International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 1 (2017)

Optimal Tuning of Fractional Order PI Controller using Particle Swarm Optimization for pH process

G. Petchinathan
Department of Electronics and Instrumentation Engineering
Sri Sairam Engineering College,
Chennai, Tamilnadu, India
gpetchi@gmail.com

K. Valarmathi
Department of Electronics and communication Engineering
P.S.R Engineering College
Sivakasi, Tamilnadu, India

T. K. Radhakrishnan
Department of Chemical Engineering
National Institute of Technology,
Trichy, India

***Abstract*— In this proposed work, Fractional Order PI (FOPI) controller is used to control the pH process using Particle Swarm Optimization algorithm (PSO). The finding of optimal parameters of FOPI controller is originated as a nonlinear optimization problem, in which the objective function is formulated as the combination of Integral Time Absolute Error (ITAE), steady-state error, rise time and settling time. The performances of the PSO-FOPI and PSO-PI controller are compared with Genetic Algorithm (GA) based FOPI and PI controller. Various numerical simulations and comparisons with GA based FOPI/PID controller show that the PSO-FOPI controller can ensure better control performan**

Keywords*— *FOPI controller; PSO; Genetic Algorithm; pH Process; Performance criterion; ITAE .

I. INTRODUCTION

The Control of pH process is a very challenging task, due to its high process nonlinearity and also plays a very important role in chemical industries such as waste water treatment, polymerization reactions, fatty acid production, biochemical processes, etc. In the recent decade, many modern control techniques including neural network control, fuzzy control, optimal control, predictive control and some hybrid intelligent control have been developed extensively. Nonetheless, the Proportional-Integral-Derivative (PID) control technique has been widely used in many process industries till now. Nowadays, to enhance the performance of PID controller, the concept of fractional calculus has been introduced in the literature. Fractional-Order Proportional-Integral-Derivative (FOPID) controller has been derived from the fractional calculus by podlubny in 1997 [1, 2].

The Fractional Order Controller is the generalization of non-integer orders of conventional controllers. Many design approaches have been reported in literature for the design of FOPID controller such as frequency domain [3, 4], pole distribution [5], state space design [6], and Quantitative feedback theory [7]. Ziegler–Nichols-type rules tuning of FOPID controllers are reported in literature [8]. Fabrizio and Antonio (2011) proposed a set of tuning rules for integer order and fractional order PID controllers for the First-Order-Plus-Dead-Time (FOPDT) model of the process to minimize the integrated absolute error with a constraint on the maximum sensitivity [9]. Evolutionary algorithms based design approach has been investigated by several authors. Evolutionary algorithm such as Genetic Algorithm (GA), Choatic Ant Swarm (CAS), Differential Evolution(DE), Chaotic multi-objective optimization, Multi-objective Pareto optimization and some hybrid optimization approaches like, differential evolution and particle swarm optimization are used for the design of FOPID controllers based on the process [10-15]. In this proposed work, the PSO is employed to design FOPI controller for pH process. The FOPI controller has three parameters: proportional gain, integral gain and integral order. The objective function for determining the optimal FOPI controller parameters is formulated as the combination of Integral Time Absolute Error (ITAE), steady-state error, rise time and settling time. The proposed work is simulated with different scenarios and their performances are compared with GA.

The concept of fractional calculus is described in section II. In section III, Particle Swarm Optimization (PSO) algorithm has been explained. Section IV discusses the tuning of FOPI controller using PSO for pH process and numerical results are presented in section V. The conclusion is given in section VI.

II. THE CONCEPT OF FRACTIONAL CALCULUS

Fractional calculus is a branch of mathematical analysis that deals with the possibility of taking real number powers of the differentiation operator and integration operator. To define fractional order differentiation and integration, the generalized differ-integrator ${}_aD_t^m$ operator is introduced in

fractional calculus. where m ($m \in R$) is the fractional order and *a* and *t* are limits of operation.

There are several definitions of fractional order differentiation and integration. The most commonly used are Riemann and Liouville (RL) and Grunwald–Letnikov (GL) definitions. The GL definition is given by

$$ {}_aD_t^m f(t) = \frac{d^m f(t)}{d(t-a)^m} = \lim_{N\to\infty}\left[\frac{t-a}{N}\right]^{-m}\sum_{j=0}^{N-1}(-1)^j\binom{m}{j} f\left(t-j\left[\frac{t-a}{N}\right]\right) \quad (1)$$

The RL definition is given by

$$ {}_aD_t^m f(t) = \frac{d^m f(t)}{d(t-a)^m} = \frac{1}{\Gamma(n-m)}\frac{d^n}{dt^n}\int_0^t (t-\tau)^{n-m-1} f(\tau)d\tau \quad (2)$$

for $n-1<m<n$ and $\Gamma(.)$ is the Gamma function.

