1} S^k \left[\frac{d^{m-1-k}}{dt^{m-1-k}}\right]_{t=0} .. (7)$$

Where n is an integer such that $n - 1 < m < n$.

The above expression becomes very simple if all the derivatives are zero:

$$\mathcal{L}\left\{\frac{d^m f(t)}{dt^m}\right\} = S^m \mathcal{L}\{f(t)\} \ldots \ldots \ldots (8)$$

The feedback control loop of a fractional order system with a fractional controller is similar to the integer order feedback control loop. [7][8].

III. BASIC IDEA OF FRACTIONAL ORDER PID CONTROLLER

Proportional Integral Derivative PID controllers belong to wide spread industrial controllers and therefore they are topics of wide effort for improvements of control system performance. One of the possibilities to improve PID controllers is to use fractional-order controllers with non-integer derivation and integration parts.

A $PI^\lambda D^\mu$controller needs tuning five parameter which gives greater freedom in tuning and more flexibility that enhance the dynamical properties of the control system.

The differential equation of fractional order controller described as[9] :

$$u(t) = Kpe(t) + KiD^{-\lambda}e(t) + KdD^{\mu}e(t) \ldots . (9)$$

Where e(t) is the error between a measured process output variable and a desired set point u(t) is the control output

Applying Laplace Transform yield transfer function of FOPID given as :

$$G_C(S) = \frac{U(S)}{E(S)} = Kp + Ki\, S^{-\lambda} + Kd\, S^{\mu}, (\lambda , \mu > 0) \ldots . (10)$$

Where

E(S) : Error Signal

U(S) : Controller Output Signal

Kp: proportional constant

Ki: integral constant

Kd : derivative constant

From Eq. (10) in case of conventional (PID) controller that to choose the values of (μ, λ) = 1 and fractional order (FOPID) become general case of (PID) controller where, (λ , μ > 0) or the orders become non integer.

IV. PARTICLE SWARM OPTIMIZATION (PSO)

The aim of using Particle Swarm Optimization algorithm is to evaluate optimal solution by simulation foraging behavior of some flocks of birds and fish. The Particle Swarm Optimization algorithm is consisted of a collection of particles each particle has a position at time and velocity. Each particle influenced by their own local best position and the global best position in which known to swarm or a close neighbor. Each iteration a particle's velocity is evaluated using:

$$v_i(t+1) = wv_i(t) + c_1 * rand() * (p_{lbest} - x_i(t)) + c_2 * rand() * (p_{gbest} - x_i(t)) \ldots . (11)$$

Where $v_i(t + 1)$ is the new velocity for the *i th* particle

c1and c2 are the weighting coefficients for the local best and global best positions respectively.

$x_i(t)$ is the *it* particle's position at time t

p_{lbest} is the *ith* particle's local best position

p_{gbest} is the global best position known to the swarm

rand ()is random variable ∈[0, 1].

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 11, May 2016

The new position is then update by the sum of the previous position and new velocity that will find best position within a particles local neighborhood at time t.:[10]

$$x_i(t+1) = x_i(t) + v_i(t+1) \ldots\ldots\ldots (12)$$

V. BACTERIAL FORAGING OPTIMIZATION ALGORITHM (BFOA)

The E-coli principle in biological behavior of bacterial foraging has been utilized for constructer of bacterial foraging optimization. The main idea of the BFOA is simulated motion the bacteria toward searching for a food and try to avoid noxious substances. The social bacteria motion modeled based on chemo taxis, swarming, reproduction, and elimination- dispersal. The goal of using BFOA ability of the algorithm to find new solution [10]. In this optimization method there are four typical behaviors that imitate nature [11]:

A- *Chemotaxis*: In this step theE-coli movement by swimming and tumbling through flagella. Verified by mathematical expression :

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \ldots\ldots (13)$$

Where Δ indicates a vector in the random direction whose elements lie in [-1, 1].

B- *Swarming*: the swarm process bacteria accomplished of through bacterium signaling to the other bacteria in order to swarming together to reach the desired location.

C- *Reproduction*: In this step two main processes occurred like die of non healthy and asexually split of healthy bacterium that in final step this keeps the swarm size constant.

D- *Elimination and Dispersal*: This step take into account a sudden disturbance in the environment, where due to it some a bacterium killed or a group is dispersed into a new place.

