International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:8, No:12, 2014

CSTR Control by Using Model Reference Adaptive Control and PSO

Neha Khanduja

***Abstract*—**This paper presents a comparative analysis of continuously stirred tank reactor (CSTR) control based on adaptive control and optimal tuning of PID control based on particle swarm optimization. In the design of adaptive control, Model reference adaptive control (MRAC) scheme is used, in which the adaptation law have been developed by MIT rule & Lyapunov's rule. In PSO control parameters of PID controller is tuned by using the concept of particle swarm optimization to get optimized operating point for minimum integral square error (ISE) condition. The results show the adjustment of PID parameters converting into the optimal operating point and the good control response can be obtained by the PSO technique.

***Keywords*—**Model reference adaptive control (MRAC), optimal control, particle swarm optimization (PSO).

I. INTRODUCTION

DURING the past decades, there has been a great advancement in the process control. Various control methods such as Adaptive control, neural control, fuzzy control and optimal control have been studied. In this paper two methods i.e. adaptive control and optimal control by using PSO has been discussed.

An adaptive controller is a mechanism of some system identification to obtain a model and then design a controller. Adaptive controller can be apply for modify its behavior in response to the changing dynamics of the process and the character of the disturbances. The base element of all the approaches is that they have the ability to adapt the controller to accommodate changes in the process. This permits the controller to maintain a required level of performance in spite of any noise, nonlinearity or fluctuation in the process. An adaptive system has maximum application when the plant undergoes transitions or exhibits non-linear behavior and when the structure of the plant is unknown. Adaptive is called a control system, which can adjust its parameter automatically to compensate for variations in the characteristics of the process it control.

Particle Swarm Optimization (PSO), first introduced by Kennedy and Eberhart, is one of the modern heuristic algorithms. It was developed through simulation of a simplified social system. It has been found to be robust in solving continuous nonlinear optimization problems. The PSO technique is used to generate a high-quality solution within shorter calculation time and stable convergence characteristic than other stochastic methods. PSO method is an excellent optimization methodology and a promising approach for solving the optimal PID controller parameters. Therefore, this study develops the PSO-PID [1], [10].

Neha Khanduja is with Bhagwan Parshuram Institute of Technology, Delhi, India (e-mail: nehakhanduja.dce@gmail.com).

II. DEVELOPMENT OF MATHEMATICAL MODELING

A perfectly mixed continuously stirred tank reactor (CSTR) as shown in Fig. 1 with first order exothermic irreversible reaction $A \rightarrow B$ is considered. In this a fluid stream is continuously fed to the reactor and other fluid stream is continuously removed from the reactor. A jacket surrounding the reactor has in feed and exit streams. The jacket is assumed to be perfectly mixed and at lower temperature than the reactor [4]-[6].

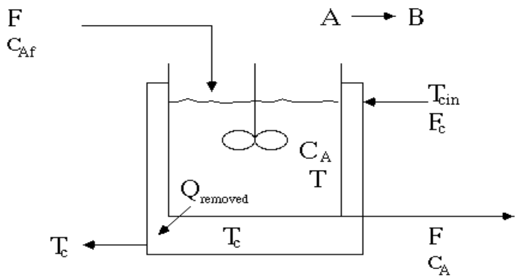

Fig. 1 Continuously Stirred Tank Reactor

Following assumptions has been made for CSTR:

- Perfect mixing (product stream values are the same as the bulk reactor fluid)
- Constant volume
- Constant parameter values

A. State Variable form of Dynamic Equations

In state variable form equations can be written as:

$$f_1(C_A, T) = \frac{dC_A}{dt} = \frac{F}{V}(C_{Af} - C_A) - r \quad (1)$$

$$f_2(C_A, T) = \frac{dT}{dt} = \frac{F}{V}\left(T_f - T\right) + \left(\frac{-\Delta H}{\rho c_p}\right) r - \frac{UA}{V\rho c_p}(T - T_j) \quad (2)$$

The reaction rate per unit volume (Arrhenius expression) is:

$$r = k_o \exp\left(\frac{-\Delta E}{RT}\right) C_A$$

where it is assumed that the reaction is first-order [5].

