International Journal of Modern Computation, Information and Communication Technology, 2018;1(1):24-32.
http://ijmcict.gjpublications.com

Research Article

A Multi Objective Hybrid Algorithm to Optimize Set-point Filter based PID Controller for a Class of Systems

L. Balachandar, B. Ashok Raja

Department of Electronics and Instrumentation Engineering, St. Joseph's College of Engineering, Chennai - 600119. India.

*Corresponding author's e-mail: bala160897@gmail.com

Abstract

This paper proposes a novel hybrid evolutionary algorithm using Particle Swarm assisted Bacterial Foraging Optimization algorithm for the closed loop automatic tuning of a set-point filter and PID controller for a class of chemical systems operating at unstable steady state. In this work, the PSO algorithm is employed in the optimization search to add the velocity parameter for the tumbling operation of the bacterial foraging algorithm, which can speed up the algorithm convergence. The need for a suitable PID controller structure for the evolutionary algorithm based search is discussed in detail. In the proposed work, the optimization process is focused to search the best possible controller parameters (K_p, K_i, K_d) and set-point filter parameter (T_f) by minimizing the multi objective performance index. The effectiveness of the proposed scheme has been confirmed through a comparative study with PSO, IPSO, BFO and the classical controller tuning methods proposed in the literature. The results show that, the proposed method provides enhanced performance in effective reference tracking with minimal ISE and IAE values. Finally the robustness of the proposed method is validated by operating the unstable systems in the presence of a measurement noise. The results testify that the PSO-BFO tuned set-point filter based PID performs well in tracking the change in reference signal even in the noisy environment.

Keywords: Particle swarm optimisation; Bacterial foraging optimization; Hybrid algorithm, Unstable systems; PID controller.

Introduction

In chemical industry, important processing units such as jacketed Continuous Stirred Tank Reactor (CSTR), Continuous Stirred Tank Bioreactor (CSTB), fermenters, and polymerization reactor are inherently open-loop unstable by design and for economical and/or safety reasons, these process loops to be operated in unstable steady state [1]. During the closed loop operation, optimized controller parameters for such systems are essential to minimize the waste and to maximize the production rate. Fine tuning the controller parameters for unstable systems is highly complex than open loop stable systems.

In recent years, many efforts have been attempted to design optimal and robust controllers for unstable chemical systems. Panda has proposed a synthesis method to design an Internal Model Controller based PID (IMC-PID) controller for a class of time delayed unstable process [2]. Chen et al. have discussed a set-point weighted PID controller tuning for time delayed unstable systems [3]. Huang and Chen have examined an auto-tuning method based PID controller for a class of second order unstable process having dead time [4]. Lee et al. have proposed a PID controller tuning methodology for integrating and unstable processes with time delay [5]. Sreeet al. has discussed a classical PID controller tuning method for stable and unstable first order systems [6]. Visioli, have discussed a classical tuning method for unstable systems. Most of these classical approaches require an approximated first or second order transfer-function model with a time delay [7]. In real time applications, the approximated model

Received: 16.04.2018; Received after Revision: 28.04.2018; Accepted: 28.04.2018; Published: 13.06.2018

parameter may be changing or subject to uncertainty. The tuning method proposed for a particular model does not provide a satisfactory result on the other models. The classical controller tuning procedures proposed for time delayed unstable system also requires complex computations to identify the optimal controller parameters. To overcome this, it is necessary to employ intelligent controller auto tuning methods to identify the best possible controller parameters for the unstable process loop.

In recent years, evolutionary algorithm based optimization is emerged as a powerful tool for finding the solutions for a variety of control engineering applications. Soft computing based PID controller parameter optimization is widely addressed by the researchers. The literature gives the application details of soft computing in PID controller tuning for a class of stable systems [8-13]. Hybridization based optimization techniques [14,15] have also been reported in PID controller tuning for stable process models. The above methods are proposed for stable systems only. For stable systems, the overshoot and the error value will be very small and it supports the PID controller tuning efficiently. For unstable systems, the controller parameter tuning seems to be difficult task and is limited due to‘d/τ’ ratio. Since the basic PID controller will not provide the optimised parameter and this may require a modified PID structure such as I-PD.

Recently, Rajinikanth and Latha [10-12] has attempted evolutionary algorithm based PID and I-PD tuning for a class of unstable process models. In this work, error minimization is highly prioritized as a performance measure and it monitors the algorithm, until the controller parameters converge to an optimized value. From their work it has been demonstrated that a BFO based PID controller tuning can be performed for the unstable system when the ‘d/τ’ ratio is below 0.2. PID based tuning results large overshoot which tends to increase the error and overshoot value, when the d/τ ratio is greater than 0.2. This phenomenon disrupts the convergence of soft computing based search.

