

Research Article

Modeling, Analysis, and Intelligent Controller Tuning for a Bioreactor: A Simulation Study

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, Anna University, MIT Campus, Chennai 600 044, India

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

Received 24 October 2012; Accepted 8 November 2012

Academic Editors: A. Gil and M. E. R. Shanahan

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.

In this paper, a novel modeling technique has been attempted to develop the mathematical model for a bioreactor functioning at multiple operating regions. The first principle mathematical equations of the reactor are used with the POLYMATH software to generate essential data for the model development. A relative analysis is also carried out with the existing models in the literature. An optimal PID controller is then designed using a multiobjective particle swarm optimization algorithm. The controller tuning procedure is individually discussed for both the stable and unstable steady state regions. The controller tuned for each region is scheduled using a set-point scheduler to achieve a complete control over the bioreactor. The effectiveness of the proposed scheme has been confirmed through a comparative study with the controller tuning methods proposed in the literature. The results show that, the proposed method provides enhanced performance in effective reference tracking and load disturbance rejection with minimal ISE and IAE. Finally the proposed method is validated on the nonlinear bioreactor model in the presence of a measurement noise. The results testify that the PSO tuned PID performs well in tracking the change in biomass concentration at the entire operating region.

1. Introduction

Bioreactor plays a vital role in chemical process industries to produce important chemical and biochemical compounds. In this system, living organisms also known as microbes are converted into marketable products such as beverages, antibiotics, vaccines, and industrial solvents [1–3]. The quality of the final product from a bioreactor depends mainly on the control loop employed to monitor and control the microbial growth based on the reference input. Apart from this, incidental external and internal disturbances in a reactor may result in reactor failure. Therefore, there is a strong financial inspiration to develop a finest control scheme that would facilitate rapid startup and stabilization of continuous bioreactors subject to redundant disturbances [4].

In the literature, a variety of methods have been discussed to implement a robust controller for the bioreactor operating at single or multiple steady-states. Kumar et al.

have examined a bioreactor with input multiplicities. With an experimental study, the mathematical models for the different steady state operating regions are developed, and a nonlinear PI controller was implemented [5]. Sivakumaran et al. have discussed recurrent neural network (RNN) based modeling method for a nonlinear bioreactor operating at single steady state and implemented a nonlinear model predictive controller to obtain satisfactory result [6]. Nagy has proposed a neural network model-based predictive control (NNMPC) strategy for a fermentation bioreactor model, and with simulation result they concluded that the proposed NNMPC provides enhanced performance compared to linear MPC and PID controller [7]. Giriraj Kumar et al. have discussed a genetic-algorithm- (GA-) based PI controller tuning for a bioreactor operating at stable steady state [8]. Bioreactor operating at unstable steady state was widely discussed by the researchers due to its complexity and instability [9–11]. A detailed analysis to realize various model

based control configurations for continuous bioreactor was presented by Zhao and Skogestad [4].

In the control literature, regardless of the considerable progress in advanced process control proposals such as sliding mode control (SMC), model predictive control (MPC), and internal model control (IMC), PID controllers are still widely employed in industrial control systems because of their structural simplicity, reputation, robust behavior, and easy implementation. Along with the system's stability, it also satisfies chief performance such as smooth reference tracking, efficient disturbance rejection, and measurement noise attenuation criteria. Most of the PID tuning approaches proposed for stable and unstable system require numerical computations to identify the optimal controller parameters [12, 13].

Recently, soft computing approach-based PID tuning has attracted the researchers due to its ability to find optimized controller parameters with a minimized computation time. The particle swarm optimization (PSO) algorithm is one of the soft computing methods introduced by Kennedy and Eberhart [14]. In recent years, it is widely considered to find optimal solutions for various engineering optimization problems [15–22]. In this paper, PSO algorithm is primarily considered for PID controller parameter optimization because (i) is an autotuning method and does not require detailed mathematical description of the process under control, and (ii) very few parameters to assign compared to other evolutionary methods such as GA and bacterial foraging optimization (BFO) [23]. PSO-tuned optimal controller implementation for stable systems are reported by Korani et al. [24]. Chang and Shih proposed an improved PSO optimization algorithm to tune the PID controller for a nonlinear inverted pendulum system [15]. Zamani et al. have discussed a multiobjective PSO algorithm to tune the controller for automatic voltage regulator (AVR) problem [25]. They have proposed a multiobjective PSO algorithm to tune the H_∞ PID controller for a nonlinear system [26]. Banu and Uma have discussed a hybrid algorithm-based PID controller implementation for a nonlinear CSTR [27]. Kanth and Latha have attempted a PSO-based PID and I-PD controller-tuning for a class of unstable systems [28, 29]. The PSO-based controller provides better servo and regulatory responses than the classical PID and modified internal model controller (IMC). They also discussed a PSO-based controller implementation for a bioreactor operating at unstable operating region.

From the literature, it is observed that a nonlinear system with multiple steady states can be effectively controlled by a gain-scheduled control scheme [27, 30]. This control scheme consists of a family of local controller and a scheduler (switching unit). The switching unit activates the controller based on the set-point in order to achieve effective control over the entire operating region of the process.

The main contributions of the work are as follows. Initially, the first principle model has been developed using the POLYMATH software generated data. A multiple objective function-based PSO algorithm has been proposed for the PID controller parameter tuning. PID tuning for the stable and unstable operating region of the bioreactor

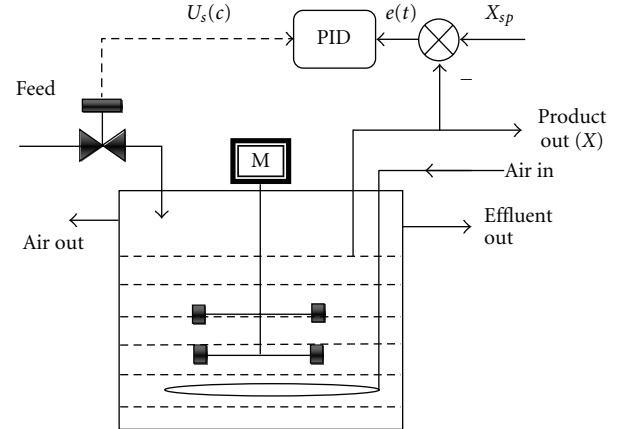


FIGURE 1: Schematic diagram of the bioreactor.

is separately addressed. The tuned PID controllers are gain scheduled to provide a complete control on the bioreactor system. The effectiveness of the proposed scheme has been validated through a simulation study by using the nonlinear bioreactor model.

2. Process Description

Bioreactor can be defined as a reactor system employed to execute a number of biological reactions in a liquid medium to form intermediate and final products. Figure 1 shows the schematic diagram of the bioreactor. The dynamic behavior of the reactor is complex, and a number of vital manufacturing processes belong to this group:



The basic reaction inside the bioreactor is



where "A" is the reactant, "k" is reaction rate constant, and "P" is the product.

Biosynthesis is widely utilized to convert the living cells (biomass/microbes) into marketable chemical, pharmaceutical, food and beverage products. Equation (2) shows the operation performed during biosynthesis. In this operation, the biomass/microbes consume nutrients from the substrate (feed) to cultivate and to produce more cells and important products [1]. During this operation the bioreactor is kept under a controlled environment with constant pH, temperature, agitation rate, and dissolved oxygen tension to attain better growth of microbes. In biosynthesis, microorganisms play an essential role in the production of industrial chemicals, enzymes, and antibiotics.

The various stages of microbial growth in a bioreactor during biosynthesis are schematically shown in Figure 2. During the initialization of the process, the living cells (microbes) are placed inside the bioreactor maintained with incubated environment, and the necessary feed (substrate)

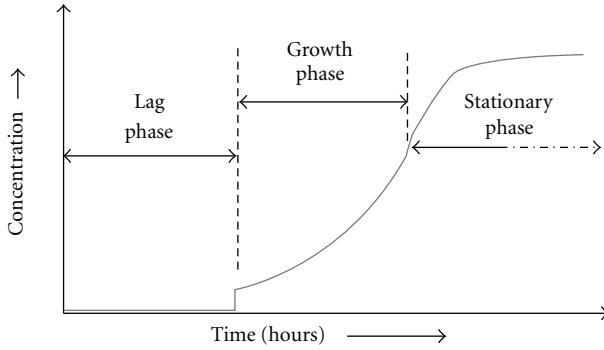


FIGURE 2: Stages of microbial growth in a bioreactor.

is gradually applied to the microbes. In the first stage (lag phase), all the living cells are allowed to adjust their performance to the new environmental conditions after the adjustment, they are getting ready to begin reproduction. In the second stage (growth phase) the cells consume nutrients and increase in size. If the environmental conditions are favorable, each cell grows, divides into two which, in turn, grow and divide and the cycle conditions. This process is rapid and the growth rates of cells are proportional to cell concentration and the nutrient. After the growth, they reach a minimum biological space called stationary phase (third stage). At this stage, the cell growth is limited due to the lack of one or more nutrients, buildup of toxic materials during biosynthesis and organic acids generated during the growth phase. Many important fermentation products are produced in stationary space. After the stationary phase, the cells will reach death phase. Death phase is mainly due to the toxic by-products and depletion of nutrient supply. In this, a decrease in live cell concentration occurs [2, 3].

