

PSO based Tuning of a PID Controller for a High Performance Drilling Machine

S.M.GirirajKumar
School of Electrical and Electronics
Engineering,
SASTRA University,Thanjavur,
Tamilnadu-613402

Deepak Jayaraj
School of Mechanical engineering,
SASTRA University,Thanjavur,
Tamilnadu-613402

Anoop.R.Kishan
School of Mechanical engineering,
SASTRA University,Thanjavur,
Tamilnadu-613402

ABSTRACT

The paper deals with optimal tuning of a Proportional-Integral-Derivative (PID) controller used in a high performance drilling system for controlling the output obtained and hence to minimize the integral of absolute errors (IAE). The main objective is to obtain a stable, robust and controlled system by tuning the PID controller using Particle Swarm Optimization (PSO) algorithm. The incurred value is compared with the traditional tuning techniques like Ziegler-Nichols and is proved better. Hence the results establishes that tuning the PID controller using PSO technique gives less overshoot, system is less sluggish and reduces the IAE.

Keywords:

PSO, Ziegler Nichols, PID controller, IAE

1. INTRODUCTION

PIDcontrol is a generic feedback control technology and it makes up 90% of automatic controllers in industrial control systems. The PID control was first placed in the market in 1939 and has remained the most widely used controller in process control until today. The basic function of the controller is to execute an algorithm based on the control engineer's input and hence to maintain the output at a level so that there is no difference between the process variable and the setpoint[1]. The popularity of PID controllers is due to their functional simplicity and reliability. They provide robust and reliable performance for most systems and the PID parameters are tuned to ensure a satisfactory closed loop performance [2]. A PID controller improves the transient response of a system by reducing the overshoot, and by shortening the settling time of a system [3]. The PID control algorithm is used to control almost all loops in process industries and is also the cornerstone for many advance control algorithms and strategies [1]. For this control loop to function properly, the PID loop must be properly tuned. Standard methods for tuning include Ziegler-Nichols Ultimate-cycle tuning [4], Cohen-Coon's [6], Astrom and Hagglund[5] and many other traditional techniques.

Although new methods are proposed for tuning the PID controller, their usage is limited due to complexities arising at the time of implementation.

The PI or PID controller implementation has been recommended for the control of processes of low to medium order, with small time delays, when parameter setting must be done using tuning rules and when controller synthesis is performed either once or more often (Isermann, 1989). However, despite decades of development work, surveys indicating the state of the art of control industrial practice report sobering results. For example, Ender (1993) states that, in his testing of thousands of control loops in hundreds of plants, it has been found that more than 30% of installed controllers are operating in manual mode and 65% of loops operating in automatic mode poorly tuned. The Handbook of PI and PID controller Tuning Rules by Aidan O.Dwyer has recorded 408 separate sources of tuning rules since the first such rule was published by Callender et al. (1935). In a striking statistic, 293 sources of tuning rules have been recorded since 1992 reflecting the upsurge of interest in the use of the PID controller recently.

Although these many tuning rules are available in literature most of the rules are applicable for a first order system with a time delay. So clearly they are not meant to be applied for higher order non linear systems. In order to apply them we may go for approximating the system to a FOPTD (first order plus dead-time) model: $K_p e^{-\tau_d s} / (\tau s + 1)$. This can primarily be done either using Taylor's approximation or Skogestad's approximation. But the word approximation itself suggests that the parameters obtained using the application of these traditional tuning rules on the approximated system will also be a compromise. The intensity of compromise depends on the magnitude of degree degradation. This approximation could itself fail if the higher order system has a complex time constant where approximation will be a tedious process. There are also certain methods that are applicable only to certain specific systems so that too reduces the acceptance of the method. These conventional PID tuning methods may be very useful for determining initial tuning parameters on the situation that open-Loop Step Response Test is executable, and that trial and error can be accepted. However they are not suitable for fine tuning in critical condition.