In the present work, the PID controller parameter tuning is proposed for a class of unstable chemical systems using Particle Swarm assisted Bacterial Foraging Optimization (PSO-BFO) algorithm. A set-point filter based PID controller proposed by Jung et al. [16] is considered in this study to evaluate the performance of the proposed method. A comparative study also carried out with basic Particle Swarm Optimization (PSO), Improved PSO Algorithm (IPSO), Bacterial Foraging Optimization (BFO) and the classical controller tuning methods.

Methodology

Particle Swarm Optimization

PSO is a population based stochastic optimization technique inspired by social behaviour of bird flocking or fish schooling, and it is widely used in engineering applications due to its high computational efficiency [17,18]. PSO algorithm is easy to implement and there are few parameters to adjust compared to other heuristic methods. It is a population based evolutionary computation technique, attempts to mimic the natural process of group communication of individual knowledge, to achieve some optimum property. In this method, a population of swarm is initialized with random positions ‘S_i’ and velocities ‘V_i’. At the beginning, each particle of the population is scattered randomly throughout the entire search space and with the guidance of the performance criterion, the flying particles dynamically adjust their velocities according to their own flying experience and their companions flying experience. Each particle remembers its best position obtained so far, which is denoted pbest (P_i^t). It also receives the globally best position achieved by any particle in the population, which is denoted as gbest (G_i^t). The updated velocity of each particle can be calculated using the present velocity and the distances from pbest and gbest. The updated velocity and the position are given in eq. (1) and (2) respectively [19].

$$V_i^{t+1} = W^t.V_i^t + C_1.R_1.(P_i^t - S_i^t) + C_2.R_2.(G_i^t - S_i^t) \quad (1)$$

$$S_i^{t+1} = S_i^t + V_i^{t+1} \quad (2)$$

$$W^t = (W_{max} - Iter) \text{ x } \frac{(W_{max} - W_{min})}{Iter_{max}} \quad (3)$$

Where, C_1, C_2 are positive constants. C_1 is the cognitive learning rate and C_2 is the global learning rate. R_1, R2 are random numbers in the range 0-1. The parameter ‘W’ is inertia weight that increases the overall performance of PSO. The larger value of ‘W’ can favour the global wide-range search and lower value of 'W' implies a higher ability for local nearby search.

Improved PSO Algorithm

Chang and Shih [20] have developed an IPSO algorithm to tune the PID controller for a non-linear inverted pendulum system. In this, an improved velocity updating equation is proposed to improve the algorithm convergence and it is given in eq. (4).

$$V_i^{(t+1)} = W^t . V_i^t + C_1 . R_1 . (pbest - S_i^t) + C_2 . R_2 . (gbest - S_i^t) + C_3 . R_3 . (ibest - S_i^t) \quad (4)$$

Where, 'ibest' represents the best particle's position among all particles in the sub-population that the i^{th} particle belongs to. C_3 and R_3 are positive constant and random number respectively.

Bacterial Foraging Optimization

Bacteria Foraging Optimization (BFO) algorithm is a biologically inspired stochastic search technique based on mimicking the foraging (methods for locating, handling and ingesting food) behavior of *E. coli* bacteria [21-23]. During foraging, a bacterium can exhibit two different actions: Tumbling or swimming. The tumble action modifies the orientation of the bacterium. During swimming (chemotactic step) the bacterium will move in its current direction. Chemotactic movement is continued until a bacterium goes in the direction of positive nutrient gradient. After a certain number of complete swims, the best half of the population undergoes reproduction, eliminating the rest of the population. In order to escape local optima, an elimination-dispersion event is carried out where, some bacteria are liquidated at random with a very small probability and the new replacements are initialized at random locations of the search space.

Chemo-taxis

This is the initial stage of BFO search. During this process, the bacteria can move towards the food location with the action of swimming and tumbling via flagella. Through swimming, it can move in a specified direction and during tumbling action, the bacteria can modify the direction of search. These two modes of operation is continuously executed to move in random paths to find adequate amount of positive nutrient gradient. These operations are performed in its whole lifetime.

Swarming

In this process, after the success towards the best food location, the bacterium which has the knowledge about the optimum path will attempt to communicate to other bacteria by using an attraction signal. The signal communication between cells in *E. coli* bacteria is represented by the following eq. (5).

$$J_{cc}(\theta, P(j,k,l)) = \sum_{i=1}^{n} J_{cc}(\theta, \theta^i(j,k,l)) = A + B \quad (5)$$

Where

$$A = \sum_{i=1}^{n}[-d_{attractant} \exp(-W_{attractant} \sum_{m=1}^{P}(\theta_m - \theta^i{}_m)^2)]$$

$$B = \sum_{i=1}^{n}[h_{repellant} \exp(-W_{repellant} \sum_{m=1}^{P}(\theta_m - \theta^i{}_m)^2)]$$

Where J_{cc} (θ, P (j, k, l)) represents objective function value, 'n' is the total number of bacterium, 'P' the total parameters to be optimised. The other parameters such as '$d_{attractant}$' are the depth of attractant signal released by a bacteria and '$W_{attractant}$' is the width of attaractant signal. The signals '$h_{repellant}$' and '$W_{repellent}$' are the height and width of repellent signals between bacterium. Attractant is the signal for food source and repellent is the signal for noxious substance.