In the above three stages (Figure 2), essential work is performed in growth and stationary phase. Since it is necessary to place a controller in this region to increase productivity the cell mass concentration (X) and substrate concentration (S) are the two process state variables available for the controller design. To optimize the model-based controller parameter, it is necessary to delineate the various stability regions in the bioreactor and to study the effect of substrate (S) on the biomass (X). The main objective for implementing the controller for the bioreactor is to maximize the production and to minimize the waste. The stoichiometry for biomass activity is very complex since it varies with environmental conditions, microorganism, and nutrients in the feed. Due to these reasons, unstructured models are mainly considered for analysis purpose. A number of studies are available in the literature for model-based control of bioreactor [8–10].

The following mathematical equations can describe a variety of industrial bioreactors [11]:

$$\text{Cell balance: } \frac{dX}{dt} = (\mu - D)X, \quad (3)$$

$$\text{Substrate balance: } \frac{dS}{dt} = D(S_f - S) - \frac{\mu X}{Y},$$

$$\text{Product balance: } \frac{dP}{dt} = -DP + (\alpha\mu + \beta)X, \quad (4)$$

$$\text{Monod kinetics: } \mu = \frac{\mu_{\max}S}{K_m + S},$$

where “ μ ” is the specific growth rate, “ X ” the biomass concentration, “ S ” the substrate concentration, and “ α ” and “ β ” are yield parameters for the product. At steady state, the variables will be $X = X_s$, $S = S_s$, and $P = P_s$.

At steady state operating region, (3) will be

$$\frac{dX_s}{dt} = 0; \quad \frac{dS_s}{dt} = 0. \quad (5)$$

If more than one steady state occurs, it is identified as trivial steady state and nontrivial steady state:

- (i) for trivial solution, $X_s = 0$ when $(\mu_s - D_s) \neq 0$,
- (ii) for nontrivial solution $X_s \neq 0$ when $(\mu_s - D_s) = 0$.

The nominal parameter and constant values considered in the mathematical equations are presented in Table 1.

3. First Principle Model Development

Figure 3 shows a generalized procedure to be followed for model development and model validation practice for nonlinear bioreactor.

Real time model development for a bioreactor is a challenging job, and sometimes it may not provide a satisfactory model due to its nonlinear nature. The operating time of the bioreactor is also large compared to the other nonlinear chemical process loops existing in the chemical industries. Generally in industries, the bioreactor may run for several days in order to convert the raw material in to products. Collecting the real time data to develop the model around the operating regions (lag, growth, and stationary phases) is time consuming. Hence, in this paper, a first principle-based model development is proposed for the bioreactor widely considered in the literature. The first principle modeling equations from (3) and (4) are considered in this work to develop the mathematical model of the bioreactor (state-space and transfer function model).

The state-space model of the system is represented by the following equations [11]:

$$\begin{aligned} \overset{o}{\dot{x}}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \quad (6)$$

where

$$A = \begin{bmatrix} \mu_s - D_s & X_s \mu'_s \\ \frac{-\mu_s}{Y} & -D_s - \frac{\mu'_s X_s}{Y} \end{bmatrix}, \quad (7)$$

$$B = \begin{bmatrix} -X_s \\ S_f - S \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}; \quad D = [0], \quad (8)$$

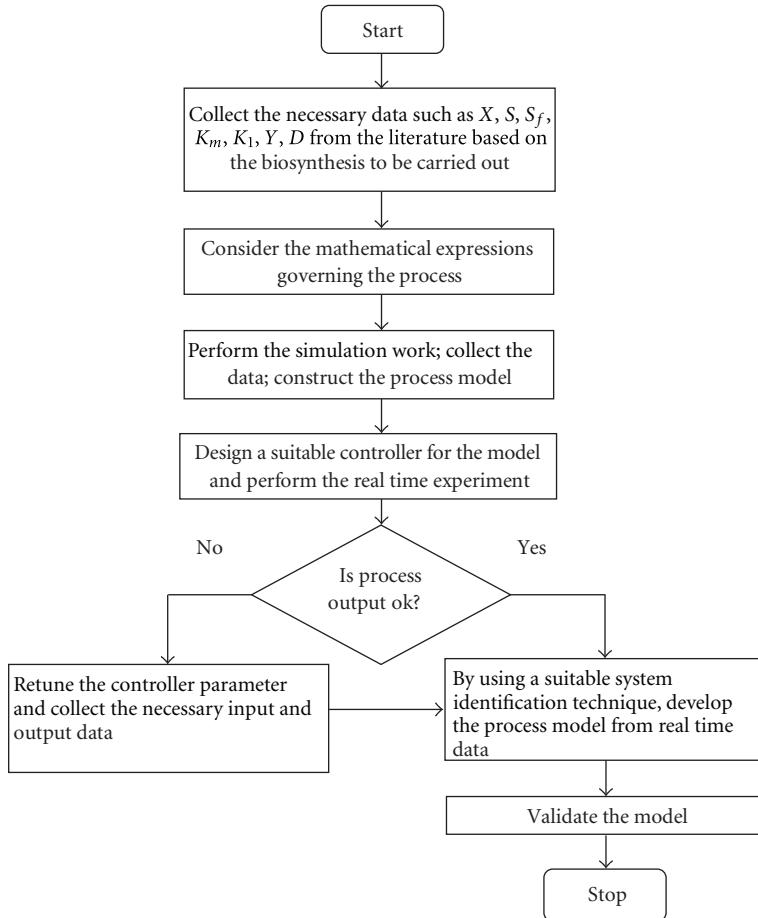


FIGURE 3: Stages in model development and validation procedure.

(where: μ_s , D_s , μ_s^1 , Y , S , S_f , and X_s are the final steady state values of the bioreactor considered during the model development).

Table 2 shows the POLYMATH software codes developed to simulate the bioreactor data at various operating levels. Each program is separately simulated for a simulation time of 10 hours, and the corresponding plots are shown in Figures 4 to 6.

Initially the “lag phase” is simulated by considering the values as represented in Table 2. The generated data from the above program is considered to construct Figure 4, and it depicts the time-related values of feed and product concentrations in the lag phase region. In this region, the microbes are allowed to adjust their performance to the new environmental conditions. Since the increase in the product concentration in this region is very minimal or approximately zero, in the simulation study the final values for “biomass concentration” and “substrate concentrations” are “0” and “3.850639,” respectively. From Table 1, it is observed that data attained from the simulation study is very close with the actual data (i.e., biomass concentration (BC) = 0; substrate concentration (SC) = 4).

Simulation study for growth phase is done with the values provided in Table 2. The created data from simulation

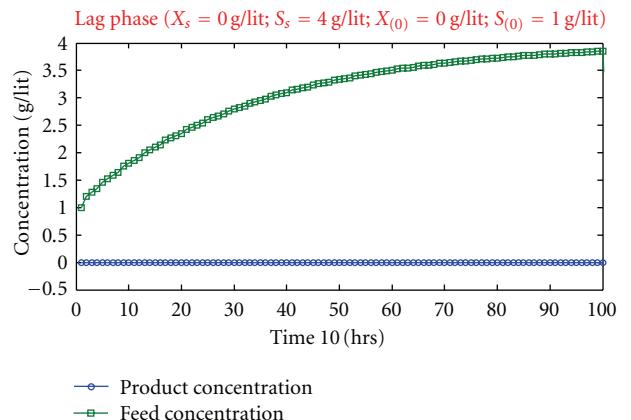


FIGURE 4: Stages of microbial growth in lag phase.

is considered to construct Figure 5. It shows the time-related values of feed and product concentrations in the growth phase region.

