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 Optimization of a PID Controller for a Linear Tank Level Control Process

P Aravind[A*] and M Valluvan[B]

[A]Department of Instrumentation and Control Engineering, Saranathan College of Engineering, Tamil Nadu,India
[B]Department of Electronic and Instrumentation Engineering, JJ college of Engineering and Technology, Tamil Nadu, India.

Accepted 01 March 2014, Available online 01 April 2014, **Vol.4, No.2 (April 2014)**

Abstract

Automatic control has played a vital role in the evolution of engineering and science. It is also important in such industrial operations as controlling temperature, pressure, humidity, flow and level in the process. This paper focus on the development and implementation of an intelligent optimization to obtain a optimum PID controller settings for level control process in cylindrical tank. System identification of the process is done by step test method and found to be First Order Plus Dead Time (FOPDT) model. At first, a Proportional Integral Derivative (PID) controller based on internal model controller (IMC) settings is designed and the results are compared with PSO based PID controller settings. The robustness of the controllers is validated by imposing both servo and regulatory disturbances. The performance comparison of the designed controller settings are analyzed by time domain specification. Better 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, Level process. MATLAB.*

1. Introduction

The extensively used PID industrial controller uses a combination of proportional, integral and derivative action on the reduce error in order control the output at reference value. Over the past half century research work on PID tuning methods are carried out which includes Ziegler-Nichols Ultimate-cycle tuning (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. Since PID control has been an lively a research topic. These technique is highly appreciated by many researchers (Yonghong Tan Xuanju and Dang Achiel Van Cauwenberghe (1999) , Dorigo M, Di Caro G.(1999)) because of the adjustments will be made in controller parameters with minimum attempt. The popularity of PID controllers is due to their functional simplicity, reliability and cost effectiveness. They provide robust and reliable performance for most systems and the PID (Kp- proportional gain, Ki- integral gain, kd – derivative gain) parameters i.e. the proportional, integral and differential constants are tuned to ensure a satisfactory closed loop performance (Astrom, K J and Hagglund .T(1984)). A PID controller improves the transient response of a system by reducing the overshoot, and by limiting the settling time of a system (Kennedy JF, Eberhart RC.(1995)).

Although new methods are proposed for tuning the PID controller, their usage is limited due to difficulties arising at the time of implementation. Owing its simple structure and their real time implementation and tuning, the research community as well as the industrial to pay attention towards computation intelligence (Muller SD, Marchetto J, Airaghi S, Koumoutsakos P (2002), JavedAlam Jan, BohumilSulc (2002), Y Zheng, Liyan Zhang, JixinQianLonghua Ma (2003)) to obtain the PID controller parameters.. The computation efficiency is the advantage of particle swarm optimization algorithms over other tuning techniques.

The advantage of optimization algorithms over other controllers is that they can be integrated in PID tuning with ease and simplicity. Optimality is just with respect to the criterion at hand and the real performance depends on the suitability of the chosen criterion (T.Bartz–Beielstein K.E. Parsopoulos and M.N. Vrahatis (2004)).

The most familiar representatives of swarm intelligence in optimization problems are:, particle swarm optimization (S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008)), artificial immune system (Castro LN, Timmis JI (2002)), food-searching activities of ants and bacterial foraging (Muller SD, Marchetto J, Airaghi S, Koumoutsakos P (2002)). A special approach of swarm intelligence based on simplified simulations of animals' social behaviours, such as fish schooling and bird flocking, 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)).

*Corresponding author: **P Aravind**

A typical standard for good control action that the system is subjected to step analysis and design of various control techniques and suitably of the best suitable controller parameters are estimated by time domain analysis in order to get a response a with minimum overshoot, minimum rise time, minimum settling time . To achieve the effective control standards, design of PSO based PID control is proposed in this work to maintain a liquid level in the tank at desired value.

In this work the process dynamics is modeled from a step response analysis by changing the inflow rate of fluid. For the developed model an IMC based PID control structure is designed and its performance measure is based on rise time, settling time and various performance indices are compared with PSO based PID controller.

In section 2, we have discussed in detail about the development of the mathematical model for the linear tank process. The tuning method of conventional techniques and the explanation of the IMC based PID and their implementations are discussed in section 3. The comparative studies and results are given in section 4. The conclusions arrived, based on the results in section 5.

2. Experimental

The real time experimental system consisting of a transparent linear tank, reservoir and water pump, current to pressure converter, compressor, Electronic Type Level Transmitter and a Personal Computer which acts as a controller forms a closed loop system.

