International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

PSO based PID Controller Design for a Liquid Flow Process

P Aravind[A*], S.Ramachandran[A] and M.Saranya[B]

[A]Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu, India
[B]Department of Electronic and Communication Engineering, Sai Vidya Institute of Technology, Karnataka India

Accepted 20 Nov 2014, Available online 28 Dec 2014, **Vol.4, No.6 (Dec 2014)**

Abstract

System identification and controller tuning are important to get a stable control system in process industries. In the proposed work, real time flow process is taken into account and process parameters are identified as First Order Plus Dead Time Model by step test method and is validated, proposed for controller design. The work focused on the development and implementation of optimization technique to obtain PID controller settings. Values of PID controller are obtained based on Internal Model Control technique and Particle swarm optimization technique. Performances of the controllers are evaluated on the basis of time domain analysis and error criterion analysis. The robustness of the controllers is validated by imposing both servo and regulatory disturbances. Best controller performance can be envisaged by the proposed methods using PSO based PID controller settings than IMC tuned PID controller.

Keywords: *PSO, Internal Model Control, PID controller, Flow Process , LabVIEW.*

1. Introduction

P,PI,PD and PID combination controllers are widely used in process industries because of its simple tuning procedures. PID controller is a three parameter control settings are proportional, integral and derivative. Over the past 50 years research work on PID tuning methods are carried out (J. G. Ziegler and N. B. Nichols(1942)), Cohen-Coon's (G.H Cohen and G.A Coon(1953)), Astrom and Hagglund (Astrom, K J and Hagglund .T(1984)) and many other techniques are also emerged. These technique is highly appreciated by many researchers because of minimum effort sufficient to get a satisfactory response.

To most process PID controller provides reliable and stable performance with minimum effort. Although new methods are proposed for tuning the PID controller. To meet system demands researchers were intended to deals with intelligent agents like human thinking ability human intelligence(Asriel U. Levin and Kumpati S. Narendra(1996),(Simon Fabri and Visakan Kadirkamanathan, (1996)). Due the complexity in their real time implementation and tuning in such techniques, the research societies as well as the industrial societies pay attention towards computation intelligence(Muller SD, Marchetto J, Airaghi S, Koumoutsakos P (2002), Javed Alam Jan, Bohumil Sulc (2002), Y Zheng, Liyan Zhang, Jixin Qian Longhua Ma (2003)). The computation efficiency is the advantage of particle swarm optimization algorithms over other tuning techniques.

Optimization tuning techniques are have advantage over other control techniques. PID tuning can be done effortlessly and integration to the PSO is simple(S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008)).

In recent years, the interest of most researchers moves towards optimization techniques. Best possible solution is obtained just with respect to the decisive factor at hand and the real performance depends on the fittingness of the chosen criterion (T.Bartz–Beielstein K.E. Parsopoulos and M.N. Vrahatis (2004)).

A unusual approach of swarm intelligence based on simplified numerical function on metaphor of social behavior of flocks of birds and schools of fish, is the particle swarm optimization (PSO) algorithm (Kennedy JF, Eberhart RC.(1995)). PSO is a self-adaptive search optimization, first introduced by Kennedy and Eberhart (Kennedy JF, Eberhart RC.(1995)).

PSO algorithm used here to obtain a optimized value for PID controller parameters. The obtained PID values from optimization technique are compared with Model control technique. To achieve the effective control standards, analysis are performed on time domain and error criterion analysis. The objective of the work is to maintain a liquid inlet flow to the tank at desired value. Effectiveness of the controller is also tested by disturbance rejection analysis.

This paper is organized as follows: In section 2, experimental setup and process model development are discussed. In section 3, tuning techniques are briefed. The comparative studies and results are given in section 4. The conclusions arrived, based on the results from section 5.