Tuning a PID controller means setting the proportional, integral and derivative values to get the best possible control for a particular process. Tuning a controller means adjusting the controller gains to satisfy the performance specifications like margin of stability, transient response and bandwidth. Although trial and error can be used, analytical approach is used to compute the gain that can minimize the performance index. This is

represented as a function of error. The commonly employed performance indices are:

- Integral Absolute Error
- Integral Squared Error
- Integral of time multiplied by absolute value of error
- Integral of time multiplied by squared error

2. Dynamic Model of a High-Performance Drilling Process

The modeling of a high-performance drilling process [7] includes the modeling of the feed drive system, the spindle system and the cutting process. In this paper, the overall plant model is obtained by experimental identification using different step shaped disturbances in the command feed. The drilling force, F , is proportional to the machining feed, and the corresponding gain varies according to the work piece and drill diameter. The overall system of the feed drive, cutting process and dynamometric platform was modelled as a third-order system, and the experimental identification procedure yielded the transfer function as:

$$G(s) = \frac{1958}{s^3 + 17.89s^2 + 103.3s + 190.8}$$

Where s is the Laplace operator, f is the command feed, and F is the cutting force. The model does have certain limits in representing the complexity and uncertainty of the drilling process. However, it provides a rough description of the process behaviour that is essential for designing a network-based PID control system

3. Particle Swarm Optimization

Optimization algorithms are another area that has been receiving increased attention in the past few years by the research community as well as the industry [8,9]. An optimization algorithm is a numerical method or algorithm for finding the maxima or the minima of a function operating with certain constraints [10].

Computational intelligence (CI) is a successor of artificial intelligence relying on Evolutionary computation, which is a famous optimization technique. Computational intelligence combines elements of learning; adaptation and evolution to create programs that are, in some sense, intelligent. Computational intelligence research does not reject statistical methods, but often gives a complementary view [11]. Computational intelligence finds its fundamental application in the area of fitness function design, methods for parameter control, and techniques for multimodal optimization. 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 [12].

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 [17].

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 [18]. Thus a common practice followed in industries for the purpose of control design and process analysis is to model the dynamics of the processes near the operating point by simpler models such as FOPTD [19].

The objective of the paper is to use the PSO algorithm in order to obtain optimal PID controller settings for a high performance drilling process, which is non-linear in nature. Every possible controller setting represent a particle in the search space which changes its parameters proportionality constant, K_p , integral constant, K_i , in order to minimize the error function (objective function in this case). The error function used here is Integral Time of Absolute errors (IAE). The tuning results of conventional techniques are discussed in section 3. Section 4 and 5 deal with the explanation of the PSO algorithm and its implementation. The comparative studies and results are given in Section 6. The conclusions arrived, based on the results is given in Section 7 followed by conclusion and reference in section 8 and 9 respectively.

4. Conventional Design Technique

Over the last fifty years, numerous methods have been developed for setting the parameters of a PID controller. In this paper the PSO tuning technique is compared with Ziegler Nichols [4] tuning method. In the 1940's, Ziegler and Nichols devised two empirical methods for obtaining controller parameters. The Ziegler-Nichols closed-loop tuning method allows you to use the ultimate gain value, K_u , and the ultimate period of oscillation, P_u , to calculate K_c . It is a simple method of tuning PID controllers and can be refined to give better approximations of the controller. Even though this method was devised in 1940, it is still one of the most widely used methods of tuning a PID controller because of its

applicability to almost all the systems irrespective of its order. Although many other methods of tuning are being developed in this field in recent years not many have proved to be as efficacious as the above mentioned one.

The ultimate gain value for the above mentioned system has been calculated to be $K_u=0.854$ and the ultimate period of oscillation is $P_u =0.6160$. Based on Ziegler-Nichols tuning method the tuning parameters has been calculated to be

$$K_p= 0.5128 \quad K_i= 1.6649 \quad K_d= 0.0395$$

The frequency response of the system with PID tuned with Ziegler-Nichols has been compared with our method of tuning in the forthcoming paragraphs

5. PSO BASED PID CONTROLLER

5.1 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 [19]. 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 p_{best} . 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.

Let D be the dimension of the search space taken into consideration and $X_i = [x_{i1}, x_{i2}, \dots, x_{iD}]^T$ denote the current position of i^{th} particle of the swarm, Then: $X_i^{pbest} = [x_{i1}^{pbest}, x_{i2}^{pbest}, \dots, x_{iD}^{pbest}]^T$ denote the best position ever visited by the particle. $X_{gbest} = [x_{i1}^{gbest}, x_{i2}^{gbest}, \dots, x_{iD}^{gbest}]^T$ represents 'gbest', i.e the best position obtained this far by any particle in the population. $V_i = [v_{i1}, v_{i2}, \dots, v_{iD}]^T$ represents the velocity of i^{th} particle. $V_{imax} = [v_{i1}^{max}, v_{i2}^{max}, \dots, v_{iD}^{max}]^T$ 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:

