International Journal of Electrical Engineering.
ISSN 0974-2158 Volume 5, Number 5 (2012), pp. 571-586

http://www.irphouse.com

Study and Comparison of PI Controller Tuning Techniques using Bacteria Foraging and Bacteria Foraging based Particle Swarm Optimization

Deepa Thangavelusamy and Lakshmi Ponnusamy

Department of EEE, College of Engineering, Anna University, Chennai, Tamil Nadu, India
E-mail: deepabalaji30@gmail.com, p_lakshmi@annauniv.edu

Abstract

This paper presents a design procedure for a Bacteria Foraging based Particle Swam Optimization (BF-PSO) tuned PI and investigating the robustness of the BF–PSO technique applied to the Quadruple Tank Process (QTP). The transfer function is derived from the open loop response. Design of a Decentralized PI Control and tuning the PI parameters using Bacteria Foraging (BF) and BF-PSO techniques are discussed. Performance index Integral Square Error (ISE) is used for designing the controllers. Results show that BF-PSO controller gives better performance for both servo and regulatory responses.

Keywords: Quadruple Tank process, Decentralized PI, BF and BF-PSO

Introduction

The multivariable laboratory process, called the Quadruple-Tank Process (QTP), consists of four interconnected liquid tanks, two pumps and two valves, [4], [11]. They are shown schematically in Fig. 1. The inputs are the voltages to the two pumps (v_1 and v_2) and the outputs are the liquid levels in the lower tanks (h_1 and h_2). The linearized dynamics of the process exhibits a multivariable zero that can be moved from one side of the complex plane to the other side by changing the position of the valves γ_1 and γ_2. This process was found to be ideally suited to illustrate many concepts in the multivariable control.

The tuning of a multiloop Proportional Integral Derivative (PID) controller has to bring loop interactions into consideration. A common way to handle this problem is to introduce a detuning factor to the Single Input Single Output (SISO) tuning constants to stabilize the multivariable closed-loop system. If the process model of the multivariable system is available, a systematic procedure to find this detuning factor is suggested in the literature for multiloop PI tuning [7]. The multivariable zero dynamics of the system can be made for both minimum phase and non-minimum phase by simply changing a valve. The location and the direction of the zero have an appealing physical interpretation. The Relative Gain Array (RGA) has a straightforward meaning for the process [5].

Designing multivariable decoupling and multiloop PI/PID controllers in a sequential manner were developed [10]. The method is based on a single-loop tuning technique developed for multivariable systems with unknown dynamics. Tan et al [12] proposed PID tuning is based on loop shaping H_∞ control. A method for auto-tuning fully cross-coupled multivariable PID controllers from decentralized relay feedback is proposed [15]. It should be noted that modern control techniques might achieve better performance than the conventional PID controller. Zhuang and Atherton [19] designed a diagonal PID controller for a Two-Input Two Output (TITO) system.

Design of frequency selective surface using Particle Swarm Optimization (PSO) technique is discussed [17], [3]. The PSO algorithm is the population based optimization algorithm which can be used to solve the minimization problem [1]. Passino [8] had discussed the control system on the E.coli that dictates how foraging should proceed. A computer program that emulates the distributed optimization process represented by the activity of social Bacterial Foraging (BF) is presented.

The Fractional Order PID [FOPID] control system has a good robustness resembling the integer PID control system, and the FOPID controller has more flexibility. A global search optimization method with BF oriented by PSO is applied for the optimization of the parameters of the FOPID controller [18].

D.H.Kim et al [2] had proposed a novel hybrid approach consisting of a Genetic Algorithm (GA) and BF and the performance is illustrated using various test functions. The proposed method is used for tuning a PID controller of Automatic Voltage Regulator (AVR) system.

A new algorithm for PID controller tuning based on a combination of the foraging behavior of E coli bacteria foraging and PSO is presented [6]. The E coli algorithm depends on random search directions, which had led to delay in reaching the global solution. The PSO algorithm may also lead to possible entrapment in local minimum solutions.

PI tuning of multivariable system using BF-PSO has not been explored much. In this paper PI controller parameters are tuned for a multivariable system (QTP) using BF-PSO algorithm for an operating point and its effectiveness is proved in real time setup.

The paper is organized as follows. A nonlinear model for the QTP based on physical data is derived in section II. Simple multi-loop PI control of the quadruple-tank process using Bacterial Foraging and BF-PSO are discussed in section III. The results and conclusions are presented in Sections IV and V respectively.