In this region, the microbes consume nutrients from the feed and increase its size. After reaching the maximum size, it will split into two and grows. This operation is rapid since

TABLE 1: Nominal parameters of bioreactor.

| Parameter | | Values |
|-----------------|--|--|
| X_s | Biomass concentration | 0; g/lit—lag phase 0.9951; g/lit—growth phase 1.5302; g/lit—stationary phase |
| S_s | Substrate concentration | 4.0; g/lit—lag phase 1.5122; g/lit—growth phase 0.1746; g/lit—stationary phase |
| D | Dilution rate | 0.3; hr^{-1} |
| X_{fs} | Feed substrate concentration | 4.0 g/lit |
| μ_{\max} | Maximum specific growth rate | 0.53; hr^{-1} |
| K_m | Substrate saturation constant | 0.12; g/lit |
| Y | Cell mass yield | 0.4 |
| K_i | Substrate inhibition constant | 0.4545; lit/g |
| $X(0)$ | Initial arbitrary value of biomass (for simulation study) | 0; g/lit—lag phase 0.125; g/lit—growth phase 1; g/lit—stationary phase |
| $S(0)$ | Initial arbitrary value of substrate (for simulation study) | 1; g/lit—lag phase 1; g/lit—growth phase 1; g/lit—stationary phase |
| α, β | Yield parameters for the product—to be selected based on the substrate and biomass (it will have value only during the real time study) | |

TABLE 2: POLYMATH program for bioreactor simulation.

| Lag phase | Growth phase | Stationary phase |
|---|---|---|
| $d(x)/d(t) = (u - D) * x$ | $d(x)/d(t) = (u - D) * x$ | $d(x)/d(t) = (u - D) * x$ |
| $x(0) = 0$ | $x(0) = 0.125$ | $x(0) = 1$ |
| $d(s)/d(t) = D * (s_f - s) - (u * x/Y)$ | $d(s)/d(t) = D * (s_f - s) - (u * x/Y)$ | $d(s)/d(t) = D * (s_f - s) - (u * x/Y)$ |
| $s(0) = 1$ | $s(0) = 1$ | $s(0) = 1$ |
| $u = ((u_m * s)/(k_m + s))$ | $u = ((u_m * s)/(k_m + s))$ | $u = ((u_m * s)/(k_m + s))$ |
| $u_m = 0.53$ | $u_m = 0.53$ | $u_m = 0.53$ |
| $k_m = 0.12$ | $k_m = 0.12$ | $k_m = 0.12$ |
| $D = 0.3$ | $D = 0.3$ | $D = 0.3$ |
| $s_f = 4$ | $s_f = 1.5122$ | $s_f = 0.1746$ |
| $Y = 0.4$ | $Y = 0.4$ | $Y = 0.4$ |
| $t(0) = 0$ | $t(0) = 0$ | $t(0) = 0$ |
| $t(f) = 10$ | $t(f) = 10$ | $t(f) = 10$ |

there will be a sudden rise in the product concentration. In the simulation study the final values for “BC” and “SC” are “0.989742” and “1.528277,” respectively. From Table 1, it is observed that data attained from the simulation study is very close with the actual data (i.e., BC = 0.9951; SC = 1.5122).

Simulation study for stationary phase is conducted using Table 2 values, and the generated data provides Figure 6. This figure depicts the time-related values of feed and product concentrations in the stationary phase region.

In this stage, the cell growth is limited due to the lack of the nutrients, buildup of toxic materials during biosynthesis, and organic acids generated during the growth phase. During

this stage, all the available microbes are converted in to useful chemical and biochemical products. In this study the final values for “BC” and “SC” are “1.526381” and “0.159153,” respectively. From Table 1, it is observed that data attained from the simulation study is very close with the actual data (i.e., BC = 1.5302; SC = 0.1746).

Figure 7 represents the combinations of the product concentration achieved in lag, growth, and stationary phases. This diagram is the exact replica of the cell growth stage discussed in Section 2 (Figure 2).

From the POLYMATH simulation, the following values are obtained for the final feed and product concentrations:

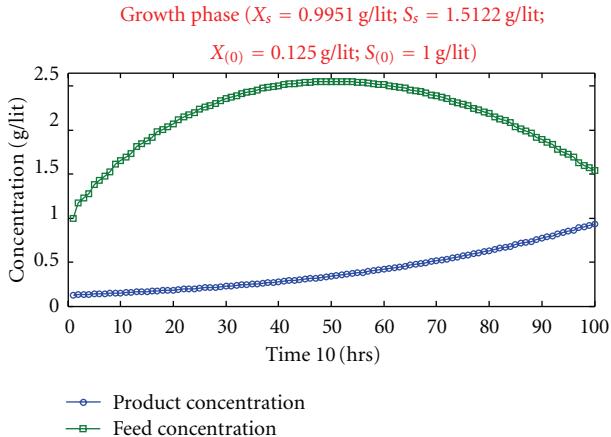


FIGURE 5: Stages of microbial growth in growth phase.

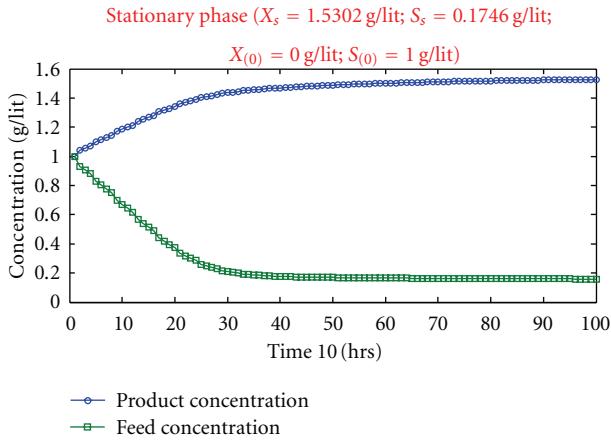


FIGURE 6: Stages of microbial growth in Stationary phase.

- (i) lag phase: SC = 3.850639 g/lit; BC = 0 g/lit,
- (ii) growth phase: SC = 1.528277 g/lit; BC = 0.989742 g/lit,
- (iii) stationary phase: SC = 0.159153 g/lit; BC = 1.526381 g/lit.

These values are implemented in (7), to develop the first principle model of the bioreactor for the growth and stationary phases, and the developed models are tabulated in Table 3. From the Table 3 values, it is confirmed that the developed model by this procedure is approximately similar to the model existing in the literature [11].

The developed mathematical models shown in Table 3 are utilized along with a delay time of "0.1" in the proposed study (the delay time = measurement of the product concentration, converting the measured concentration into the value acceptable by the control loop) for the PSO-based PID controller tuning.

The POLYMATHE program considered in this simulation study also provides the following information about the bioreactor considered in this study.

Figure 8 shows the specific growth rate of microbes with respect to the substrate (feed) concentration. Each

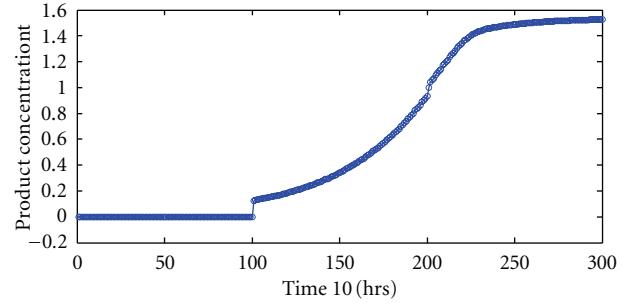


FIGURE 7: Stages of microbial growth in a bioreactor.

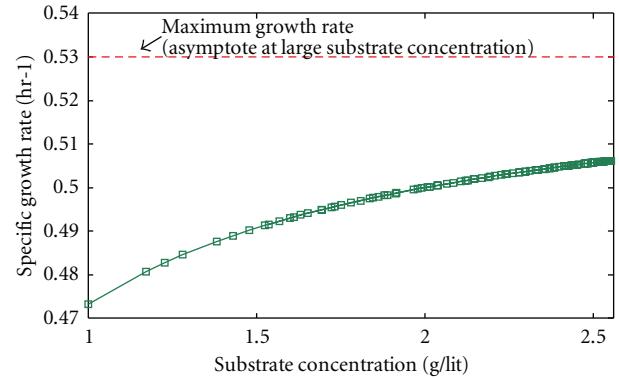


FIGURE 8: Variation of specific growth rate with respect to feed concentration.