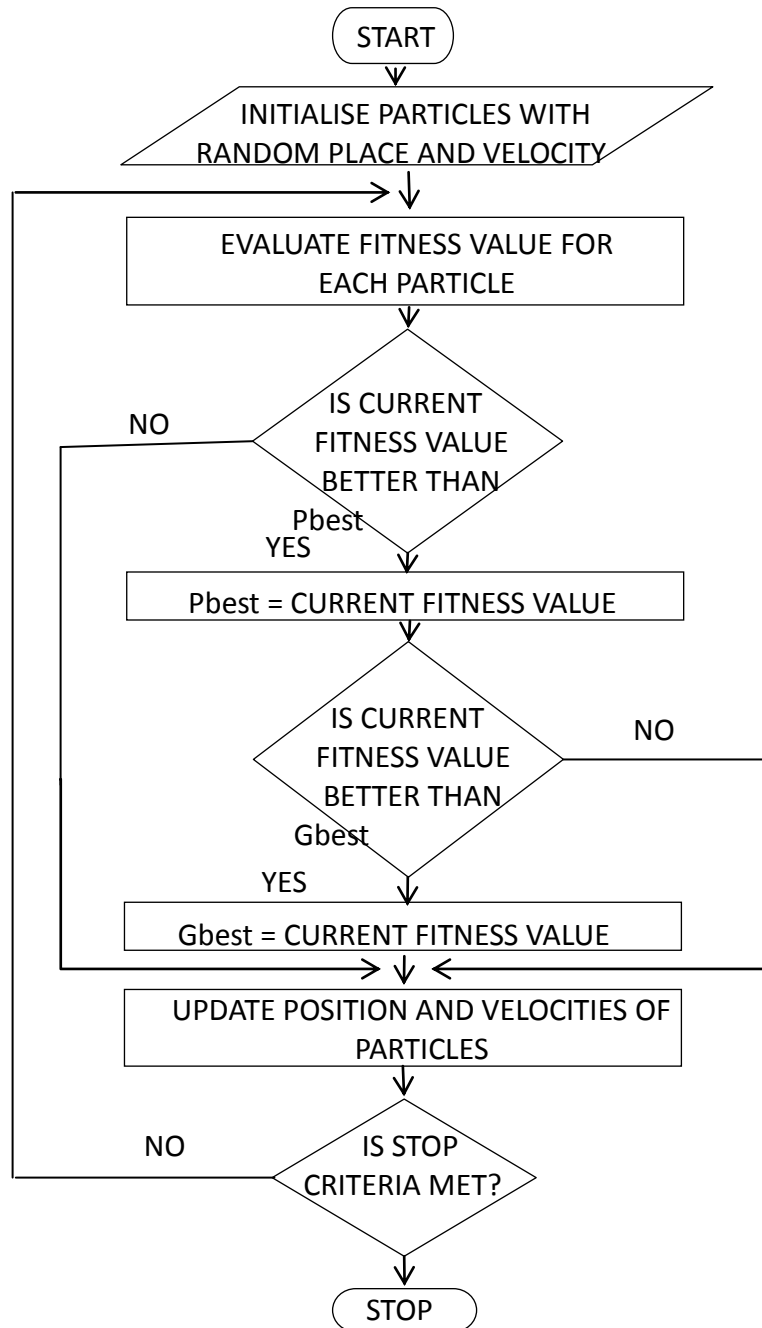
$$V_{id} = w * v_{id} + c1 * r1 * (x_{id}^{pbest} - x_{id}) + c2 * r2 * (x_{id}^{gbest} - x_{id}) \quad \text{----- (2)}$$

$$V_{id} = v_{d}^{max}, v_{id} > v_{d}^{max} \\ -v_{d}^{max}, v_{id} < -v_{d}^{max} \quad \text{----- (3)}$$

$$X_{id} = X_{id} + V_{id} \quad \text{----- (4)}$$

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, in figure.1.



6. IMPLEMENTATION OF PSO ALGORITHM

The optimal values of the PID controller parameters K_p , K_i and K_d , is found using PSO. All possible sets of controller parameter values are particles whose values are adjusted so as to minimize the objective function, which in this case is the error criterion, which is discussed in detail. For the PID controller design, it is

ensured the controller settings estimated results in a stable closed loop system.