Reproduction

In swarming process, the bacteria accumulated as groups in the positive nutrient gradient and which may increase the bacterial density. Later, the bacteria are sorted in descending order based on its health values. The bacteria which have the least health will expire and the bacteria with the most health value will split into two and breed to maintain a constant population.

Elimination-dispersal

Based on the environmental conditions such as change in temperature, noxious surroundings, and availability of food, the population of a bacteria may change either gradually or suddenly. During this stage, a group of the bacteria in a restricted region (local optima) will be eliminated or a group may be scattered (dispersed) into a new food location in the search space. The dispersal possibly flattens the chemo-taxis advancement. After dispersal, sometimes the bacteria may be placed near the good nutrient source and it may support the chemo-taxis, to identify the availability of other food sources. The above procedure is repeated until the optimized solutions are achieved.

Hybrid Optimization Algorithm

This algorithm was proposed by Korani et al. [23] to improve the performance of BFO. In this method, the cost function (ISE) is applied for both the PSO and BFO algorithm. The PSO algorithm monitors the BFO to achieve a minimum convergence time with optimized parameters. In hybrid algorithm, after undergoing a chemotactic step, each bacterium gets mutated by a PSO operator. The PSO operator considers only the 'social' component and eliminates the 'cognitive' component. In this algorithm, due to the information sharing between the PSO and BFO (Tumble – step 4.3), the hybrid algorithm can provide the optimized solutions with minimal convergence time compared to a conventional BFO algorithm.

Hybrid optimization algorithm is defined as follows:

Step 1 % Assign values for the BFO parameters %: Initialize: dimension of search space (D); number of bacteria (n); No. of chemo tactic steps (N_c); No. of reproduction steps (N_{re}); No. of elimination-dispersal events (N_{ed}); No. of bacteria reproduction (n_r); Probability for elimination – dispersal (P_{ed}); Random swim direction vector (Δ_i); run length vector (C_i). % Assign values for the PSO parameters %: Initialize: swarm (N) and step size; learning rate (C_1, C_2) ;inertia weight (W_{min}, W_{max}); Initialize random values (R_1, R_2).
% Iterative algorithm for optimization %:
Generate initial values for K_p, K_i, K_d and T_f .
Begin with the calculation of the cost function (J_1). For any i^{th} bacteria at the j^{th} chemotactic, k^{th} reproduction and i^{th} elimination stage is $\theta^i(j,k,l)$ and its cost function value is given by J (i, j, k, l).

Step 2 Elimination-dispersal loop: l = l + 1

Step 3 Reproduction loop : k = k+1

Step 4 Chemotaxis loop: j = j+1

4.1 For bacteria (i) = 1,2, . . . n; calculate J (K_p, K_i, K_d) and compute fitness function J [(w_1.CF) + (w_2 . t_s) + (w_3 . E_{ss}) +(w_4. M_p)]

4.2 Let, J_{last} = J (K_p, K_i, K_d)

4.3 Tumble (PSO-Velocity Update):

$$V_i^{t+1} = W^t.V_i^t + C_1.R_1.(P_i^t - S_i^t) + C_2.R_2.(G_i^t - S_i^t)$$

4.4 Move: $\theta^i(j+1, k, l) = \theta^i(j, k, l) + C_i \dfrac{\Delta_i}{\sqrt{\Delta_i^T \times \Delta_i}}$

Calculate J (i, j+1, k, l)

4.5 Swim

Let m=0 (counter for swim length)

While m < N_s

m =m+1

If J (K_p, K_i, K_d) < J_{last} , then $J_{last} = J_1$ (i, j+1, k, l) and

$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C_i \dfrac{\Delta_i}{\sqrt{\Delta_i^T \times \Delta_i}}$

Calculate J (i, j+1, k, l) using θ^i (j+1, k, l)

Else m =N_s

4.6 Repeat the above procedures for the bacterium (i+1) till all the bacterium undergo chemotaxis.

Step 5 If j < N_c go to step 4.3 and continue chemotaxis

Step 6 % Reproduction %

a) For the given k and i , and for each i = 1,2 ,… ,n, let

$ITAE^i_{health} \sum_{j=1}^{Nc+1} J(i,j,k,l)$ is the health of i^{th} bacterium.