Physical Model

A schematic diagram of the process is shown in Fig. 1. The target is to control the level in the lower two tanks with two pumps. The process inputs are input voltages to the two pumps and the outputs are the level measurements. Mass balances and Bernoulli's law yield the following simple nonlinear equations [4]

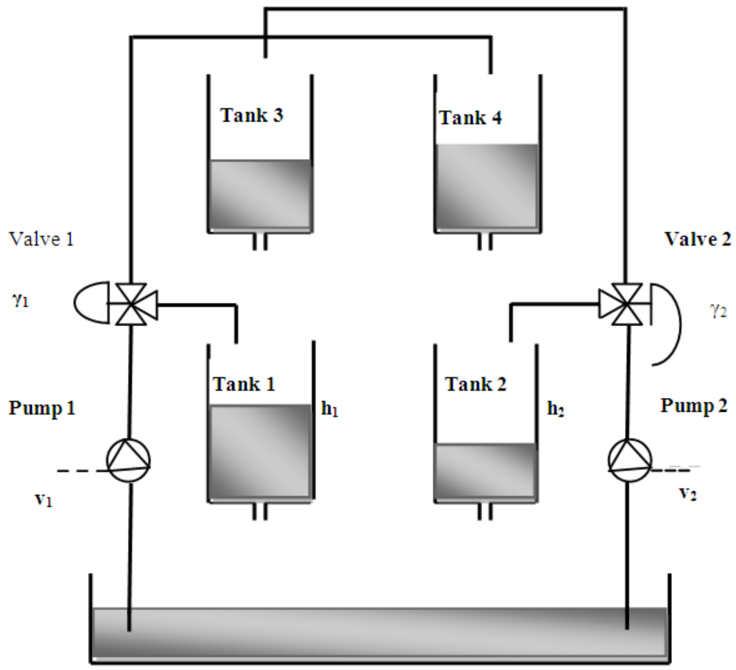

Figure 1: Schematic diagram of the quadruple-tank process.

$$
\begin{aligned}
\frac{dh_1}{dt} &= -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1 k_1}{A_1} v_1 \\
\frac{dh_2}{dt} &= -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2 k_2}{A_2} v_2 \\
\frac{dh_3}{dt} &= -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{\left(1-\gamma_2\right) k_2}{A_3} v_2 \\
\frac{dh_4}{dt} &= -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{\left(1-\gamma_1\right) k_1}{A_4} v_1
\end{aligned}
\tag{1}
$$

where

A_i : Cross-section of Tank i
a_i : Cross-section of the outlet hole i
h_i : Water level i

The voltage applied to pump i is v_i and the corresponding flow is $k_i v_i$. The parameters γ_1, γ_2 ε (0,1) are determined from the position of the valves set prior to an experiment. The flow to Tank 1 is $\gamma_1 k_1 v_1$ and the flow to Tank 4 is $(1-\gamma_1)k_1 v_1$ and the same applies to Tank 2 and Tank 3. The acceleration of gravity is denoted as g. The measured level signals are $k_c h_1$ and $k_c h_2$.

The linearized state-space equation is given by

$$\frac{dx}{dt} = \begin{bmatrix} -\frac{1}{T_1} & 0 & \frac{A_3}{A_1 T_3} & 0 \\ 0 & -\frac{1}{T_2} & 0 & \frac{A_4}{A_2 T_4} \\ 0 & 0 & -\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_4} \end{bmatrix} x + \begin{bmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_2} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{bmatrix} x \tag{2}$$

where the time constants are

$$T_i = \frac{A_i}{a_i}\sqrt{\frac{2h_i^0}{g}}, i=1,\ldots 4 \tag{3}$$

The corresponding transfer function matrix is

$$G(s) = \begin{bmatrix} \frac{\gamma_1 c_1}{1+sT_1} & \frac{(1-\gamma_2)c_1}{(1+sT_3)(1+sT_1)} \\ \frac{(1-\gamma_1)c_2}{(1+sT_4)(1+sT_2)} & \frac{\gamma_2 c_2}{1+sT_2} \end{bmatrix} \tag{4}$$

where $c_1=T_1k_1k_c/A_1$ and $c_2=T_2k_2k_c/A_2$.

Figure 2: Experimental setup of the quadruple-tank process.

