International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 11, November - 2013

Evaluations of Optimum Value of PID Controller Gains Using Hybrid Bacterial Swarm Optimization

Mohammad Faisal Hashim[1], Mohd. Sanawer Alam[2]

Electronics Instrumentation & Control Engineering, Azad Institute of Engineering & Technology, Lucknow-226002, India

Abstract

In the control system problems it is challenging to find the PID parameters in the initial stage, and there fine tuning during the system run condition. In this paper we have test the application of hybrid bacterial foraging and particle swarm optimization algorithm named as bacterial swarm optimization (BSO) based PID parameter tuning of close loop controller. The objective is dependent on globally minimal error squared error integral criteria of the step response of second order and higher order plants cascaded with PID controller by our proposed method. In this algorithm parameters are found by evolutionary methods with consideration of the globally optimal solution for control applications. The Kp, Ki and Kd gains are calculated by the PSO and (BSO) methods for plants with all the poles of the transfer function located in the left half of the s-plane. The performance of both algorithms is analyzed on transfer functions of a second order and higher order plants.

***Keywords**: bacterial foraging, chemotxis, close loop controller, PID parameters, PSO, optimization.*

1. Introduction

The PID controller contributes to satisfactory operation of the system by upholding the parameters within the acceptable limits and maintains balance the effects due to variations in response. They are applied in most systems because of their ability to adapt with system dynamics. The PID parameters are properly tuned to get secure dynamic performance and sustainable utilization of system resources. The conventional of tuning K_P, K_I and K_D parameters are Ziegler-Nichols, simplex method, orthogonal test, Newton method, quadratic method etc. These methods need complete information of the system behavior and other pre-requisite knowledge. The conventional methods are not suitable in giving better results and do not provide suitable control during changes in operating conditions.

In all control domain advancements are including the use of intelligent computations based on Neural Network (NN), Fuzzy Logic (FL) and Evolutionary Algorithms (EA) in association with the design of efficient controllers. The evolutionary search algorithms have flexibility and adaptability to the allotted task and they have robustness along with the global search characteristics [6]. The article is based on intelligent computing techniques for optimum tuning of PID parameters for improving the controller efficiency. The optimized values are used in the PID controller in control loop second and higher order systems. EA provides excellent results for complex optimization problems in comparison to conventional methods [1].

2. Bacterial Foraging Optimization Algorithm

According to evolution theory only those species survives which have better fitness. Fitness in a general case involves food searching and motile behavior hence those species have higher probability of survival which can show better capability of food searching. The qualities of stronger species are transferred in the next offspring's because stronger species has reproduction quality of better offspring's. By designing a model of food searching strategy of the species, [3] we develops the optimization techniques for the nonlinear system. The Bacteria Foraging Optimization is also a similar algorithm based on food searching behavior of E. coli bacteria. It is divided into following parts:

1. Chemotaxis
2. Swarming
3. Reproduction
4. Elimination dispersal

Bacterial movement for searching of food consists of two modes swimming and tumbling. Both modes on combining together called as chemotaxis. The swimming related movements indicate a movement in a predefined direction, and if the movement of bacteria is in a different direction then the predefined one it is taken as tumbling.

The process of swarming can be described by the behavior of bacteria that congregates them into a group such that they always perform movements in a high bacterial density. The fittest bacterium attracts other bacteria to converge all of them at a desired location. For this purpose penalty function based on distances of bacterium from the fittest one is added with cost function such that the penalty function should reaches to zero on achieving desired optimum location. During the process of reproduction bacteria are divided into two groups. The healthier half are retained and other part of remaining half are eliminated.

The last stage of elimination and dispersal represents the occurrence of a sudden unforeseen incidence which can bring changes in the evolution process. Such kind of unforeseen causes are generally occurrence of unknown events that may replace a set of bacteria to a location nearer to the food location. This process is very helpful in minimizing the behavior of stagnation mostly found parallel search algorithms. [3-5].

