"Bodh", BPIT's International Journal of Technology & Management
ISSN: 2454-8421, Volume 1, Issue 1, July-Dec, 2015, *Page 65-69*

Optimal Control of CSTR

Neha Khanduja
Deptt.of Electrical and Electronics Engineering
Bhagwan Parshuram Institute of Technology
Delhi, India
nehakhanduja.dce@gmail.com

Simmi Sharma
Deptt.of Electrical and Electronics Engineering
Bhagwan Parshuram Institute of Technology
Delhi, India
Simmit08@gmail.com

Abstract— **Sometimes conventional feedback controllers may not perform well online because of the variation in process dynamics due to nonlinear actuators, changes in environmental conditions and variation in the character of the disturbances. To overcome the above problem, this paper deals with the designing of a controller for a second order system with optimal design of PID control based on particle swarm optimization.The mathematical model of experimental system had been approximate near the operating point for the PSO algorithm to adjust PID parameters for the minimum integral square of error (ISE) condition. The results show the adjustment of PID parameters converting into the optimal point and the good control response base on the optimal values by the PSO technique.**

Keywords— ***PIDControl,optimal control,particle swarm optimization(PSO)***

I. INTRODUCTION

During the past decades, great advancement has been made in the process control. Numerous control methods such as PID Control,Adaptive control, neural control, fuzzy control and optimal control have been studied. Among them, the best known is the proportional-integral-derivative (PID) controller, which has been widely used in the industry because of its simple structure and robust performance in a wide range of operating conditions. Unfortunately, it has been quite difficult to tune properly the gains of PID controllers because many industrial plants are often burdened with problems such as high order, time delays, and nonlinearities.

Over the years, several heuristic methods have been developed for the tuning of PID controllers. The first method used the classical tuning rules proposed by Ziegler and Nichols. Generally, it is always hard to determine optimal or almost optimal PID parameters with the Ziegler-Nichols method in many industrial plant..Other than original works done by Ziegler and Nichols, a great number of methods have been proposed to obtain optimal gains of the PID such as by Cohen and Coon in 1953, Åström and Hägglund in 1984 or by Zhuang and Atherton in 1993.To obtain the optimal parameter tuning, it is highly desirable to increase the capabilities of PID controllers by adding new features.

Many Artificial Intelligence (AI) techniques have been employed to improve the controller performance for a wide range of plants while retaining their basic characteristics. Artificial Intelligence techniques such as Neural Network, Fuzzy Logic have been widely applied to proper tuning of PID control parameters.

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, and has been found to be robust in solving continuous nonlinear optimization problems. The PSO technique can 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 [1,2,3,12].

II. DEVELOPMENT OF MATHEMATICAL MODELLING

The examined reactor has real background and graphical diagram of the CSTR reactor is shown in Figure 1. The mathematical model of this reactor comes from balances inside the reactor. Notice that: a jacket surrounding the reactor also has feed and exit streams. The jacket is assumed to be perfectly mixed and at lower temperature than the reactor. Energy passes through the reactor walls into jacket, removing the heat generated by reaction. The control objective is to keep the temperature of the reacting mixture T, constant at desired value .The only manipulated variable is the coolant temperature [4, 5].

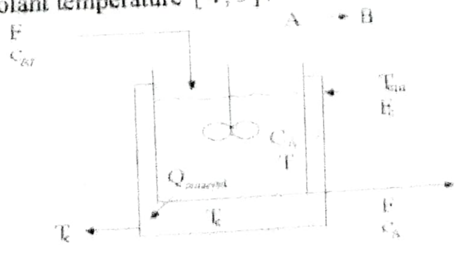


Fig. 1. Continuously Stirred Tank Reactor

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

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

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

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 [5,6,7].

B. *Steady-State Solution*

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

$$f_1(C_A, T) = 0 = \frac{F}{V}(C_{Af} - C_A) - k_o \exp\left(\frac{-\Delta E}{RT}\right) C_A$$

$$f_2(C_A, T) = 0 = \frac{F}{V}(T_f - T) + \left(\frac{-\Delta H}{\rho c_p}\right) k_o \exp\left(\frac{-\Delta E}{RT}\right) C_A - \frac{UA}{V\rho c_p}(T - T_j)$$

The linear model of the system is obtained as:

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

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	$10e^{15}$
-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. PID CONTROLLER

The PID controller is used to improve the dynamic response as well as to reduce or eliminate the steady-state error. The Derivative controller adds a finite zero to the open-loop plant transfer function and improves the transient response. The integral controller adds a pole at the origin, thus increasing system type by one and reducing the steady state-error due to a step function to zero.