All the bacteria are sorted according J^i_{health} (Ascending order)

b) The bacteria with the highest J_{health} die and the other bacteria with minimum values split and the copies that are made are placed at the same location as their parent

Step 7 If k < N_{re}
go to Step 4.2 to start the next generation in the chemotactic loop. Else go to Step 3

Step 8 % Elimination – Dispersal %
For I = 1,2,…,n; a random number is generated and if rand ≤ P_{ed} , Then eliminate the bacteria and disperse it to a random location.
Else do not eliminate the bacteria

Step 9 If l < N_{ed} go to Step 2 Else STOP.

PID Tuning Procedure

Preliminaries

A generalized close loop control system is depicted in fig. 1. The controller 'G_c(s)' has to provide closed loop stability, smooth reference tracking and load disturbance rejection [24].

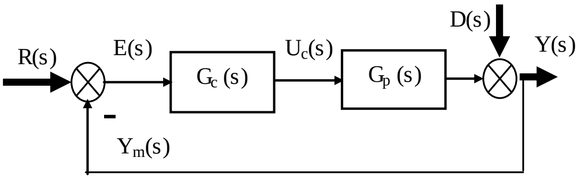

Fig. 1. Block diagram of a closed loop control system

Closed loop response of the above system with set point 'R(s)' and disturbance 'D(s)' can be expressed as;

$$Y(s)=\left[\frac{G_p(s)G_C(s)}{1+G_p(s)G_C(s)}\right]R(s)+\left[\frac{1}{1+G_p(s)G_C(s)}\right]D(s) \quad (6)$$

The final steady state response of the system for the reference tracking and the disturbance rejection is presented in eq. (3) and eq. (4) correspondingly.

$$Y_R(\infty)=\lim_{t\to\infty} s\,Y_R(s)=\lim_{t\to\infty} s \text{ x} \left[\frac{G_p(s)G_C(s)}{1+G_p(s)G_C(s)}\right]\left(\frac{A}{s}\right)=A \quad (7)$$

$$Y_D(\infty)=\lim_{t\to\infty} s \text{ x} \left[\frac{1}{1+G_p(s)G_C(s)}\right]\left(\frac{L}{s}\right)=0 \quad (8)$$

Where: A = amplitude of reference signal
D = disturbance

To achieve a satisfactory Y_R (∞) and Y_D (∞), it is necessary to have optimally tuned values for K_p, K_i and K_d. In this study, a non-interacting form of parallel PID controller is considered to achieve the preferred response.

The parallel PID structure is given below:

$$G_C(s)=K_p\ e(t)+K_i\int_0^T e(t)\,dt+K_d\frac{de(t)}{dt} \quad (9)$$

$$G_{PID}(s)=K_p\left[1+\frac{1}{T_i\,s}+T_d\,s\right] \quad (10)$$

Where: $K_p / T_i = K_i$; $Kp*T_d = K_d$.

Performance Criterion

In closed loop systems, the main objective of the controller is to make the peak overshoot (M_p), settling time (t_s) and final steady state error (E_{ss}), as small as possible. In soft computing based approach, the Cost Function (CF) is used to appraise the performance of the closed loop system during the optimization search. Integral Time Absolute Error (ITAE) criterion shown in eq. (11) is preferred as the 'CF'.

$$ITAE=\int_0^t t\,.\left|e(t)\right|\,dt=\int_0^t t\,.\left[r(t)-y(t)\right]dt \quad (11)$$

Where e(t) = error, r(t) = reference input, and y(t) = process output.

The multiple objective functions for controller optimization was first proposed by Zamani et al. [25] for stable systems. In this method, along with CF, values like M_p, t_s, E_{ss}, rise time (t_r), Gain Margin (GM) and Phase Margin (PM) were considered in the performance criterion. This approach can work good for stable system models. For unstable models, the peak overshoot (M_p) is unavoidable and also the values like GM and PM cannot be obtained during the optimization search. Since, in this work we proposed a simple performance criterion with four functions, such as CF, M_p, t_s, and E_{ss} as presented in eq. (12).

$$J(K_p, K_i, K_d) = (w_1.CF) + (w_2.t_s) + (w_3.E_{ss}) + (w_4.\ M_p) \quad (12)$$

Where:
w_1. w_2, w_3, w_4- weighting parameters (range is from 0 - 1),

- CF - ITAE
- M_p, T_s and E_{ss} are additional performance index obtained from the process output as in fig. 2.

Eq. (12) shows a multi objective criterion and has four terms accompanied by a weighting factor 'w'. The above equation can work fine for a class of stable and unstable process models.