The Laplace transforms of the RL fractional differ-integral operation for zero initial conditions are given as follows

$$ L\left\{{}_0D_t^m f(t)\right\} = s^m F(s) \quad (3)$$

In practice, fractional-order differential equations do not have exact analytic solutions. Hence, the transfer functions involving fractional orders of *s* are approximated with typical (integer order) transfer functions. Several possible approximation methods are available in literature [3]. One of the well known approximations is Oustaloup which uses the recursive distribution of poles and zeros and is given by [16]

$$ s^m = k\prod_{n=1}^{N}\frac{1+\left(s/\omega_{zn}\right)}{1+\left(s/\omega_{pn}\right)}, \qquad m>0 \quad (4)$$

The approximation is valid in the frequency range [ω_l , ω_h], gain k is adjusted so that, the approximation shall have a unit gain at 1 rad / sec, the number of poles and zeros N is chosen beforehand. The best performance of approximation strongly depends on N, where as low values result in simpler approximations but also cause the appearance of a ripple in both gain and phase behaviors. Such ripples can be functionally removed by increasing N, but approximation becomes computationally heavier. Frequencies of poles and zeros in Eqn. (4) are given by

$$ \alpha = \left(\omega_h/\omega_l\right)^{\frac{m}{N}} \quad (5)$$

$$ \eta = \left(\omega_h/\omega_l\right)^{\frac{(1-m)}{N}} \quad (6)$$

$$ \omega_{z1} = \omega_1\sqrt{\eta} \quad (7)$$

$$ \omega_{zn} = \omega_{p,n-1}\eta, \qquad n = 2,.....,N \quad (8)$$

$$ \omega_{pn} = \omega_{z,n-1}\alpha, \qquad n = 1,.....,N \quad (9)$$

The case m < 0 can be dealt with inverting (4). For $|m|>1$, the approximation becomes dissatisfied. So, it is common to separate the fractional order of *s* as follows

$$ s^m = s^n s^\delta, \qquad m = n+\delta, \quad n\in Z, \quad \delta\in[0,1] \quad (10)$$

As a result, only the second term (s^δ) has to be approximated.

III. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) algorithm is a population based stochastic technique inspired by social behaviour of bird flocking or fish schooling [17]. Nevertheless, unlike GA, PSO has no evolution operators such as crossover and mutation. Compared to GA, it is easy to implement and there are few parameters to adjust. The PSO consists of a swarm of particles moving in a D dimensional search space where certain quality measures and fitness are being optimized. The position and velocity of each particle are represented by a position vector X_i $(x_{i1}, x_{i2}, x_{i3},..., x_{iD})$ and velocity vector V_i $(v_{i1}, v_{i2}, v_{i3},..., v_{iD})$, respectively. Each particle remembers its own best position in a vector P_i $(p_{i1}, p_{i2}, p_{i3},..., p_{iD})$. Where *i* is the index of that particle. The best position vector among all the neighbours of a particle is then stored in the particle as a vector P_g $(p_{g1}, p_{g2}, p_{g3},..., p_{gD})$. The velocity and position of each particle can be manipulated according to the following equations.

$$ V_{im}^{(t+1)} = wV_{im}^{(t)} + c_1r_1(p_{im} - x_{im}^{(t)}) + c_2r_2(p_{gm} - x_{im}^{(t)}) \quad (11)$$

$$ X_{im}^{(t+1)} = X_{im}^{(t)} + V_{im}^{(t+1)}, \quad m = 1,2,...,D \quad (12)$$

Where, *w* is inertia weight and also decreases linearly from 0.9 to 0.4 during a run. In this study, inertia weight (*w*) is set 0.9 [18]. Also c_1 and c_2 are the acceleration constants which influence the convergence speed of each particle and are often set to 2.0, according to the past experiences [19]. In addition to that, r_1 and r_2 are random numbers in the range of [0, 1].