The continuous form of a PID controller, with input $e(.)$ and output $(.)u_{pid}$, is generally given as :

$$u_{pid}(t) = k_p\left[e(t) + \frac{1}{T_i}\int_0^t e(\tau)\,d\tau + T_d\frac{d}{dt}e(t)\right] \qquad (4)$$

where *kp* is the proportional gain, *Ti* is integral time constant and *Td* is the derivative time constant. We can also rewrite as

$$u_{pid}(t) = k_p e(t) + k_i\int_0^t e(\tau)\,d\tau + k_d\frac{d}{dt}e(t) \qquad (5)$$

where $k_i = k_p / T_i$ is the integral gain and $k_d = k_p T_d$ is the Derivative gain.In simple form, the PID controller transfer function is

$$C(s) = k_p + \frac{k_i}{s} + k_d s \qquad (6)$$

A. *Ziegler Nichols Tuning*

In 1942, Ziegler and Nichols [9], described simple mathematical procedures, for tuning the PID controllers.Both the techniques make a priori assumption on the system model, but do not require the system model to be specifically known. Ziegler-Nichols formulae for specifying the controllers are based on the plant step response.

1) *Open Loop Response*

The open-loop method is typical for a first-order system with transportation delay. The response is characterized by 2 parameters, L the time-delay and T the timeconstant.These are found by drawing a tangent to the step response at its point of inflection and noting its intersections with the time axis and steady-state value.

2) Closed Loop Response

The closed-loop method targets plant that can be rendered unstable under proportional control. The technique is designed to result in a closed loop system with 25% overshoot [8,12].

IV. PARTICLE SWARM OPTIMIZATION(PSO)

Particle swarm optimization is an extremely simple algorithm that seems to be effective for optimizing a wide range of functions. It is viewed as a mid-level form of A-life or biologically derived algorithm, occupying the space in nature between evolutionary search, which requires cons, and neural processing, which occurs on the order of milliseconds. Social optimization occurs in the time frame of ordinary experience - in fact, it *is* ordinary experience. In addition to its ties with A-life, particle swarm optimization has obvious ties with evolutionary computation. Conceptually, it seems to lie somewhere between genetic algorithms and evolutionary programming. It is highly dependent on stochastic processes, like evolutionary programming.

PSO is derived from the social-psychological theory, and has been found to be robust in complex systems. Each particle is treated as a valueless particle in g-dimensional search space, and keeps track of its coordinates in the problem space associated with the best solution (evaluating value) and this value is called *pbest*. The overall best value and its location obtained so far by any particle in the group that was 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. As 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 of 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 shown in the following formulas:

$$v_{j,g}^{(t+1)} = w.v_{j,g}^{(t)} + c_1 * rand\,() * (pbest_{j,g} - x_{j,g}^{(t)}) + c_2 * rand\,() * (gbest_g - x_{j,g}^{(t)}) \quad (7)$$

$$x_{j,g}^{(t+1)} = x_{j,g}^{(t)} + v_{j,g}^{(t+1)} \quad (8)$$

j=1, 2…... n; g=1, 2... 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 were often set to be 2.0 according to past experience. This 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 equation (8) below. Suitable selection of *w* provides a balance between global and local explorations, thus requiring less iteration on average to find a sufficiently optimal solution.

$$w = \frac{w_{max} - w_{min}}{iter_{max}} * iter \quad (9)$$

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[8,9,10].

A. Optimal Tuning of PID Controller Using PSO

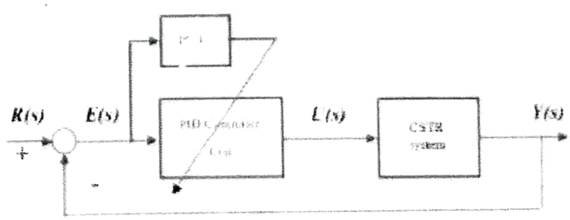


Fig.2 Block diagram of optimal PID controllers with PSO for CSTR

The control system with a set of optimal PID parameters can obtain an excellent response output show in Fig.2.The value

of fitness function defined by optimization algorithm would be the minimum.
Performance characteristic of evaluation function includes overshoot, rise time, settling time and static error time. The evaluation function as in (9), to compute the evaluation value of each particle in swarm according to control performance. can obtain an excellent response output.

The sequence of steps to study the PSO for the CSTR system is given below:
STEP 1: Specify the lower and upper bounds of Kp,Ki,Kd ·Initialize randomly 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.
STEP 3: Compare each particle's new fitness value with its personal best position's 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_i1 of each particle according to equation (3), and update the particle position matrix according to equation (4).
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 [12].
A suitable performance index is the integral of the square of the error, ISE, which is defined as

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

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:

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

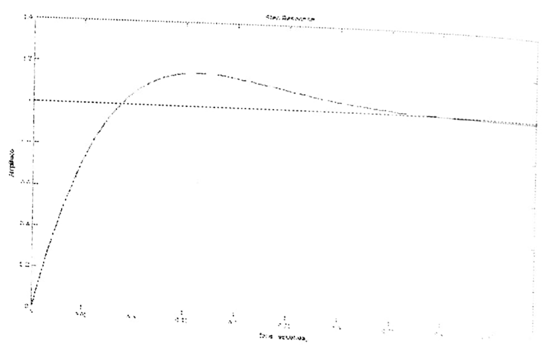
Other performance criteria include evaluation of rise tim settling-time and peak overshoot. Rise time is the time take for the response to rise from 0 to 100% for the first tim Settling time is defined as the time taken by the response reach and stay within specified error limit. Peak Oversho is the ratio of maximum peak value measured fro maximum value to the final value [12].