2. Experimental

The real time experimental setup consisting of a transparent linear tank, reservoir and water pump, current

*Corresponding author: **P Aravind**

to pressure converter, compressor, Differential Pressure Transmitter and a Personal Computer which acts as a controller forms a closed loop system.

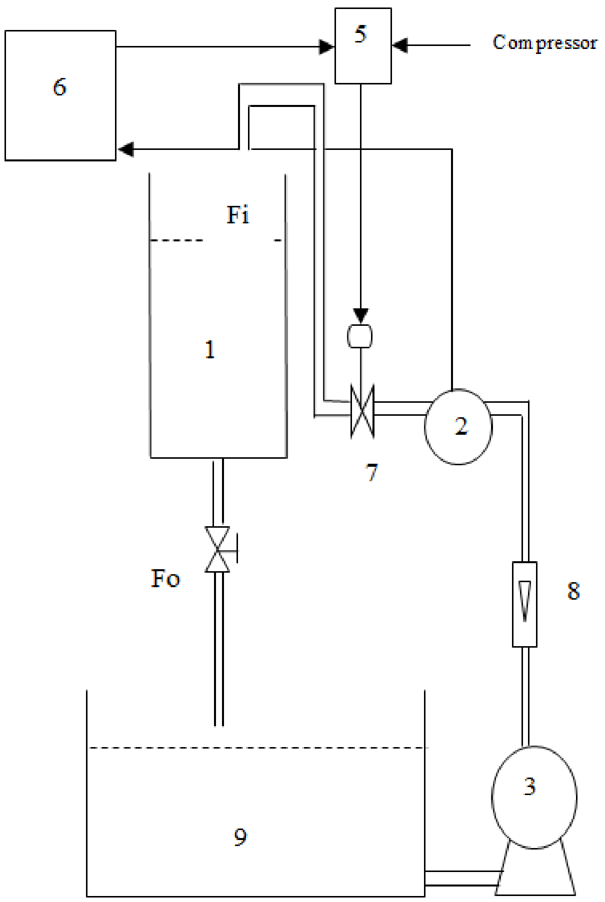

Fig.1. Piping and Instrument diagram of Experimental Setup

1	Process Tank
2	FT
3	Pump
4	Reservoir
5	I to P convertor
6	Computer (controller)
7	Control Valve
8	Rotameter
9	Storage Tank
Fi, Fo	Inflow and Outflow

Table 1 Technical details of the Experimental Setup

Part Name	Details
Tank	Transparent body- Cylindrical
Differential Pressure Transmitter (FT)	Capacitance-Range (25-250)mbar cm, Output 4–20mA
Pump	Centrifugal 0.5 HP
Control valve	Size ¼ Pneumatic actuated, Type: Air to close, Input 3 – 15 psi
Rotameter	Range 10 - 100 LPH
I/P converter	Input 4-20 mA, Output 3-15 psi
Pressure gauge	Range 0 - 30 psi

2.1 Step Test Method

The process is identified by the Step Test method and is commonly known as Process Reaction or Transient Response method. The basic approach is to open the feedback loop, so that no control action occurs. Initially the process is set at manual mode and step test is performed by varying the inflow rate. The open loop step response is obtained and the transfer function is found as FOPDT model (D. R. Coughanowar (1991), (S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008).

For a change in step function the PRC method produces a response, from the response parameters like dead time (τ_d), the time taken for the response to change (τ), and the ultimate value that the response reaches at steady state, τ = 63.2 % of the maximum value are measured and Sunderasan Kumaraswamy(SK) method (S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008)) is used to develop model from the obtained response. As per the structure of the curves, the FOPTD model is given by,

$$G(s) = \frac{k_p e^{-\tau_d(s)}}{\tau s+1} \qquad (1)$$

Where K is the process gain; τ is the first order time constant; τ_d is the delay time.