3. Particle Swarm Optimization (PSO)

The PSO model [2, 4] consists of a swarm of particles, initialized with a random population of candidate solutions moving iteratively in a problem space in exploring new solutions. Particle position are represented using a position-vector X^{ik} where (i means i^{th} particle), and a velocity-vector V^{ik}. Each particle has best position $P^{i\ Lbest}$ .The best position vector of complete swarm is stored as P^{i}_{Global} vector. In each iteration k, the velocity is updated using the previous velocity. The new velocity is determined by:

$$C_1R_1\,(P^{i}_{Lbest} - X^{i}_{k}) + C_2R_2 + (P^{i}_{Global} - X^{i}_{k}) \qquad (1)$$

The new position is calculated as from previous position and the new velocity given as:

$$X^{i}_{k+1} = X^{i}_{k} + V^{i}_{k+1} \qquad (2)$$

R1 and R2: random numbers. A particle move to next position, using the values of the best past position, and the values related the global best particle position in the swarm.

4. BSO Hybrid Algorithm and Its Application to Controller Parameters:

In order to improve dynamics of system response problems PSO was introduced. In BSO for further improvement of the performance foraging steps of Ecoli bacteria, chemotactic, reproduce and disperse, are added. In this paper state of particles during foraging is evaluated in our fitness function. Then bacteria with poor errors are eliminated and remaining ones are reproduced, and after that their position updated by PSO in each iteration.

5. Optimal Tuning Of PID Control Parameter:

A closed loop control system is illustrated in Fig. 1 where r(t), e(t) and y(t) are respectively the reference, error and controlled signals.

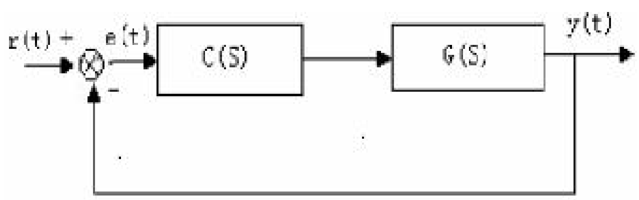

Figure 1: A common feedback control system

G(s): plant transfer function and C(s): PID controller transfer function.

K_p, K_i, K_d are proportional, integral and derivative control parameters gain that has to be found for optimum solution. The plant used here are second order and higher order systems.An optimum system is designed by adjusting PID parameters in the system to meet the required specification. The objective in the BSO and PSO is to seek a set of PID parameters such that the feedback control system has minimum peak over shoot.

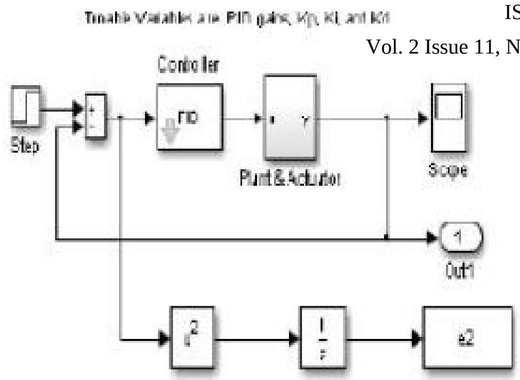

Figure 2: Simulink model for closed loop PID controller

6. Results and Discussions:

In this paper we have developed a hybrid optimization technique including Bacterial Foraging oriented by PSO for determining optimize solution of PID Controller Parameters.
The results consist of response having global minimum error in the step response for a plant governed by PID controller. Usually approximation methods are used for designing PID controllers but they do not guaranty globally best solution.
We have drive the values of PID parameters using PSO and Bacterial Swarm Optimization hybrid method (BSO) on a second order system and high order system representing a plant transfer function. The cost function is taken as the squared of the integral error. In each iteration algorithm selected the parameter which gives minimum cost. The Block diagram of our control system is shown in fig 2. Results are calculated using PSO and BSO algorithm implemented in MATLAB 10. The Simulink model of close loop control system (fig 2) is design using fig 1. The transfer functions considered for the plants are given as

$$H_1(s) = \frac{1}{s^2 + 0.1s + 1} \qquad (3)$$

$$H_2(s) = \frac{1}{s^4 + 3s^3 + 7s^2 + 5s} \qquad (4)$$

Initially both algorithms are tested on second order system with two kinds of cases named as case 1a and case 1b. In case 1a $H_1(s)$ is tuned without integrator. In case 1b $H_2(s)$ is tuned considering all the three control parameters (P, I, D). Fig 3 shows the step response using BSO and PSO for case 1a and fig 4 for case 1 b.