1	Process Tank
2	Level Transmitter
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
Level Transmitter (LT)	Electronic-Range 0–90 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

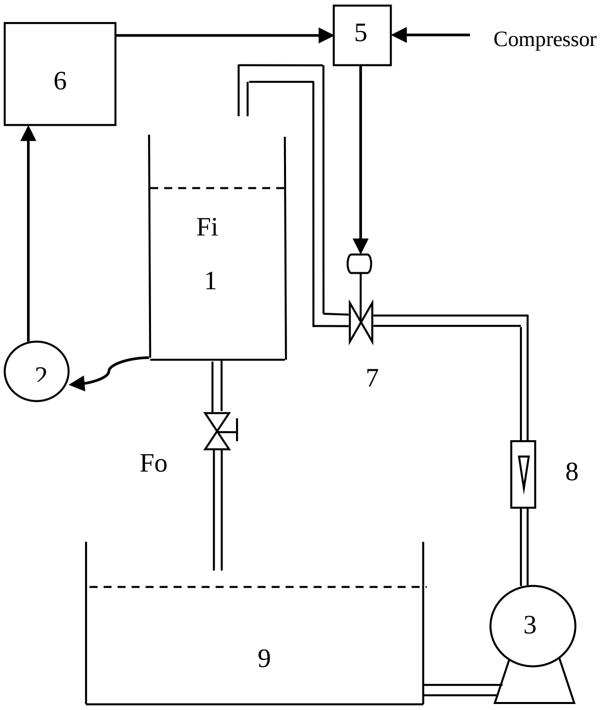

Fig.1. Piping and Instrument diagram of Experimental Setup

2.1 Step Test Method

Step response based methods are most commonly used for system identification. A large number of graphical methods are available in literature and they have been used effectively in real time applications to obtain the model. At first, the inlet valve is at fully opened condition and outlet valve is set to a particular restriction. The open loop step response is obtained by varying the inflow rate, the experimental results are noted in terms of time and height or level. The models are identified by process reaction curve method (PRC) (D. R. Coughanowar (1991)) and SunderasanKumaraswamy(SK) method (S. M. GirirajKumar, R. Sivasankar, T.K. Radhakrishnan, V. Dharmalingam and N. Anantharaman (2008)) method. 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 SunderasanKumaraswamy(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,

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

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

The calculated models are validated with real time results is presented in figure 2 (P Aravind, M Valluvan, B Sangeetha (2014)) and model obtained through SK method of identification is effectively matches to real time response. From the response of the real time system we obtain the mentioned constants for SK method and thereby we get the FOPTD models for the real time linear tank process as,

$$G(s) = \frac{1.7e^{-27.99s}}{203.68s+1} \quad (2)$$

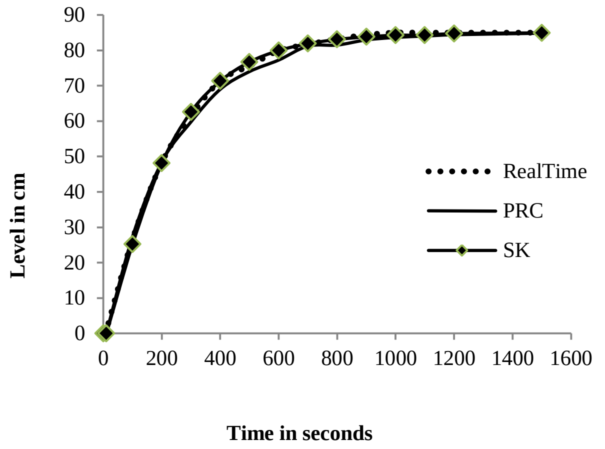

Fig.2. Comparison of real time and simulated responses

3. Controller Design

This paper addresses the implementing of PID parameters in two design modes. The IMC based PID control technique and PSO based PID controller. With these methods, tuning of PID parameters is performed to achieve a robust design with the desired response time. PID controller is tuned by manually adjusting design criteria in two design modes. The tuner computes PID parameters that robustly stabilize the system.

3.1 IMC based PID

In order to arrive at a PID equivalent form for process with a time delay, we must apporimate the dead time 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 (3and 4),

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

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

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

3.2Optimization 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=100
Number of iterations=100
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 *gbest.* 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 pbest, goto Step5 else goto step8.
Step6: Pbest equal to current fitness value
Step7: If current fitness value is better than Gbest, goto to Step8 else goto step 8
Step9: Gbest is equal to current fitness value.
Step10: Update position and velocity of particles
Step11: Goto step10 if stop criteria met else goto 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 100 particles considered as potential solution is chosen. Therefore the best values for 100 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 , Ki and Kd for 100 iterations