The calculated model is validated with real time results is presented in figure 2 (Srimathi.R, Sivaranjani.T, Suganya.S, Pradibaa.S, Aravind.P, (2014)). From the response of the real time system we obtain the mentioned constants and thereby we get the FOPTD models for the real time process as,

$$G(s)=\frac{0.74e^{-2s}}{15s+1} \qquad (2)$$

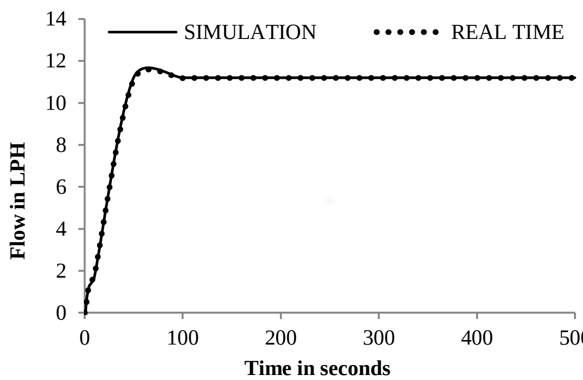

Fig.2. Comparison of real time and simulated responses

3. Controller Design

Two control techniques are implemented in the process to obtain robust controller. The IMC based PID control technique and PSO based PID controller. IMC is a model based approach and PSO is an optimization algorithm used to obtain a PID values of controller.

3.1 IMC based PID

In order to turn up the process with a time delay model to a PID equivalent form, the dead time must be apporimate using pade approximation method.(W. B. Bequette (2003))

$$G(s) = \frac{k_p e^{-\tau_d(s)}}{\tau s+1} \quad (3)$$

First order pade approximation for dead time,

$$e^{-\tau_d(s)} = \frac{-0.5\tau_d s+1}{-0.5\tau_d s+1} \quad (4)$$

PID controller parameters are identified by solving the equation (3 and 4),

$$K_p = \frac{\tau+0.5\tau_d}{K_p(\tau_c+0.5\tau_d)} \quad (5)$$

$$T_i = \tau+0.5\tau_d \quad (6)$$

$$T_d = \frac{\tau\,\tau_d}{2\tau+\tau} \quad (7)$$

3.2 Optimization Using PSO

PSO is a robust stochastic optimization technique based on the movement and cooperation of swarms. The application of PSO algorithm was put onward by several researchers who developed computational simulations of the movement of organisms such as schools of fish and flocks of birds. Such simulations were heavily based on manipulating the distances between individuals, i.e., the synchrony of the behavior of the swarm was seen as an effort to keep an optimal distance between them. Sociobiologist Edward Osbourne Wilson outlined a link of these simulations for optimization problems. PSO, originally developed by Kennedy and Eberhart in 1995, is a population-based swarm algorithm (Kennedy JF, Eberhart RC.(1995)).

3.3. Selection of PSO parameters

To start up with PSO, certain parameters need to be defined. Selection of these parameters decides to a great extent the ability of global minimization.

Population size=50
Number of iterations=50
Velocity constant, c1=1.2
Velocity constant, c2=2.

3.3.1. Particle Swarm Optimization

The 'swarm' is initialized with a population of random solutions. 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. The performance of each particle is measured according to a pre-defined fitness function, which is related to the problem being solved.

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 (S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008)). 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 G_{best}. Each particle moves in the search space with an adaptive velocity. 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 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.

velocity = w *velocity + c1*(R1.*(L_b_position-current_position)) + c2*(R2.* (g_b_position-current_position)) (8)

where c1 and c2 are positive constants, represent the cognitive and social parameter respectively; R1 and R2 are random numbers uniformly distributed and w is inertia weight to balance the global and local search ability. In general the PSO technique can be given by the following algorithm,

3.3.2. Algorithm

Step1: Start the program
Step2: Initialize particles with random place and velocity
Step3: Evaluate fitness value for each particle
Step4: If current fitness value is better than P_{best}, goto Step5 else go to step8.
Step5: P_{best} equal to current fitness value
Step6: If current fitness value is better than G_{best}, goto to Step7 else go to step 8
Step7: G_{best} is equal to current fitness value.
Step8: Update position and velocity of particles
Step9: Go to step10 if stop criteria met else go to step3.