6.1 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. The maximum velocity affects the ability of escaping from local optimization and refining global optimization. The size of swarm balances the requirement of global optimization and computational cost. Initializing the values of the parameters is as per table.1.

Table.1.PSO selection parameters

Population size	100
Number of iterations	100
Velocity constant,c1	2
Velocity constant,c2	2

6.2. Performance Indices for the PSO Algorithm

The objective function considered is based on the error criterion. The performance of a controller is best evaluated in terms of error criterion. A number of such criteria are available and in the proposed work, controller’s performance is evaluated in terms of:

- Integral of Absolute Errors (IAE) criterion, given by

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

The IAE weights the error with time and hence emphasizes the error values over arrange of 0 to T, where T is the expected settling time

- Integral Square of Errors (ISE) criteria. The error criterion is given by the equation:

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

- Integral of Time multiplied by Absolute Errors (ITAE) criterion, given by:

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

The time is considered as, t=0 to t=Ts, where Ts is the settling of the system to reach steady state condition for a unit step input.

- Mean Square Error(MSE) criterion, given by:

$$MSE(\hat{\theta}) = E[(\hat{\theta} - \theta)^2]$$

6.3 Termination Criteria

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, in this case is nothing but reciprocal of the magnitude of the objective function, since we consider for a minimization of objective function. In this paper the termination criteria is considered to be the attainment of satisfactory fitness value which occurs with the maximum number of iterations as 100.

For each iteration the best among the 100 particles considered as potential solution are chosen Therefore the best values for 100 iterations is sketched with respect to iterations, and are as shown in Figs. 2,3 and 4.

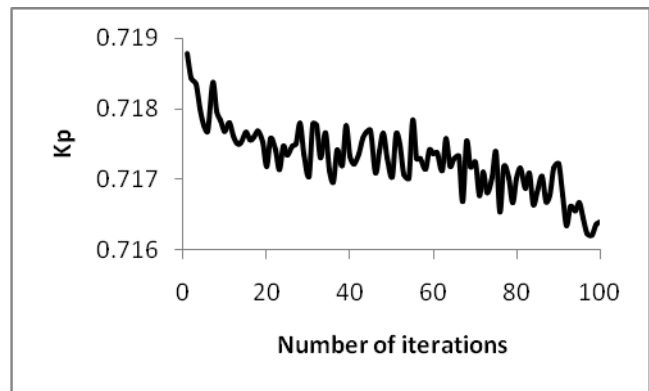


Fig. 2. Best solutions of Kp for 100 iterations.

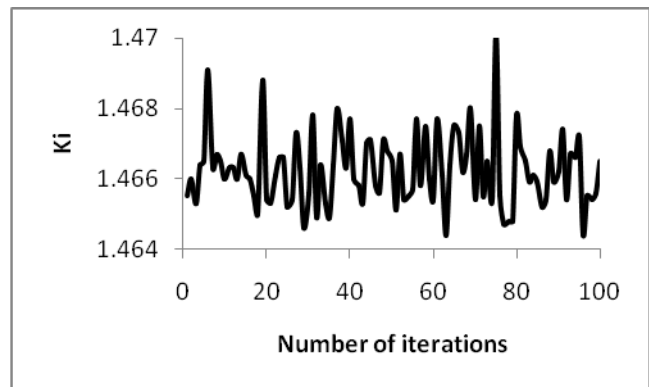


Fig. 3. Best solutions of Ki for 100 iterations

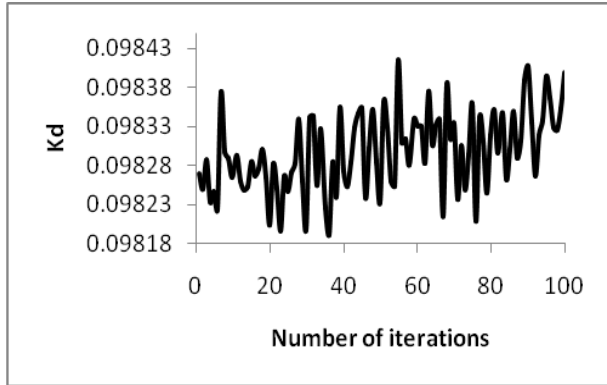


Fig. 4. Best solutions of Kd for 100 iterations.