V. SIMULATION RESULTS

A. PID Controller

TABLE II. PID TUNING PARAMETERS USING ZEIGLER-NICHOLAS METHOD

Tuning Method	K_p	K_i	k_d
ZNT	9.2675	37.911	-.678



B. 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 follows

$$\min_{K:\text{stablizing}} W(K) = (1-e^{-\beta}).(M_p + E_{ss}) + e^{-\beta}.(t_s - t_r) \qquad (10)$$

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

The fitness function is reciprocal of the performance criterion, in the other words:

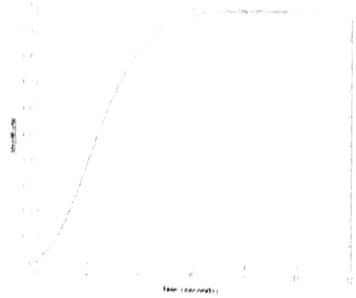
$$f = \frac{1}{W(K)}$$

3) Robustness of PSO Algorithm

To check the robustness of PSO-PID controller, values of PID controller is calculated for different iterations and conclude
which among these gives the best fitness function [15].

TABLE III. OPTIMIZATION OF PID TUNING PARAMETERS OF PSO

K_p	K_i	K_d
.0978	.9075	-.1375



C. *Comparitive Analysis of Performance Indices[15]*

TABLE IV. OPTIMIZATION OF PID TUNING PARAMETERS OF PSO

Performance Index	Z-N tuned PID Controller	PSO-PID Controller
Rise time (sec.)	.36	4.47
Peak time (sec.)	1.0	17.0
Maximum Overshoot (%)	7.1	0
Settling time (sec.)	1.4	8.65
ISE	.16	.003

VI. CONCLUSION

A thorough comparative analysis has been carried out on CSTR performance with different controllers. It has been shown that the individual controllers have their own merits and demerits. The choice of selection of controller for a particular application should be based on typical requirement. When the requirement is of simplicity and ease of application, a Z-N tuned PID controller is of a good choice. When the requirement is of both intelligent response and good steady state performance with minimum overshoot and least error, optimization based PSO-PID controller is a better choice. The major impact of PSO is on integral square error and peak overshoot. Both are minimized by PSO-PID controller. In future the same problem can be solved by adopting other evolutionary algorithms like ant colony algorithm, bacteria foreaging algorithm etc [15].

REFERENCES

[1] Gaing, Z.L. (2004). A Particle Swarm Optimization approach for optimum design of PID controller in AVR system. *IEEE Transactions on Energy Conversion, Vol.19(2), 384-391.*

[2] Skogestad, S, "Simple Analytical rules for Model Reduction and PID Controllers Tuning", Journal of Process Control, 13, 2003,pp.291-309.

[3] K.J. Astrom, & T. Hagglund, The future of PID control *Control Engineering Practice*, pp.1163 - 1175, 2001.

[4] Rahul Upadhyay and Rajesh Singla," Analysis of CSTR Temperature Control with Adaptive and PID Controller (A Comparative Study)", IACSIT International Journal of Engineering and Technology, Vol.2, No.5, October 2010.

[5] Dr. M.J.Willis,"continuous stirred tank reactor models",.Deptt. of Chemical and Process Engineering, University of Newcastle,March2010.

[6] K.Prabhu,Dr.V.MuraliBhaskaran,"optimization of control loop using adaptive method", International Journal Of Engineering and Innovative Technology,Volume1,Issue3,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] G. Glan Devadhas, S. Pushpakumar,"an intelligent design of PID controller for continous stirred tank reactor",world applied science journal 14(5), pp.698-703,2011.

[9] 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 Optimisation Techniques 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 of 5th 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] Priyank Jain , M.J. Nigam," Design of a Model Reference Adaptive Controller Using Modified MIT Rule for a Second Order System",Advance in Electronic and Electric Engineering , ISSN 2231-1297, Volume 3, Number 4, pp. 477-484,2013.

[14] Padmayoga.R, Shanthi.M,, Yuvapriya.T," Modeling and Control of Chemical Reactor Using Model Reference Adaptive Control",International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Special Issue 4, May 2014

[15] NehaKkhanduja,"CSTR control using model reference adaptive control & bio inspired optimization Technique",M.tech thesis,D.T.U.,July,2013.