3.3.3. Termination criteria

Optimization algorithm will automatically terminate execution either when the number of iterations gets over or with the attainment of acceptable fitness value. Fitness value, in this case is nothing but reciprocal of the error, since we consider for a minimization of objective function. In this paper the termination criteria is considered to be the attainment of maximum number of iterations. For each iteration the best among the 50 particles considered as potential solution is chosen. Therefore the best values for 50 iterations for the model is sketched and shown in figure3-5 with respect to iterations for Kp, Ki and Kd.

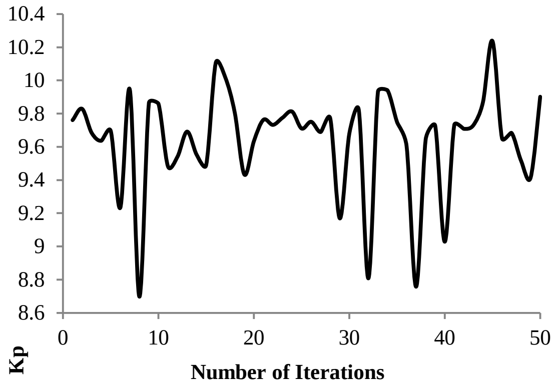

Fig.3 Best solutions of Kp for 50 iterations

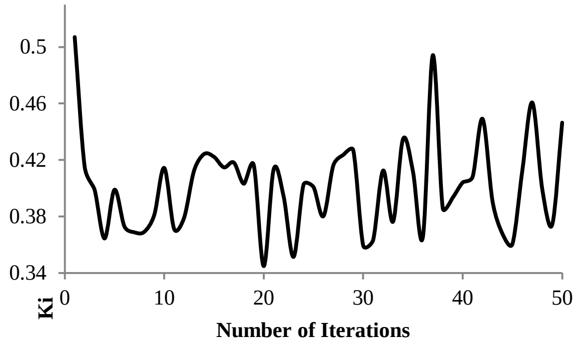

Fig.4. Best solutions of Ki for 50 iterations

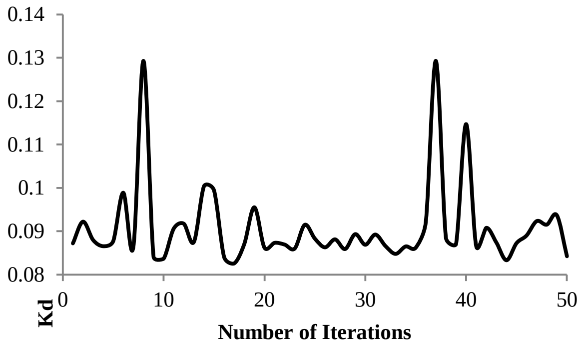

Fig.5. Best solutions of Kd for 50 iterations

The PID controller was formed based upon the respective parameters for 50 iterations, and the G_{best} (global best) solution was selected for the set of parameters.

Table 2 Tuned Gain values of controller

Controller	IMC	PSO
K_p	7.149	9.6623
K_i	0.44681	0.4969
K_d	6.70576	0.08427

Table 3 Comparison of Time Domain Analysis and Performance Indices:

Specifications	IMC	PSO
Rise Time (seconds)	9	4
Peak Time(seconds)	15	5
Settling Time (seconds)	50	25

The Table 3 shows the performance estimation of proposed controllers based on time domain analysis and performance index. The response curve of the PSO based controller has the advantage of having better closed loop time constant, which enables the controller act faster with minimum rise time and settling time. Figure 6 shows The response of IMC controller is more sluggish than the PSO based controller.

