

Research Article

Controller Parameter Optimization for Nonlinear Systems Using Enhanced Bacteria Foraging Algorithm

V. Rajinikanth¹ and K. Latha²

¹Department of Electronics and Instrumentation Engineering, St. Joseph's College of Engineering, Chennai 600 119, India

²Department of Instrumentation Engineering, MIT Campus, Anna University, Chennai 600 044, India

Correspondence should be addressed to V. Rajinikanth, rajinisjeeie@gmail.com

Received 29 April 2012; Revised 23 July 2012; Accepted 16 August 2012

Academic Editor: Serafin Moral

Copyright © 2012 V. Rajinikanth and K. Latha. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An enhanced bacteria foraging optimization (EBFO) algorithm-based Proportional + integral + derivative (PID) controller tuning is proposed for a class of nonlinear process models. The EBFO algorithm is a modified form of standard BFO algorithm. A multiobjective performance index is considered to guide the EBFO algorithm for discovering the best possible value of controller parameters. The efficiency of the proposed scheme has been validated through a comparative study with classical BFO, adaptive BFO, PSO, and GA based controller tuning methods proposed in the literature. The proposed algorithm is tested in real time on a nonlinear spherical tank system. The real-time results show that, EBFO tuned PID controller gives a smooth response for setpoint tracking performance.

1. Introduction

In control literature, despite of significant developments in advanced process control schemes such as model predictive control (MPC), internal model control (IMC), and sliding mode control (SMC), PID controllers are still widely used in industrial control system where reference tracking and disturbance rejection are a major task.

The key merits of the PID controllers over the advanced control techniques are as follows: (i) available in a variety of structures such as series, parallel, and so forth; (ii) provides an optimal and robust performance for a variety of processes; (iii) supports online/offline tuning and retuning based on the performance requirement of the process under control (iv) simple structure and it can be easily implementable in analog or digital form; (v) along with the basic and the modified structures, it also supports the one degree of freedom (1DOF), 2DOF, and 3DOF controller structures.

Most of the real-time chemical process loops such as continuous stirred tank reactor (CSTR), biochemical reactor, spherical tank system, and conical tank system are nonlinear in nature. These nonlinear processes can be modelled as linear processes (stable or unstable process model with a

delay time) around the operating region. The linear model is then efficiently controlled by employing a PID controller. The precision and performance of the PID controller mainly rely on three controller parameters such as proportional gain (K_p), integral gain (K_i), and derivative gain (K_d). In recent years, a number of tuning rules have been proposed for the PID controllers to enhance the performance of the process to be controlled. Maher and Samir [1] have discussed the robust stability criterion for a class of unstable systems under model uncertainty. Vijayan and Panda [2] have proposed a setpoint filter PID controller for a class of stable and unstable process models. Padmasree and Chidambaram [3] have provided a detailed review on the methods of controller tuning for a class of time delayed unstable system. Jhunjhunwala and Chidambaram [4] have examined an optimized PID controller tuning for nonlinear systems such as the biochemical reactor and the CSTR process, modelled as a time delayed unstable system around the operating region.

The classical PID tuning methods proposed by most of the researchers for the stable and unstable processes require an approximated first or second order transfer-function model with a time delay. The tuning procedure anticipated

for one particular process model will not provide the fitting response for other process models. Hence, soft computing-based PID controller tuning is widely proposed by the researchers to design a robust PID controller. Chiha et al. [5] proposed a multiobjective ant colony optimization (ACO) algorithm-based PID controller tuning for a class of time delayed stable process models. Zamani et al. [6, 7] have discussed a multi objective particle swarm optimization (PSO) algorithm-based “ H_∞ ” and fractional order PID controller design for stable systems. Chang and Shih [8] presented an improved PSO-based PID controller design for a nonlinear system. Hassan and Mobayen [9] have proposed genetic-algorithm (GA-), PSO- and ACO-based controller design for a rotary inverted pendulum. Kanth and Latha [10, 11] have presented a relative work with PSO-, improved PSO- and parallel PSO-based PID tuning for a class of unstable process models.

In this paper, bacteria foraging optimization- (BFO-) based PID tuning is proposed for a class of nonlinear process models. BFO algorithm is a nature inspired metaheuristic search technique, introduced by Passino [12] to design an adaptive controller for a tank liquid level control problem. It is a biologically inspired computation technique based on mimicking the foraging activities of *Escherichia coli* (*E. coli*) bacteria. In this algorithm, a collection of artificial bacteria cooperates to find the best possible solutions in the “ D ” dimensional search space during the optimization exploration. The previous research has reported the superiority of the BFO-based search for finding the optimum solution for a class of engineering problems.

Dasgupta et al. [13, 14] proposed an adaptive BFO (ABFO) algorithm to find optimal parameters for a variety of engineering optimization problems. Chen et al. [15, 16] proposed a cooperative BFO (CBFO) algorithm and ABFO algorithm. With a comparative study, they proved that the ABFO provides superior performance compared to PSO and GA. Das et al. [17, 18] provided the comprehensive analysis on the BFO algorithm. The discussion by Biswas et al. [19] provides the insight on the reproduction operator performance in BFO. A detailed study on the optimal controller parameter tuning for a class of process models are examined by Biswas et al. [20], Roy et al. [21], and Ghosh et al. [22]. Ali and Majhi [23] examined the BFO algorithm to tune the PID controller parameter for a class of stable process models. Korani et al. [24] presented a comparative study between the PSO, BFO, and hybrid-algorithm-tuned PID control for stable systems. Recently, Kanth and Latha [25] discussed about the BFO-tuned I-PD controller performance on a class of time delayed unstable process models.

In the present work, we propose an enhanced bacteria foraging optimization (EBFO) algorithm to identify optimised PID controller parameters for a class of stable and unstable process models by maintaining guaranteed accuracy in the optimized value. The need for multiobjective performance index in order to improve the exactness of the proposed controller is also discussed. The method is finally tested on a real time nonlinear spherical tank system.

Further, a detailed description of enhanced bacterial foraging optimization algorithm is provided in Section 2.

Section 3 presents the outline of the PID controller structure, problem formulation and the cost-function-based controller tuning. Section 4 discusses the simulated results on different nonlinear process models and the real-time test on a nonlinear spherical tank system. Section 5 deals about the conclusion of the present research work.

2. Bacteria Foraging Algorithm

Bacteria foraging optimization (BFO) algorithm is a new division of metaheuristic algorithm. It is a population-based optimization technique developed by inspiring the foraging manners of *E. coli* bacteria [12]. The basic operations of BFO algorithm is briefly discussed below.