Fig.2 shows the experimental setup of the QTP consisting of four interconnected tanks with common water source. This setup is interfaced with a window - based PC via interfacing modules and USB ports. This setup consists of a water supply tank with two positive displacement pumps for water circulation, two pneumatic control valves, four transparent process tanks fitted with level transmitters and rotameters (0-440 lph). Process signals from the level transmitters are interfaced with the PC and it sends outputs to the individual control valves through interfacing units using LabVIEW software. Tanks 1 and 2 are mounted below the other two tanks 3 and 4 for receiving water flow by gravity. Each tank outlet opening is fitted with a valve. Both pumps 1 and 2 takes water by suction from the reservoir. Flow from the pumps is split to top and bottom tanks by manually adjusting the valves. Ratio of flow split between the top and bottom tanks, substantially alters the dynamics of the system. Pump 1 discharges water to tank 1 and tank 4 simultaneously and the flows are indicated by rotameters 1 and 4. Similarly, pump 2 discharges water to tank 2 and tank 3 and the flows are indicated by rotameters 2 and 3. Tanks 1 and 2 also receive water by gravity flow from tank 4 and tank 3, respectively. The parameters of QTP are given in Table 1.

Table 1: Process Parameter Values of Fig.1

i	$A_i(cm^2)$	$a_i(cm^2)$	h_i^0 (cm)
1	176.71	2.01	6.34
2	176.71	2.01	8.31
3	176.71	2.01	3.06
4	176.71	2.01	4.16

The time constants are T_1=42.48 sec, T_2=55.64 sec, T_3=39.86 sec and T_4=55.68 sec.

The transfer function matrix is given in (5)

$$G(s)=\begin{bmatrix} \dfrac{0.3811}{42.48S+1} & \dfrac{0.2334}{(42.48S+1)(39.86S+1)} \\ \dfrac{0.1998}{(55.68S+1)(55.64S+1)} & \dfrac{0.3934}{55.68S+1} \end{bmatrix} \quad (5)$$

Relative Gain Array (RGA)

The RGA was introduced by Ed Bristol as a measure of interaction in multivariable control systems [16]. The RGA Λ is defined as

$$\Lambda = G(0) \times G^{-T}(0) \quad (6)$$

where $\times$ denotes the element by element matrix multiplication and $^{-T}$ inverse transpose.

Properties of RGA

Sum of rows and columns property of the RGA

Each row of the RGA sums to 1.0 and each column of the RGA sums to 1.0.

(ie) $\lambda_{11}+\lambda_{12}=1$ $\quad\quad$ $\lambda_{11}+\lambda_{21}=1$

$\lambda_{12}+\lambda_{22}=1$ $\quad\quad$ $\lambda_{21}+\lambda_{22}=1$

Use of RGA to determine variable pairing

It is desirable to pair output i and input j such that λ_{ij} is as close to 1 as possible.

The RGA depends only on the valve settings and not on other physical parameters.

RGA $\Lambda=\begin{bmatrix} 1.4515 & -0.4515 \\ -0.4515 & 1.4515 \end{bmatrix}$

From RGA h_1 is paired with u_1 and h_2 is paired with u_2

Decentralized PI Control

The decentralized controller structure is shown in Fig.3 and the decentralized control law [5] u=diag{C_1,C_2}(r-y) . The QTP is considered as minimum phase process (the process does not have RHP zeros or time delays).

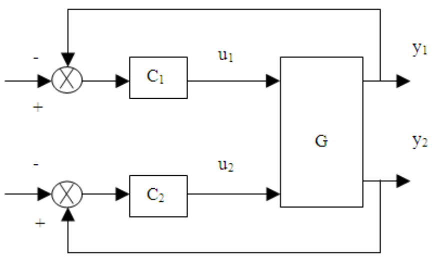

Figure 3: Decentralized control structure with two PI controllers

PI controllers have the form [9]

$$C_l(s) = K_l\left(1+\frac{1}{T_{il}s}\right), \quad l = 1,2 \tag{7}$$

So the direct synthesis controller for a first order process gives

$$K_l = \frac{T_{il}}{K_P T_c}$$

$$T_c = 0.5T_{il} \tag{8}$$

Particle Swarm Optimization (PSO)

PSO is a robust stochastic optimization technique based on the movement and intelligence of swarms. PSO applies the concept of social interaction to problem solving. It was developed in 1995 by James Kennedy and Russell Eberhart. It uses a number of agents that constitute a swarm, moving around in the search space, looking for the best solution. Each particle is treated as a point in an N-dimensional space which adjusts its "flying" according to its own flying experience as well as the flying experience of other particles. Each particle keeps track of its coordinates, in the solution space, which are associated with the best solution, that has achieved so far by that particle. This value is called personal best, pbest. Another best value that is tracked by the PSO is the best value obtained so far by any particle in the neighborhood of that particle. This value is called gbest. The basic concept of PSO lies in accelerating each particle toward its pbest and the gbest locations, with a random weighted acceleration at each time step as shown in Fig 4.