The following Table 4 exhibits the error criterion analysis of obtained two controller settings. From the Table 4 based upon the error criterions ISE, IAE, ITAE analysis PSO based PID controller settings gives better result than IMC PID controller settings.

Table 4 Comparison of Error Criterion Analysis

Specifications	IMC	PSO
ISE	0.042	0.034
IAE	0.033	0.024
ITAE	0.01582	0.0052

4. Results and Comparison

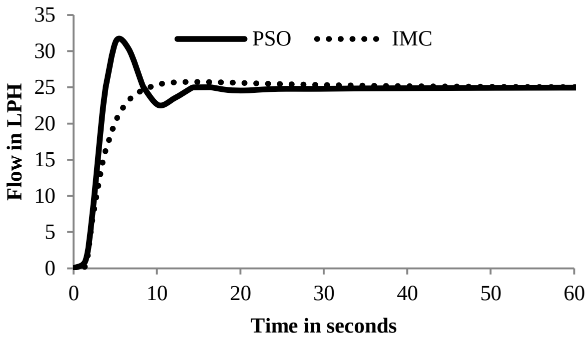

Fig. 6 Comparison of PID Settings for a setpoint of 25 LPH.

The controller parameters are calculated and implemented for set point 25 LPH and shown in figure 6. The servo response of the system was observed by giving set points of 10 LPH, 15 LPH. The corresponding variation of flow from a reference value is noted. The responses for all the set points with controller settings are presented in the Fig 4.

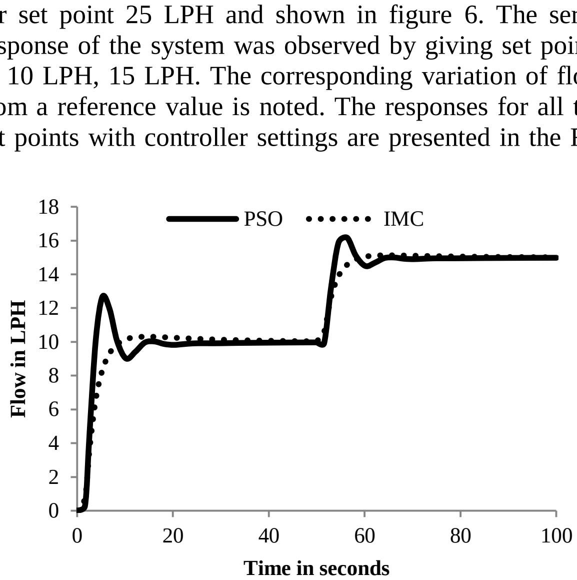

Fig. 7 Servo Response of a Process

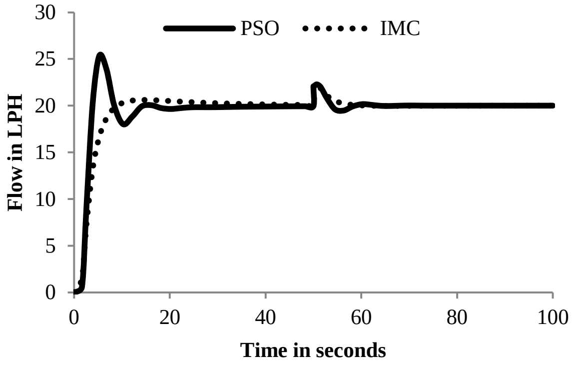

Fig.8 Load Change Response of a Process for PID Controller

Figure 8 clearly states that how fast the PSO based controller reacts to disturbance compare to IMC based controller. A process is disturbed at the time of 50 seconds with 10% of setpoint , the proposed PSO based controller reacts faster and process variable attains steady state quicker than IMC based controller.