TABLE 3: Mathematical model of the bioreactor.

| Developed model | Available model (Wayne Bequette, 2003 [11]) |
|---|---|
| $A = \begin{bmatrix} 0 & -0.064 \\ -0.77 & -0.1298 \end{bmatrix}$ | $A = \begin{bmatrix} 0 & -0.068 \\ -0.75 & -0.1302 \end{bmatrix}$ |
| $B = \begin{bmatrix} -0.9883 \\ 2.5104 \end{bmatrix}$ | $B = \begin{bmatrix} -0.9951 \\ 2.4878 \end{bmatrix}$ |
| $C = \begin{bmatrix} 1 & 0 \end{bmatrix}; D = [0]$ | $C = \begin{bmatrix} 1 & 0 \end{bmatrix}; D = [0]$ |
| $A = \begin{bmatrix} 0 & 0.8738 \\ -0.77 & -2.499 \end{bmatrix}$ | $A = \begin{bmatrix} 0 & 0.9056 \\ -0.75 & -2.564 \end{bmatrix}$ |
| $B = \begin{bmatrix} -1.5272 \\ 3.8195 \end{bmatrix}$ | $B = \begin{bmatrix} -1.5301 \\ 3.8255 \end{bmatrix}$ |
| $C = \begin{bmatrix} 1 & 0 \end{bmatrix}; D = [0]$ | $C = \begin{bmatrix} 1 & 0 \end{bmatrix}; D = [0]$ |

microbe (cell) has its own size, and after reaching the maximum growth, the microbe starts the reproduction. Later it consumes the feed and continues the reproduction until it reaches the elimination phase.

Figure 9 shows the relationship between the specific growth rate and the biomass concentration. Initially the growth rate rapidly increases due to the transition of the microbe from the lag phase to the growth phase. After reaching the maximum growth rate, due to the reproduction operation, the biomass concentration increases.

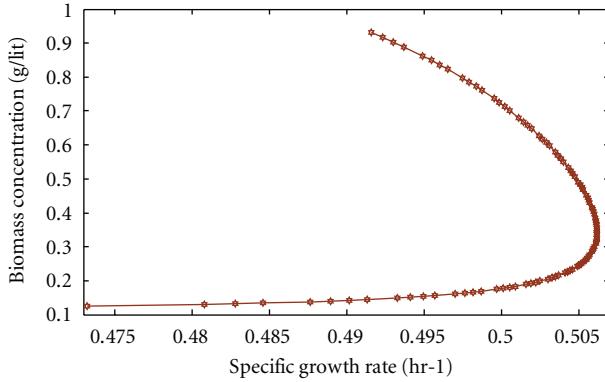


FIGURE 9: Relationship between the product concentration and specific growth rate.

Figure 10 depicts the relationship between the feed and the biomass concentration. Initially the biomass concentration linearly increases with respect to the feed concentration. After reaching the maximum growth rate in microbes, even though the feed is less, due to the reproduction operations, the concentration in the biomass increases.

Figure 11 shows the specific growth rate of microbes with respect to time. In this a simulation period of 10 hrs is shown. In growth phase, the rate of growth linearly increases with respect to time. When it attains the maximum growth rate, the microbes researches their saturation level. Following the saturation, some of the microbe (cell) may expire due to the reasons such as lack of food ageing. Hence, after saturation, the growth rate of microbe rapidly decreases with respect to time.

Figure 12 shows the relationship between the specific growth rate and the “pH” value inside the bioreactor. Form the diagram, it is observed that the microbes can effectively grow when the surrounding “pH” is $6 < \text{pH} < 7.5$.

4. PSO Algorithm

Particle swarm optimization (PSO) algorithm is a population-based evolutionary computation technique developed by the inspiration of the social behavior in bird flocking or fish schooling [14]. It has become one of the most powerful soft computing methods for solving optimization problems. It attempts to mimic the natural process of group communication of individual knowledge, to achieve some optimum property. In PSO algorithm, a population of swarm is initialized to move in a “ D ” dimensional search space. Each particle in swarm has a position represented by a vector “ $\mathbf{S}_i^t = (s_{i1}, s_{i2}, \dots, s_{id})$ ” and velocity represented by a vector “ $\mathbf{V}_i^t = (v_{i1}, v_{i2}, \dots, v_{id})$ ”. At the beginning, each particle in the swarm population is scattered randomly throughout the entire search space “ d ” and with the guidance of the performance criterion, the flying particles dynamically regulate their velocity and position according to their own flying experience and their companions flying experience. Each particle remembers its best position obtained so far, which is denoted p_{best}^t . It also receives the globally best

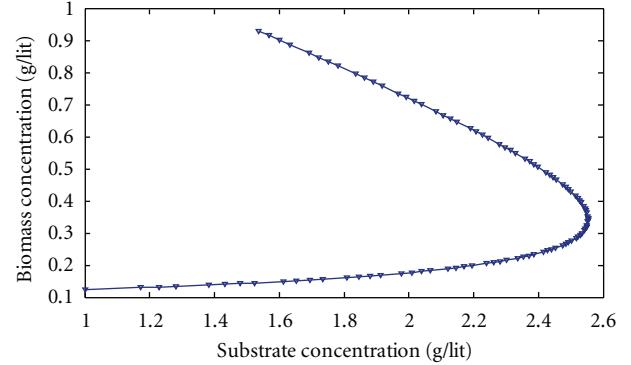


FIGURE 10: Relationship between the product and feed concentration.

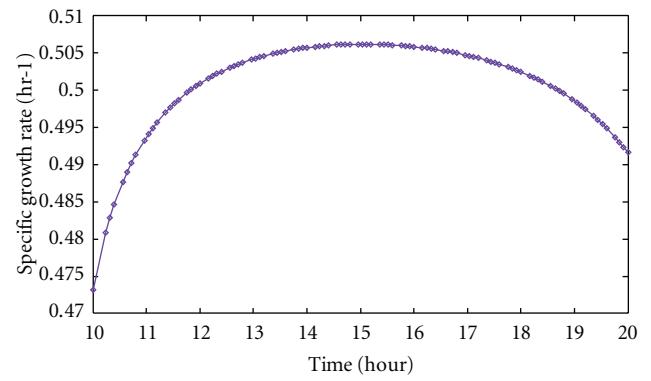


FIGURE 11: Growth rate of microbes.

position achieved by any particle in the population, which is denoted as g_{best}^t (G_i^t) [17, 20, 21, 24]. The updated velocity of each particle can be calculated using the present velocity and the distances from p_{best} and g_{best} . The updated velocity and the position are given in (6) and (8), respectively. Equation (7) shows the inertia weight:

$$\begin{aligned} V_i^{t+1} &= W^t \cdot V_i^t + C_1 \cdot R_1 \cdot (P_i^t - S_i^t) + C_2 \cdot R_2 \cdot (G_i^t - S_i^t), \\ S_i^{t+1} &= S_i^t + V_i^{t+1}, \\ W^t &= W_{\max} - \left(\text{Iter} \times \left[\frac{(W_{\max} - W_{\min})}{\text{Iter}_{\max}} \right] \right), \end{aligned} \quad (9)$$

where C_1 and C_2 are positive constants known as acceleration constants. “ C_1 ” is the cognitive learning rate and “ C_2 ” is the global learning rate. R_1 and R_2 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 ” (i.e., W_{\max}) can favor the global wide-range search, and lower value of “ W ” (i.e., W_{\min}) implies a higher ability for local nearby search (see Pseudocode 1).

5. PSO-based PID Tuning

The PID tuning process is to find that the optimal values for K_p , K_i , and K_d form the search space that minimizes the

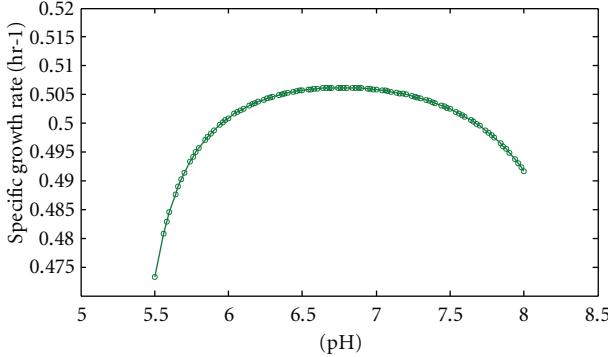


FIGURE 12: Growth rate of microbes with respect to pH.

objective function (11). During this search, the performance criterion “ $J(K_p, K_i, K_d)$ ” guides the optimization algorithm to get appropriate value for the controller parameters.