Chemotaxis. During foraging operation (tracing, handling, and ingesting food), an *E. coli* bacterium moves towards the food location with the aid of swimming and tumbling by using 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 operations are 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 in the direction of the best food position, the bacterium which has the knowledge about the optimum path to the food source 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 equation:

$$J(\theta, D(j, k, l)) = \sum_{i=1}^N J_{cc}(\theta, \theta^i(j, k, l)) = A + B, \quad (1)$$

where

$$A = \sum_{i=1}^N \left[-d_{\text{attract}} \exp \left(-W_{\text{attract}} \sum_{m=1}^D (\theta_m - \theta_m^i)^2 \right) \right], \quad (2)$$

$$B = \sum_{i=1}^N \left[h_{\text{repell}} \exp \left(-W_{\text{repell}} \sum_{m=1}^D (\theta_m - \theta_m^i)^2 \right) \right],$$

where θ is the location of the global optimum bacterium till the j th chemotactic, k th reproduction, and l th elimination stage and “ θ_m ” is the m th parameter of global optimum bacteria.

Where $J(\theta, D(j, k, l))$ represents objective function assessment, “ N ” is the total number of bacterium, and “ D ” the total parameters to be optimised. The other parameters such as “ d_{attract} ” are the depth of attractant signal released by a bacteria and “ W_{attract} ” is the width of attractant signal. The signals “ h_{repell} ” and “ W_{repell} ” are the height and width of repellent signals between bacterium (attractant is the signal for food source and repellent is the signal for noxious reserve).

Reproduction. In swarming process, the bacteria build up as groups in the positive nutrient gradient and which may

increase the bacterial concentration. After the congregation the bacteria are sorted in descending order based on its health values. The bacteria which have the least health will perish 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 accessibility of food, the population of a bacteria may change either steadily or abruptly. 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 “ D ” dimensional search space. The dispersal possibly flattens the chemotaxis 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 procedures are repeated until the optimized solutions are achieved.

2.1. Enhanced Bacteria Foraging Algorithm. The parameters of the basic BFO algorithms are defined in the following. D : the dimension of search space (the search boundary is $-100 < 0 < +100$), N : the total number of artificial *E. coli* bacteria, N_c : total number of chemotaxis steps, N_s : swim length during the search, N_{re} : total number of reproduction steps, N_{ed} : total number of elimination-dispersal events, N_r : number of reproduced bacteria, P_{ed} : the probability that each bacterium will be eliminated/dispersed, and n : the run length.

In the basic BFO algorithm, the fitness of each bacterium is determined from the average value of the entire chemotactic performance index before the reproduction operation. In the proposed EBFO algorithm, the bacterium with the maximum health is retained [12].

The health of the bacterium can be found by the following relation

$$J_{\text{health}}^i = \sum_{j=1}^{N_c+1} J(i, j, k, l), \quad (3)$$

where $i = 1, 2, \dots, N$.

In the proposed algorithm, the retained bacterium is used to guide the reproduced bacteria towards the nutrient source. Due to this process, along with the accuracy in optimization, the iteration time can be reduced.

In the literature there is no apparent guide line to allocate the parameters for the BFO algorithm. In the proposed EBFO algorithm, we assigned the limitations for the algorithm parameters by considering the various stages of bacterium growth discussed in the book by El-Mansi and Bryce [26].

Stages of bacteria growth in a controlled environment are shown schematically in Figure 1.

- (i) *Lag phase:* Amendment of the cells to new environment take place and it is getting ready to begin reproduction.
- (ii) *Growth phase:* In this stage with the help of chemotaxis and swarming practice, the cells can reach the

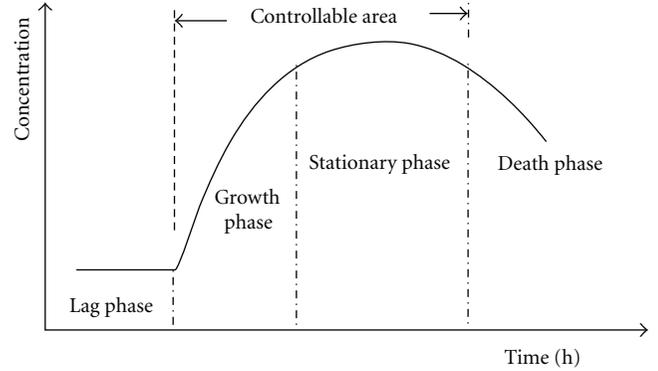


FIGURE 1: Stages of cell growth in an incubated environment.

location of food source. The growth rate is proportional to the cell concentration and the nutrient quantity. When the cell reaches the sufficient food location, the growth rate is rapid. When the cell reaches the maximum growth, it begins reproduction.

- (iii) *Stationary phase:* After growth and reproduction, the cell will reach a minimum biological space called stationary phase. Due to the lack of one or more nutrients, build up of toxic materials and organic acids generated during the growth phase, cell growth is restricted.

- (iv) *Death phase:* It is mainly due to the toxic by-products and depletion of nutrient supply. In this, a decrease in live cell concentration occurs. The cell with a minimum health is eliminated.

The above process repeats until there exist a controlled environment such as constant temperature and pH.

In bacteria foraging algorithm, the total number of bacterium considered for the optimization practice plays a vital role in maintaining the optimization accuracy and algorithm convergence. The larger number of bacterium can offer an agreeable accuracy, but sometimes it may increase the computation time. In this paper, we performed a number of trials to fix the range of the bacteria group size. When the *E. coli* is placed in a controlled environment, it will reach the food source with the action of tumbling or swimming. The first half in the group swim towards the food and rest in the group tumbles. The bacterium which enters into the nutrient environment first, may grow earlier and starts the reproduction operation. Around 25% of cells may die due to lack of nutrients and build up toxic materials. The probability of bacterial elimination mainly depends on the bacteria at the noxious environment, initial population of the bacteria, and the bacterial with the reproduction process. The bacteria are living organism, which will act fast at toxic environment compared to the nutrient source. This process may help to fix the values for the attract and repel signal strength.