PID controller tuning

The PID tuning process is employed to find the best possible values for K_p, K_i and K_d and the set-point filter parameter 'T_f' form the three dimensional search space by minimizing the objective function (Eq. 12). During this search, the performance criterion 'J (K_p, K_i, K_d)' guides the hybrid algorithm to get appropriate values for the controller parameters. In the literature, there is no clear guide line to assign the algorithm parameters for the evolutionary algorithm. In this study, we propose a simple method to assign the parameters for BFO algorithm in order to reduce the convergence time during the optimization search.

Optimal algorithm parameters for optimization search

- The best possible value for number of bacteria (N_b) is between 10 – 20 (for stable and unstable systems)

- Number of chemotactic steps (Nc) = N_b /2
- Length of swim (Ns) = number of reproduction steps (Nr) = number of elimination-dispersal events (Ned) ≈ Nc / 2
- Number of bacteria reproductions (Nsr) = N_b/2
- probability for bacteria eliminated/dispersed (Ped) = 0.25
- The three dimensional search space is defined as: K_p: ± 30%; K_i : ± 30% and K_d : ± 30% . If the search does not converge with an optimal K_p, K_i, K_d values, increase the search range by 5% and begin a new search.
- The steady state error (E_{ss}) in the process output is assigned as zero.
- There is no guideline to specify the values for CF, Overshoot (M_p) and settling time (t_s). In this, 't_s' is preferred as <75% of the maximum simulation time and the 'M_p' as <50% of the reference signal.

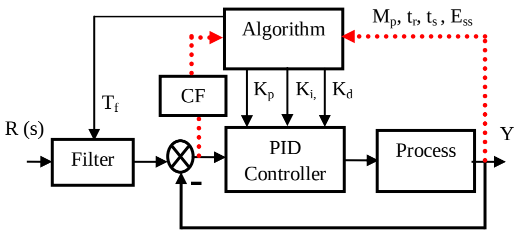

Fig. 2. Optimization algorithm based controller tuning

The following parameters are assigned for the algorithm parameters:

PSO: Dimension of search space is three (ie. K_p, K_i, K_d); number of swarm and bird step is considered as twelve; the cognitive (C_1) and global (C_2) search parameter is assigned the value of 2, the minimum (W_{min}) and maximum (W_{max}) inertia weight is set to be 0.2 and 14 respectively. For IPSO the parameter C_3 is set as 2.

BFO: Dimension of search space is three; number of bacteria is chosen as twelve; number of chemotactic steps is set to six; number of reproduction steps, length of a swim and number of elimination-dispersal events are considered as three; number of bacterial reproduction is assigned as six, probability for bacteria eliminated/dispersed is set as '0.25'.

Other parameters are assigned as follows: $d_{attractant}$ = 0.3, $W_{attractant}$ = 0.5, $h_{repellant}$ =0.5 and $W_{repellent}$ = 0.5.

Results and discussion

The results of simulations of five examples to illustrate the effectiveness of the proposed hybrid optimization algorithm tuned set-point filter with PID control design method are given in this section.

Process 1

The first order plus delayed time unstable process with the following transfer function model is considered (Eq. 13).

$$G_p(s) = \frac{4e^{-2s}}{4s-1} \tag{13}$$

The process has a gain (K) = 4, process time constant (τ) = 4 and time delay (d) =2. For this process d/τ is 0.5. Many studies have proposed different PID settings for the above model and the values are clearly presented in the literature. The classical PID settings are presented in table 1. The evolutionary algorithm based controller tuning is proposed for the system as in fig. 2.

The final convergence of the controller parameters for the hybrid algorithm is shown in fig. 3 and the optimised K_p, K_i, K_d values are tabulated in Table 1. Fig. 4 shows the convergence of the CF for the various evolutionary search algorithms. Fig. 5 depicts the servo response of the process with classical PID settings proposed in the literature. Fig. 6 shows the reference tracking performance of the evolutionary methods. The observation is that, the hybrid method provides a good result for reference tracking performance.

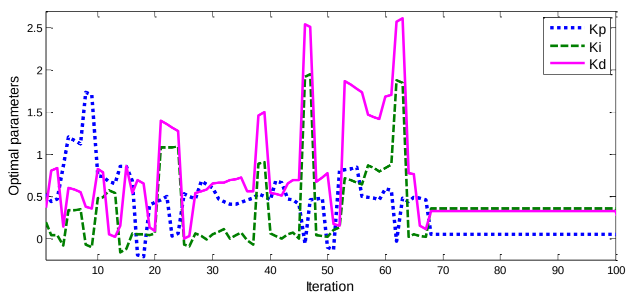

Fig. 3. Convergence of PID parameters with hybrid algorithm

From table 1, it is observed that, the hybrid algorithm based tuning has less number of iteration (67) and it also shows a good performance measure in reference tracking (such as: ISE, IAE, M_p and T_s) compared to PSO, IPSO and BFO algorithms.