In the literature, there is no clear guide line to assign the algorithm parameters for the PSO algorithm. In this study, before proceeding with the PSO-based controller tuning, we assigned the following parameters.

Dimension of search space “ D ” is three (i.e., K_p , K_i , K_d); the number of swarm and bird steps is considered as twelve; the cognitive (C_1) and global (C_2) search parameter is assigned the value of 2, the inertia weight “ W_{\min} ” and “ W_{\max} ” is assigned the value 0.2 and 1, respectively.

In this study, a noninteracting form of parallel PID controller is considered to control the nonlinear system.

Parallel PID structure

$$K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right) = \left(K_p + \frac{K_i}{s} + K_d s \right), \quad (10)$$

where: $\tau_i = K_p/K_i$, $\tau_d = K_d/K_p$.

5.1. Controller Tuning. The controller tuning process is employed to find the best possible values for K_p , K_i , and K_d . In order to achieve the superior accuracy during the optimization search, it is necessary to assign appropriate performance index which guides the PSO algorithm.

In this work we considered the following performance criterion (11) with six parameters, ISE, IAE, M_p , t_r , t_s , and E_{ss} :

$$\begin{aligned} J(K_p, K_i, K_d) = & (w_1 \cdot \text{IAE}) + (w_2 \cdot \text{ISE}) + (w_3 \cdot M_p) \\ & + (w_4 \cdot t_s) + (w_5 \cdot t_r) + (w_6 \cdot E_{ss}), \end{aligned} \quad (11)$$

where

$$\text{IAE} = \int_0^T |e(t)| dt = \int_0^{100} |r(t) - y(t)| dt, \quad (12)$$

$$\text{ISE} = \int_0^T e^2(t) dt = \int_0^{100} [r(t) - y(t)]^2 dt, \quad (13)$$

$$M_p = y(t) - r(t), \quad (14)$$

where t_r isrise time (time required for $y(t)$ to reach 100% of its setpoint at the first instant), t_s issettling time – time required for $y(t)$ to reach and stay at $r(t)$ [i.e., $y(t) = r(t)$], E_{ss} issteady state-error, T is simulation time, where w_1, w_2, \dots, w_6 are weighting functions used to set the priority of the multi-objective performance index parameters, and the value of “ w ” varies from 0 to 10. The performance criterion $J(K_p, K_i, K_d)$ guides the PFO algorithm to get appropriate values for the controller parameters.

Figure 13 shows the block diagram of the MOPSO-based PID controller tuning proposed for the bioreactor model.

5.2. Optimization Search. Prior to the optimization search, it is necessary to assign the parameters for the PSO algorithm and the multi-objective performance index criterion.

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) number of swarm and bird steps is considered as twelve, the cognitive (C_1) and global (C_2) search parameter is assigned the value of 2, the inertia weight “ W_{\min} ” and “ W_{\max} ” is assigned the value 0.2 and 1, respectively,
- (iii) the maximum iteration for generation (Iter) is set to 250,
- (iv) boundaries for the three dimensional search space is assigned as

$$\begin{aligned} \text{Value 1} &= -25\% < K_p < +50\% \quad (\text{i.e., } -2.5 < K_p < 5.0), \\ \text{Value 2} &= -20\% < K_i < +20\% \quad (\text{i.e., } -2.0 < K_i < 2.0), \\ \text{Value 3} &= -20\% < K_d < +30\% \quad (\text{i.e., } -2.0 < K_d < 3.0), \end{aligned} \quad (15)$$

- (v) the weighting values are assigned as $w_1 = w_2 = w_3 = w_4 = 10$ and $w_5 = w_6 = 5$,
- (vi) maximum simulation time is selected as 100 sec,
- (vii) the “ t_r ” is preferred as <25% of the maximum simulation time. The simulation time should be selected based on the process time delay,
- (viii) the overshoot (M_p) range is selected as <100% of the reference signal,
- (ix) the “ t_s ” is preferred as <50% of the maximum simulation time,
- (x) the E_{ss} is assigned as zero,
- (xi) the reference signal is considered as unity (i.e., $R(s) = 1$).

5.3. Comparative Study. In order to evaluate the performance of the proposed algorithm, a comparative analysis is done with the most successful soft computing methods like PSO and BFO.

```

Step 1 % Assign values for the PSO parameters %
    Initialize: swarm size ( $N$ ) and step size; learning rate ( $C_1, C_2$ ) dimension for the search
    space ( $d$ ); inertia weight ( $W$ );
    % Initialize random values and current fitness %
     $R_1 = \text{rand}(d, N)$ ;  $R_2 = \text{rand}(d, N)$ ; Current fitness =  $0 * \text{ones}(N, 1)$ 
Step 2 % Initialize Swarm Velocity and Position %
    Current position =  $10 * (\text{rand}(d, N) - 0.25)$ 
    Current velocity =  $0.5 * \text{rand}(d, N)$ 
Step 3 Evaluate the objective function of every particle and record each particle's  $P_i^t$  and  $G_i^t$ .
    Evaluate the desired optimization fitness function in " $d$ "—dimension variables
Step 4 Compare the fitness of particle with its  $P_i^t$  and replace the local best value as given below.
    for  $i = 1 : N$ 
        If current fitness ( $i$ ) < local best fitness ( $i$ );
        Then local best fitness = current fitness; %
    Replacement %
        local best position = current position ( $i$ ); %
    Replacement %
        end
    Step 5 Compare the fitness of particle with its  $G_i^t$  and replace the global best value as given below.
    for  $i = 1 : N$ 
        If current fitness ( $i$ ) < global best fitness ( $i$ );
        Then global best fitness = current fitness; %
    Replacement %
        global best position = current position ( $i$ ); %
    Replacement %
        end
Step 6 Update the current velocity and position of the particles according to (6) and (8)
Step 7 Repeat step 2 to 6 until the predefined value of the performance index has been reached.
    Record the optimized  $K_p, K_i, K_d$  values.

```

PSEUDOCODE 1: Pseudocode for multiple objective PSO-based PID tuning.

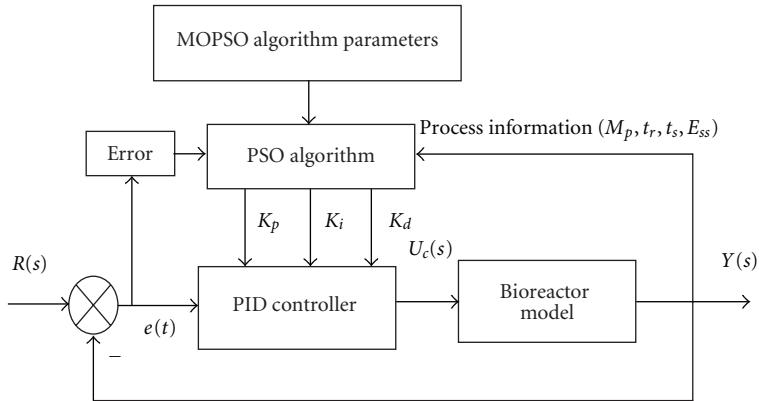


FIGURE 13: Block diagram of the MOPSO algorithm-based PID controller tuning.

PSO. The simulation is carried out by using the PSO algorithm attempted by RajniKanth and Latha [28]. 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 steps 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 fixed 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; the number of chemotaxis steps

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 [29].

6. Results and Discussions

6.1. Set-Point Scheduling. A complete control scheme of the nonlinear bioreactor operating at multiple steady states is schematically presented in Figure 14. This system has a

family of classical PID controller (PID1 for unstable region and PID2 for stable region) and a gain scheduler switch. The switch considered in this study is an electronic switch, and the switching time by this device is negligible.

In the bioreactor, during the lag phase, a manual control is employed to monitor and control the biomass activity (0 to 10 hr in Figure 7). During the growth phase, the cell growth is very rapid, and its concentration is the function of the time and the substrate feed rate (from 10 to 20 hr in Figure 7). At this region, in order to increase the production rate and to reduce the wastage, it is essential to implement an optimally tuned controller. All the necessary products are produced in the stationary phase (from 20 to 30 hr in Figure 7).

In order to maintain the product quality, it is necessary to implement another controller in this region since the PID tuned for growth phase (unstable state) will not provide the robust performance in the stationary phase (stable steady state). This can be avoided by implementing a gain scheduler along with a separate PID controller based on the operating regions.