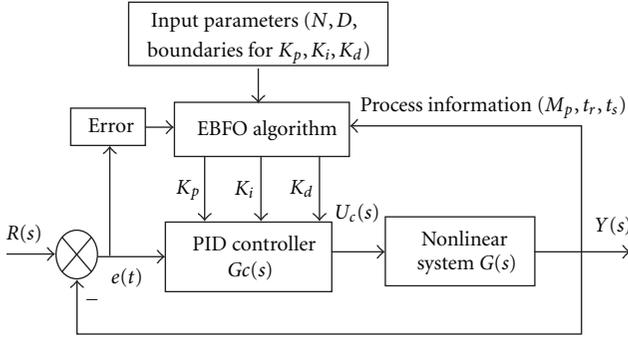


FIGURE 3: EBFO algorithm-based PID controller tuning.

The following performance criterion (9) with five parameters, such as ISE, IAE, M_p , t_r , and t_s are considered in this work:

$$J(K_p, K_i, K_d) = (w_1 \cdot \text{IAE}) + (w_2 \cdot \text{ISE}) + (w_3 \cdot M_p) + (w_4 \cdot t_s) + (w_5 \cdot t_r), \quad (10)$$

where

$$\begin{aligned} \text{IAE} &= \int_0^T |e(t)| dt = \int_0^{100} |r(t) - y(t)| dt, \\ \text{ISE} &= \int_0^T e^2(t) dt = \int_0^{100} [r(t) - y(t)]^2 dt, \\ M_p &= y(t) - r(t), \end{aligned} \quad (11)$$

t_r = rise time (time required for $y(t)$ to reach 100% of its setpoint at the first instant), t_s = settling time—time required for $y(t)$ to reach an stay $\text{atr}(t)$ [i.e., $y(t) = r(t)$], and T = time considered for error calculation.

Where w_1, w_2, \dots, w_5 are weighting functions used to set the priority of the MOPI parameters and the value of “ w ” varies from 0 to 10. The performance criterion $J(K_p, K_i, K_d)$ guides the EBFO algorithm to get appropriate values for the controller parameters.

3.2. Optimization Search. Prior to the optimization search, it is necessary to assign the parameters for EBSO and MOPI.

Figure 3 shows the basic block diagram of the EBFO-based PID controller tuning.

In this study, the following values are assigned.

- (i) Dimension of the search space (D) = 3 (i.e., K_p, K_i, K_d).
- (ii) The total number of *E. coli* bacteria = 10.
- (iii) Boundaries for the three dimensional search space is assigned as

$$\begin{aligned} \text{Value 1} &= -25\% < K_p < +50\% \quad (\text{i.e., } -2.5 < K_p < 5.0) \\ \text{Value 2} &= -20\% < K_i < +20\% \quad (\text{i.e., } -2.0 < K_i < 2.0) \\ \text{Value 3} &= -20\% < K_d < +30\% \quad (\text{i.e., } -2.0 < K_d < 3.0) \end{aligned} \quad (12)$$

- (iv) The weighting function values are assigned as $w_1 = w_2 = w_3 = 10, w_4 = w_5 = 5$.
- (v) Maximum simulation time is 100 sec.
- (vi) The “ t_r ” is preferred as <25% of the maximum simulation time. The simulation time should be selected based on the process time delay.
- (vii) The overshoot (M_p) range is selected as <100% of the reference signal.
- (viii) The “ t_s ” is preferred as <50% of the maximum simulation time.
- (ix) The reference signal is considered as unity (i.e., $R(s) = 1$).
- (x) For each process example, five trials are carried out and the finest set of values among the trials is selected as the best optimized controller value.

3.3. Comparative Study. In order to evaluate the performance of the proposed EBFO algorithm, a comparative analysis is done with most successful soft computing methods such as PSO, BFO, adaptive BFO (ABFO), and GA.

PSO. The simulation is carried out by using the PSO algorithm attempted by Kanth and Latha [10, 11]. The following algorithm parameters are considered: dimension of search space is three (i.e., K_p, K_i, K_d); number of swarm and bird step is considered as 25; the cognitive (C_1) and global (C_2) search parameter is assigned the value of 2 and 1.5, respectively, the inertia weight “ W ” is set as 0.7.

BFO. For the basic BFO algorithm, the following values are considered: dimension of search space is three; number of bacteria is chosen as ten; number of chemotaxis step is set to five; number of reproduction steps and length of a swim is considered as four; number of elimination-dispersal events is two; number of bacteria reproduction is assigned as five; probability for elimination-dispersal has a value of 0.2 [10, 11, 25].

ABFO. The following values are considered for the ABFO algorithm: dimension of search space is three; number of bacteria (S) = 100, number of chemotaxis step (N_c) = 100, $N_s = 12, N_{ed} = 4, N_{re} = 16, P_{ed} = 0.25, d_{attractant} = 0.1, W_{attractant} = 0.2, h_{repellant} = 0.1, W_{repellant} = 10$, and $\lambda = 400$ [13].

GA. In the GA-based search, the following parameters are assigned: population size is set to 20, generation size is chosen as 150, crossover probability is selected as 50%, and mutation probability is set as 0.2%. Roulette wheel based selection criterion is considered in this study.

*The MOPI proposed in this paper is utilised for all the evolutionary algorithms.

TABLE 1: PID controller values of EBFO algorithm and the performance index values for bioreactor model (five trials).

Trial	Iteration	K_p	K_i	K_d	IAE	ISE	M_p	t_r	t_s
1	51	-0.9330	-0.0171	-0.0628	8.828	77.93	0.721	1.97	62.5
2	57	-0.6289	-0.0672	-0.0386	2.537	6.439	0.718	2.39	27.6
3	67	-0.4418	-0.0503	-0.0714	3.390	11.49	0.775	2.95	30.3
4	72	-0.5928	-0.0730	-0.0288	2.336	5.456	0.759	2.48	18.1
5	59	-0.7117	-0.0391	-0.0433	4.359	19.00	0.638	2.3	38.2

4. Results and Discussions

Most of the real time chemical process systems exhibit multiple steady states due to nonlinear phenomena occurring in the systems. Around the operating region, such systems are adequately represented by a stable or unstable process model with a delay [4]. In order to show the efficiency of the EBFO tuned PID controller for the nonlinear system, approximated linear chemical (bioreactor and CSTR) process models are considered from the literature.

4.1. Process Model 1. The proposed PID controller is tested on a nonlinear bioreactor model discussed by Wayne-Bequette [28].

The following mathematical equations can describe a variety of industrial bioreactors. Equations (13)–(15) describes the balancing conditions and (16) depicts the specific growth rate.