B. Steady-State Solution

The steady-state solution is obtained when $\frac{dC_A}{dt} = 0$ and $\frac{dT}{dt} = 0$, that is:

$$f_1(C_A,T) = 0 = \frac{F}{V}(C_{Af} - C_A) - k_o \mathrm{e}^{-\Delta E/\mathrm{RT}} C_A$$

The linear model of the system is obtained as:

$$\mathrm{X}= \begin{bmatrix} -\frac{F}{V} - k_s & -C_{As}k_s' \\ \frac{-\Delta H}{\rho c_p} k_s & -\frac{F}{V} - \frac{UA}{V\rho c_p} + \left(\frac{-\Delta H}{\rho c_p}\right) C_{As}k_s' \end{bmatrix} \begin{bmatrix} C_A \\ T \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{UA}{V\rho c_p} \end{bmatrix} [T_j] \qquad (3)$$

TABLE I
REACTOR PARAMETERS

Reactor Parameter	Description	Values
F/V(hr-1)	Flow rate*reactor volume of tank	1
K_o(hr-1)	Exponential factor	$10e^{15}e^{12}$
-ΔH (kcal/kmol)	Heat of reaction	6000
E(kcal/kmol)	Activation energy	12189
ρC_P (BTU/ ft^3)	Density*heat capacity	500
Tf(K)	Feed temperature	312
C_{Af}(lbmol/ft^3)	Concentration of feed stream	10
$\frac{UA}{V}$	Overall heat transfer coefficient/reactor volume	1451
Tj(K)	Coolant Temperature	300

III. MODEL REFERENCE ADAPTIVE CONTROL

Model reference adaptive system is an important adaptive controller. It can be considered as an adaptive servo system in which the desired performance is expressed in terms of a reference model, which gives the desired response to a command signal as shown in Fig. 2. The system has two loops. One is normal feedback loop which consist of the process and the controller and other is parameter adjustment loop that changes the controller parameters. The parameters are changed on the basis of feedback from error, which is the difference between the output of system and the output of reference model [2]. The ordinary feedback loop is called the inner loop and the parameter adjustment loop is called the outer loop. The mechanism for adjusting the parameters in a model reference adaptive system can be obtained in two ways by using a gradient method or by applying Lyapunov stability theory [3]. MRAC is composed of four parts: a plant containing unknown parameters, a reference model for compactly specifying the desired output of control system, a feedback control law containing adjustable parameters [1], [3].

A. MIT Rule

The MIT rule, also known as the gradient method, changes the parameters based upon the gradient of the error, with respect to that parameter. The parameters are changed in the direction of the negative gradient of the error. This means that if the error, with respect to a specified parameter, is increasing then by the MIT rule the value of that parameter will be decreased. This control system consists of a reference model, an adjustment mechanism and a controller. The reference model describes the desired input/output dynamics of the closed loop. The controller derives the control signal so that the plant's closed-loop characteristics from the command signal to the plant output are the same as the dynamics of the reference model. The convergence of the modeling error to zero for any given command signal is assured when plant output exactly follows the output of the model.

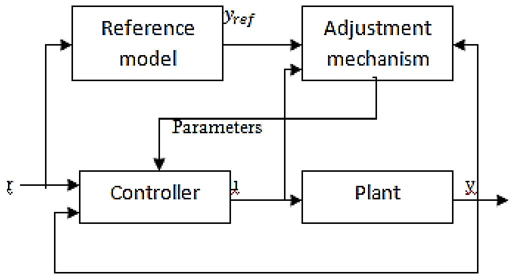

Fig. 2 Model Reference Adaptive Control

The modeling error e is given by:

$$e = Y - Ym \qquad (4)$$

The controller parameters are adjusted with the loss function J (θ):

$$J(\theta) = \frac{1}{2} e^2$$

To minimize J, the parameters can be changed in the direction of negative gradient of J. The rate of change of controller parameters (θ) with respect to time is defined by (5) where the adaptation gain is defined by γ [2], [3], [6], [7].