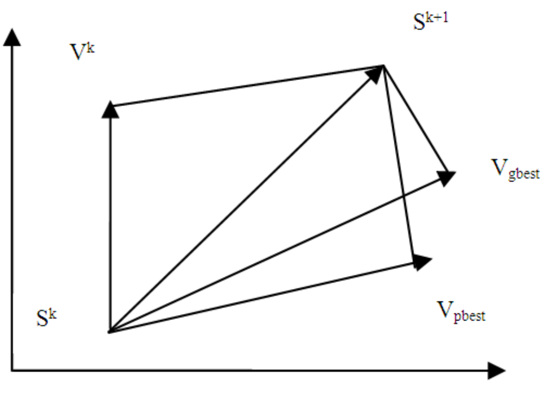

Figure 4: Concept of modification of a searching point by PSO in N-dimensional space

where

S^k Current searching point.

S^{k+1}	Modified searching point
V^k	Current velocity
V^{k+1}	Modified velocity
V_{pbest}	Velocity based on pbest
V_{gbest}	Velocity based on gbest

Each particle tries to modify its position using various informations, such as current positions, current velocities, the distance between the current position and pbest, and the distance between the current position and the gbest.

The modification of the particle's position can be mathematically modeled according to the following equation:

$$V_{ik}+1 = wV_{ik} + c_1 rand1(\ldots) \times (pbest_i - s_{ik}) + c_2 rand2(\ldots) \times (gbest - s_{ik}) \quad (9)$$

where,

V_{ik}	velocity of agent i at iteration k,
W	weighting function
c_j	Weighting factor,
Rand	Uniformly distributed random number between 0 and 1,
s_{ik}	Current position of agent i at iteration k,
$pbest_i$	pbest of agent i,
gbest	gbest of the group.

The following weighting function is usually utilized in the equation (9)

$$w = w_{Max} - \left[(w_{Max} - w_{Min}) \times iter\right] / N \quad (10)$$

Where

w_{Max}	Initial weight
w_{Min}	Final weight
N	Maximum iteration number
Iter	Current iteration number

The new position is then determined by the sum of the previous position and the velocity

$$s_{ik}+1 = s_{ik} + V_{ik}+1 \quad (11)$$

The flow chart of a general PSO algorithm [14] is developed. The optimal values of the conventional PI controller parameters K_p and K_i are found using PSO. Certain parameters of PSO need to be defined. The objective function (F) considered, is based on the error criterion (12). The controller performance is evaluated in terms of Integral Square Error (ISE) given by,

$$F = ISE \times beta \quad (12)$$

where beta = 10

Basic Bacterial Foraging optimization

Natural selection has a tendency to eliminate animals having poor foraging strategies and favor the ones with successful foraging strategies to propagate their genes as they are more likely to reach a successful reproduction. Poor foraging strategies are either completely eliminated or transformed into good ones, after many generations are produced. This activity of foraging inspired the researchers to utilize it as a novel optimization tool. The foraging process can be subdivided into four sections [13].

- Chemotaxi
- Swarming
- Reproduction
- Elimination and Dispersal.

The following BF parameters are selected for the training cycle for the QTP

Number of bacteria	s=10
Number of chemotactic steps	$N_c=4$
Limits the length of a swim	$N_s=4$
Number of reproduction steps	$N_{re}=4$
Number of elimination-dispersal events	$N_{ed}=2$
Number of bacteria reproductions (splits) per generation	$S_r=s/2$
The probability that each bacteria will be eliminated/dispersed	$P_{ed}=0.25$

BF-PSO Algorithm

BF-PSO Algorithm is a new algorithm that combines BF with PSO and is given in Fig 5. It makes full use of the ability of BF Algorithm to acquire new solution in the dispersed process and the parallel search ability of PSO possessed by swarm intelligent algorithm [6].