Cell balance:

$$\frac{dX}{dt} = (\mu - D_s)X, \quad (13)$$

Substrate balance:

$$\frac{dS}{dt} = D_s(S_f - S) - \frac{\mu X}{Y}, \quad (14)$$

Product balance:

$$\frac{dP}{dt} = -D_s P + (\alpha\mu + \beta)X, \quad (15)$$

Monod kinetics:

$$\mu = \frac{\mu_{\max} S}{K_m + S + K_I S^2}, \quad (16)$$

For substrate inhibition model, the following parameters are considered [28]: $\mu_{\max} = 0.53 \text{ hr}^{-1}$, $K_m = 0.12 \text{ g/lit}$, $K_I = 0.4545 \text{ lit/g}$, $Y = 0.4$. The steady state dilution rate is $D_s = 0.3 \text{ h}^{-1}$ and the feed substrate concentration is $S_{\text{fss}} = 4.0 \text{ g/lit}$. The nonlinear process has the three steady state operating points for a dilution rate of 0.3 h^{-1} . Transfer function model can be obtained by applying linearization technique [4].

In this study a benchmark unstable bioreactor model is considered. The dilution rate is taken as the manipulated variable to control the cell mass concentration.

For the unstable operating point, the linearized model for the unstable bioreactor is

$$G_p(s) = \frac{-0.9951s - 0.2985}{s^2 + 0.1302s - 0.0509} = \frac{-5.8644}{5.89s - 1}. \quad (17)$$

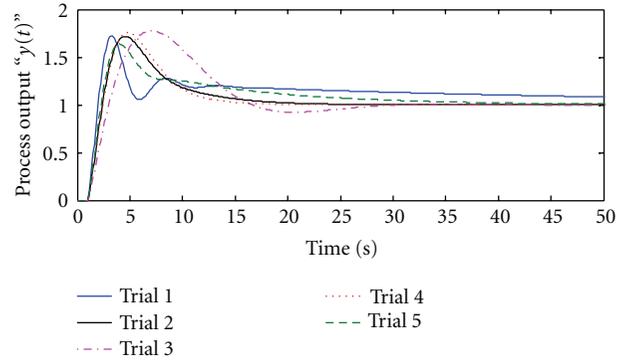


FIGURE 4: Servo response of bioreactor model with EBFO-based PID controller.

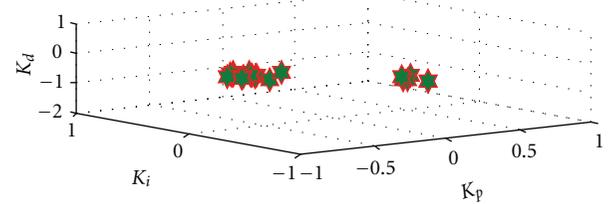


FIGURE 5: Final converged bacteria positions in three-dimensional space.

First part of (17) represents a second order model and the later part shows a reduced first order model. The delay time for both the model is considered as "1."

The EBFO-based PID controller tuning is proposed for the second order model as in Figure 2.

Five trials are performed during the optimization search. The convergence of the EBFO algorithm toward the global optimal solution of the controller parameters are presented in Table 1. Figure 4 depicts the qualitative comparison of the servo response for the trial values presented in Table 1. Among them, Trial 2 value shows the better result compared to other trial values. It also satisfies most of the performance criterion compared to other trial values (Table 1).

The position of the *E. coli* bacteria in the three dimensional search space for Trial 2 is depicted in Figure 5. In this search operation, the artificial bacterium finds the best possible controller parameters by minimizing the MOPI.

Figure 6 graphically represents the optimised K_p , K_i , and K_d values for Trial 2. The search value is converging at 57th iteration.

TABLE 2: Controller values and the performance comparison for bioreactor model.

Method	Iteration	K_p	K_i	K_d	IAE	ISE	M_p	t_r	t_s
GA	63	-0.9104	-0.0655	-0.0428	2.603	6.776	0.838	1.84	51.9
PSO	71	-0.4066	-0.0501	-0.1197	3.404	11.58	0.808	3.16	43.1
BFO	85	-0.5374	-0.0702	-0.0537	2.429	5.900	0.751	2.57	27.0
ABFO	66	-0.6113	-0.0714	-0.0518	2.388	5.704	0.720	2.40	28.3
EBFO	57	-0.6289	-0.0672	-0.0386	2.537	6.439	0.718	2.39	27.6

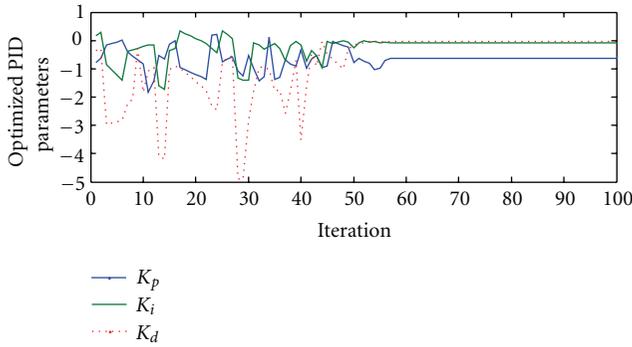


FIGURE 6: Final convergence of controller parameters for Trial 2.

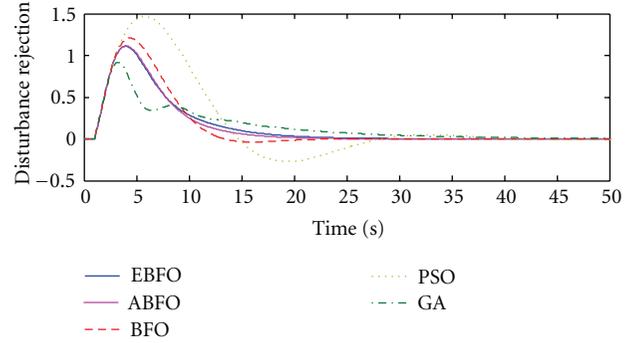


FIGURE 8: Disturbance rejection response of bioreactor model.

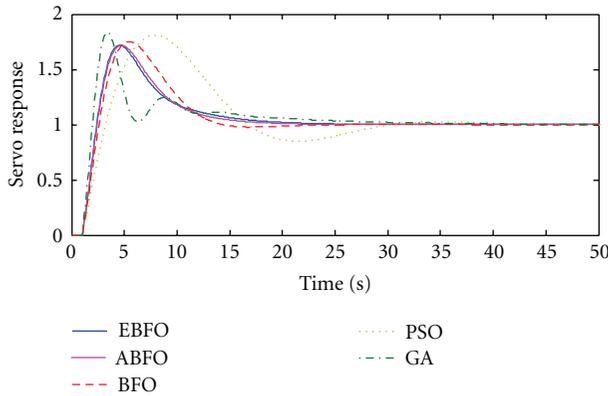


FIGURE 7: Servo response of bioreactor model with EBFO, ABFO, BFO, PSO, and GA tuned PID controller.