$$\frac{d\theta}{dt} = -\gamma \frac{dJ}{d\theta} = -\gamma e \frac{\partial e}{\partial \theta}$$

A. Lyapunov Rule

An alternative approach to the MIT rule is to use a Lyapunov based method, which avoids the stability problems present in the gradient approaches. In order to derive an update law using Lyapunov theory, the following Lyapunov function is defined as:

$$V = \frac{1}{2}\gamma e^2 + \frac{1}{2b}(b\theta_1 - b_m)^2 + \frac{1}{2b}(b\theta_2 + a - ba_m)^2 \qquad (6)$$

The time derivative of V can be found as:

$$\dot{V} = \gamma e \dot{e} + \dot{\theta}_1 (b\theta_1 - b_m)^2 + \dot{\theta}_2 (b\theta_2 + a - ba_m) \qquad (7)$$

Moreover, its negative definiteness would guarantee that the tracking error converge to zero along the system trajectories [6], [7].

B. Adaption Law

The adaption law attempts to find a set of parameters that minimize the error between the plant and the reference model outputs. For this purpose the parameters of the controller are inclemently adjusted until the error has reduced to zero. In this paper MIT rule approach and Lyapunov rule approaches have been used [6], [7].

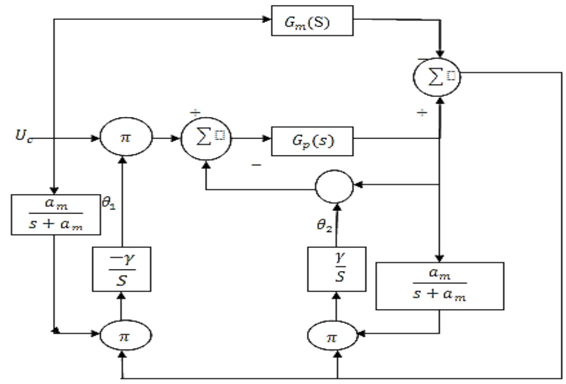

Fig. 3 Block diagram of MIT Rule

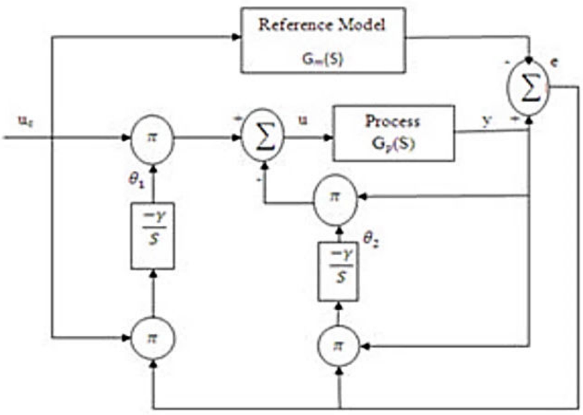

Fig. 4 Block diagram of MRAC based on Lyapunov Theory

IV. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a computational algorithm technique based on swarm intelligence. This method is motivated by the observation of social interaction and animal behaviors such as fish schooling and bird flocking. It mimics the way they find food by the cooperation and competition among the entire population [13]. A swarm consists of individuals, called particles, each of which represents a different possible set of the unknown parameters to be optimized. The "swarm" is initialized with a population of random solutions [14]. In a PSO system, particles fly around in a multi-dimensional search space adjusting its position according to its own experience and the experience of its neighboring particle. The goal is to efficiently search the solution space by swarming the particles towards the best fitting solution encountered in previous iterations with the intention of encountering better solutions through the course of the process and eventually converging on a single minimum or maximum solution [15]. The performance of each particle is measured according to a pre-defined fitness function, which is related to the problem being solved. The use of PSO has been reported in many of the recent works [16] in this field. PSO has been regarded as a promising optimization algorithm due to its simplicity, low computational cost and good performance [12]. The model of the process under study is very important for its tuning as the accuracy of the tuned controller parameters is greatly dependent on the degree of accuracy of the system model with that of the real system. As per the fundamentals it is possible to approximate the actual input-output mathematical model of a very-high-order, complex dynamic process with a simple model consisting of a first or second order process combined with a dead-time element [16].