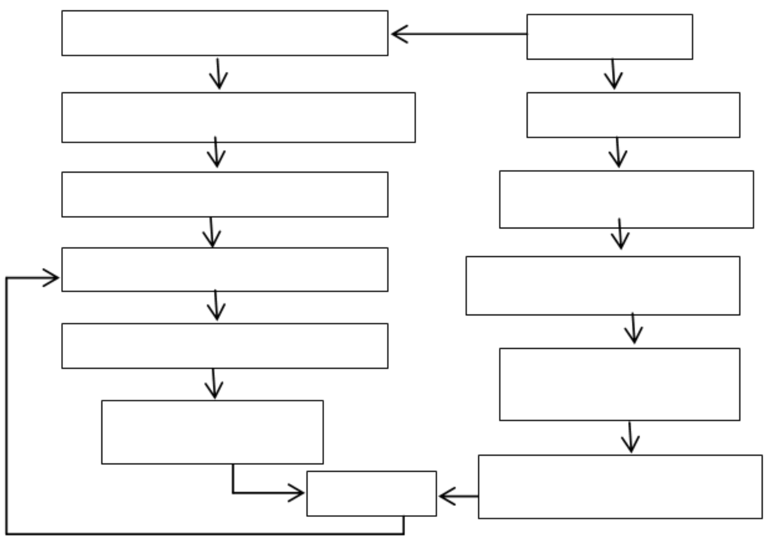

Figure 5: Bacteria foraging oriented by PSO algorithm flow chart

The main steps of BF-PSO Algorithm are summarized as follows

Step 1: Initialization
Initialize the parameters of PSO and BF are c_1,c_2 ,w,w_{max}, s , N_c , N_s , N_{re} , N_{ed} and P_{ed}.

Step 2: Reproduction
For i=1: N_{re}.

Step 3: Chemotaxis
For i=1: N_c , invoke targets to calculate.

Step 4: Tumbling
Invoke targets to calculate, if the number of bacteria tumbling step is less than N_s, update variables and current position of PSO.

Step 5: Dispersal
Invoke targets to calculate, if rand > P_{ed} , obtain locate initial points again.

Step 6: To update velocity, position and local optima and return step 3.

Step 7: Return step 2.

Step 8: Output the optimization results.
The following BF-PSO parameters are selected for the training cycle for the QTP

s : 20
N_c : 10
N_s : 4
N_{re} : 4
N_{ed} : 2
P_{ed} : 0.25
c_1 : 1.2
c_2 : 1.2

The controller parameter values are tabulated in Table 2 for Decentralized, BF and BF-PSO based PI.

Table 2: Controller Parameter Values

Type of Controller	Controller parameters			
	K_1	K_2	K_{i1}	K_{i2}
Decentralized PI	10.01	20.34	0.12	0.09
BF based PI	10.3	21	0.23	0.13
BF-PSO based PI	11	22.5	0.18	0.22

Results

Experimental results are carried out to evaluate the proposed control method by utilizing the LabVIEW software. The performance of the different control strategies are compared based on ISE and ITAE for the two controlled outputs h_1 and h_2. The design of the disturbance is also satisfactory for characterizing the performance of the three different control strategies. Decentralized PI controller and tuning the PI parameters using BF and BF-PSO are designed and implemented in the experimental QTP. The step responses for level h_1 and h_2 of the QTP are shown in Fig 6 and 7 respectively.

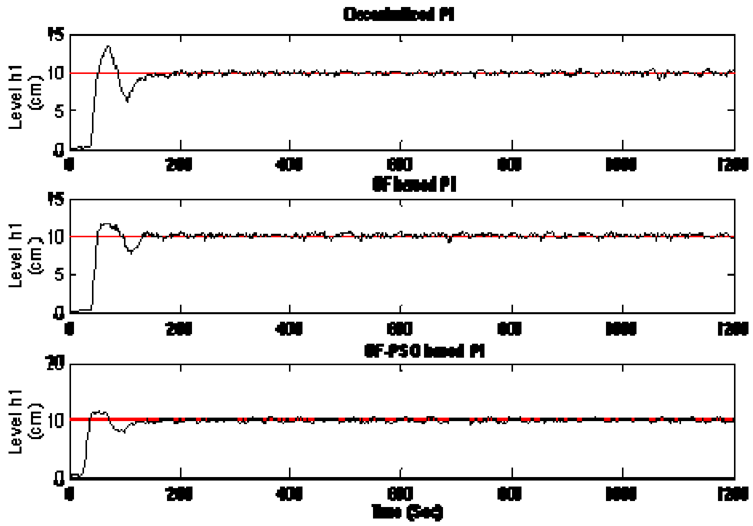

Figure 6: Closed loop Responses of the water level (h_1)

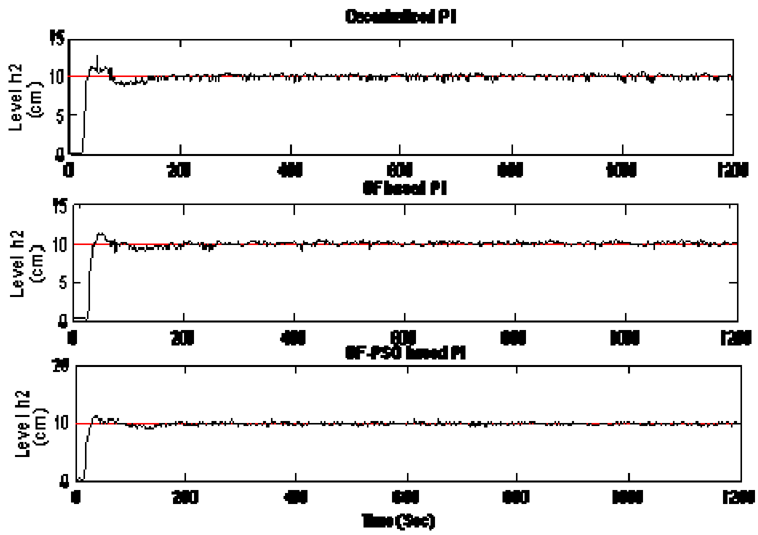

Figure 7: Closed loop Responses of the water level (h_2)