VI. OPTIMAL HYBRID BACTERIAL FORAGING-PARTICLE SWARMALGORITHM (BF-PS)

The latest two intelligent optimization are bacterial foraging and particle swarm optimization. Each has such properties like population. In order to enhance bacterial foraging algorithm we have done hybridization between two methods to ensure access to reach optimal value by the best velocity. The benefit of particle swarm optimization algorithm in this method is the ability to exchange social information while bacteria foraging has ability in finding new solution by chemo taxis.. In the hybrid algorithm the feature of this hybrid algorithm is that the unit length random direction of tumble behavior can be expressed by the global best position and the local best position of each bacterium. In the chemo taxis loop the tumble direction is evaluated by:

$$v_i(t+1) = wv_i(t) + c_1 * rand * (pbest - pcurrent)$$
$$+ c_2 * rand * (gbest - pcurrent) \ldots\ldots (14)$$

Where

pbest- is the local best position of each bacterium

gbest- is the global best bacterium

The scenario of the steps for BF-PSO performance it as software given below:

[**Step 1**] Initialize the parameters P ,s ,N_s ,N_c ,N_{re},N_{ed},p_{ed},C(i)(i=1,2,3,………,s) ,θ^i, w, c1, c2, R1, R2.

Where

P –Dimension of the search space.

s –Number of bacteria in the population.

Ns –Swimming length after which tumbling of Bacteria will be undertaken in chemotactic loop.

N_c –The number of iterations to be undertaken in chemotactic loop, always N_c>N_s.

N_{re} –Maximum no. of reproduction steps.

N_{ed} –the maximum no. of Elimination and dispersal events to be imposed over Bacteria.

P_{ed} –Probability with which elimination and dispersal will continue.

θ^i–Position of the *ith*(i= 1,2, …….,s) bacterium.

C(i) –Step size of the *ith* bacterium taken in random direction, specified by tumble

w: PSO parameters.

C1,C2: PSO random parameter.

R1,R2 : PSO random parameter.

[**Step 2**] Elimination and dispersal loop: l = l+1

[**Step 3**] Reproduction loop: k = k+1

[**Step 4**]Chemotaxis loop: j = j+1

[**Substep 4.1**] For i= 1,2,3,….,S take a chemotactic step for bacterium I as follows

[**substep4.2**] Evaluate the cost function (i, j, k, l).

[Substep 4.3]J_{last}= J(i,j,k, l) store This value best cost function in J_{last} since the program may find a better value less than it is found in another tumble and swim

[**Substep 4.4**] The best cost for each bacterium will be selected to be the local bestJ_{local}

J_{local}= J_{last}(i,j,k,l)

[**substep4.5**] Tumble: Let

[**Substep 4.6**] update position and cost function

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$$

[**Substep 4.7**] Evaluate the cost function J(i,j+1,k,l)

[Substep 4.8] Swim:

(i) Let m = 0 (initial counter for swim length)

(ii) while m <N_s (if the bacteria have not climbed too long)

- Let m = m+1
- If J(i, j+1,k, l) <J_{last} (if doing better)
- Let J_{last}= J(i, j+1,k, l)
- Update position and cost function

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$$

use this θ^i(j+1,k, l) to compute new cost function J(i,j+1,k, l)

- Evaluate the current position and local cost for each bacteria

$p_{current}(i,j+1,k,l) = \theta^i(i,j+1,k,l)$
$J_{local}(i,j+1,k,l) = J_{last}\ (i,j+1,k,l)$
Else
$J_{local}(i,j+1,k,l) = J_{last}\ (i,j+1,k,l)$
$p_{current}(i,j+1,k,l) = \theta^i(i,j+1,k,l)$

- let m = N_s. This is the end of while statement

[**Substep 4.9**] go to next bacterium
[**Substep 4.10**] evaluate the local best position (pbest) for each bacteria and global best position (gbest).
[**Substep 4.11**] evaluate the new direction for each bacterium

$$v_i(t+1) = wv_i(t) + c_1 * rand * (pbest - pcurrent) + c_2 * rand * (gbest - pcurrent)$$