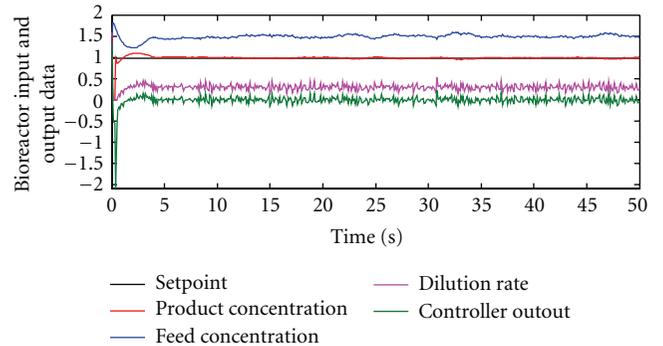


FIGURE 9: Variation of Biomass concentration and substrate concentration with respect to noise.

After finding the optimized controller value using EBSO algorithm; ABFO, classical BFO, PSO, and GA based PID controller tuning is attempted for the unstable bioreactor model. For each algorithm, five trials are performed and the best value among the trial is tabulated in Table 2. From Table 2, it is observed that the number of iteration of EBFO is smaller compared to other algorithms. The ISE and IAE value by the ABFO algorithm is lesser, but the other parameters such as M_p , t_r , and t_s are greater compared to EBFO algorithm.

From Table 2 and Figure 7 values show that, the overall performance of EBFO and ABFO are approximately similar. Disturbance rejection performance of the above methods are tested by applying a load disturbance of 50% (i.e., $D(s) = 0.5$). From Figure 8, the observation is that, the disturbance rejection performance of EBFO and ABFO are identical.

The performance of the EBFO tuned PID controller is tested on the nonlinear bioreactor model developed using the non linear equations (13)–(16). The objective is to maintain the concentration of biomass/product (S_B) based on the setpoint, by adjusting the substrate/feed concentration (S_S). In order to test the robustness of the proposed controller, measurement noise (noise power of 0.001 with a sampling time of 0.1 sec) is introduced in the feedback loop. A feed dilution rate of 0.4 g/lit is considered in this study.

Figure 9 shows the variation of biomass concentration, substrate concentration, dilution rate and the controller output for the servo response. In this a setpoint of 0.995103 g/lit is considered for the biomass concentration. The response confirms that, the proposed scheme works well even in the noisy environment and helps to provide a smooth reference tracking performance.

TABLE 3: Controller values and the performance comparison for isothermal CSTR.

Method	Iteration	K_p	K_i	K_d	IAE	ISE	M_p	t_r	t_s
GA	138	1.1507	0.0074	4.9331	40.51	1641	0.741	42.2	423.7
PSO	144	1.2804	0.0207	8.0426	14.56	212.1	1.015	38.6	305.1
BFO	169	1.4538	0.0111	5.9842	27.01	729.5	0.905	37.7	351.4
ABFO	216	1.2660	0.0205	8.5283	14.68	215.5	1.034	38.9	284.8
EBFO	151	1.3882	0.0155	9.0343	19.42	377.0	0.843	37.2	177.3

4.2. *Process Model 2.* Isothermal Continuous Stirred Tank Reactor (CSTR) considered by Liou and Yu-Shu [29] has the following transfer function model:

$$\frac{dc}{dt} = \frac{nQ}{mV}(C_f - C) - \left[\frac{K_1 C}{(K_2 C + 1)^2} \right]. \quad (18)$$

From (18), it can be observed that, the differential equation which represents the concentration of the product with respect to time is nonlinear.

The parameter values of the CSTR are given by [3]: flow rate (Q) = 0.03333 lit/sec; volume (V) = 1 lit, $K_1 = 10$ lit/s; and $K_2 = 10$ lit/mol, $n = m = 0.75$. Linearizing the nonlinear model equation around the operating region with, concentration (C_f) = 3.288 mol/lit, gives two stable steady states at $C = 1.7673$ mol/lit and $C = 0.01424$ mol/lit. When $C = 1.304$ mol/lit, the CSTR provides an unstable steady state and it can be mathematically represented by the following unstable transfer function model (with a measurement delay of 20 sec):

$$G(s) = \frac{\Delta C(s)}{\Delta C_f(s)} = \frac{3.3226 \exp^{-20s}}{(99.69s - 1)}. \quad (19)$$

Equation (19) depicts a first order model and the controller setting for this model is proposed as discussed in Section 3.2 with the following values.

- (i) The total number of *E. coli* bacteria = 20 (same value is assigned for the agents in BFO, PSO, and GA)
- (ii) The numerator of the transfer function has a positive sign (i.e., +3.3266), since the lower boundary of the controller parameter search is assigned as zero. The three dimensional search space is assigned as follows:

$$\begin{aligned} \text{Value 1} &= -0\% < K_p < +20\% \quad (\text{i.e., } 0 < K_p < 2.0) \\ \text{Value 2} &= -0\% < K_i < +10\% \quad (\text{i.e., } 0 < K_i < 1.0) \\ \text{Value 3} &= -0\% < K_d < +100\% \quad (\text{i.e., } 0 < K_d < 10.0) \end{aligned} \quad (20)$$

The delay time of the process is 20 sec. Since the maximum simulation time is selected as 500 sec.

Five trials are performed using the evolutionary algorithms and the best value among the trial is tabulated in Table 3. Table 3 shows the optimized controller values and its performance measure for the isothermal CSTR model. Even though the iteration value of EBFO algorithm is large compared to the GA- and PSO-based methods, other

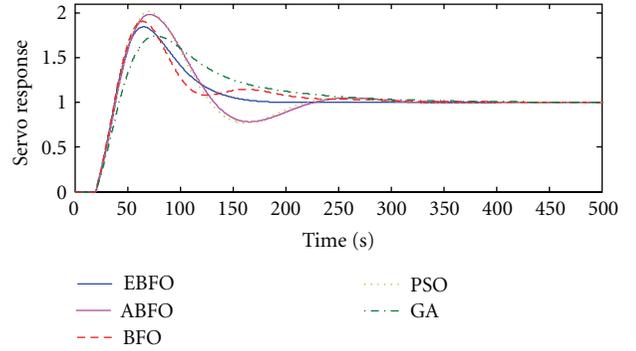


FIGURE 10: Servo response of CSTR model with EBFO, BFO, PSO, and GA tuned PID controller.

parameters such as ISE, IAE, t_r , and t_s are lesser than other algorithms. The ISE and IAE value by the ABFO algorithm is smaller compared to GA, BFO, and EBFO. From Figure 10, the observation is that, the EBFO-based method provides significantly improved result compared with BFO, PSO, and GA.