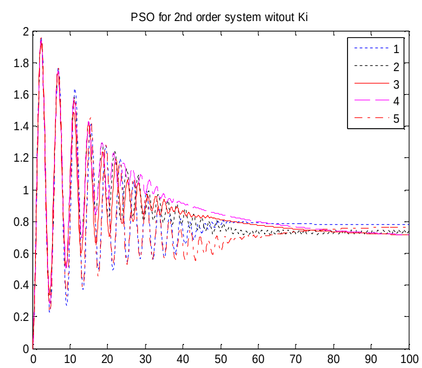

Figure 3 (a)

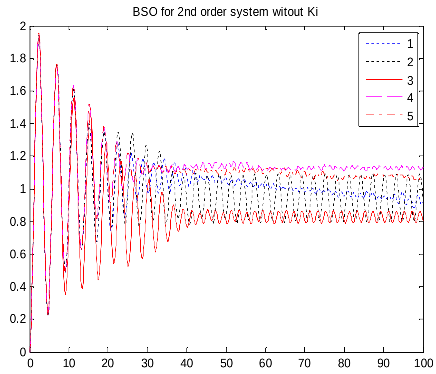

Figure 3 (b)

Figure 3: (a) BSO & (b) PSO response for H1(s) for best 5 results for case 1(a)

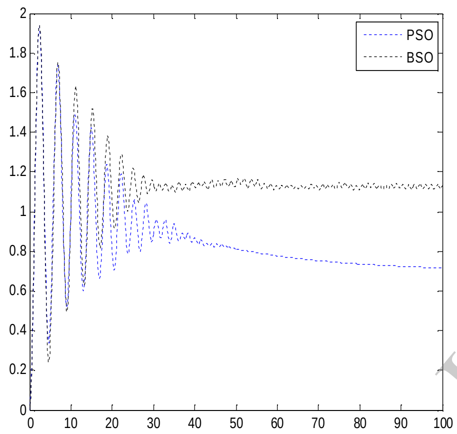

Figure 4: BSO (-.-) and PSO(_) response for H1(s) without considering Ki gain case 1(a)

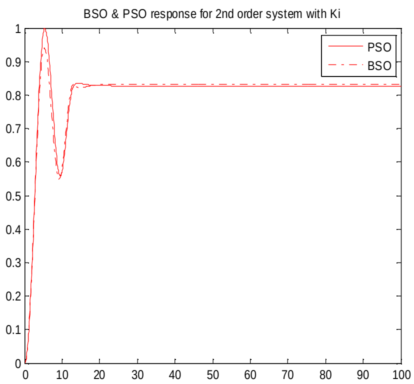

Figure 5: BSO (-.-) and PSO (_) response for H1(s) with considering Ki gain for case 1(b).

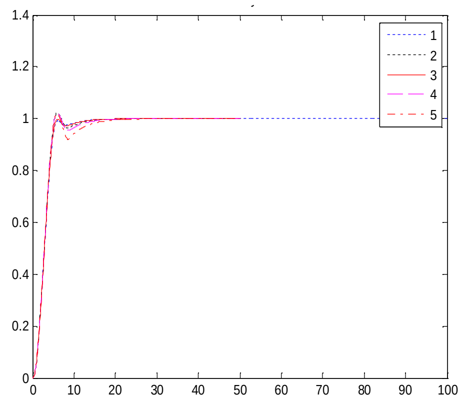

Figure 6 (a)