[**Substep 4.12**]
Delta = v_{i+1}
[**Step 5**] If j <N_c, go to [Step 4]. In this case, continue chemotactic
[**Step 6**] Reproduction:
[**Substep 6.1**] For the given k and l and for each i= 1,2,……,S let

$$J^i_{health} = \sum_{j=1}^{Nc+1} J(i,j,k,l)$$

The health of the bacterium i (a measure of how many nutrients it got over its lifetime and how successful it was at avoiding noxious substances). Sort bacteria so as to ascending cost J_{health} (higher cost mean slower health).
[**Substep 6.2**] The S_r= S/2 bacteria with the highest J_{health} values die and this process is performed by the copies that are made are placed at same location as their parent.
[**Step 7**] If k <N_{re}, go to the [Step 3]. Since in this case the specified reproduction steps are not reached, start the next Generation of the chemotactic loop.
[**Step 8**] with the probability p_{ed}, elimination-dispersal, For i= 1,2,….,S, each bacterium, which results in keeping number of bacteria in the population constant. To do this, if a bacterium is eliminated, simply disperse one to a random location on the optimization domain.
[**Step 9**] If l <N_{ed} then go to [Step 2], otherwise end.

VII. SIMULATION OF POSITION CONTROL SYSTEM WITH AUTO TUNED FOPID CONTROLLER PARAMETER VIA PSO-BF OPTIMIZATION

The position control system with unity feedback control system simulated by MTLAB simulation which shown in fig.1. Where FOPID controller (Gc(s)) implemented by using FOMCON toolbox, the integral of square error (ISE) is the cost function, and Gp(s) is physical servomotor position transfer function

$$Gc(s) = Kp + \frac{Ki}{S^{\lambda}} + Kd\,s^{\mu} \quad \ldots\ldots\ldots(15)$$

$$Gp(s) = \frac{0.6461}{0.013s^2 + s} \ldots\ldots\ldots\ldots\ldots\ldots(16)$$

$$ISE = \int_0^{\infty} e^2(t)\,dt \ldots\ldots\ldots\ldots\ldots\ldots(17)$$

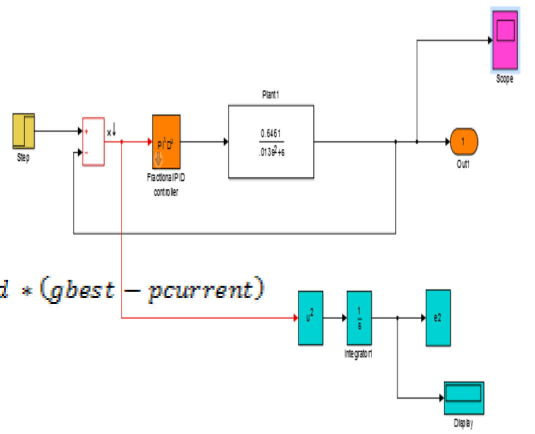

Fig 1

The result of the tuned parameter's and ISE given in Table 1. For FPID while Table 2 is for conventional PID

Table (1): Parameters of FOPID Controller Obtained by BF-PSO algorithm

Parameter of FOPID controller					Objective function
Kp	**Ki**	**Kd**	**λ**	**μ**	**ISE**
31.1438	**0.2312**	**-0.0344**	**0.2708**	**0.1861**	**0.03134**

Table (2): Parameters of FOPID Controller Obtained by BF-PSO algorithm

Parameter of PID controller			Objective function
Kp	**Ki**	**Kd**	**ISE**
25.6536	**-0.0328**	**0.2812**	**0.04116**

The step response for both PID and FOPID shown in Fig.2.The concluded time domain specifications for both PID and FPID given in table 3 with no overshoot.

Table (3):Step Response Specification of PID & FOPID Controllers Obtained by BF-PSO algorithm.

Controller	Tr (sec)	Ts (sec)
PID	**$135.72*10^{-3}$**	**$180*10^{-3}$**
FOPID	**$74.94*10^{-3}$**	**$124*10^{-3}$**

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 11, May 2016

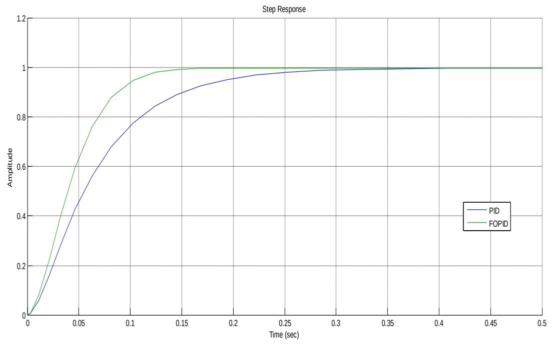

Fig.2

In Fig.3 a step response shown the closed loop system with FOPID and initial parameters value before BF-PSO implementation and with final values of the tuned parameters with PSO-BF