4.3. *Process Model 3.* Continuous stirred tank reactor (CSTR) with nonideal mixing considered by Liou and Yu-Shu [29] is considered in this study. Linearizing the nonlinear model equation around the operating region with, concentration (C_f) = 3.288 mol/lit, $C_e = 1.8$ mol/lit and $C = 1.304$ mol/lit gives the following unstable transfer function model:

$$G(s) = \frac{\Delta C_e(s)}{\Delta C_f(s)} = \frac{2.21(1 + 11.133s)\exp^{-20s}}{(98.3s - 1)}. \quad (21)$$

The process model has one unstable pole and a stable zero. The time delay " θ " in the system is considered as 20 sec. The unstable system with a zero may produce a large overshoot or inverse response. Since during the optimization search, the boundary for the overshoot " M_p " is disconnected. For this process, the simulation study is performed as discussed in Section 3.2 with a simulation time of 500 sec.

Five trials are performed on this process model for each evolutionary algorithm and the best value among the trial is tabulated in Table 4. Figure 11 shows the servo response of the CSTR model with EBFO, ABFO, BFO, PSO, and GA tuned PID controller. Even though the overshoot is large compared to other methods; the overall performance of the present tuning method provides a better result with significantly reduced performance criterion values.

TABLE 4: Controller values and the performance comparison for process model 3.

Method	Iteration	K_p	K_i	K_d	IAE	ISE	M_p	t_r	t_s
GA	331	0.1970	0.0009	0.5109	41.32	1707	0.991	36.5	647.2
PSO	295	0.1270	0.0012	0.5338	31.08	965.9	0.869	43.7	406.1
BFO	324	0.1790	0.0019	0.3941	19.6	385.3	1.190	37.2	370.4
ABFO	360	0.1135	0.0014	0.5211	26.64	709.7	0.976	37.3	371.3
EBFO	319	0.2108	0.0025	0.8100	14.92	222.5	1.243	34.9	335.9

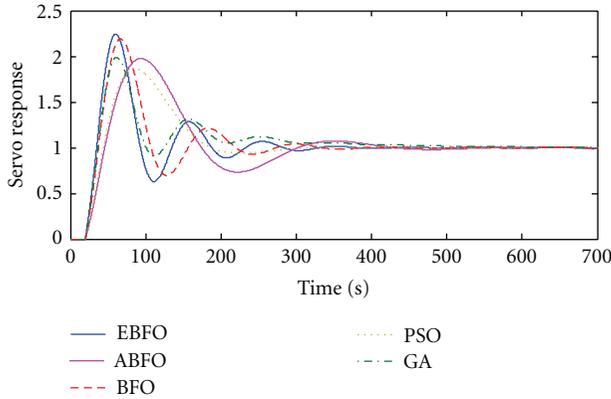


FIGURE 11: Servo response of process model 3.

4.4. Real-Time Implementation. In this work, to evaluate the effectiveness of the proposed algorithm, a PID controlled nonlinear spherical tank system is considered. The spherical tank setup shown in Figure 12 is an image of a liquid storage structure widely used in oil and gas industries.

Modelling and control of nonlinear spherical tank system was widely addressed by the researchers. Faccin and Trierweiler [30] proposed a multimodel PID controller design for a spherical tank system. They also proposed a simple model identification technique using the first principle analysis. Madhavasarma and Sundaram [31] proposed a model-based controller tuning for the nonlinear spherical tank system. They developed an approximate stable first order plus dead time (FOPDT) model around the operating region and proposed an IMCPID controller. Nithya et al. [32] proposed a black box system identification technique and developed a FOPDT model. GA-tuned fuzzy logic controller (FLC) was proposed for the identified stable FOPDT model. Figure 12 shows the real time experimental setup of the spherical tank system and its specifications. The objective is to maintain the fluid level (h) by adjusting the inlet flow rate (F_{in}).

The first principle modelling equations for the tank is given below:

$$\begin{aligned} \frac{dV}{dt} &= F_{in} - F_{out}, \\ A \frac{dh}{dt} &= F_{in} - F_{out}, \\ \frac{dh}{dt} &= \frac{F_{in} - \beta \sqrt{h}}{\Pi * h * (D_t - h)}, \end{aligned} \quad (22)$$

Nonlinear process loop (spherical tank)



Specification

Tank diameter: 50 cm Rotameter : 100–1000 lph (1.667–16.67 lpm)
 I / P converter: ABB—model TEIP 11.
 Input: 4–20 mA and output: 3–15 psi
 DPT : Rosemount make HART enabled system, output: 4–20 mA
 Control valve with smart positioner: current: 4–20 mA and working pressure : 19.9199.56 psi
 Interfacing unit: VI-microsystems make MATLAB compatible, VISA enabled, serial DAQ system

FIGURE 12: Real-time spherical tank setup and its specifications.

where F_{in} = inlet flow rate, F_{out} = outlet flow rate, V = tank volume, A = area of tank, h = head, D_t = diameter of the tank based on the head, and β = outlet flow capacity coefficient.

The linearised transfer function of the system around the operating point can be developed by neglecting the wall thickness of the tank.

In this setup, a personal computer (PC) loaded with the MATLAB software allows the user to monitor and control the working process. The DAQ supports 4 analog input (voltage in the range of 1 to 5V or current in the range of 4 to 20 mA), 4 analog/digital input and 4 analog/digital output channels. A communication link between the process loop and the monitoring PC is established by the DAQ module through Universal Serial Bus (USB).