PSO is derived from the social-psychological theory, and has been found to be robust in complex systems. In PSO Each particle is considered as a valueless particle in g-dimensional search space, and keeps track of its coordinates in the problem space associated with the best evaluating value and this value is called *pbest.* The overall best value and its location obtained so far by any particle in the group is tracked by the global version of the particle swarm optimizer *gbest*. The PSO concept consists of changing the velocity of each particle toward its pbest and gbest locations at each time step. for example, the jth particle is represented as $x_j = (x_{j.1}, x_{j.2}, \ldots, x_{j.g})$ in the g-dimensional space. The best previous position of the jth particle is recorded and represented as $pbest_j = (pbest_{j.1}, pbest_{j.2} \ldots pbest_{j.g})$. The index of best particle among all particles in the group is represented by the $gbest_g$. The rate the position change (velocity) for particle j is represented as $v_j = (v_{j.1}, v_{j.2} \ldots v_{j.g})$. The modified velocity and position of each particle can be calculated using the current velocity and distance from $pbest_{j.g}$ to $gbest_g$ as:

$$v_{j.g}^{(t+1)} = w.v_{j.g}^{(t)} + c_1 * rand\,() * (Pbest_{j.g} - x_{j.g}^{(t)}) + c_2 * rand\,() * (gbest_g - x_{j.g}^{(t)})$$

$$x_{j.g}^{(t+1)} = x_{j.g}^{(t)} + v_{j.g}^{(t+1)}$$

$$j = 1, 2 \ldots\ldots n$$
$$g = 1, 2 \ldots m$$

where n: number of particles in a group; m: number of members in a particle; t pointer of iterations (generations); $v_{j.g}^{(t)}$ velocity of particle *j* at iteration *t, W* inertia weight factor;c_1, $c_2$acceleration constant; *rand ( )* random number between 0 and 1;$x_{j.g}^{(t)}$ current position of particle *j* at iteration *t*; $pbest_j$ pbest of particle j; gbest gbest of the group.

The parameter v_g^{max} determined the resolution, or fitness, with which regions were searched between the present position and the target position. If v_g^{max} is too high, particles might fly past good solutions but if v_g^{max} is too low, particles may not explore sufficiently beyond local solutions.

The constant c_1 and c_2 represent the weighting of the stochastic acceleration terms that pull each particle toward *pbest* and *gbest.* c_1 and c_2 has been set to be 2.0. This is done because low values allow particle to fly far from the target region before being tugged back while high values result in abrupt movement toward or past target regions. Generally, the inertia weight w is set according to (8):

$$w=\frac{w_{\max}-w_{min}}{iter_{max}} * iter \quad (8)$$

Suitable selection of w provides a balance between global and local explorations, thus it requires less iteration on average to find a sufficiently optimal solution where $iter_{max}$ is the maximum number of iterations or generations and iter is the current number of iterations.

It is a very simple concept, and paradigms can be implemented in a few lines of computer code. It requires only primitive mathematical operators, and is computationally inexpensive in terms of both memory requirements and speed. Early testing has found the implementation to be effective with several kinds of problems [8]-[10], [13]-[16].

A. Optimal Tuning of PID Controller Using PSO

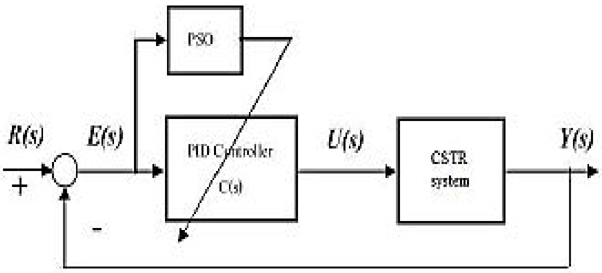

Fig. 5 Block diagram of optimal PID controllers with PSO for CSTR

The parameters of PID controller are tuned by PSO to get optimal PID parameters and by applying PSO-PID controller an excellent output response can be obtained.