The above response is the closed loop response of the level h_1 for decentralized PI, BF and BF-PSO based PI controllers. The BF-PSO controller settles quickly. The performance index (ISE and ITAE) of BF-PSO is less when compared to decentralized PI and Bacteria foraging and shown in Table 3.

Table 3: Performance Comparison of Various Controllers (Tank 1)

Type of Controller	Level Output of Tank1 (H_1)		
	Peak over shoot (%)	*ISE*	*ITAE*
Decentralized PI	30.6	23.06	886.08
BF based PI	17	17.34	863.76
BF-PSO based PI	13	16.93	819.75

The responses of the closed-loop system for level h_2 are shown in Fig 7. Here the ISE for BF-PSO is 12, ITAE is also less when compared to Decentralized and BF controllers are given in Table 4.

Table 4: Performance Comparison of Various Controllers (Tank 2)

Type of Controller	Level Output of Tank1 (H_2)		
	Peak over shoot (%)	*ISE*	*ITAE*
Decentralized PI	27.6	17.36	471.86
BF based PI	14	13	439.15
BF-PSO based PI	10	12	162.03

In order to test the robustness of the proposed design procedure of BF-PSO controller, experimental results are found out for the servo and regulatory operations. The set point tracking responses of the water level of h_1 and h_2 for the decentralized, BF and BF-PSO are given in Fig 8 and 9 respectively. At 1200th sec, the set point is increased from 10cm to 12cm and at 2400th sec the set point is decreased from 12cm to 10cm. After that the set point is increased to 16cm at 3600th sec, and the response is plotted. The performance comparison of the set point tracking of the controllers for level h_1 and h_2 are given in Tables 5 and 6 respectively.

Table 5: Performance Comparison of Set point Changes (Tank 1)

Type of Controllers	Set point (10 cm)		Set point (12 cm)		Set point (10 cm)		Set point (16 cm)	
	Peak Over Shoot (%)	*ISE*	*Peak Over Shoot (%)*	*ISE*	*Under Shoot (%)*	*ISE*	*Peak Over Shoot (%)*	*ISE*
Decentralized PI	30.6	23.1	7.92	0.5	6.76	0.23	18.4	1.06
BF based PI	29	17.3	24	0.7	12.43	0.25	17.9	1.08
BF-PSO based PI	13	16.9	7	0.4	14.27	0.2	16.2	1.04

Table 6: Performance Comparison of Set point Changes (Tank 2)

Type of Controllers	Set point (10 cm)		Set point (12 cm)		Set point (10 cm)		Set point (16 cm)	
	Peak Over Shoot (%)	*ISE*	*Peak Over Shoot (%)*	*ISE*	*Under Shoot (%)*	*ISE*	*Peak Over Shoot (%)*	*ISE*
Decentralized PI	28	17	7.9	0.3	8.6	0.23	19	1.29
BF based PI	14	13	51.6	0.7	8.2	0.36	15.4	1.21
BF-PSO based PI	10	12	7.75	0.3	14.7	0.23	16.4	0.75

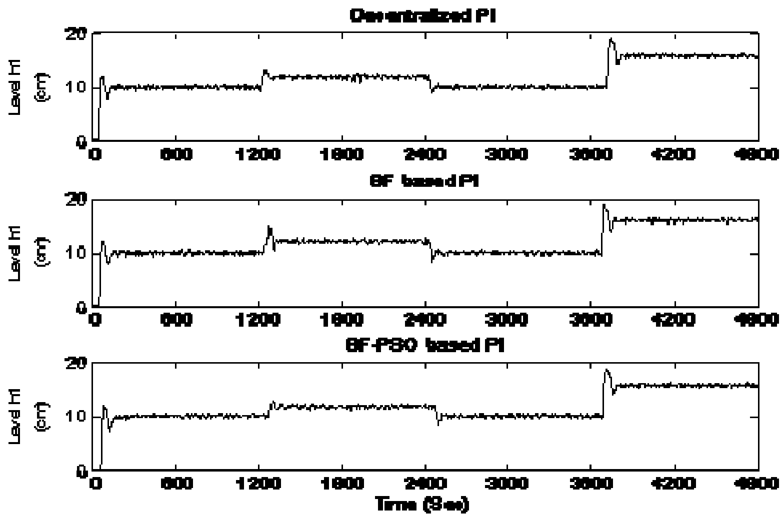

Figure 8: Setpoint tracking for the Responses of the water level (h_1)