A level of 18 cm is considered as the operating point. To develop the transfer function around this operating

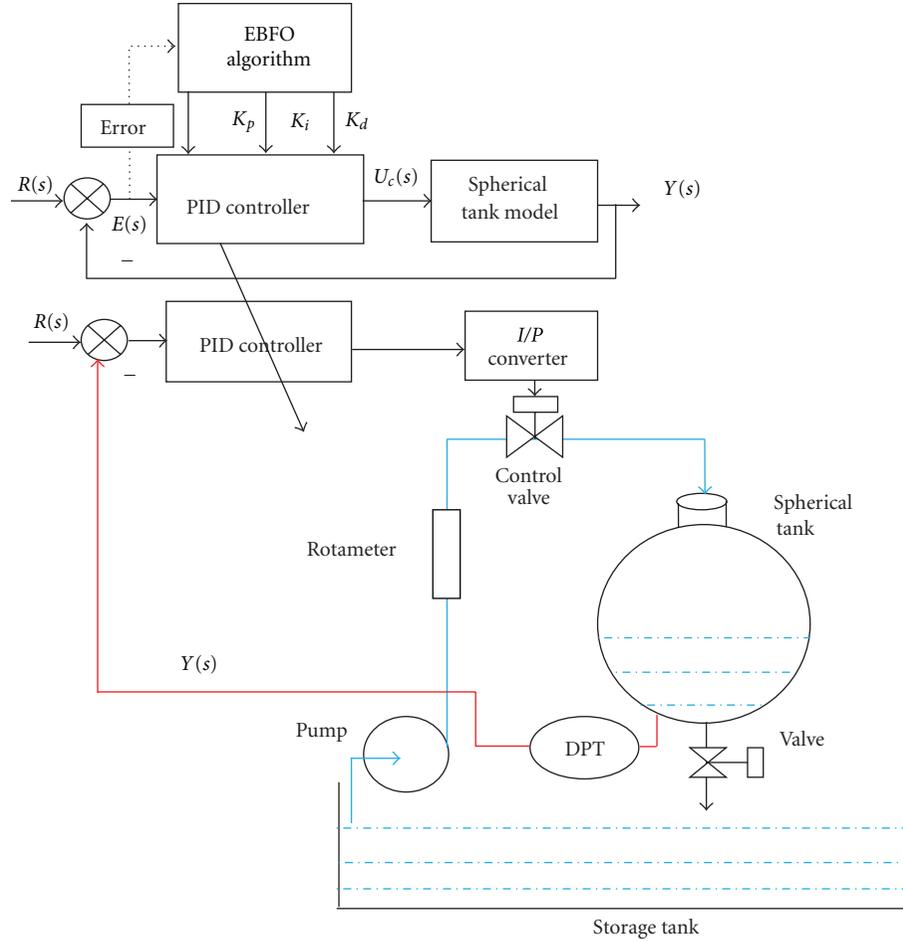


FIGURE 13: EBFO-based PID tuning and its implementation on the real time spherical tank.

point, open-loop step response test is performed with an inlet flow rate $F_{in} = 9$ lpm. At this operating region, 29.549% of tank is filled with the liquid (19.34 lit) and 70.451% is with the air, the model identified around this operating point using MATLAB software is given in the following:

$$G_m(s) = \frac{3.6215e^{-\theta s}}{330.46s + 1} = \frac{3.6215e^{-11.7s}}{330.46s + 1}. \quad (23)$$

With the experimental result, the delay by the control valve and the DPT level transmitter is accounted as 11.7 sec.

Figure 13 shows the EBFO-based PID controller tuning procedure and its real time implementation on the spherical tank level control problem. The EBFO algorithm proposed in Section 3.2 is considered for the search. Initially the search is performed on the tank model using (23).

A simulation time of 500 sec is considered in the EBFO-based PID tuning procedure and the algorithm is converged at 57th iteration with the following values:

$$K_p = 7.2074, \quad K_i = 1.1603, \quad K_d = 2.0441. \quad (24)$$

Initially, the controller parameter optimization is searched with the process model, and later the identified PID values are transferred to the real time controller installed in the process loop through the monitoring and control program developed in MATLAB Simulink with ODE 45 solver. The Simulink program is directly interfaced with the real time process system through DAQ module. It is enabled with National Instruments VISA serial communication interface. The module supports ASCII data format with a sampling time of 0.1 sec and a baudrate of 38400. With this, Monitoring and control of the real-time process can be easily established with MATLAB software. In real time implementation, the maximum controller output is set as 90% in order to reduce the thrust on the control valve which allows the flow rate to the tank.

Figure 14 shows the variation of level based on the setpoint and the corresponding controller output during the real time study. The ISE and IAE values are obtained as 491.34 and 215.93, respectively, during the real time study.

From this study, it can be noted that the proposed EBFO algorithm presents a smooth servo response for the spherical tank level control problem and it can be easily implemented in real time using a MATLAB supported real time process loop.

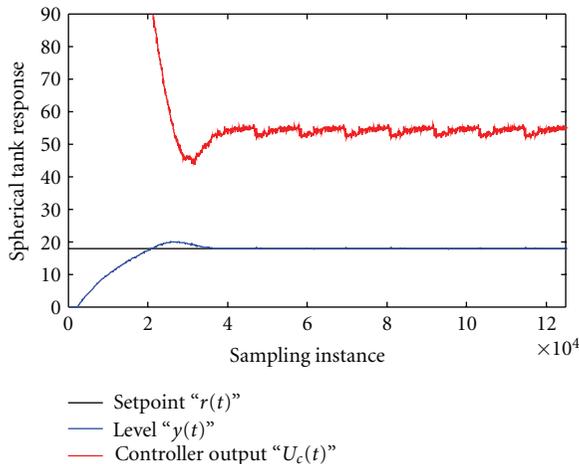


FIGURE 14: Response of spherical tank system for a setpoint of 18 cm (real-time data).

5. Conclusion

In this work, we proposed an enhanced bacterial foraging optimization (EBFO) algorithm to tune the parallel form of PID controller for a class of nonlinear process models. In order to minimize the algorithm complexity, guidelines are provided to select the BFO algorithm parameters. A comparative study is performed with the basic BFO-, ABFO-, PSO-, and GA-based PID controller tuning methods proposed in the literature. The study confirms that the EBFO-tuned PID controller provides improved overall performance compared to other algorithms considered for study. Finally the proposed algorithm is implemented in real time for a spherical tank level control problem. The real time result shows that, the EBFO-tuned PID controller gives a smooth response for reference tracking and maintains the level based on the reference signal.