PID controller was formed based upon the respective parameters for 100 iterations, and the global best solution was selected for the set of parameters, which had the minimum error. A sketch of the error based on IAE criterion for 100 iterations is as given in Fig. 5.

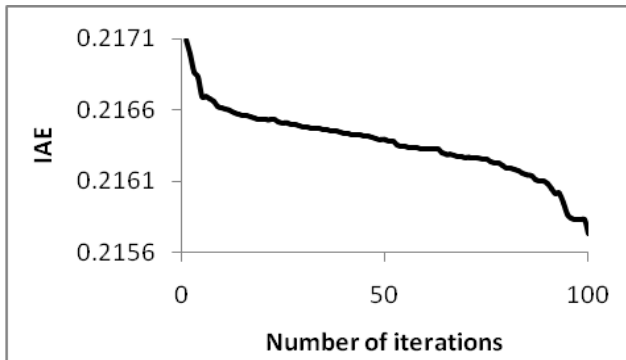


Fig. 5. IAE values for 100 iterations.

It was seen that the error value tends to decrease for a larger number of iterations. As such the algorithm was restricted to 100 iterations for beyond which there was only a negligible improvement. Based on PSO for the application of the PID tuning we get the PID tuning parameters for the model as

$$K_p = 0.7164 \quad K_i = 1.4665 \quad K_d = 0.0984.$$

7. RESULTS AND COMPARISON

Analysis shows that the design of proposed controller gives a better robustness, and, the performance is satisfactory over a wide range of process operations. Simulation results show performance improvement in time domain specifications for a step response. Using the PSO approach, global and local solutions could be simultaneously found for better tuning of the controller parameters.

The PID value which was obtained by the PSO algorithm is compared with that of the one derived from Zeigler-Nichols method in various perspectives, namely robustness and stability

performances. All the simulations were implemented using MATLAB.

7.1 Performance related to steady state conditions

A reference input of unit step was given to the closed loop system. The above procedure was implemented for the controller for which the PID values were tuned by Zeigler-Nichols as well as SA algorithm. The response curves obtained are as shown in the Fig.6

A comparison of time domain specifications peak overshoot, peak time, rise time and settling time are tabulated as given in table (2). It is found very clearly that the PSO based controller drastically reduces the overshoot by a large value. Settling time, Rise Time and Peak Time have also improved. Henceforth outperforms that of the traditionally tuned controller with Zeigler-Nichols criterion.

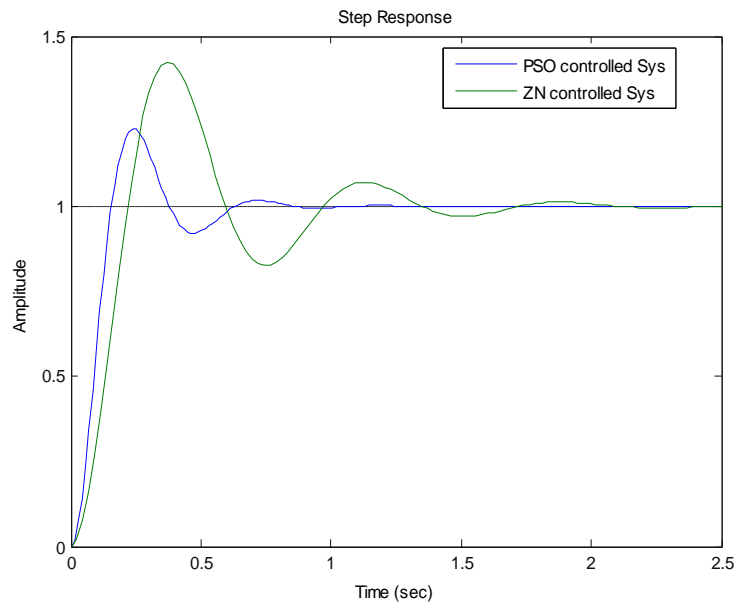


Fig.6. Unit step response with Z-N and PSO based controller

Table.2. Comparison of time domain specifications

Type of controller	Zeigler-Nichols	PSO
Peak time(sec)	0.4	0.2
Peak overshoot (%)	43	18
Rise Time(sec)	0.223	0.17
Settling time(sec)	2.7	1.4

7.2 Robustness Investigation

The PID controllers tuned by the PSO based method should not be compared only with their time domain responses but also with its performance index from the four major error criterion techniques of Integral Time of Absolute Error (ITAE) ,Integral of Absolute Error(IAE) ,Integral Square of Error(ISE)and Mean Square Error (MSE).Robustness of the controller is defined as its ability to tolerate a certain amount of change in the process parameters without causing the feedback system to go unstable.