IV. Tuning of FOPI Controller Using PSO for P^H Process

A. FOPI Controller

The FOPID controller is a generalization of integer order PID controller, which can be called as $PI^\lambda D^\mu$ controller. The continuous transfer function of such a controller is given by

$$ G_c(s) = K_P + K_I s^{-\lambda} + K_D s^\mu, \qquad (\lambda,\mu>0) \quad (13)$$

The time domain equation for the FOPID controller output is given by

$$ u(t) = K_P e(t) + K_I D^{-\lambda} e(t) + K_D D^\mu e(t) \quad (14)$$

From Fig.1, it can be seen that FOPID controller generalizes the integer order PID controller and expands it from point to plane. This expansion can give more flexibility to the design of controller in achieving control objectives.

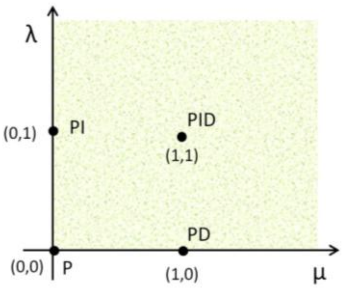

Fig. 1. Plane of FOPID controller

The integer order PID controller is obtained when λ= μ=1. If λ = 1 and μ =0, normal PI controller is obtained. In this proposed work, FOPI controller is employed for the control of pH process. The output equation and continuous transfer function of FOPI controller are obtained from the Eqns. 16 and 17 as follows:

$$u_1(t) = K_P e(t) + K_I D^{-\lambda} e(t) \tag{15}$$

$$G_{c_1}(s) = K_P + K_I s^{-\lambda}, \qquad (\lambda > 0) \tag{16}$$

B. pH process modelling

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

pH is the measurement of the acidity or alkalinity of a solution containing a proportion of water. It is mathematically defined for dilute solution, as the negative decimal logarithm of the hydrogen ion concentration $[H^+]$ in the solution, that is

$$pH = -\log_{10}[H^+] \tag{17}$$

The pH process consists of neutralization of two monoprotic reagents of a weak acid (acetic acid) and a strong base (sodium hydroxide). The method uses mass balances on components called reaction invariants of the solution in the Continuous Stirred Tank Reactor (CSTR) as shown in Fig.2. The CSTR has two inlet streams: the influent process stream, the titrating stream and one effluent stream at the output. The model of the pH neutralization process used in this work follows the method proposed by McAvoy et al (1972) and is given below [20]. Assumption of perfect mixing is general in the modeling of pH processes. Material balances in the reactor can be given by

$$V\frac{dx_a}{dt} = F_a C_a - (F_a + F_b)x_a \tag{18}$$

$$V\frac{dx_b}{dt} = F_b C_b - (F_a + F_b)x_b \tag{19}$$

Where F_a is the flow rate of the influent stream, F_b is the flow rate of the titrating stream, C_a is the concentration of the influent stream, C_b is the concentration of the titrating stream, x_a is the concentration of the acid solution, x_b is the concentration of the basic solution and V is the volume of the mixture in the CSTR.

The Eqns. 18 and 19 describe the concentration of the acidic and basic components x_a and x_b which change dynamically subject to the input streams F_a and F_b. Consider acetic acid (weak acid) denoted by HAC and is neutralized by a strong base NaOH (sodium hydroxide) in water. The reactions are

$$H_2O \Leftrightarrow H^+ + OH^- \tag{20}$$

$$HAC \Leftrightarrow H^+ + AC^- \tag{21}$$

$$NaOH \Leftrightarrow Na^+ + OH^- \tag{22}$$

According to the electro neutrality condition, the sum of the charges of all ions in the solution must be zero, i.e.

$$[Na^+] + [H^+] = [AC^-] + [OH^-] \tag{23}$$

Here, [X] denotes the concentration of the X ion. The equilibrium relations also hold water and the acetic acid.

$$K_a = \frac{[AC^-][H^+]}{[HAC]} \tag{24}$$

defining, $X_a = [HAC] + [AC^-]$ and $X_b = [NA^+]$ using the Eqns. (23) and (24)

$$[H^+]^3 + [H^+]^2\{K_a + X_b\} + [H^+]\{K_a(X_b - X_a) - K_w\} - K_w K_a = 0 \tag{25}$$

Let $pH = -\log_{10}[H^+]$ and $pK_a = -\log_{10} K_a$.