References

- [1] R. A. Maher and R. Samir, "Robust stability of a class of unstable systems under mixed uncertainty," *Journal of Control Science and Engineering*, vol. 2011, no. 970962, 8 pages, 2011.
- [2] V. Vijayan and R. C. Panda, "Design of a simple setpoint filter for minimizing overshoot for low order processes," *ISA Transactions*, vol. 51, no. 2, pp. 271–276, 2012.
- [3] R. Padma Sree and M. Chidambaram, *Control of Unstable Systems*, Narosa Publishing House, New Delhi, India, 2006.
- [4] M. K. Jhunjhunwala and M. Chidambaram, "Pid controller tuning for unstable systems by optimization method," *Chemical Engineering Communications*, vol. 185, no. 1, pp. 91–113, 2001.
- [5] I. Chiha, N. Liouane, and P. Borne, "Tuning PID controller using multiobjective ant colony optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2012, Article ID 536326, 7 pages, 2012.
- [6] M. Zamani, N. Sadati, and M. K. Ghartemani, "Design of an H^∞ PID controller using particle swarm optimization," *International Journal of Control, Automation and Systems*, vol. 7, no. 2, pp. 273–280, 2009.
- [7] M. Zamani, M. Karimi-Ghartemani, N. Sadati, and M. Parniani, "Design of a fractional order PID controller for an AVR using particle swarm optimization," *Control Engineering Practice*, vol. 17, no. 12, pp. 1380–1387, 2009.
- [8] W. D. Chang and S. P. Shih, "PID controller design of nonlinear systems using an improved particle swarm optimization approach," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 11, pp. 3632–3639, 2010.
- [9] I. Zadeh Hassan and S. Mobayen, "Controller design for rotary inverted pendulum system using evolutionary algorithms," *Mathematical Problems in Engineering*, vol. 2011, Article ID 572424, 17 pages, 2011.
- [10] V. Kanth and K. Latha, "Bacterial foraging optimization algorithm based pid controller tuning for time delayed unstable systems," *Mediterranean Journal of Measurement and Control*, vol. 7, no. 1, pp. 197–203, 2011.
- [11] V. Kanth and K. Latha, "Optimization of PID controller parameters for unstable chemical systems using soft computing technique," *International Review of Chemical Engineering*, vol. 3, no. 3, pp. 350–358, 2011.
- [12] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [13] S. Dasgupta, S. Das, A. Abraham, and A. Biswas, "Adaptive computational chemotaxis in bacterial foraging optimization: an analysis," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 919–941, 2009.
- [14] S. Dasgupta, S. Das, A. Biswas, and A. Abraham, "Automatic circle detection on digital images with an adaptive bacterial foraging algorithm," *Soft Computing*, vol. 14, no. 11, pp. 1151–1164, 2010.
- [15] H. Chen, Y. Zhu, and K. Hu, "Cooperative bacterial foraging optimization," *Discrete Dynamics in Nature and Society*, vol. 2009, Article ID 815247, 17 pages, 2009.
- [16] H. Chen, Y. Zhu, and K. Hu, "Adaptive bacterial foraging optimization," *Abstract and Applied Analysis*, vol. 2011, Article ID 108269, 27 pages, 2011.
- [17] S. Das, S. Dasgupta, A. Biswas, A. Abraham, and A. Konar, "On stability of the chemotactic dynamics in bacterial-foraging optimization algorithm," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 39, no. 3, pp. 670–679, 2009.
- [18] S. Das, A. Biswas, S. Dasgupta, and A. Abraham, "Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications," *Studies in Computational Intelligence*, vol. 203, pp. 23–55, 2009.
- [19] A. Biswas, S. Das, A. Abraham, and S. Dasgupta, "Stability analysis of the reproduction operator in bacterial foraging optimization," *Theoretical Computer Science*, vol. 411, no. 21, pp. 2127–2139, 2010.
- [20] A. Biswas, S. Das, A. Abraham, and S. Dasgupta, "Design of fractional-order $PI^{\lambda}D^{\mu}$ controllers with an improved differential evolution," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 2, pp. 343–350, 2009.
- [21] G. G. Roy, P. Chakraborty, and S. Das, "Designing fractional-order $PI^{\lambda}D^{\mu}$ controller using differential harmony search algorithm," *International Journal of Bio-Inspired Computation*, vol. 2, no. 5, pp. 303–309, 2010.
- [22] A. Ghosh, S. Das, A. Chowdhury, and R. Giri, "An ecologically inspired direct search method for solving optimal control problems with Bézier parameterization," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 7, pp. 1195–1203, 2011.

- [23] A. Ali and S. Majhi, "Design of optimum PID controller by bacterial foraging strategy," in *Proceedings of IEEE International Conference on Industrial Technology (ICIT '06)*, pp. 601–605, December 2006.
- [24] W. M. Korani, H. T. Dorrah, and H. M. Emara, "Bacterial foraging oriented by particle swarm optimization strategy for PID tuning," in *Proceedings of the 8th IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '09)*, pp. 445–450, December 2009.
- [25] V. Kanth and K. Latha, "I-PD controller tuning for unstable system using bacterial foraging algorithm: a study based on various error criterion," *Applied computational Intelligence and Soft Computing*, vol. 2012, Article ID 329389, 10 pages, 2012.
- [26] E. M. T El-Mansi and C. F. A. Bryce, *Fermentation Microbiology and Biotechnology*, Taylor and Francis, Indian reprint, 2003.
- [27] M. A. Johnson and M. H. Moradi, *PID Control: New Identification and Design Methods*, Springer, London, UK, 2005.
- [28] B. Wayne-Bequette, *Process Control—Modeling, Design and Simulation*, Prentice-Hall, New Delhi, India, 2003.
- [29] C. T. Liou and C. Yu-Shu, "The effect of nonideal mixing on input multiplicity in a CSTR," *Chemical Engineering Science*, vol. 46, no. 8, pp. 2113–2116, 1991.
- [30] F. Faccin and J. O. Trierweiler, "A novel toll for multi-model PID controller design," in *Proceedings of the 7th IFAC symposium on Dynamics and Control of Process Systems*, pp. 251–256, 2004.
- [31] P. Madhavasarma and S. Sundaram, "Model based tuning of a non linear spherical tank process with time delay," *Instrumentation Science and Technology*, vol. 36, no. 4, pp. 420–431, 2008.
- [32] S. Nithya, N. Sivakumaran, T. Balasubramanian, and N. Anantharaman, "Design of controller for nonlinear process using soft computing," *Instrumentation Science and Technology*, vol. 36, no. 4, pp. 437–450, 2008.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