The set-point-based gain scheduler considered in this work is a comparator assisted electronic switch, which activates the corresponding PID controller based on the available set-point value. The scheduler unit continuously monitors the reference signal (biomass concentration) applied to the bioreactor and selects the suitable local controller unit to execute a complete control over the entire operating region as depicted in Figure 14.

6.2. Growth Phase Model. The second-order unstable transfer function model of the bioreactor with a measurement delay of “0.1” is shown in (12):

$$G(s) = \left(\frac{-0.9951s - 0.2987}{s^2 + 0.1302s - 0.051} \right) e^{-0.1s}. \quad (16)$$

This process model has two stable poles and an unstable pole. The PSO-based PID controller tuning is proposed for the model with the multi-objective performance criterion represented in (11). To examine the performance of the MOPSO algorithm, five trials are performed. The best solution among the trials is chosen as the optimal solution. The final convergence of the optimized controller parameters is shown in Figure 15, and the K_p , K_i , K_d values are summarized in Table 3.

It shows that the proposed MOPSO-based tuning has less number of iteration compared to BFO algorithm [29]. Figure 16 depicts the servo response of the growth phase model. From this response, it is observed that, the proposed method provides a reduced value of “ M_p ” and “ t_s ” compared to PSO and BFO tuned PID controller. Through a simulation time of 100 min, the MOPSO algorithm provides an enhanced result in reference tracking with reduced ISE and IAE than a single objective PSO algorithm (Table 4) [28].

The regulatory response is studied with a load disturbance of 0.3 (30% of setpoint) introduced at 75th min as in Figure 17. The controller successfully rejects disturbance at 87th min and allows the system to track the setpoint from 88th min onwards. The controller also provides a

smooth output. From the result, the observation is that the proposed PID controller provides a very stable and smooth response for both the set-point tracking and load disturbance elimination.

6.3. Stationary Phase Model. The second-order stable transfer function model of the bioreactor with a measurement delay of 0.1 hr is depicted in (13):

$$G(s) = \left(\frac{-1.53s - 0.4588}{s^2 + 2.564s + 0.6792} \right) e^{-0.1s}. \quad (17)$$

The MOPSO-based PID controller tuning is proposed for the above model, and the ultimate convergence of the controller parameters (best solution among the five trials) is shown in Figure 18, and the optimized values are presented in Table 5.

A step change of 1 g/lit is introduced to the bioreactor model, and the corresponding reference tracking performance of the controller (PID2) is presented in Figure 19. A relative analysis with the previously published work is also carried out [8]. The proposed controller provides improved performance with reduced values for t_r , t_s , ISE, and IAE compared to other methods such as ZN, GA, and Skogestad.

A load disturbance of 0.3 (30% of setpoint) is introduced at 75th min to test the regulatory performance of the controller. From Figure 20, it can be inferred that the PSO-tuned PID controller is able to reject the load disturbance quickly (77th min) and maintains the biomass concentration based on the given setpoint. A summary of the performance comparisons between the proposed and the existing methods are presented in Table 4. The MOPSO-tuned PID provides superior performance with reduced ISE and IAE values in entire operating region. (In Table 5 the suffix “ s ” is for setpoint and “ L ” is the load disturbance.)

6.4. Controller Implementation on Bioreactor Model. The performance of the proposed method is examined on the nonlinear bioreactor model, developed using the first principle equations from (3) to (5). Simulation studies have been carried out on the nonlinear model to demonstrate the reference tracking and disturbance rejection performance of the proposed MOPSO-tuned PID controller in the growth phase and stationary phase of the biomass.

Initially, a step change in the biomass concentration of 0.9951 g/lit has been introduced in the system. To study the regulatory performance, a 20% change in the dilution rate is applied at 300th sampling instant, and the corresponding change in system parameters such as substrate concentration, dilution rate, and controller output (PID1) is observed (Figures 21 and 22). The observation is that change in substrate concentration is more compared to other variables. Later a step change of 1.5302 g/lit has been applied at 700th sampling instant to study the performance in the stationary phase. This change in setpoint, activates the local PID controller (PID 2) with the help of the set-point scheduler. From Figure 21, the observation is that though there is a small overshoot due to the process nonlinearity, the proposed controller provides a smooth setpoint tracking in the entire

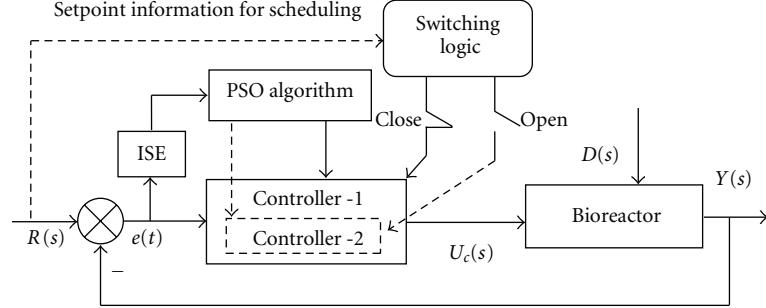


FIGURE 14: Bioreactor control with local PID controllers and setpoint scheduler.

TABLE 4: Performance evaluation for unstable region.

| Algorithm | PID parameters | | | Iteration number | Reference tracking | |
|-----------|----------------|---------|---------|------------------|--------------------|-------|
| | K_p | K_i | K_d | | ISE | IAE |
| MOPSO | -0.5499 | -0.0638 | -0.2018 | 58 | 7.143 | 2.673 |
| BFO | -0.5374 | -0.0702 | -0.0537 | 85 | 5.903 | 2.429 |
| PSO | -0.491 | -0.0501 | -0.1201 | 52 | 11.54 | 3.397 |

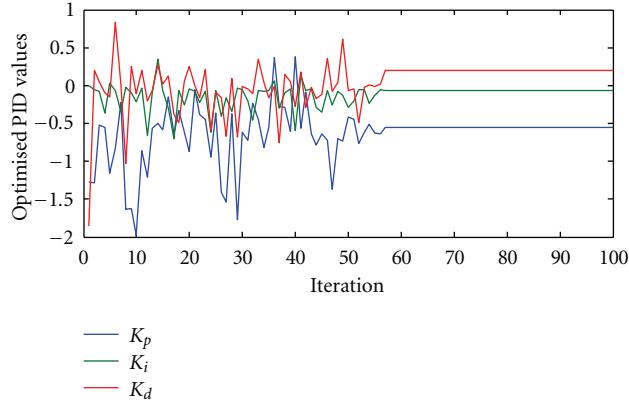


FIGURE 15: Convergence of controller parameters for unstable operating region.

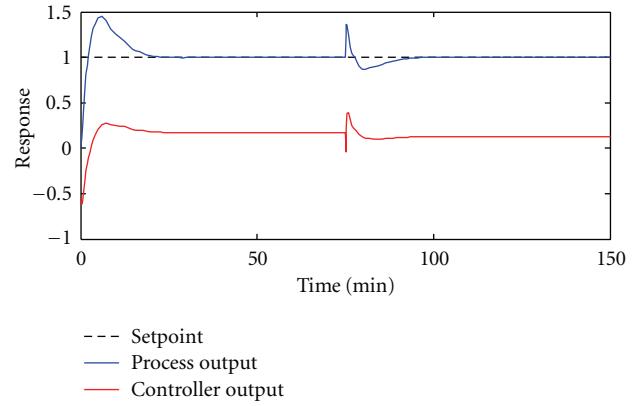


FIGURE 17: Load disturbance rejection performance for unstable operating region.

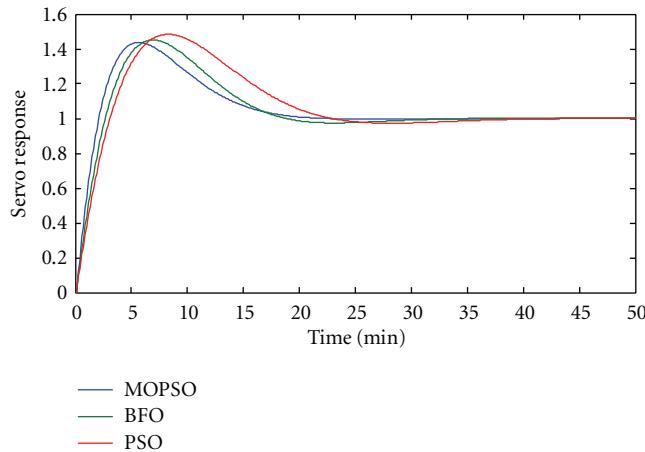


FIGURE 16: Reference tracking performance for unstable operating region.