Performance characteristic of evaluation function includes overshoot, rise time, settling time and static error time. The evaluation function is defined by (9), to compute the evaluation value of each particle in swarm according to control performance.

The sequence of steps to study the PSO-PID controller for the CSTR system is given below:

Step 1. Specify the lower and upper bounds of Kp, Ki, and Kd also randomly initialize the particles of the swarm including swarm size, iteration, acceleration constant, inertia weight factor, the position matrix x_j and the velocity matrix v_j and so on.

Step 2. Calculate the evaluation value of each particle using the evaluation function given in 9.

Step 3. Compare each particle's new fitness value with its personal best position fitness value, and update the personal best position pbest.

Step 4. Search for the best position among all particles personal best position, and denote the best position as gbest.

Step 5. Update the velocity v_i of each particle according to (6), and update the particle position matrix according to (7).

Step 6. Update control parameter.

Step 7. If the number of iterations reaches the maximum, then stop. The latest gbest is regarded as the optimal PID controller parameter. Otherwise, go to step 2 [11], [12].

B. Performance Indices

A performance index is a quantitative measure of the performance of the system. A system is considered an optimal control system when the system parameters are adjusted so that the index reaches an extreme value, commonly a minimum value [7].

A suitable performance index is the integral of the square of the error, ISE, which is defined as:

$$ISE = \int_0^T e^2(t)dt$$

ISE is more suitable to minimize initial large amount of errors. The squared error is mathematically more convenient for analytical and computational purposes.

Another readily instrumented performance criterion is the integral of the absolute magnitude of the error, IAE, which is written as:

$$IAE = \int_0^T |e(t)|dt$$

This index is particularly useful for computer simulation studies. To reduce the contribution of the large initial error to the value of the performance integral, as well as to emphasize errors occurring later in response, the integral of time multiplied by absolute error, ITAE has been proposed, which is defined as:

$$ITAE = \int_0^T t|e(t)|dt$$

Other performance criteria include evaluation of rise time, settling-time and peak overshoot [12].

V. Simulation Results

A. MIT Rule

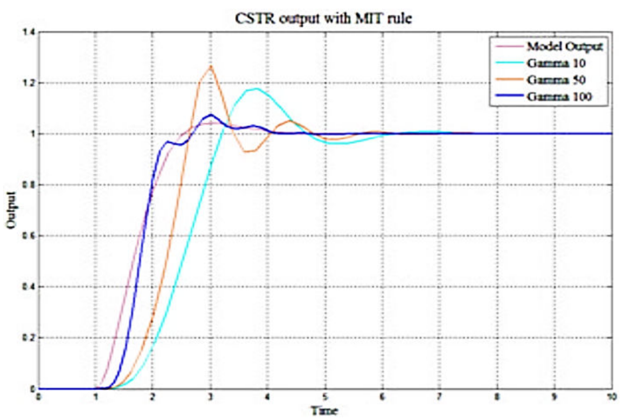

Fig. 6 CSTR output with MIT' rule for different values of adaption Gain

B. Lyapunov'sRule

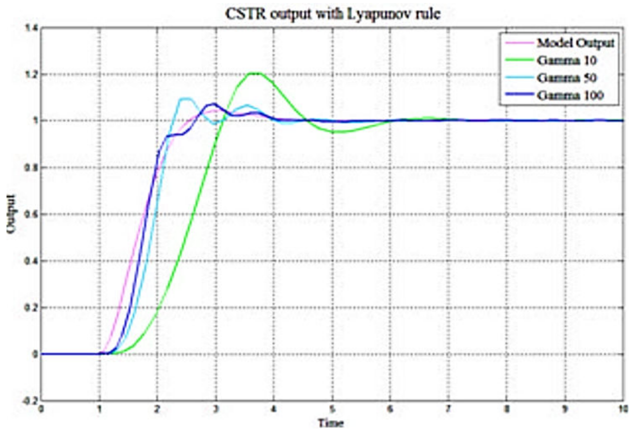

Fig. 7 CSTR output with Lyapunov's for different values of adaption Gain

C. Particle Swarm Optimization

1. PSO Parameters

- Weight / Inertia of the system - 0.5.
- Acceleration constants, $C_1$and C_2 - 1.5.
- Swarm population - 100.
- Dimension of the search-space - 3 (K_p, K_i, K_d)

