ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014

PSO based controller design for a first order process with time delay

C. Ishwarya[1], S. Nandhini[2], T. Bhuvaneshwari[3], S. M. Girirajkumar[4]

Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Trichy, India[1, 2, 3]

Professor, Dept. of Instrumentation and Control Engineering, Saranathan College of Engineering, Trichy, India[4]

Abstract: This Paper deals with the tuning of PID controller using conventional methods and computational technique like Particle Swarm Optimization. The main objective is to prove that the response obtained is more stable, robust and efficient when PID is tuned using PSO. The obtained value is compared with conventional methods like Ziegler Nicholas, Cohen-coon and Internal Model Control. The criteria used for comparison include time domain specifications, Performance index, robustness of the system, servo and regulatory responses.

Keywords: PID, PSO, ZN, IMC, Controller, Tuning

I. INTRODUCTION

Modern trends have emerged in the field of industrial automation to surpass the needs of the end user. Controller takes the complete credit in maintaining the process in any production firm. Composite controllers are used in place of a two point controller as the former will have a continuous and complete over the process [4]. Proportional, integral and derivative type of controller are not used individually, rather they are used individually rather they are used in composite form [1].

P controller is mostly used in first order processes with single energy storage to stabilize the unstable process. As the proportional gain factor K increases, the steady state error of the system decreases [4]. However, despite the reduction, P control can never manage to eliminate the steady state error of the system. We can use this controller only when our system is tolerable to a constant steady state error.

P-I controller is mainly used to eliminate the steady state error resulting from P controller. However, in terms of the speed of the response and overall stability of the system, it has a negative impact. This controller is mostly used in areas where speed of the system is not an issue [2]. The aim of using P-D controller is to increase the stability of the system by improving control since it has an ability to predict the future error of the system response. In order to avoid effects of the sudden change in the value of the error signal, the derivative is taken from the output response of the system variable instead of the error signal [4].

P-I-D controller has the optimum control dynamics including zero steady state error, fast response (short rise time), no oscillations and higher stability [3]. The necessity of using a derivative gain component in addition to the PI controller is to eliminate the overshoot and the oscillations occurring in the output response of the system.

One of the main advantages of the P-I-D controller is that it can be used with higher order processes including more than single energy storage. The above mentioned features can be obtained only when the PID controller is tuned using a suitable technique [4].

Computational intelligence (CI) is an optimization technique, combining various elements of learning such as adaptation and evolution to create programs that are intelligent and effective [1]. Computational intelligence research does not reject statistical methods, but often gives a complementary view. The importance of CI lies in the fact that these techniques often find optima in complicated optimization problems more quickly than the traditional optimization methods [2].

Particle Swarm Optimization (PSO) is CI strategy, motivated from the simulation of birds' social behavior and the approach and technique followed by them to search their food. PSO is widely used in many control engineering applications and has proved to provide optimum solutions in many such cases [1]. PSO has been regarded a promising optimization algorithm due to its simplicity, low computational cost and good performance [2].

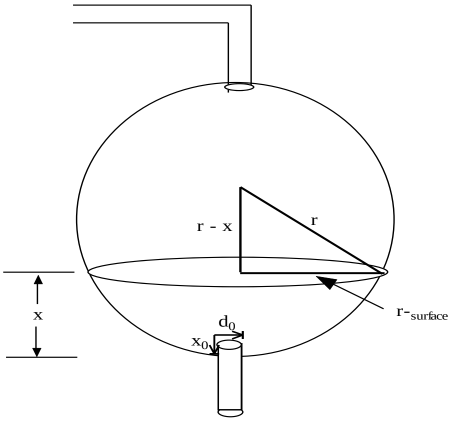
The main objective of this paper is to tune the PID controller using conventional techniques like Zeigler-Nichols, Cohen-Coon and IMC and using computational technique namely PSO. The obtained results will be compared based on the error criteria such as IAE, ISE, ITAE and MSE.

II. SYSTEM DECRIPTION

The spherical tank level process model as suggested by K. K. Tan, et.al., in Closed-loop automatic tuning of PID controller for nonlinear systems of chemical engineering science, 2002 is considered here in which the control input f_{in} is the input flow rate (m^3/s) and the output x is the fluid level (m) in the spherical tank. Let, r, d_0 and x_0 is the radius of spherical tank, thickness (diameter) of pipe (m) and initial liquid level height respectively. Assume '$r_{surface}$' radius on the surface of the fluid varies according to the level of fluid in the tank.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014



$Length^2 + Height^2 = Hypotenuse^2$ (By Pythagoras theorem)
where, Length = $r_{surface}$;
Height = radius of tank (r) – fluid level (x) and
Hypotenuse = radius of tank (r)