For the proposed model the comparison of performance index were done and are listed as per the given Table 3.

Table 3: Comparison of performance index obtained for Z-N and PSO

Performance index	ZN	PSO
ITAE	1.1824	0.2157
IAE	3.4496	1.9047
ISE	1.8844	1.2291
MSE	0.0369	0.0241

From these values obtained it is clearly visible that the error magnitude obtained for Z-N is far too high as compared to the proposed tuning method based on PSO algorithm.

8. CONCLUSION

In this paper, a systematic design method aiming at enhancing PID control for complex processes is proposed. It is shown analytically and graphically that there is a substantial improvement in the time domain specification in terms of lesser rise time, peak time, settling time as well as a lower overshoot. The performance index for various error criterions for the proposed controller using PSO algorithm is proved to be less than the controller tuned by Ziegler Nichols method.

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, starting the design with a reduced amount of information about the controller (type and allowable range of the parameters), but keeping sight of the behavior of the control system. The performance of the above said method of tuning a PID controller can even be proved to be better than the method of tuning the controller after approximating the system to a FOPTD model, and using the traditional techniques, regarding which a rich literature is available. 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.

9. REFERENCES

1. M Araki: Control systems, Robotics and Automation-Vol II – PID Control –,Kyoto University ,Japan
2. Kim Dong Hwa and Park Jin Ill: Intelligent PID Controller Tuning of AVR system using GA and PSO: Springer-Verlag Berlin Heidelberg: ICIC 2005, Part II, LNCS 3645, pp 366-375.(2005).
3. K.J.Astrom, T.Hagglund, The future of PIDcontrol, Control Eng.Pract.9(11)(2001)1163–1175.
4. Ziegler, G. and Nichols, N. B, 1942.Optimum settings for automatic controllers, Trans. ASME, 64,759-768.
5. Astrom, K J.,Hagglund .T,1984, Automatic tuning of simple regulators with specifications on phase and amplitude margins, Automatica, 20,645-651.
6. G.H Cohen and G.A Coon: Theoretical Consideration of Retarded Control , Trans ASME 75,pp.827/834,(1953)
7. Haber ,Rodolf., Toro1, Raúl M. and. Alique, José R, 'Using Simulated Annealing for Optimal Tuning of a PID Controller for Time-Delay Systems. An Application to a High-Performance Drilling Process', Springer.
8. K.E.Parsopoulos and M.N.Vrahatis,“Particle swarm optimizer in noisy and continuously changing environment”,Indianapolis.IN(2001)
9. Hirotaka Yoshida, Kenichi Kawata, yoshikazuFukuyana, yosuke Nakanishi, “A particle swarm optimization for reactive power and voltage control considering voltage stability”, IEEE international conference on intelligent system applications to power systems(ISAP'99), Rio de Janeiro, April 4-8 (1999).
10. Zwe-Lee Gaing, “ A particle swarm optimization approach for optimum design of PID controller in AVR system”, IEEE transactions on energy conversion,November 6,(2002).
11. JavedAlam Jan and BohumilSulc, “Evolutionary computing methods for optimizing virtual reality process models”, International Carpathian control conference ICC'2002, Malenovice Czech republic, May 27-30, (2002).
12. K.E. Parsopoulos and M.N. Vrahatis, “Recent approaches to global optimization problems through Particle Swarm Optimization, Natural Computing 1: 235–306, (2002).
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, JixinQianLonghua 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. Jun Zhao, Tianpeng Li and JixinQian “Application of Particle Swarm Optimization Algorithm on Robust PID

- controller Tuning”, Springerlink-Verlag Berlin Heidelberg , pp. 948-957, (2005).
17. Haibing Hu, Qingbo Hu, Zhengyu Lu, DehongXu “Optimal PID controller Design in PMSM Servo System via Particle Swarm Optimization” 31st Annual Conference of IEEE, Industrial Electronics Society.
19.
 18. RohitRamachandran, S.Lakshminarayanan and G.P Rangaiah”Process identification using open-loop and closed loop step responses”Journal of the institute of Engineers,Singapore vol.45,Issue6 (2005).