The equation for titration is given by

$$X_b + 10^{-pH} - 10^{-pH^{-14}} - \frac{X_a}{1 + 10^{pKa - pH}} = 0 \tag{26}$$

Where, K_a and K_w are dissociation constant of acetic acid at 25ºC ($K_a = 1.778*10^{-5}$ & $K_w = 10^{-14}$). Fig.3 shows the titration curve drawn between acid / base and pH using the Eqn. 26. It is strictly static nonlinear relation between the states X_a, X_b and the output pH variable, and it manifests itself as the familiar titration curve of the neutralization process.

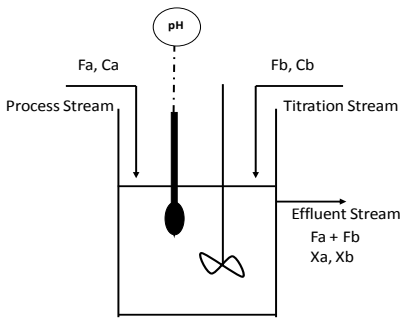

Fig. 2. pH Neutralization Process

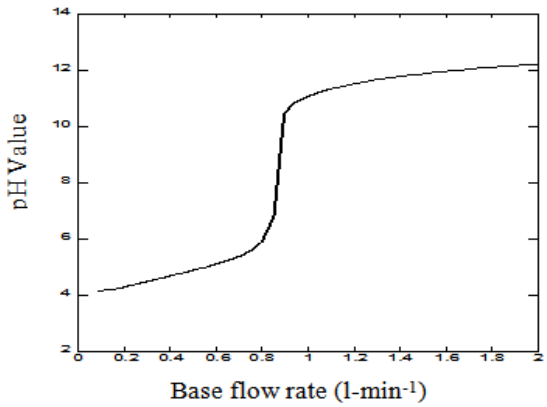

Fig. 3. Titration curve for weak acid neutralized by strong base

C. Performance Criterion

To formulate the objective function for optimum design of FOPI controller, fitness of each bat must be evaluated based on certain performance criterion. Several performance criteria are proposed in the literature for the design of controllers such as Mean Square Error (MSE), Integral of Squared Error (ISE), Integral of Time Squared Error (ITSE), Integral of Absolute Error (IAE) and Integral Time Absolute Error (ITAE) [21-23]. However, there are some drawbacks associated with these criteria as mentioned in Gaing; 2004, Zamani et al; 2009 and Tang et al; 2012[24, 25, 11]. The IAE and ISE criteria produce a response relatively to small overshoot and long settling time. Even though ITSE criterion can overcome these disadvantages, it cannot guarantee desirable stability margin [24]. Das et al; 2011 have stated that, ITAE criterion is capable of providing closed loop response with low overshoot and fast response for FOPID controller tuning [26]. The equation for Integral Time - Absolute Error (ITAE) is given by,

$$ITAE = \int_0^\infty t|e(t)|dt \tag{27}$$

In this work, a new performance criterion in the time domain is proposed for design of FOPI controller for the control of pH process. This performance criterion consists of ITAE, rise time (t_r) , settling time (t_s) and steady-state error (E_{ss}). The ITAE has to compute numerically and the time limit for the integral is set from *0* to *T* which is chosen sufficiently as large with the intention that, error is negligible for *t* is greater than *T*. The proposed performance criterion *J(k)* is defined as

$$J(k) = (W_1 * ITAE) + (W_2 * E_{ss}) + (W_3 * t_r) + (W_4 * t_s) \tag{28}$$

Where, *k* is the parameter of FOPI controller $\{k_P, k_I, \lambda\}$

The importance of each criterion in Eqn.31 is determined by a weight factor W_j. The user can set the weight factors properly in order to attain the desired requirements. In this study, weight factors selections are W_1=1000 , W_2=100 and W_3=W_4=1. A change in W_j gives improvement in the corresponding characteristics at the expense of degrading other characteristics.

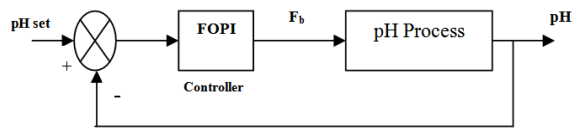

Fig. 4. Block diagram of pH process with FOPI controller

D. FOPI Controller design using PSO

The PSO algorithm is employed to determine the optimal parameters $\{k_P, k_I, \lambda\}$ of FOPI controller by optimizing a proposed performance criterion. Here, FOPI controller is used to improve the transient step response of pH process. Initially, objective function is formulated as minimization of J(k) as mentioned in Eqn. 30. To make sure the stability of closed loop, the objective function is penalized with a penalty function M(k) and is given by

$$M(k) = \begin{cases} Q & if \quad k \ is \ unstable \\ 0 & otherwise \end{cases} \tag{29}$$

where, Q is a large positive real number.