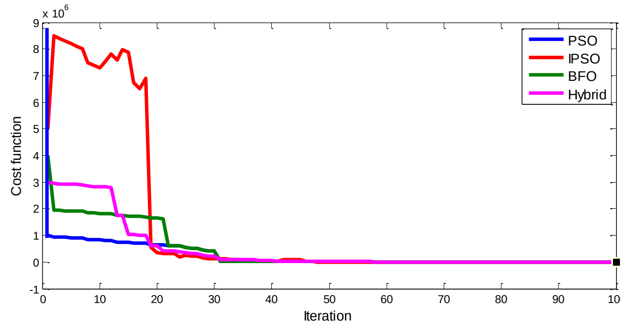

Fig. 4. Convergence of cost function

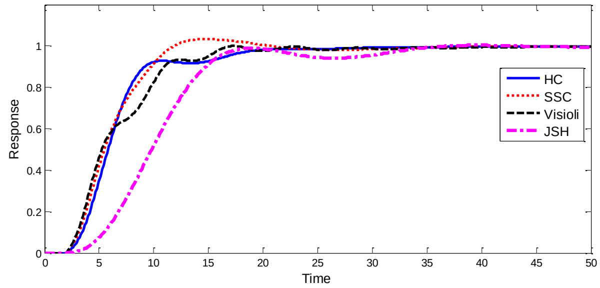

Fig. 5 Servo responses for process 1 with classical PID parameters

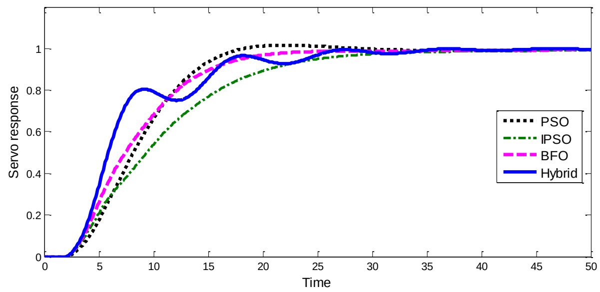

Fig. 6. Servo responses for process 1 with optimized PID parameters

Process 2

The second order delayed unstable process with the following transfer function is considered (Eq. 14). It has one unstable pole and a stable pole.

$$G_p(s) = \frac{\exp^{-s}}{(2s-1)(0.5s+1)} \qquad (14)$$

Previous studies have proposed different PID settings for the above model. Fig. 7 shows the servo response of the previous work reported in literature. In this diagram, the method proposed by LLP provides the smooth reference tracking performance compared to PC and HC. Fig. 8 shows the reference tracking performance of the present study. The response produced by the PSO and IPSO algorithm is more oscillatory compared to BFO and PSO-BFO methods.

From fig. 8 and table 1, it is inferred that, the proposed hybrid method can be used to get an optimal controller parameter with lesser convergence time to provide a smooth reference tracking performance than other optimization algorithms.

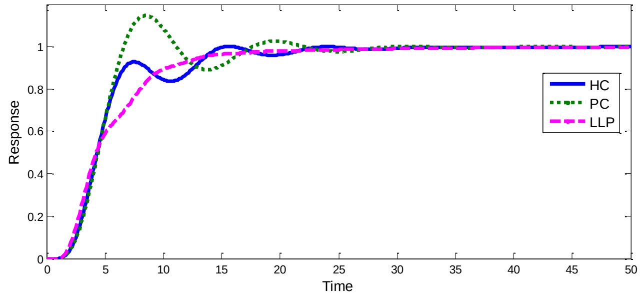

Fig. 7. Servo responses for Process 2 with conventional PID parameters

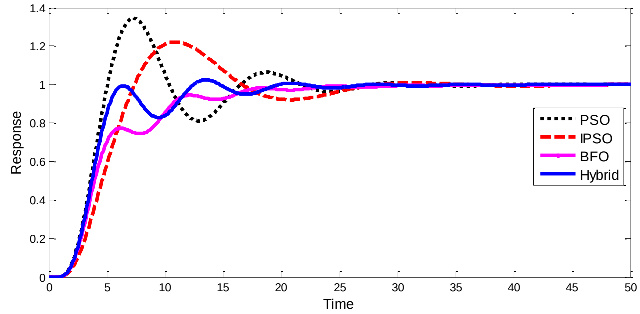

Fig. 8. Servo responses for Process 2 with soft computing based PID parameters (see online version for colours)

Process 3

The third order delayed unstable process with the following transfer function is considered (Eq. 15). It has one unstable pole and two stable poles.

$$G_p(s) = \frac{\exp^{-0.5s}}{(5s-1)(0.5s+1)(2s+1)} \qquad (15)$$

The classical PID parameters proposed by previous studies are presented in Table 1. The PSO- BFO tuned controller and filter gains and the final iteration numbers are provided in table 1. The hybrid algorithm based controller parameter search value is converging at 59th iteration. Fig. 9 and 10 shows the servo response of the process 3 with classical and algorithm based set-point filter based PID controller respectively.