Conclusions

In the proposed work, a development based optimization algorithm is achieved through swarm intelligence. The PID controller parameters are obtained using PSO and compared with traditional tuning method. The obtained results which exhibit the effectiveness of the Particle Swarm Optimization tuning technique. The results shown that there is a significant improvement in the time domain specification in terms of lesser rise time and settling time with the application PSO based PID settings. The performance of the proposed controller is also analyzed by applying set point change and load change are shown in figure7 and 8.The error criterion analysis is also states that PSO based PID controller gives better results than IMC based PID controller and are tabulated in Table 4.

Reference

P Aravind, M Valluvan, (2014), PSO Based Optimization of a PID Controller for a Linear Tank Level Control Process, International Journal of Current Engineering and Technology, Vol.4, No.2, PP. 486-490.

Srimathi.R, Sivaranjani.T, Suganya.S, Pradibaa.S, Aravind.P, (2014), Comparative Analysis of PID Controller for a Flow Process, International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, Volume. 2, Issue 3, PP.1186-1188.

P.Aravind, S.M.Giriraj Kumar, (2013), Optimal Tuning of PI Controller using Swarm Intelligence for a Nonlinear Process, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol 2, Issue 12, PP. 5949-5958.

P.Aravind, S.M.Giriraj Kumar, (2013), Performance Optimization of PI Controller in Non Linear Process using Genetic Algorithm, International Journal of Current Engineering and Technology, Vol.3, No.5, PP. 1968-1972

S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008), Particle Swarm Optimization Technique Based Design of PI Controller for a Real-Time Non-Linear Process, Instrumentation Science & Technology, Volume 36, Issue 5.

Jun Zhao, Tianpeng Li and Jixin Qian (2005), Application of Particle Swarm Optimization Algorithm on Robust PID controller Tuning, Springerlink-Verlag Berlin Heidelberg , pp. 948-957.

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

W. B. Bequette (2003), Process Control Modeling and Simulation, Prentice Hall.

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

Javed Alam Jan, Bohumil Sulc (2002), Evolutionary computing methods for optimizing virtual reality process models, International Carpathian control conference ICCC'2002, Malenoice, Czech Republic, May 27-30.

Muller SD, Marchetto J, Airaghi S, Koumoutsakos P (2002), Optimization based on bacterial chemotaxis. IEEE Trans Evolut Comput ; PP 6–29.

Castro LN, Timmis JI (2002). Artificial immune systems: a new computational intelligence approach. London, UK: Springer-Verlag.

Kennedy JF, Eberhart RC, Shi Y.(2001), Swarm intelligence. San Francisco: Morgan Kaufman.\

Dorigo M, Di Caro G.(1999), The ant colony optimization metaheuristicIn:Corne D, Dorigo M, Glover F, editors. New ideas in optimization. McGraw-Hill; PP. 11–32.

Yonghong Tan Xuanju and Dang Achiel Van Cauwenberghe (1999), Generalised nonlinear PID controller based on neural networks, IEEE, Information decision and control.

Asriel U. Levin and Kumpati S. Narendra(1996), Control of nonlinear dynamical systems using Neural Networks- Part II : observability, identification and control, IEEE Transactions on Neural Networks, Vol. 7, No. 1, 1996.

Simon Fabri and Visakan Kadirkamanathan,(1996), Dynamic structure neural networks for stable adaptive control of nonlinear systems, IEEE Transactions on Neural Networks, Vol. 7, No. 5, 1996.

Kennedy JF, Eberhart RC. (1995), Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks, vol. 4. Perth, Australia; PP. 1942– 48.

D. R. Coughanowar (1991), Process Systems Analysis and Control, Tata McGraw Hill.

Astrom, K J.;. Hagglund .T (1984), Automatic tuning of simple regulators with specifications on phase and amplitude margins, Automatica, 20,645-651.

G.H Cohen and G.A Coon (1953), Theoretical Consideration of Retarded Control , Trans ASME 75, PP.827-834.

J. G. Ziegler and N. B. Nichols (1942), Optimum settings for automatic controllers, Trans. Amer. Soc. Mech. Eng., vol. 64, pp. 759–768.