Therefore, fitness function for this optimization problem is obtained as

$$f(k) = J(k) + M(k) \tag{30}$$

The design steps of PSO-FOPI controller for pH process are as follows.

1. Randomly initialize the population (position points and velocities)
2. Calculate the values of the performance criterion in Eqn.30 for each initial particle k of the population.
3. Compare each particle's evaluation value with its personal best Pi. The best evaluation value among the Pi is denoted as Pg.
4. Modify the member velocity of each particle k according to Eqn. 11.
5. Modify the member position of each particle k according to Eqn. 12.
6. If the number of iterations reaches the maximum, then go to Step7, otherwise go to Step2.
7. The latest Pg is the optimal controller parameter.

V. Numerical Results

A. Related Parameters

The model parameters of the pH process are listed in Table I. In this study, acid flow rate (F_a) is kept constant as 0.192 l - min $^{-1}$ and base flow rate (F_b) is the manipulated variable for the control of pH. The lower and upper bounds of the each FOPI controller parameter are $0 \le k_P, k_I \le 50$ and $0 \le \lambda \le 2$ .In

Oustaloup approximation, $\omega_l = 0.001\omega_{gc}$ and $\omega_h = 1000\,\omega_{gc}$, where, ω_{gc} is the gain cross over frequency and the order of approximation (N) is set to 5. The block diagram and MATLAB – Simulink diagram of pH process model with FOPI controller are shown in Figs.4 and 5 respectively.

TABLE I. MODEL PARAMETERS FOR THE P^H PROCESS

Parameter	Description	Value
V	Volume of the Continuous Stirred Tank Reactor	7.4l litre
F_a	Flow rate of the influent stream	0.24 l min^{-1}
F_b	Flow rate of the titrating stream	0-0.8 l min^{-1}
C_a	Concentration of the influent stream	0.2 g mol l^{-1}
C_b	Concentration of the titrating stream	0.1 g mol 1^{-1}

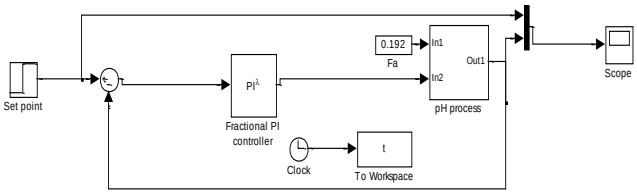

Fig. 5. MATLAB- Simulink diagram of pH process with FOPI controller

B. *Performance of PSO-FOPI Controller*

The performance of PSO-FOPI controller is also analyzed in this section. The performance of proposed PSO-FOPI controller is compared with GA based FOPI controller. The parameters for the GA training are as follows: number of generations = 30, population size = 10, crossover probability = 0.8 and mutation probability = 0.08. The PSO parameters are selected as follows: dimension (d) =3, population size=50, maximum number of bird step = 5, cognitive factor (C_1) = 1.2, social acceleration factor (C_2) = 1.2, inertia weight factor (w) = 0.9, maximum time limit for integral is set to T=4 sec and Q in Eqn. 29 is set to 10^{10}. The GA and PSO algorithm are executed for ten runs to get best controller parameters using the performance criterion mentioned in Eqn.30.

TABLE II. PERFORMANCE MEASURES AND BEST CONTROLLER PARAMETERS OF VARIOUS FOPI CONTROLLERS

Set point	Controller	K_p	K_i	λ	ITAE	t_r	t_s	Ess
7	GA-FOPI	7.283	26.00	1.79	13.63	0.033	0.221	0.211
	PSO - FOPI	30.108	33.83	0.44	0.89	0.054	0.171	0.029
9	GA-FOPI	14.467	43.05	0.01	1.58	0.279	0.497	0.009
	PSO - FOPI	29.368	25.12	1.03	1.42	0.196	0.414	0.018

Figs.6 and 7 shows the step response of the pH process controlled by GA-FOPI and PSO-FOPI controller for the set point of pH 7 and 9 respectively. The performance measures and the best controller parameters in the time domain are listed in Table II. It can be seen from Figs.6,7 and Table II that the performance of PSO-FOPI controller is better than the GA-FOPI controller for both the set point changes