Therefore, $(r - x)^2 + r_{surface} = r^2$

$$r_{surface}^2 = r^2 - (r - x)^2$$
$$= r^2 - (r^2 - 2rx + x^2)$$
$$= r^2 - r^2 + 2rx - x^2$$
$$= 2rx - x^2$$
$$r_{surface} = \sqrt{2rx - x^2}$$

Now the Dynamic model of the spherical tank is given by

$$\frac{\delta}{\delta t}\left[\int_0^{x_1} A(x)\,\delta x\right] = f_{in}(t) - a\sqrt{2gx - x_0}$$

Where $A(x)$ = area of cross section of tank = $\pi\, r_{surface}^2 = \pi\,(2rx - x^2)$

a = area of cross section of pipe = $\pi\left(\frac{d_0}{2}\right)^2$

Rewriting the equation at time t + δt

$$A(x)\ \delta x = f_{in}\ \delta t - a\sqrt{2g(x - x_0)}$$

Where, $A(x)\ \delta x$ = Amount of water;
$f_{in}\ \delta t$ = Input flow rate and
$a\sqrt{2g(x - x_0)}\ \delta t$ = Output flow rate

Combining the above equations we have

$$\frac{\delta x}{\delta t} = \frac{f_{in}\,\delta t - \frac{\pi\, d_0^2}{4}\sqrt{2g\,(x - x_0)}}{\pi\,(2rx - x^2))}$$

By applying $lim_{d \to 0}$ in the above equation, we have δx δt= dx dt

Therefore

$$\frac{dx}{dt} = \frac{f_{in}\,\delta t - \frac{\pi\, d_0^2}{4}\sqrt{2g\,(x - x_0)}}{\pi\,(2rx - x^2)}$$

The above equation shows the process dynamics model of the spherical tank level system and this model representation is considered for simulation studies.

In simulation platform, level in the Spherical Tank system is kept at a steady state each of different operating points of 20%, 40%, 60% and 80%. A step size of 5% level for each operating point is applied and the variation of level against time for each operating point is recorded separately until a new steady state is attained. From the recorded data, the model parameters such as process gain (K_p) time constant (τ_p) and time delay (t_d) are computed and tabulated in the following table. From the table, the worst case model parameters such as larger process gain (K_p), smaller time constant (τ_p) and larger delay (t_d) are considered and these parameters are taken for design of controllers.

Operating Point (% of level)	**K_p**	**τ_p**	**t_d**
20	0.864	***96.45***	***17.85***
40	1.23	219	8
50	1.38	252.75	7.75
60	1.52	258.9	8.9
80	***1.76***	174.5	13.25

The identified model is given by

$$G(s) = \frac{1.76}{96.45s + 1}\, e^{-17.85s}$$

III. CONVENTIONAL TUNING METHODS

ZN tuning technique:

ZN method was proposed by John G. Ziegler and Nathaniel B. Nichols in 1942. ZN method is one of the most widely used tuning techniques as it involves simple algorithm for its implementation. Tuning a controller using ZN technique involves determining the values of ultimate gain (Ku) and ultimate period (Tu). These values were found using Bode plot and root locus.

$Tu = 2\pi/\ \omega_{c0}$
ω_{c0} = 0.121 rad /sec Ku= 6.71
Tu= 51.9sec

PID parameters	Kp	τi	τd
ZN tuning Formula	0.6Ku	0.5Tu	Tu/8
ZN based Tuned values	4.026	0.1551	43.53

Cohen- Coon tuning technique:

The Cohen-Coon method of controller tuning corrects the slow, steady-state response given by the Ziegler-Nichols method when there is a large dead time (process delay) relative to the open loop time constant; a large process delay is necessary to make this method practical because otherwise unreasonably large controller gains will be predicted. This method is only used for first-order models with time delay; due to the fact that the controller does not instantaneously respond to the disturbance.

PID parameters	Kp	τi	τd
C-C tuning Formula	(τ/Kp.td)((4/3)+(td/4τ))	td((32+6(td/τ))/(13+8(td/τ)))	td(4/(11+2(td/ τ)))
C-C based Tuned values	4.22	0.103	26.49

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014

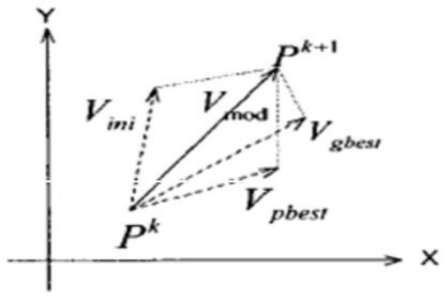
IMC tuning technique:

IMC was introduced by Garcia and Morari in the year 1982. Design of IMC based controller depends on the complexity of the model and the performance requirements stated by the designer. The proposed IMC structure provides valuable insight regarding controller tuning effects on both performance and robustness.