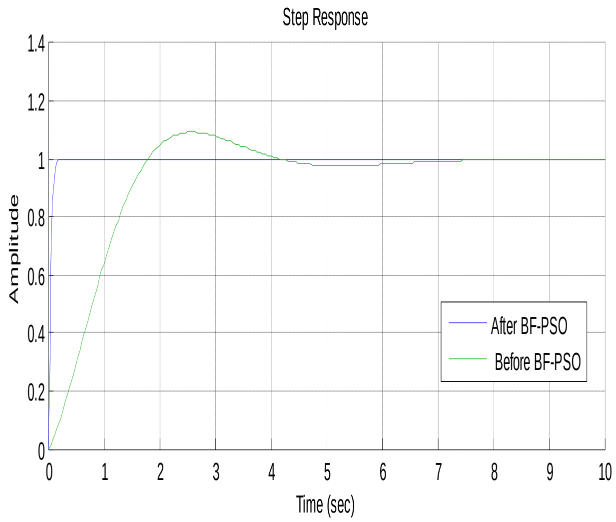

Fig. 3

VIII. DISTURBANCE EFFECT ON THE SYSTEM

In order to test the robustness of the system a disturbance has been applied to the output of the system of both controllers. The application of the disturbance was at 5 second with value 0.3 shown in fig.4. The response for system FOPID controller with disturbance shown inFig.5 and response for System PID controller with disturbance shown inFig.6in which the FOPID controller has rejected of the disturbance with fast time. In The Table (4) shown the time duration has been give for both controller.

The Table (4) shown the time duration for performance of System FOPID Controller and PID controller with Disturbance