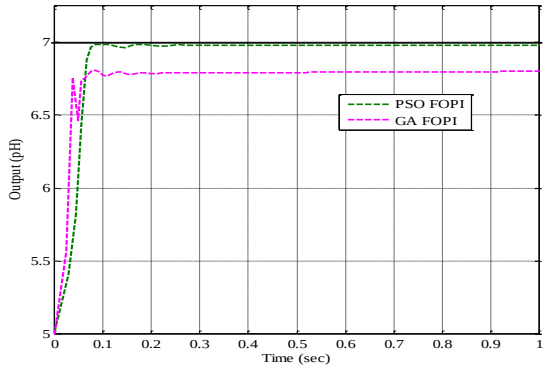

Fig. 6. Step response of pH process controlled by PSO-FOPI and GA- FOPI Set point of pH =7

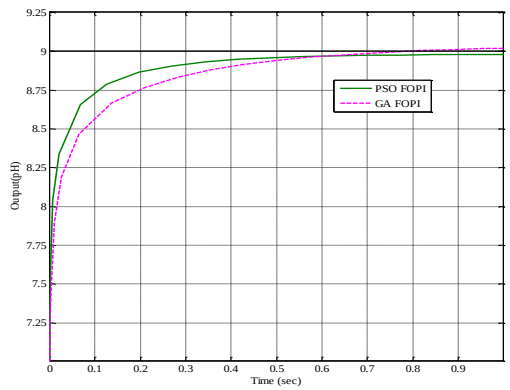

Fig. 7. Step response of pH process controlled by PSO-FOPI and GA- FOPI Set point of pH =9

C. *Comparision with other PI controller*

The performance of PSO-FOPI controller is also compared with GA-PI and PSO-PI controllers. The step responses of pH process controlled by GA-PI, PSO-PI and PSO-FOPI controller for the set point of pH 7 and 9 are shown in Figs. 8 and 9. The performance measures and the best controller parameters in the time domain are listed in Table III. As shown in Figs. 8, 9 and Table III, the performance of PSO-FOPI controller has better performance with smaller ITAE, rise time, settling time and steady state error compared to GA-PI and PSO-PI controller for the both the set points.

TABLE III. PERFORMANCE MEASURES AND BEST CONTROLLER PARAMETERS OF VARIOUS PI CONTROLLERS WITH PSO-FOPI CONTROLLER

Set point	Controller	K_p	K_i	λ	ITAE	t_r	t_s	Ess
7	GA-PI	25.318	33.38	-	14.51	0.063	0.523	0.187
	PSO - PI	23.192	35.83	-	0.978	0.068	0.167	0.022
9	PSO - FOPI	30.108	33.83	0.441	0.897	0.054	0.171	0.029
	GA-PI	10.899	5.76	-	4.43	0.344	2.062	0.041

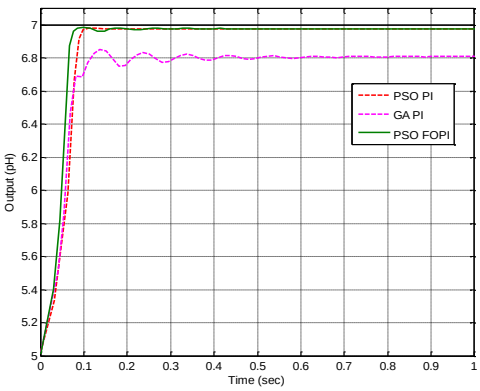

Fig. 8. Step response of pH process with various PI controllers and PSO-FOPI controller Set point of pH =7

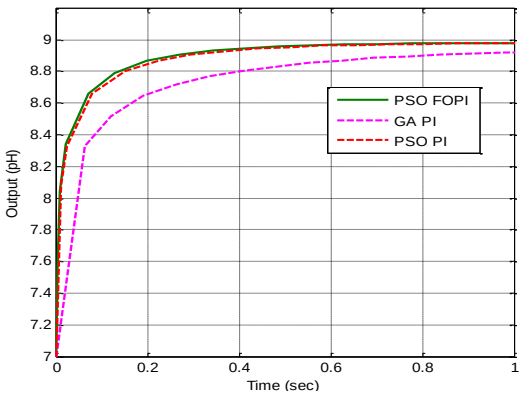

Fig. 9. Step response of pH process with various PI controllers and PSO-FOPI controller Set point of pH =9