IV. PARTICLE SWARM OPTIMIZATION

In PSO algorithm, the system is initialized with a population of random solutions, which are called particles, and each potential solution is also assigned a randomized velocity. PSO relies on the exchange of information between particles of the population called swarm. Each particle adjusts its trajectory towards its best solution (fitness) that is achieved so far. This value is called pbest. Each particle also modifies its trajectory towards the best previous position attained by any member of its neighborhood. This value is called gbest. Each particle moves in the search space with an adaptive velocity.

The concept of PSO is briefly explained in the following figure ,where Pk is the current position,Pk+1 is the modified position, vini is the initial velocity, vmod is the modified velocity ,vpbest is velocity considering pbest and vbest is the velocity considering gbest.



The fitness function evaluates the performance of particles to determine whether the best fitting solution is achieved. During the run, the fitness of the best individual (hopefully) improves over time and typically tends to stagnate towards the end of the run. Ideally, the stagnation of the process coincides with the successful discovery of the global optimum.

Let *D* be the dimension of the search space taken into consideration and Xi = [xi1, xi2,….xiD]^T denote the current position of ith particle of the swarm, Then: Xi pbest =[xi1pbest,xi2pbest,….xiDpbest] ^{T} denote the best position ever visited by the particle. Xgbest =[x i1gbest,x i2gbest,….x i2gbest] ^{T} represents 'gbest',i.e the best position obtained this far by any particle in the population. Vi=[vi1,v i2,….v iD] T represents the velocity of ith particle. Vimax = [vi1max, vi2max ….viDmax] T denotes the upper bound on the absolute value of the velocity with which the particle can move at each step.

The position and velocity of the particles is adjusted as per the following equation:

Vid = w* v id + c1.*r1 * (x idpbest -x id) + c2 *r2 *(x gbest –x id)

$$V\,id = \begin{cases} vdmax, & v\,id > vdmax \\ -vdmax, & v\,id < -vdmax \end{cases}$$

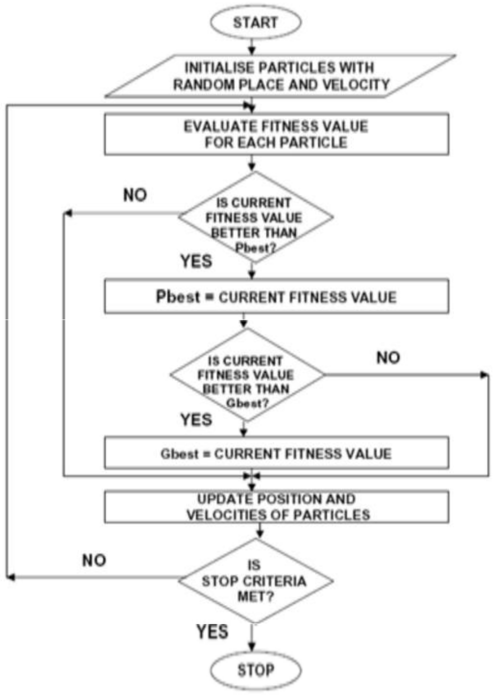
$$xid = xid + vid$$

where, c1 and c2 are positive constants, represent the cognitive and social parameter respectively;

r1 and r2 are random numbers uniformly distributed in the range [0,1];

w is inertia weight to balance the global and local search ability.

In general the PSO algorithm can be given by the following flowchart.



The objective functions considered are based on the error criterion. The performance of a controller is best evaluated in terms of error criteria. A number of such criteria are available and in this paper, controller's performance is evaluated in terms of ITAE.

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

Termination of optimization algorithm can take place either when the maximum number of iterations gets over or with the attainment of satisfactory fitness value. Fitness value is the reciprocal of the error.

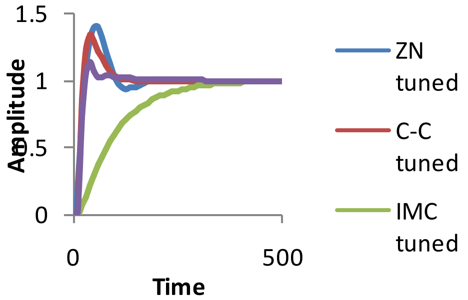
PID parameters	Kp	Ki	Kd
PSO tuned values	3.9756	0.039	20.9052

V. RESULTS AND COMPARISON

Response of the system was observed by applying a unit step input with a PID controller tuned using the proposed conventional methods and particle swarm optimization. The following graph shows the comparative analysis of all the four methods:

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014



Time domain specifications:

Time domain specifications such as rise time, peak overshoot, settling time and offset are found from the above graph and tabulated as follows:

Rise time = 63.2% of the final value
Peak overshoot = 1- (Maximum value of the first peak observed in the response graph)
Settling time= time required to get settled at the set value without oscillations
Offset= steady state error (set value – settled value)

Time domain specifications	ZN tuning	C-C tuning	IMC tuning	PSO tuning
Rise time	20	19	105	21
Peak overshoot	0.4055	0.3416	---	0.133
Settling time	338	288	---	268
Offset	0	0	0.0035	0

Performance index:

The performance of the system can be analyzed using various error criteria such as IAE, ISE, ITAE and MSE.