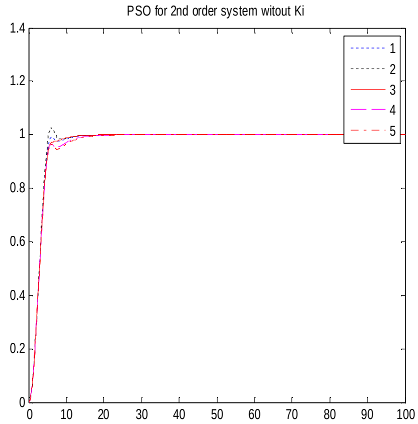

Figure 6 (b)

Figure 6: (a) BSO and (b) PSO(___) response for H2(s) for best 5 results.

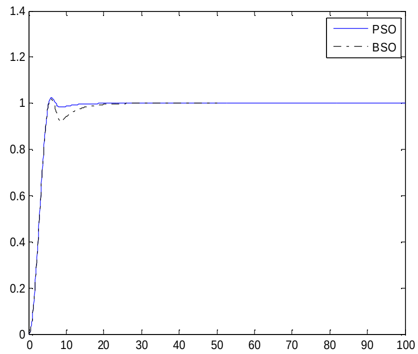

Figure 7: BSO (-.-) and PSO (___) response for H1(s) with considering Ki gain for case 1(b).

ISSN: 2278-0181
Vol. 2 Issue 11, November - 2013

7. Conclusion and Future Scope:

The optimum PD and PID parameters are calculated for 2^{nd} order and higher order plants with transfer function H1 and H2 using PSO and BSO algorithms with minimizing cost related to the square of integral error of steady state response with minimum peak over shoot. Results obtained by using (BSO) algorithm are shown in two cases . Step response for case 1(a) and 1(b) indicates that the peak over shoot using BSO is getting lower on comparison to PSO.Tuning results using PSO and BSO in case 1(a) shows very high oscillatory response . These Kp Ki and Kd parameters are randomly initialized in a fixed same range for case 1(b) here it is found that the oscillations are almost eliminated and steady state error improves with including Ki. The performance of PID controller for BSO in terms of minimum peak and steady state error is found better than PSO as shown in figure 5 for 2^{nd} order system and figure 7 for higher order system.

PSO has the higher cost function as compared to BSO.In all cases (BSO) results in a lower overshoot compared to PSO.In future we can check our results for considering other performance parameters of system response like settling time , rise time and steady state error. We can also consider performance indices as an objective to see system performance. There are many recent modifications are present in PSO these can also be included to generate our BSO algorithm for tuning of PID controller.

Reference:

[1] Back, T., Fogel, D.B. and Michalewicz, Z. “Handbook of Evolutionary Computation”, Oxford Univ. Press and Institute of Physics, New York, 1997.

[2] B. Luitel, G. K. Venayagamoorthy, Differential Evolution Particle Swarm Optimization for Digital Filter Design, IEEE Congress on Evolutionary Computation (CEC 2008), PP. 3954-3961, 2008.

[3] D. H. Kim, A. Abraham, and J. H. Cho “A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Global Optimization”, Information Sciences, vol. 177, no. 18, September 2007, pp. 3918–3937.

[4] D. Karaboga, D.H. Horrocks, N. Karaboga, A. Kalinli, Designing digital FIR filters using Tabu search algorithm, IEEE International Symposium on Circuits and Systems, 1997, vol.4, pp.2236-2239.

[5] S. Mishra, M. Tripathy, and J. Nanda, “Multi-machine Power System Stabilizer Design by Rule Based Bacteria Foraging”, Int. J. of Electric Power Systems Research vol. 77, no. 1, 2007, pp. 1595–1607.

[6]. Thomas Back, Ulrich Hammel, and Hans-Paul Schwefel, “Evolutionary Computation: Comments on the History and Current State”, IEEE Transactions on Evolutionary Computation, Vol. 1, No. 2, pp. 3-17, 1997.