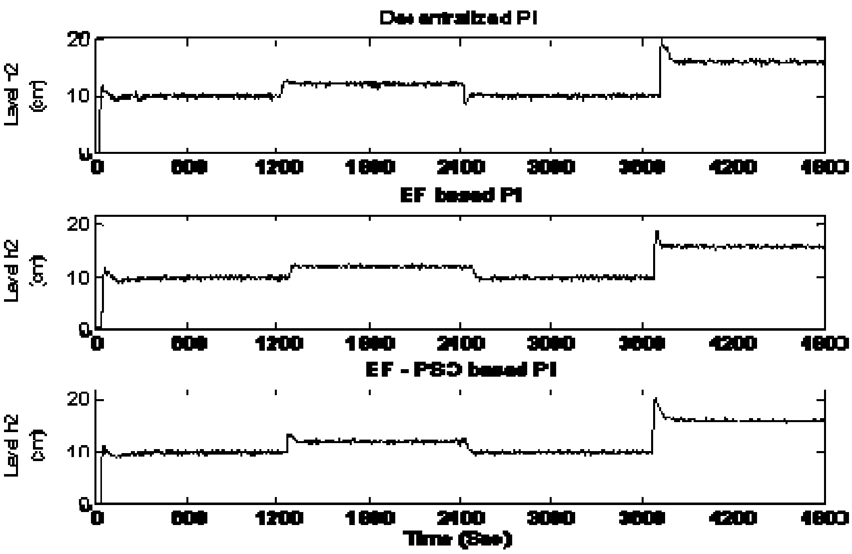

Figure 9: Setpoint tracking for the Responses of the water level (h_2)

Fig 10 and 11 shows the regulatory response of water levels h_1 and h_2. Initially the level of tank 1 and tank 2 are maintained at a steady state of 10cm. After 20 minutes, a sudden external disturbance (1000ml of water) is appended in tank 1 and tank 2 at 1200 sec. From the above response BF-PSO settles quickly and the overshoot is also less.

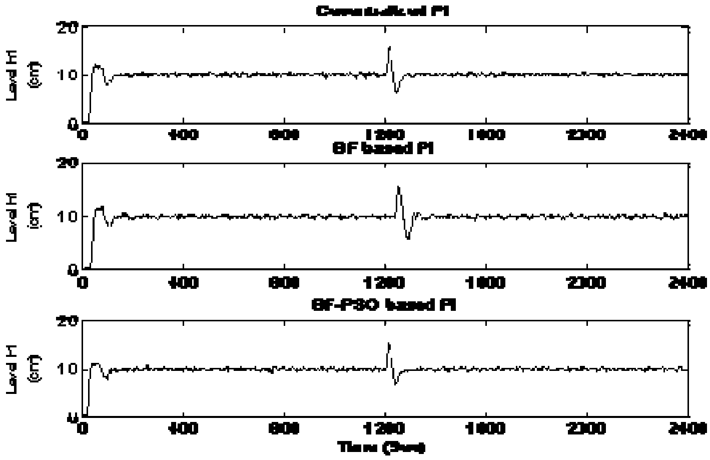

Figure 10: Regulatory Responses of the water level (h_1)

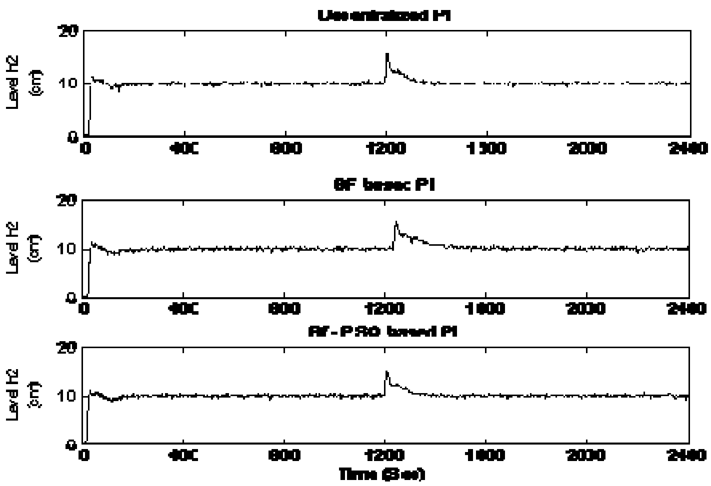

Figure 11: Regulatory Responses of the water level (h_2)

The system response of the levels h1 and h2 (Fig. 9- 12) show the robustness of the BF-PSO for both servo and regulatory operations.