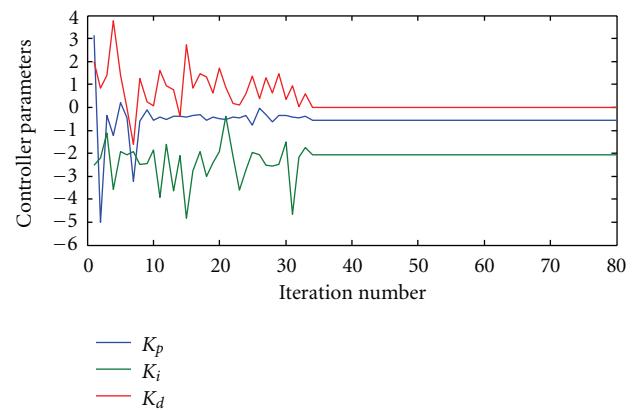


FIGURE 18: Convergence of PID parameters for stable operating region.

TABLE 5: Performance evaluation for stable region.

| Method | Iter | K_p | K_i | K_d | ISE_s | IAE_s | ISE_L | IAE_L |
|-----------|------|---------|---------|---------|---------|---------|---------|---------|
| ZN | — | -1.6722 | -1.8580 | -0.3762 | 0.6343 | 0.796 | 0.991 | 0.995 |
| Skogestad | — | -0.3268 | -0.3268 | -0.1634 | 20.50 | 4.528 | 31.95 | 5.652 |
| GA | — | -1.2440 | -1.3980 | -0.2427 | 1.120 | 1.058 | 1.751 | 1.323 |
| MOPSO | 34 | -0.5612 | -2.0712 | 0.0133 | 0.2863 | 0.5351 | 0.5098 | 0.714 |

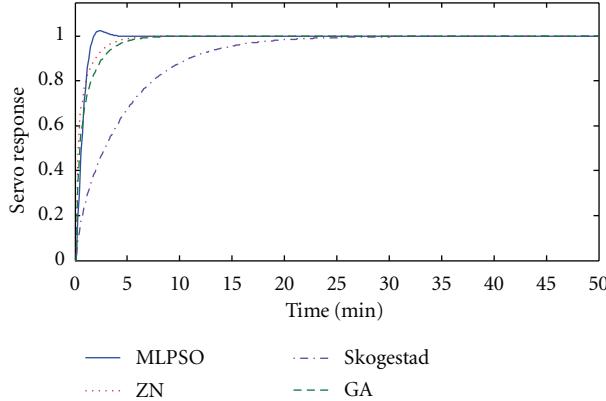


FIGURE 19: Servo response for stable operating region.

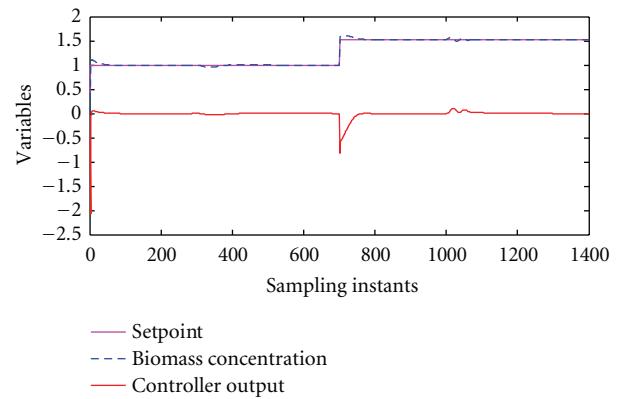


FIGURE 21: Change in biomass concentration and control signal for load disturbance.

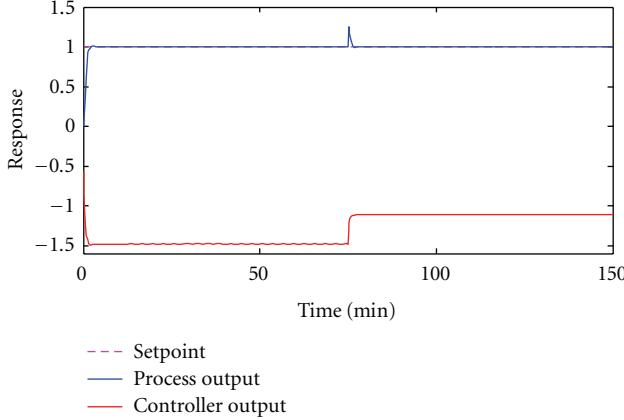


FIGURE 20: Load disturbance rejection performance for stable operating region.

operating region. The regulatory performance is then studied by introducing 20% change in the dilution rate at 1000th sampling instant. The corresponding change in the substrate concentration and the dilution rate is described in Figure 22. Due to the presence disturbance, the dilution rate and the substrate concentration are more oscillatory from 1000th to 1150th sampling instant. Later the response reaches a very smooth value up to 1400th sampling instant.

The transition between the controller (from PID1 to PID2) is initiated by the set-point scheduler during the 700th sampling instant (Figure 21). Due to this changeover, a spiky change in the controller output is observed at 700th sampling instant. This effect disappears quickly and the controller output reaches a steady-state value at 783rd sampling instant.

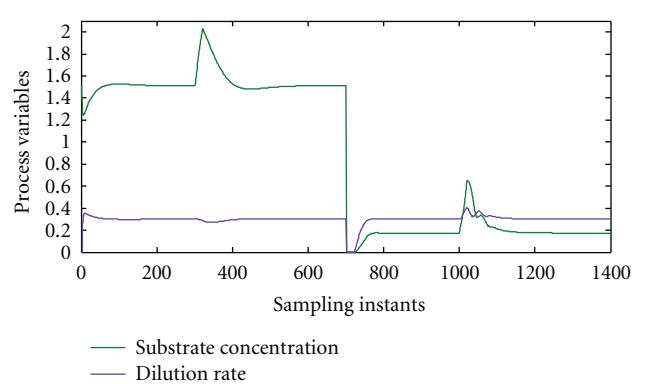


FIGURE 22: Variation of process variables in the presence of load disturbance.

The robustness of the proposed control scheme is then tested by introducing a measurement noise (noise power of 0.001; sampling time of 0.1 sec).

Figure 23 shows the variations of biomass concentration and Figures 24 and 25 depict the variation of substrate concentration, dilution rate, and controller signal in the presence of measurement noise. The reference tracking response of the set-point scheduled bioreactor shows that the proposed scheme works well in the noisy environment. In the presence of measurement noise, the dilution rate and the substrate concentration are more oscillatory from 0th to 500th sampling instant. After 500th sampling instant, the system enters into stationary phase since the effect of measurement noise is minimal. From these observations, it is concluded that the proposed controller scheme is robust,

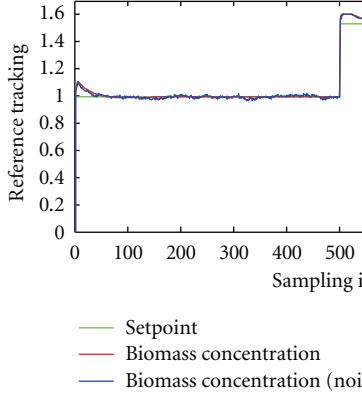


FIGURE 23: Variation of biomass concentration for multiple operating regions.

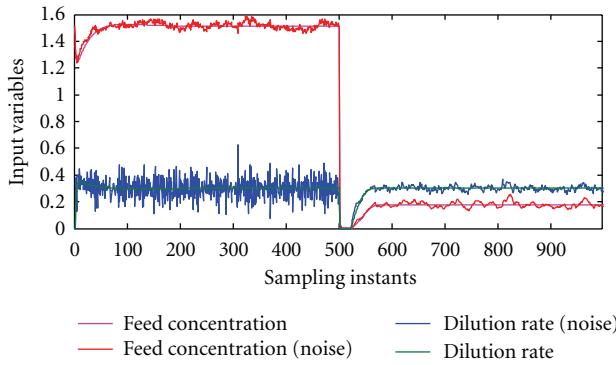


FIGURE 24: Variation of feed concentration and dilution rate for multiple operating regions.

and it helps to provide a better output for setpoint tracking in the entire operating region of the bioreactor.