2. Calculation of Fitness Function

A set of good control parameters *P,I* and *D* can yield a good step response that will result in performance criteria minimization in the time domain. These performance criteria in the time domain include the overshoot, rise time, settling time, and steady-state error. Therefore, the performance criterion is defined as:

$$W(K) = \left(1 - e^{\beta}\right)(M_p + E_{ss}) + e^{-\beta}(T_s - T_r) \tag{9}$$

where K is [P, I, D], and β is the weighting factor. The performance criterion W (K) can satisfy the designer requirement using the weighting factor β value weighting factor is chosen as 1 in this application.

The fitness function is reciprocal of the performance criterion. It can be written as:

$$F = \frac{1}{W(K)}$$

3. Robustness of PSO Algorithm

To check the robustness of PSO-PID controller, values of PID controller is calculated for different iterations and then calculate the best fitness function [12], [16].

TABLE II
OPTIMIZATION OF PID TUNING PARAMETERS OF PSO

K_p	K_i	K_d
.0978	.9075	-.1375

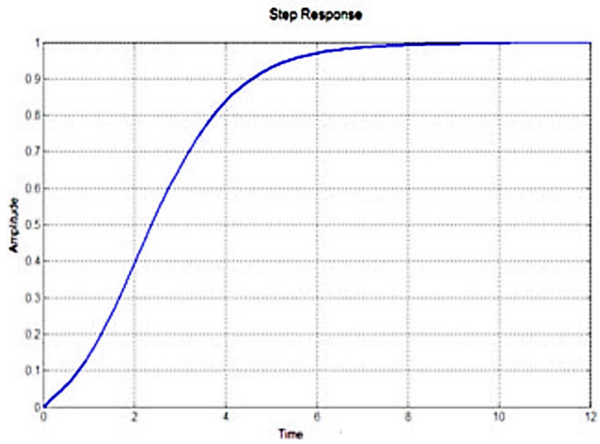

Fig. 8 CSTR output with PSO-PID controller

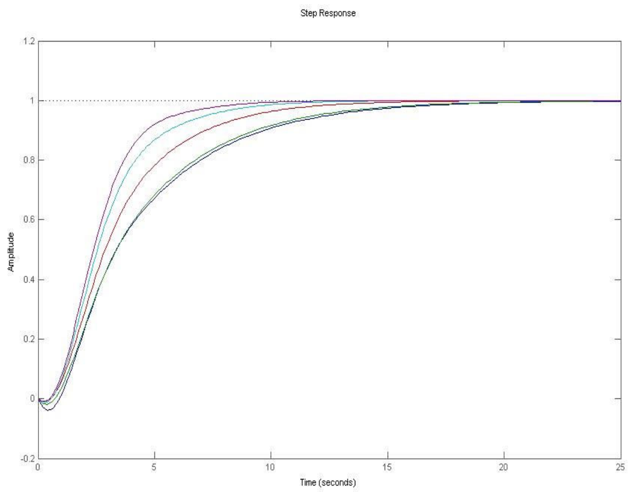

Fig. 9 Plot of PSO parameters for different iterations

TABLE III
COMPARISON OF PERFORMANCE INDICES

Performance specification	MIT Rule			Lyapunov Rule			PSO
	γ = 10	γ = 50	γ = 100	γ = 10	γ = 50	γ = 100	
Rise time (sec.)	3.27	2.65	2.78	3.1	2.3	2.6	4.47
Peak time (sec.)	3.83	3.0	3.01	3.8	2.4	3	17.00
Maximum Overshoot (%)	17.83	26.7	7.5	20.3	9.6	7.3	0
Settling time (sec.)	7.0	6.2	4.5	6.4	4	3.9	8.65
ISE	.405	.21	.02	.37	.04	.014	.003