Performance index	ZN tuning	C-C tuning	IMC tuning	PSO tuning
IAE	262.6504	259.8823	21.065	200.8164
ISE	280.7967	212.6228	576.3877	189.8551
ITAE	187.29	111.62	536.07	47.7975
MSE	0.0561	0.0425	0.1254	0.038

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

$$ITAE = \int_0^T t|e(t)|\, dt \qquad MSE = 1/t \int_0^T |e(t)|^2\, dt$$

Robustness estimation:

The robustness investigation for the process is analyzed by calculating the performance index to the transfer function model whose parameters such as process gain, time constant and propagation delay are deviated by ±20 %. The altered model which possesses the uncertainties is given by,

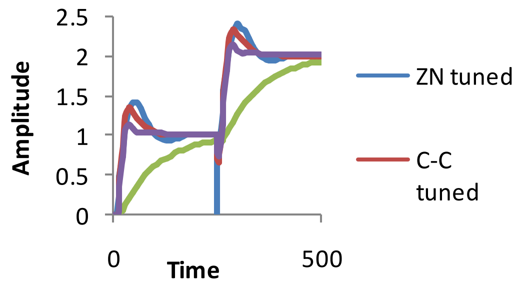
$$\frac{1.7952e^{-18.207s}}{96.45s+1}$$

Performance index for the uncertain model can be tabulated as follows:

Performance index	ZN tuning	C-C tuning	IMC tuning	PSO tuning
IAE	261.9736	262.8138	20.66	205.8652
ISE	303.4157	220.7634	568.1072	195.4156
ITAE	182.33	110.08	514.34	50.77
MSE	0.0607	0.0441	0.1136	0.0391

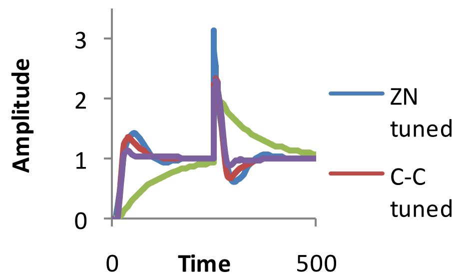
Servo response:

Servo response of the system was obtained by giving a step change to the input or set point. The obtained response by implementing the three proposed tuning strategies can be plotted as follows:



Regulatory response:

Regulatory response of the system can be obtained by disturbing the system i.e. by applying a unit step change in the load side. Obtained regulatory responses are plotted as follows for the mentioned tuning strategies:



ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 2, Issue 3, March 2014

VI. CONCLUSION

It is obvious from the presented results that the response of the system with a PSO tuned PID controller significantly outmatches the responses of the system with conventionally tuned PID controllers. Rise time and settling time of the system is notably lower for a PSO tuned controller than its conventional counterpart. The values of all errors are lower for a PSO tuned controller. System with controller tuned using particle swarm optimization is more robust for uncertain models. Better servo and regulatory responses are obtained if the controller is tuned using PSO technique.

The various results presented prove the betterness of the PSO tuned PID settings than ZN, C-C and IMC tuned ones. The simulation responses for the models reflect the effectiveness of the PSO based controller in terms of time domain specifications. The performance index under the various error criterions for the proposed controller is always less than the conventionally tuned controller.

PSO presents multiple advantages to a designer by operating with a reduced number of design methods to establish the type of the controller, giving a possibility of configuring the dynamic behavior of the control system with ease. So this method of tuning can be applied to any system irrespective of its order and can be proved to be better than the existing traditional techniques of tuning the controller.

REFERENCES

[1] S.M. Girirajkumar, Deepak Jayraj, Anoop. R. Kishan, PSO based tuning of a PID controller for a High performance drilling machine
[2] S. M Girirajkumar, Bodla Rakesh, N. Anantharaman, Design of controller for a pressure process using particle swarm optimization techniques, in Journal of Current Sciences, 2010
[3] S. Sathishbabu, Chapter 3 of Materials and Methods, in Thesis submitted in Annamalai University named Real time implementation of iterative learning control strategies for a class of non- linear systems
[4] Sena Temel, Semih Yagli, Semih Goren, Report on Discrete time Control systems, Middle East Technical University.
[5] Jun Zhao, Tianpeng Li and Jixin Qian "Application of Particle Swarm Optimization Algorithm on Robust PID controller Tuning", Springerlink-Verlag Berlin Heidelberg , pp. 948-957, (2005).