7. Conclusion

Modeling and the model-based robust controller implementation for the nonlinear chemical system is a challenging work. This paper presents a novel method for modeling and model-based PID controller implementation for a nonlinear bioreactor. First principle equation-based model for the bioreactor is developed using POLYMATH software. The developed model is then considered for PSO algorithm-based PID controller tuning. The proposed method implements a time domain associated multiobjective performance criterion to guide the PSO algorithm in order to obtain optimized PID parameters. Two different local PID controllers are tuned separately for both the stable and unstable operating regions and implemented using a set-point-assisted gain scheduler to track the biomass concentration based on the reference signal. From the extensive simulation studies, it can be observed that the proposed scheme provides enhanced result in setpoint tracking and disturbance rejection. This controller also shows a robust performance in the presence of measurement noise. From the

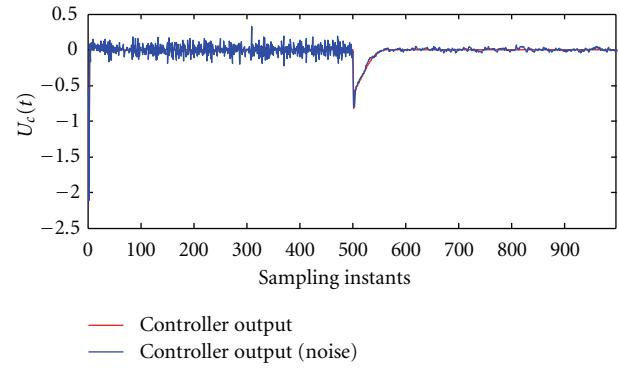


FIGURE 25: Variation of controller signal for multiple operating regions.

study, it can be concluded that the proposed method helps to provide an approximated mathematical model using the modelling equations governing the systems. The proposed controller scheme can be considered as an alternative to other gain scheduled controllers widely employed in nonlinear system control.

Nomenclature:

| | |
|--------------|----------------------------|
| C : | Positive constants (0–2) |
| C_f : | Feed concentration |
| C_r : | Reactor concentration |
| $e(t)$: | Error |
| g_{best} : | Global best position |
| IAE: | Integral absolute error |
| ISE: | Integral squared error |
| K_p : | Proportional gain |
| K_i : | Integral gain |
| K_d : | Derivative gain |
| N : | Filter constant (10) |
| P_{best} : | Local best position |
| PV : | Measured output |
| Q : | Inlet flow rate |
| R : | Random number (0–1) |
| $r(t)$: | Reference input |
| S : | Position of particle |
| SP : | Setpoint |
| τ_i : | Integral time constant |
| τ_d : | Derivative time constant |
| $U_c(s)$: | PID controller output |
| V : | Velocity of particle |
| W : | Inertia weight of particle |
| $y(t)$: | Measured variable |
| $Y(s)$: | Process output. |

Superscripts

| | |
|-----------|---------------------------|
| t : | Iteration number |
| $t + 1$: | Updated iteration number. |

References

- [1] A. K. Jana, *Chemical Process Modeling and Computer Simulation*, Prentice-Hall, New Delhi, India, 2008.
- [2] E. M. T. El-Mansi and C. F. A. Bryce, *Fermentation Microbiology and Biotechnology*, Taylor and Francis, 2003.
- [3] H. Scott Fogler, *Elements of Chemical Reaction Engineering*, Prentice Hall, New Delhi, India, 2006.
- [4] Y. Zhao and S. Skogestad, "Comparison of various control configurations for continuous bioreactors," *Industrial and Engineering Chemistry Research*, vol. 36, no. 3, pp. 697–705, 1997.
- [5] S. V. S. Kumar, V. R. Kumar, and G. P. Reddy, "Nonlinear control of bioreactors with input multiplicities—an experimental work," *Bioprocess and Biosystems Engineering*, vol. 28, no. 1, pp. 45–53, 2005.
- [6] N. Sivakumaran, T. K. Radhakrishnan, and J. S. C. Babu, "Identification and control of bioreactor using recurrent networks," *Instrumentation Science and Technology*, vol. 34, no. 6, pp. 635–651, 2006.
- [7] Z. K. Nagy, "Model based control of a yeast fermentation bioreactor using optimally designed artificial neural networks," *Chemical Engineering Journal*, vol. 127, no. 1–3, pp. 95–109, 2007.
- [8] S. M. Giriraj Kumar, R. Jain, N. Anantharaman, V. Dharmalingam, and K. M. M. Sheriffa Begam, "Genetic algorithm based PID controller tuning for a model bioreactor," *Indian Institute of Chemical Engineers*, vol. 50, no. 3, pp. 214–226, 2008.
- [9] I. Ananth and M. Chidambaram, "Closed-loop identification of transfer function model for unstable systems," *Journal of the Franklin Institute*, vol. 336, no. 7, pp. 1055–1061, 1999.
- [10] S. Pramod and M. Chidambaram, "Closed loop identification of transfer function model for unstable bioreactors for tuning PID controllers," *Bioprocess Engineering*, vol. 22, no. 2, pp. 185–188, 2000.
- [11] B. Wayne Bequette, *Process Control: Modeling, Design and Simulation*, Prentice Hall, 2003.
- [12] A. O'Dwyer, *Handbook of PI and PID Controller Tuning Rules*, Imperial College Press, London, UK, 3rd edition, 2009.
- [13] R. Padma Sree and M. Chidambaram, *Control of Unstable Systems*, Narosa Publishing House, New Delhi, India, 2006.
- [14] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [15] 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.
- [16] M. Khalilzadeh, F. Kianfar, A. S. Chaleshtari, S. Shadrokh, and M. Ranjbar, "A modified PSO algorithm for minimizing the total costs of resources in MRCPS," *Mathematical Problems in Engineering*, vol. 2012, Article ID 365697, 18 pages, 2012.
- [17] P. Umapathy, C. Venkatasesaiah, and M. S. Arumugam, "Particle Swarm Optimization with various inertia weight variants for optimal power flow solution," *Discrete Dynamics in Nature and Society*, vol. 2010, Article ID 462145, 15 pages, 2010.
- [18] R. F. Abdel-Kader, "Particle swarm optimization for constrained instruction scheduling," *VLSI Design*, vol. 2008, Article ID 930610, 7 pages, 2008.
- [19] R. M. Chen and C. M. Wang, "Project scheduling heuristics-based standard PSO for task-resource assignment in heterogeneous grid," *Abstract and Applied Analysis*, vol. 2011, Article ID 589862, 20 pages, 2011.
- [20] M. S. Arumugam and M. V. C. Rao, "On the optimal control of single-stage hybrid manufacturing systems via novel and different variants of particle swarm optimization algorithm," *Discrete Dynamics in Nature and Society*, vol. 2005, no. 3, pp. 257–279, 2005.
- [21] M. S. Arumugam and M. V. C. Rao, "On the performance of the particle swarm optimization algorithm with various inertia weight variants for computing optimal control of a class of hybrid systems," *Discrete Dynamics in Nature and Society*, vol. 2006, Article ID 79295, 17 pages, 2006.
- [22] T. S. Zhan and C. C. Kao, "Modified PSO method for robust control of 3RPS parallel manipulators," *Mathematical Problems in Engineering*, vol. 2010, Article ID 302430, 25 pages, 2010.
- [23] V. Rajinikanth and K. Latha, "Bacterial foraging optimization algorithm based pid controller tuning for time delayedunstable systems," *Mediterranean Journal of Measurement and Control*, vol. 7, no. 1, pp. 197–203, 2011.
- [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 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '09)*, pp. 445–450, December 2009.
- [25] 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.
- [26] 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.
- [27] U. S. Banu and G. Uma, "Fuzzy gain scheduled continuous stirred tank reactor with particle swarm optimization based PID control minimizing integral square error," *Instrumentation Science and Technology*, vol. 36, no. 4, pp. 394–409, 2008.
- [28] V. Rajinikanth and K. Latha, "Identification and control of unstable biochemical reactor," *International Journal of Chemical Engineering and Applications*, vol. 1, no. 1, pp. 106–111, 2010.
- [29] V. Rajinikanth 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.
- [30] R. Garduno-Ramirez and K. Y. Lee, "Multiobjective optimal power plant operation through coordinate control with pressure set point scheduling," *IEEE Transactions on Energy Conversion*, vol. 16, no. 2, pp. 115–122, 2001.