VI. CONCLUSION

The proposed adaptive and PSO-PID controller is tested by using Matlab Simulink program and their performance is compared. It is clear from Table II that a PSO-PID controller is best implemented. Also, the PID controller parameters obtained from PSO algorithm gives better tuning result as compared to MIT rule and Lyapunov's rule of adaptive controller. The major impact of PSO is on integral square error and peak overshooting. Both are minimized by PSO-PID controller. PSO-PID is a very simple concept, and paradigms can be implemented in a few lines of computer code. It requires only primitive mathematical operators, and is computationally inexpensive in terms of both memory requirements and speed. In future the same problem can be solved by adopting other evolutionary algorithms like ant colony algorithm, bacteria foraging algorithm etc.

REFERENCES

[1] Rahul Upadhyay, Rajesh Singla, "application of adaptive control in a process control", 2nd international conference on education technology and computer (ICETE), 2010. (IEEE).

[2] R. Aruna, M. Senthil Kumar, "Adaptive Control for interactive thermal process "proceedings of ICTECT, 2011. (IEEE)

[3] Karl J. Astrom and Bjorn Witten mark, Adaptive control, second edition, Pearson Education, 2001.

[4] K. A. and J. C. Kantor, "An exothermic continuous stirred tank reactor is feedback equivalent to a linear system," Chem. Eng. Commum., vol. 37, no. 1, 1985.

[5] Dr. M. J. Willis, "continuous stirred tank reactor models", Deptt. of Chemical and Process Engineering, University of Newcastle, March 2010.

[6] K. Prabhu, Dr. V. Murali Bhaskaran, "optimization of control loop using adaptive method", International Journal Of Engineering and Innovative Technology, Volume 1, Issue 3, March 2012.

[7] S. Jegan, K. Prabhu, "Temperature control of CSTR process using adaptive control", International Conference on Computing and Control Engineering (ICCCE), 2012.

[8] Mohammad Ali Nekoui, Mohammad Ali Khamene and Mohammad Hosein Kazemi, "Optimal Design of PID controller for a CSTR System Using Particle Swarm Optimization", International Power Electronics and Motion Control Conference (EPE-PEMC), 2010.

[9] G. Glan Devadhas, S. Pushpakumar, S. V. Muruga Prasad, "Intelligent Computation of Controller Using Optimization Technique for a Nonlinear Chemical Process", International Journal of Research and Reviews in Soft and Intelligent Computing (IJRRSIC) Vol. 1, No.3, September 2011.

[10] J. Kennedy, & R.C. Eberhart, Particle swarm optimization, *IEEE Proceedings International Conference on Neural Networks (ICNN'95), No.4,* Perth, Australia, pp.1942–1948, 1995.

[11] Liu, Y., Zhang, J. and Wang, S. (2004). Optimization design based on PSO algorithm for PID controller. *Proceedings of5th World Congress on Intelligent Control and Automation, Vol. 3, 2419-2422.*

[12] Geetha. M, Balajee. K. and Jovitha Jerome, "Optimal Tuning of Virtual Feedback PID Controller for a Continuous Stirred Tank Reactor (CSTR) using Particle Swarm Optimization(PSO) Algorithm, IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012) March, 2012

[13] Thomas Beielstein, K. E. Parsopoulos and Michael N. Vrahatis, "Tuning PSO Parameters through Sensitivity Analysis, Technical Report of the Collaborative Research Center 531 Computational Intelligence CI--124/02, University of Dortmund, January (2002).

[14] Y Zheng, Liyan Zhang, Jixin Qian Longhua Ma "Robust PID Controller Design using PSO" International Symposium on Intelligent Control IEEE Oct (2003)

[15] T. Bartz–Beielstein K.E. Parsopoulos and M.N. Vrahatis, "Analysis of Particle Swarm Optimization Using Computational Statistics", International conference on numerical analysis and applied mathematics ICNAAM-(2004).

[16] Neha Khanduja,"CSTR control using model reference adaptive control & bio inspired optimization Technique", M. techthesis, D. T. U., July, 2013.