D. Robustness

Finally, robustness of PSO-FOPI controller is demonstrated with the disturbance rejection problem for set point 7 and 9. The task of FOPI controller is to maintain a constant pH value of 7 and 9, when some unmeasured input disturbances act on the acid input stream. To maintain a constant pH value 7, the first input unmeasured disturbance is created by increasing the acid flow rate by 1 l-min^{-1} from 1.26s to 1.33s. This is equivalent to 5 times of normal acid flow rate. The second disturbance is done by reducing the acid flow rate by 1 l-min^{-1} from 1.98s to 2.08s. Then to maintain a constant pH value 9, the first input unmeasured disturbance is created by increasing the acid flow rate by 3 l-min^{-1} from 1.26s to 1.33s. This is equivalent to 15 times of normal acid flow rate. The second disturbance is done by reducing the acid flow rate by 3 l-min^{-1} from 1.98s to 2.08s. From Figs.10 and 11, it can be seen that the PSO-FOPI controller reacts quickly with a change of the base stream compared to PSO-PI controller while a deviation in the pH value arises. Moreover, it can be observed that the disturbance does not change the pH value outlying from the set point. The PSO-FOPI controller is more robust to disturbances than the PSO-PI controller.

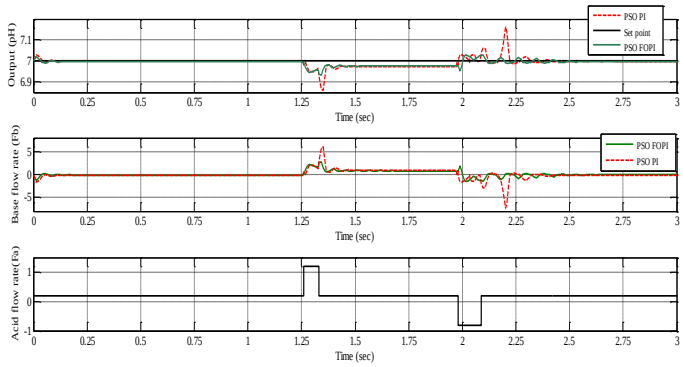

Fig. 10. Simulation result of disturbance rejection problem with PSO-FOPI and PSO-PI for the set point 7

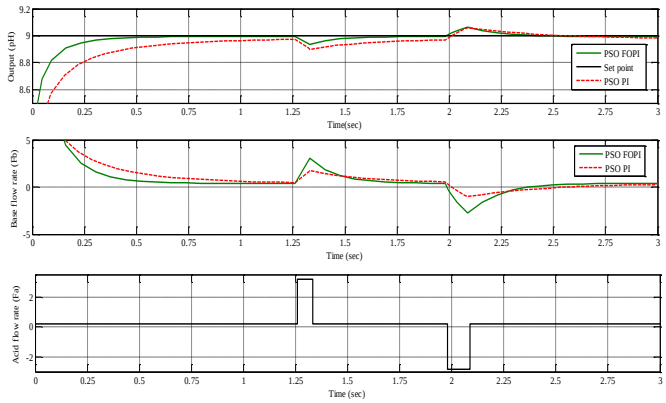

Fig. 11. Simulation result of disturbance rejection problem with PSO-FOPI and PSO -PI for the set point 9

VI. Conclusions

In this work, a design method for tuning of FOPI controller parameters using PSO is presented for the control of pH process. The proposed method determines the optimal parameters of FOPI controller by solving the optimization problem for minimizing the objective function comprising ITAE, rise time, settling time and steady state error. The graphical and numerical results of simulation show that the designed PSO-FOPI controller has better performance than GA based FOPI and PI controllers. In addition, from Fig. 8 and 9, it can be concluded that PSO-FOPI controller is more robust to input unmeasured disturbances.

References

[1] Podlubny, I., Dorcak, L., Kostial, I. (1997). On Fractional Derivatives, Fractional-Order Dynamic Systems and PIλDμ controllers, Proceedings of the 36th Conference on Decision & Control, San Diego, California, USA, December 1997.

[2] Podlubny, I. (1999). Fractional-Order Systems and PIλDμ Controllers, IEEE Transactions on Automatic Control, Vol. 44, No. 1, January 1999, pp. 208-214.