From the Tables 3-6 the performance index (ISE) of Decentralized PI, BF based PI and BF-PSO based PI controllers are analyzed. It is inferred that the performance of BF-PSO based PI is better than the other two controllers.

Conclusion

The performance/robustness trade-off comparison among the decentralized, BF and BF-PSO controllers are designed to control the liquid level of the laboratory QTP. The BF-PSO responses are compared with decentralized PI and BF responses. From these responses it is observed that the ISE and ITAE values are low with BF-PSO controller than with decentralized PI and BF controller. The results show that BF-PSO controller performance is better and is robust for both servo and regulatory responses. The design of BF-PSO controller is tested for an operating condition and the servo and regulatory responses are proved and established.

References

[1] Del Valle.Y, Venayagamoorthy G.K., S.Mohagheghi, Hernandez J.-c. and Harley R.G. Partcle Swarm Optimization: Basic concepts, Variants and Applications in power systems. IEEE Transactions on Evolutionary computations 2008; 12, (2): 171-195.

[2] Dong Hwa Kim, Ajith Abraham and Jae Hoon Cho. A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Global Optimization (in Chinese), Information Sciences 2007;177: 3918–3937.

[3] Genovesi S, Mittra R., Monorchio A. and Manara G. Particle Swarm Optimization for the Design of Frequency Selective Surfaces. IEEE Transactions on Antennas and Wireless Propagation Letters 2006; 5: 277-279.

[4] Johansson K.H, Alexander Horch and Olle Wijk Anders Hansson. Teaching multivariable control using quadruple tank process. Proceedings of the 38th conference on Decision and control 1999; 807-812.

[5] Johansson K.H. The Quadruple – Tank process: A multivariable laboratory process with an adjustable zero. IEEE Transactions on Control Systems Technology 2000; 8 (3): 456-465.

[6] Korani W.M, Dorrah H.T and Emara H.M. Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning. IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA) 2009; 445 – 450.

[7] Luyben W.L. Simple method for tuning SISO controllers in multivariable systems. Industrial & Engineering Chemistry Process Design and Development 1986; 25: 654-660.

[8] Passino K.M. Biomimicry of Bacterial Foraging for Distributed optimization and control. IEEE Control System Magazine 2002; 22, (3): 52-67.

[9] Reddya B.C, Chidambaram M and Darwish M.K.Al-Gobaisib. Design of centralized controllers for a MSF desalination plant Desalination 1997; 113, (1): 27-38.

[10] Shiu S.J. and Huang S. H. Sequential design method for multivariable decoupling and multiloop PID controllers. Industrial & Engineering Chemistry Research 1998; 37: 107–119.

[11] Shneiderman D, Palmor Z.J. Properties and control of the quadruple tank process with multivariable dead times. Journal of Process Control 2010; 18-28.
[12] Tan W., Liu J.Z and Tam P. K. S. PID tuning based on loop-shaping H_∞ control. IEE Proceedings on Control Theory and Applications 1998; 145, (4): 485–490.
[13] Vahedi H, Hosseini S.H and Noroozian R. Bacterial foraging algorithm for security constrained optimal power flow. 7th International conference on the European Energy Market (EEM) June 2010; 1-6.
[14] Venkata Rao R. Advanced modeling and optimization of manufacturing processes: International research and development, Springer series in advanced manufacturing, Springer, 2010.
[15] Wang Q.G., Zou Q., Lee T.H. and Bi Q. Autotuning of multivariable PID controllers from decentralized relay feedback. Automatica 1997; 33: 319–330.
[16] Wayne Bequette B, Process control Modeling, Design and Simulation, Prentice Hall of India, New Delhi 2004.
[17] Xin-SheYang. Engineering Optimization, An Introduction with Metaheuristic Applications, John Wiley & Sons, Inc., Hoboken, New Jersey, 2010.
[18] Youxin Luo and Jianying. The controlling parameters tuning and its application of Fractional Order PID Bacterial Foraging–based Oriented by Particle swam Optimization. IEEE international conference on Intelligent Computing and Intelligent Systems Nov 2009; 4-7.
[19] Zhuang M. and Atherton D.P. PID controller design for a TITO system. IEE Proceedings on Control Theory and Applications 1994; 141: 111–120.