From table 1, it is noted that, the result by the proposed method performs superior than the classical and other optimization algorithm with a lesser value of ISE, IAE and settling time (T_s) values.

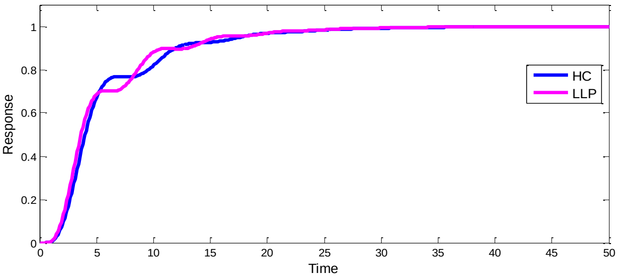

Fig. 9. Servo responses for process 3 with optimized PID parameters

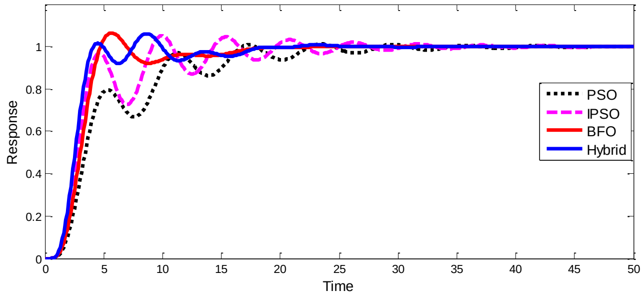

Fig. 10. Servo responses for Process 3 with optimized PID parameters

Table 1. Controller parameters, filter parameters and the performance measure for the simulation study

Process	Method	Iteration	K_p	K_i	K_d	T_f	ISE	IAE	M_p	T_s
Process 1	HC	-	0.5650	0.0460	0.3435	12.276	29.27	5.410	0.000	21.4
	SSC	-	0.5480	0.0493	0.5611	11.117	25.60	5.060	0.033	20.7
	Visioli	-	0.6240	0.0540	0.7245	11.551	21.30	4.615	0.000	23.7
	JSH	-	0.3840	0.0127	0.0000	30.301	293.9	17.14	0.000	35.3
	PSO	76	0.4403	0.0217	0.3133	20.290	122.0	11.04	0.016	28.4
	IPSO	68	0.5110	0.0230	0.5887	22.217	103.6	10.18	0.000	27.1
	BFO	82	0.5190	0.0302	0.5017	17.185	65.30	8.081	0.000	35.4
	Hybrid	67	0.6361	0.0471	0.3257	13.505	27.66	5.260	0.000	24.6
Process 2	HC	-	1.7920	0.1442	0.8602	12.425	47.71	6.907	0.000	24.8
	PC	-	1.5860	0.1322	0.7597	12.000	56.89	7.542	0.146	29.4
	LLP	-	1.9490	0.1616	1.6099	12.063	38.03	6.167	0.000	23.1
	PSO	61	1.6502	0.2161	1.0061	7.6363	21.39	4.625	0.345	27.7
	IPSO	53	1.4462	0.1142	1.0644	12.664	74.91	8.655	0.221	28.3
	BFO	78	2.0772	0.1865	1.2290	11.138	28.25	5.316	0.000	23.9
	Hybrid	49	1.9518	0.2103	1.0643	9.2810	22.49	4.742	0.024	21.1
Process 3	HC	-	6.1860	0.8628	9.1058	7.1700	1.336	1.156	0.000	37.8
	LLP	-	7.1440	1.0688	11.823	6.6840	0.873	0.934	0.000	36.2
	PSO	64	8.4588	1.3620	10.545	6.2106	0.541	0.735	0.008	42.7
	IPSO	59	9.0323	2.1186	13.056	4.2633	0.224	0.474	0.051	34.7
	BFO	75	6.2270	1.5474	10.595	4.0242	0.418	0.646	0.065	19.7
	Hybrid	59	6.9045	2.0603	15.036	3.3512	0.235	0.485	0.061	19.1

Conclusions

In the present work, an attempt has been made for tuning a set-point filter (pre-filter) based PID controller structure for a class of unstable process models using Particle Swarm Optimization assisted Bacterial Foraging Optimization (PSO-BFO) based hybrid optimization algorithm with minimizing the multiple objective performance criterion. A comparative study with the basic PSO, IPSO, BFO and classical PID tuning methods proposed in the literature has been discussed. The hybrid method tuned controller provides an enhanced reference tracking performance with minimal cost function. It also provides improved time domain specifications and robust performance for the unstable system with perturbed process parameters uncertainty.