[3] Vinagre, B. M., & Podlubny, I. (2000). Some approximations of fractional order operators used in control theory and applications. Fractional Calculus & Applications and Analysis, 3, 231–248.

[4] Monje, C. A., Vinagre, B. M., Feliu, V., & Chen, Y. (2008). Tuning and auto-tuning of fractional order controllers for industry applications. Control Engineering Practice, 16, 798–812.

[5] Petras,I.(1999).The fractional order controllers: Methods for their synthesis and application. Electrical Engineering Journal, 50(9–10), 284–288.

[6] Dorcak, L., Petras,I., Kostial,I and Terpak,J.(2001).State-space controller designfor the fractional-order regulated system in Proceedings of ICCC (pp. 15–20).

[7] Cervera, J., Banos, A., Monje, C., Vinagre, B., (2006). Tuning of fractional PID controllers by using QFT in 32nd annual conference on IEEE industrial electronics, IECON 2006 (pp. 5402–5407).

[8] Valerio, D., and SaDaCosta,J.(2006).Tuning of fractional PID controllers with Ziegler–Nichols-type rules. Signal Processing Journal, 86(10), 2771–2784.

[9] Fabrizio P, Antonio V (2011). Tuning rules for optimal PID and fractional-order PID controllers. Journal of Process Control 21, 69–81.

[10] Zhang Y, Li J (2011). Fractional-order PID controller tuning based on genetic algorithm, in proceedings of International Conference on Business Management and Electronic Information (BMEI), Volume 3, 764 - 767 .

[11] Tang Y, Mingyong Cui, Changchun Hua, Lixiang Li, Yixian Yang (2012), Optimum design of fractional order PIλDμ controller for AVR system using chaotic ant swarm. Expert Systems with Applications 39 , 6887–6896.

[12] Biswas, Das, Abraham, SambartaDasgupta (2009). Design of fractional-order PIλDμ controllers with an improved differential evolution. Engineering Applications of Artificial Intelligence 22, 343–350.

[13] Pan I, Das S (2013). Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization, Electrical Power and Energy Systems 51 106–118.

[14] Hajiloo A, N. Nariman-zadeh, Ali Moeini (2012). Pareto optimal robust design of fractional-order PID controllers for systems with probabilistic uncertainties, Mechatronics 22 788–801.

[15] Maione G, Punzi A (2013). Combining differential evolution and particle swarm optimization to tune and realize fractional-order controllers, Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences, 19:3, 277-299

[16] Oustaloup, A. (1991). La commande CRONE: commande robuste d'order non entier.Hermès, Paris.

[17] Kennedy,J.,& Eberhart,R.(1995),Particle swarm optimization .In Proceedings of IEEE international conference on neural networks Vol. 4,pp.1942–1947.

[18] Shi, Y., & Eberhart,R.(1998).A modified particle swarm optimizer. In Proceedings of IEEE international conference on evolutionary computation pp. 69–73.

[19] Eberhart,R., & Shi,Y.(2001), Particle swarm optimization :Developments, applications and resources. In Proceedings of IEEE international conference on evolutionary computation pp. 81–86.

[20] McAvoy T.J. and Hsu, E. (1972) 'Dynamics of pH in Controlled Stirred Tank Reactor', Ind. Eng. Chem. Process Des. Dev., Vol. 1, 68, pp.114-120.

[21] Petchinathan G, G. Saravanakumar, K. Valarmathi, and D. Devaraj (2011), Hybrid PSO - Bacterial Foraging Based Intelligent PI Controller Tuning for pH Process. In proceedings of the InConINDIA 2012, AISC 132, pp. 515–522.

[22] Maiti, Acharya, Chakraborty, Konar , Janarthanan (2008) Tuning PID and PIλDδ Controllers using the Integral Time Absolute Error Criterion, proceedings of 4th International Conference on Information and Automation for Sustainability (ICIAFS) 457-462.

[23] Zhuang, M, Atherton, D.P (1991), Tuning PID controllers with integral performance criteria, in proceedings of international Conference on control 1991

[24] Gaing, Z. L. (2004). A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Transactions on Energy Conversion, 19, 384–391.

[25] Zamani, M., Karimi-Ghartemani, M., Sadati, N., Parniani, M. (2009). Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Engineering Practice: Vol. 17. pp. 1380–1387.

[26] Das S, S Saha, S Das, A Gupta (2011), On the selection of tuning methodology of FOPID controllers for the control of higher order processes, ISA transactions-elsevier 1-27.