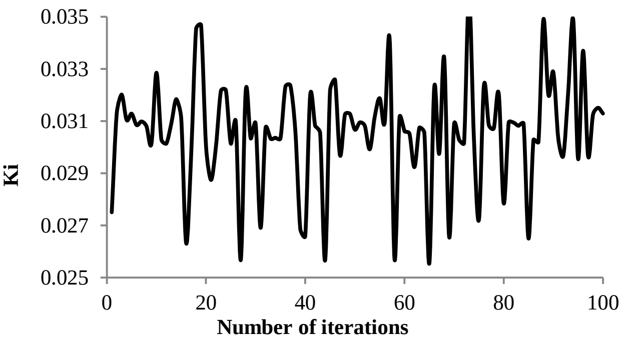

Fig.4. Best solutions of Kp , Ki and Kd for 100 iterations

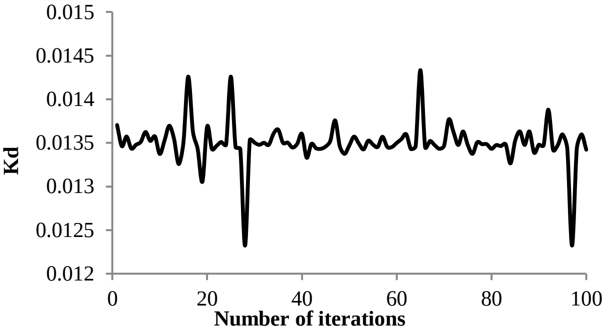

Fig.5 Best solutions of Kp , Ki and Kd for 100 iterations

The PID controller was formed based upon the respective parameters for 100 iterations, and the gbest (global best) solution was selected for the set of parameters.

Table 2: Tuned Gain values of controller

Controller	IMC	PSO
K_p	3.668	5.588
K_i	0.017	0.031
K_d	39.95	0.013645

Table 3: Comparison of Time Domain Analysis and Performance Indices:

Specifications	IMC	PSO
Rise Time (seconds)	150	70
Peak Time(seconds)	225	100
Settling Time (seconds)	400	150
ISE	22.45	11.27
IAE	48.89	25.09

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

4. Results and Comparison

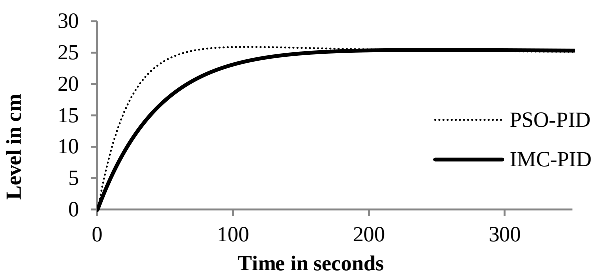

Fig.6 Comparison of IMC and PSO-PID for a setpoint of 25 cm.

The controller parameters are calculated and implemented for set point 25 cm and shown in figure 6. The servo response of the system was observed by giving set points of 15 cm, 27 cm. The corresponding variation of level from a reference value is noted. The responses of the tank for all the set points with controller settings are presented in the Fig 4.

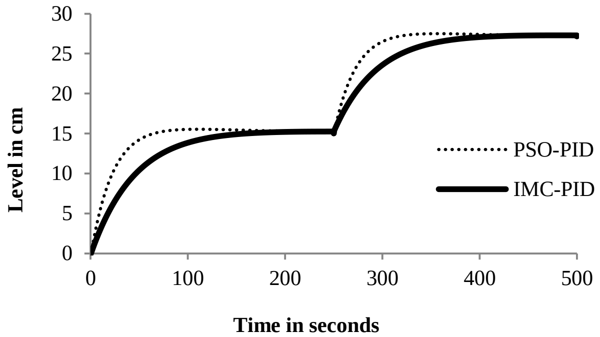

Fig. 4 Servo Response of a Process

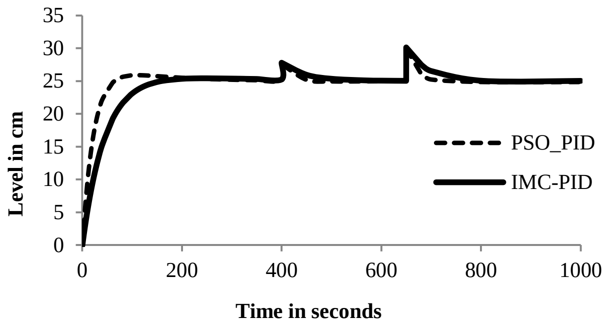

Fig.5 Load Change Response of a Process for PID Controller

Figure 5 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 400 seconds with 10% and in 650 seconds with 20% of setpoint , the proposed PSO based controller reacts faster and process variable attains steady state quicker than IMC based controller.

Conclusions

In this work we have presented scenario based optimization algorithm through swarm intelligence. The PID controller parameters are obtained using PSO. The design, implementation and testing of such PSO based PID controller parameters are discussed in detail and compared with traditional tuning method. The simulation results are obtained which demonstrate the efficiency and effectiveness of the proposed tuning technique.

It is shown graphically that there is a substantial 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 setpoint change and load change and are presented in figure 4 and 5.

References

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

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

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

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.

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

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

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

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

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

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

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

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.

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

P Aravind, M Valluvan, B Sangeetha (2014),Model Based Controller Design for Level Process, International Journal for Computer Applications, Volume 88- Number 18.

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

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