Conflicts of interest

Authors declare no conflict of interest.

References

[1] Padma Sree R, Chidambaram M. Control of unstable systems. India: Narosa Publishing House; 2006.

[2] Panda RC. Synthesis of PID controller for unstable and integrating processes. Chem Eng Sci. 2009;64;2807-16.

[3] Chen CC, Huang HP, Liaw HJ. Set-Point weighted PID controller tuning for time-delayed unstable processes. Ind Eng Chem Res. 2008;47:6983-90.

[4] Huang HP, Chen CC. Auto-tuning of PID controllers for second order unstable process having dead time. J Chem Eng Jpn. 1999;32:486-97.

[5] Lee Y, Lee J, Park S. PID controller tuning for integrating and unstable processes with time delay. Chem Eng Sci. 2000;55:3481-96.
[6] Sree RP, Srinivas MN, Chidambaram M. A simple method of tuning PID controllers for stable and unstable FOPDT systems. Comput Chem Eng. 2004;28:2201-18.
[7] Visioli A. Optimal tuning of PID controllers for integral and unstable processes. IEE Proc Control Theory Appl. 2001;148:180-4.
[8] Rajinikanth V, Latha K. Bacterial foraging optimization algorithm based PID controller tuning for Time Delayed unstable system. The Mediterranean Journal of Measurement and Control. 2011;7:197-203.
[9] Rajinikanth V, Latha K. Optimization of pid controller parameters for unstable chemical systems using soft computing technique. International Review of Chemical Engineering. 2011;3:350-8.
[10] Rajinikanth V, Latha K. Setpoint weighted PID controller tuning for unstable system using heuristic algorithm. Archives of Control Sciences. 2012;22:481-505.
[11] Rajinikanth V, Latha K. Controller parameter optimization for nonlinear systems using enhanced bacteria foraging algorithm. Applied Computational Intelligence and Soft Computing. 2012;2012:Article ID 214264. 2012. https://doi.org/10.1155/2012/21426.
[12] Rajinikanth V, Latha K. I-PD Controller Tuning for unstable system using bacterial foraging algorithm: A study based on various error criterion. Applied Computational Intelligence and Soft Computing. 2012;2012:Article ID: 329389. https://doi.org/10.1155/2012/329389.
[13] Rajinikanth V, Latha K. Tuning and retuning of PID controller for unstable systems using evolutionary algorithm. ISRN Chemical Engineering. 2012;2012:Article ID: 693545. 2012. https://doi.org/10.5402/2012/69354.
[14] Kim DH, Abraham A, Cho JA. A hybrid genetic algorithm and bacterial foraging approach for global optimization. Information Sciences. 2007;177:3918-37.
[15] Anguluri R, Abraham A, Snasel V. A hybrid bacterial foraging-PSO algorithm based tuning of optimal FOPI speed controller. Acta Montanistica Slovaca. 2011;16:,55-5 .
[16] Jung CS, Song HK, Hyun JC. A direct synthesis tuning method of unstable first-order-plus-time-delay processes. J Process Control. 1999;9:265-69.
[17] Kennedy J, Eberhart R. Particle swarm optimization. Proceedings of IEEE International Conference on Neural, pp. 1942- 1948. (1995).
[18] Fu T, Ya-Ling S. An improved multi-objective particle swarm optimisation algorithm. International Journal of Modelling Identification and Control. 2011;12:66-71.
[19] Biswas A, Dasgupta S, Das S, Abraham A. Synergy of PSO and bacterial foraging optimization – A comparative study on numerical benchmarks. Innovations in Hybrid Intelligent Systems. 2007;44:255-63.
[20] Chang WD, Shin SP. PID controller design of nonlinear systems using an improved particle swarm optimization approach. Communications in Nonlinear Science and Numerical Simulation. 2010;15:3632-39.
[21] Passino KM. Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control Systems Magazine. 2002:52-67.
[22] Gollapudi SVRS, Pattnaik SS, Bajpai OP, Devi S, Bakwad KM. Velocity modulated bacterial foraging optimization technique. Applied Soft Computing. 2011;11:154-65.
[23] Korani WM, Dorrah HT, Emara HM. Bacterial foraging oriented by particle swarm optimization strategy for PID tuning. Proceedings of the 8th IEEE International Conference on Computational Intelligence in Robotics and Automation. 2008: p. 445-50.
[24] Johnson MA, Moradi MH. PID Control: New Identification and Design Methods. London; Springer-Verlag Limited: 2005. p. 47-107.
[25] Zamani M, Sadati N, Ghartemani MK. Design of an Hα PID Controller Using Particle Swarm Optimization. Int J Control Autom Syst. 2009;7:273-80.

****