Controller	Time (msec)
FOPID	227
PID	744

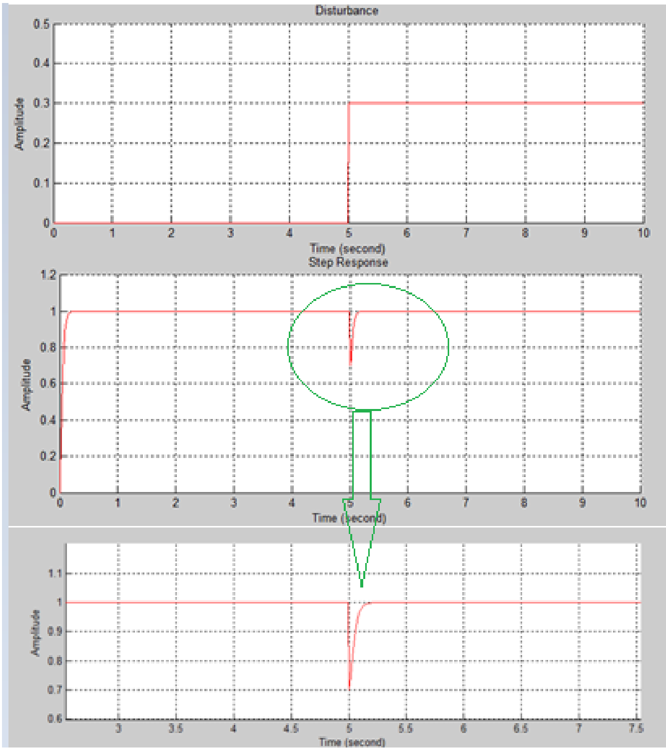

Fig 4

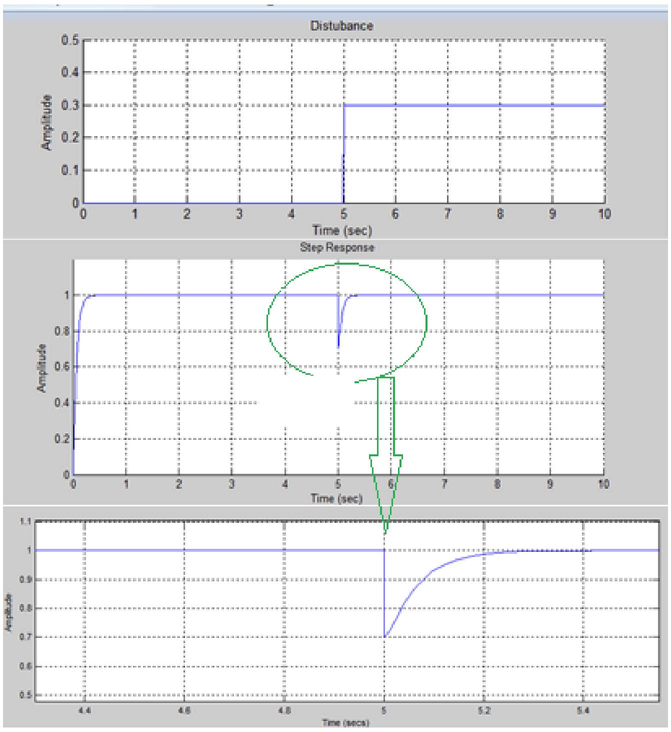

Fig. 5

IX. CONCLUSION

In this paper, hybridization has been done for both PSO and BF techniques that minimizing special objective function. The hybrid BF-PSO optimization algorithm has been implemented for tuning a both PID and FOPID controller parameters which minimizing ISE object function .The

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 11, May 2016

BF-PSO optimization has been done online through application test signal in the input and performing the optimization in sub m file and the results of computation has been fed to controller GUI directly in the end of computation program. A comparison of step response for the system with FOPID controller and with the conventional PID controller showed that the step response at FOPID better than conventional PID, and the system with FOPID controller increase robustness of the system which has fast rejection of the disturbance than with PID controller.

REFERENCES

[1] S. Karad, S. Chatterji and P. Suryawanshi,"Performance Analysis of Fractional Order PID Controller with the Conventional PID Controller for Bioreactor Control," International Journal of Scientific Engineering Research,vol. 3,issue6, pp. 1- 6, 2012.

[2] Ammar A. Aldair, Weiji j. Wang," Design of Fraction Order Controller Based on Evolutionary Algorithm for a Full Vehicle Nonlinear Active Suspension System ", International Journal of Control, Autmoation,andSystem,Vol. 3,no. 4, December 2010.

[3] D. S. Karanjkar, S. Chatterji and P.R. Venkateswaran, "Trends in Fractional Order Controllers," International Journal of Emerging Technology and Advanced Engineering, vol. 2, issue 3, pp. 383-389, 2012.

[4] Mazin Z. Othman , Emad A. Al-Sabawi", Design of Fractional Order PID Controller Based on Genetic Algorithms" , Al-Rafadain Engineering Journal,Vol. 20 Issue 4, p11-20,2012.

[5] A. Rajasekhar, S. Das, A. Abraham, "Fraction Order PID Controller Design for speed Control of Chopper Fed DC motor Drive using Artificial Bee Colony Algorithm",IEEE,pp. 259-266,2013.

[6] Abdelelah K. Mahmood, Bassam F. Mohammed, "Digital Fractional Order PID Controller Design And Realization", Al-Rafidain Engineering, Vol.22,No.4,pp.57-64, May 2014.

[7] R. Caponetto, G. Dongola, L. Forturna, I. Petráš, "Fractional Order System: Modeling and Control Application", Singapore, Word Scientific Series on Nonlinear Science, 2010.

[8] C. A. Monje, Y. Q. Chen, B. M. Vinagre, D. Xue and V. Feliu, "Fractional-order Systems and Controls: Fundamentals and Applications", London, Springer-Verlag, 2010.

[9] D. Xue, C. Zhao and Y. Q. Chen, "Fractional Order PID Control of a DC- Motor with Elastic Shaft: a Case Study", IEEE, pp.3182-3187, 2006.

[10] Jason Brownlee," Clever Algorithms Nature-Inspired Programming Recipes", LuLu., January 2011.

[11] S. Das, A. Biswas, S. Dasgupta, A. Abraham," Foundations of computational Intelligence ",Sprinnger-Verlag Berlin Heidelberg, Volume 3,2009.

AUTHOR BIOGRAPHY

Abdelelah Kidher Mahmood: born On July-1955, Kirkuk-IRAQ, he received Ph.D. in Control Engineering from Saint Peters burg. University Russia in 1992. M.Sc., Pg. Dip, B.Sc. in Electronic and Communication, Department of ElectricalEngineering,University of Mosul, IRAQ, 1981, 1979, and1978, respectively. Assistance Professor from 2008, Lecturer1993-2008, Assistance Lecturer 1984-1993. His research interest includes Control Engineering, Fuzzy Logic, Intelligent Techniques, and Real Time Digital Control, Fractional order controller BF, PSO Optimization.

Buraq Mahmood Abawi: born on June -1981, Mosul-IRAQ, he received B.Sc. in Electrical Engineering from University of Mosul, IRAQ in 2005, Pg. Dip. Student in power & machine, Department of Electrical Engineering, University of Mosul, IRAQ.