Arid Zone Journal of Engineering, Technology and Environment. August, 2012; Vol. 8, 125-131
Copyright© Faculty of Engineering, University of Maiduguri, Nigeria.
Print ISSN: 1596-2490, Electronic ISSN: 2545-5818
www.azojete.com.ng

PROPORTIONAL–INTEGRAL–DERIVATIVE (PID) CONTROLLER TUNING USING PARTICLE SWARM OPTIMIZATION ALGORITHM

Bassi, J.S.[1*], Omizegba, E.E. [2] and Mshelia, P.Y.[1]
([1]*Department of Computer Engineering, University of Maiduguri, Maiduguri, Nigeria*
[2]*Electrical and Electronics Engineering Programme, Abubakar Tafawa Balewa University, Bauchi, Nigeria*)
*Corresponding author, e-mail address: sjbassi74@yahoo.co.uk

Abstract
The proportional-integral-derivative (PID) controllers are the most popular controllers used in industry because of their remarkable effectiveness, simplicity of implementation and broad applicability. However, manual tuning of these controllers is time consuming, tedious and generally lead to poor performance. This tuning which is application specific also deteriorates with time as a result of plant parameter changes. This paper presents an artificial intelligence (AI) method of particle swarm optimization (PSO) algorithm for tuning the optimal proportional-integral derivative (PID) controller parameters for industrial processes. This approach has superior features, including easy implementation, stable convergence characteristic and good computational efficiency over the conventional methods. Ziegler- Nichols, tuning method was applied in the PID tuning and results were compared with the PSO-Based PID for optimum control. Simulation results are presented to show that the PSO-Based optimized PID controller is capable of providing an improved closed-loop performance over the Ziegler- Nichols tuned PID controller Parameters. Compared to the heuristic PID tuning method of Ziegler-Nichols, the proposed method was more efficient in improving the step response characteristics such as, reducing the steady-states error; rise time, settling time and maximum overshoot in speed control of DC motor.

Keywords: *PID Controller*, particle swarm optimization Algorithm, Ziegler- Nichols method, simulation

1. Introduction

The PID controller is regarded as the workhorse of the process industry. Today, many industrial processes are controlled using proportional-integral-derivative (PID) controllers. The popularity of the PID controllers can be attributed to their good performance in a wide range of operating conditions, functional simplicity, which allows Engineers to operate them in a simple, straightforward manner and familiarity, with which it is perceived amongst researchers and practitioners within the process control industries (Pillay and Govender, 2007). In spite of its widespread use, one of its main short-comings is that there is no efficient tuning method for this type of controller (Åström and Hägglund, 1995).

Several methods have been proposed for the tuning of PID controllers. Among the conventional PID tuning methods, the Ziegler–Nichols method (Ogata, 1987) may be the most well known technique. For a wide range of practical processes, this tuning approach works quite well. However, sometimes it does not provide good tuning and tends to produce a big overshoot. Therefore, this method usually needs retuning before applied to control industrial processes. To enhance the capabilities of traditional PID parameter tuning techniques, several intelligent approaches have been suggested to improve the PID tuning, such as those using genetic algorithms (GA) (Mahony *et al.*, 2000; Krishnakumar and Goldberg, 1992; Varsek *et al.*, 1993) and the particle swarm optimization (PSO) (Gaing, 2004). With the advance of computational methods in the recent times, optimization algorithms are often proposed to tune the control parameters in order to find an optimal performance (Gaing, 2004). It has been asserted that more than half of the industrial controllers in use today utilize PID or modified PID control schemes (Ogata, 2005). This wide spread acceptance of the PID controllers is largely attributed to their simplicity and robust performance in wide range of operating conditions. One major problem faced in the deployment of PID controllers is the proper tuning of gain values (Visioli, 2001). Over the years, various heuristic techniques were proposed for tuning the PID controller. Among the earliest methods is the classical Ziegler-Nichols tuning procedure, however, it is difficult to determine optimal or near optimal parameters with this because most industrial plants are often very complex having high order, time delays and nonlinearities (Kwok *et al.*, 1993; Gaing, 2004 and Krohling and Rey, 2001).

This paper attempts to develop an artificial intelligence(AI) automatic PID tuning scheme using PSO algorithm that can automatically acquire (or re-adapt) the PID parameters during plant operation in a routine way . The result is expected to show the effectiveness of the modern optimization such as PSO in control engineering applications. PSO algorithm is a stochastic algorithm based on principles of natural selection and search algorithm. There, are many evidences of intelligence for the posed domains in animals, plants, and generally living systems. For example, ants foraging, birds flocking, fish schooling, bacterial chemotaxis are some of the well-known examples in category.

2. Overview of Particle Swarm Optimization (PSO) Algorithm

PSO is optimization algorithm based on evolutionary computation technique. The basic PSO algorithm is developed from research on swarm such as fish schooling and bird flocking. After it was firstly introduced in 1995 (Kennedy and Eberhart, 1995), a modified PSO was then introduced in 1998 to improve the performance of the original PSO algorithm. A new parameter called inertia weight is added. This is a commonly used PSO algorithm where inertia weight is linearly decreasing during iteration in addition to another common type of PSO algorithm which was reported by Clerc (1999) and Eberhart and Shi (2000). The later is the one used in this paper.

In PSO, instead of using genetic operators, individuals called as particles are “evolved” by cooperation and competition among themselves through generations. A particle represents a potential solution to a problem. Each particle adjusts its flying according to its own flying experience and its companion flying experience. Each particle is treated as a point in a D-dimensional space. The *i*th particle is represented as $X_I = (X_{i!}, X_{i2}, ..., X_{iD})$. The best previous position (giving the minimum fitness value) of any particle is recorded and represented as $P_I = (P_{i!}, P_{i2}, ..., P_{iD})$, this is called *pbest*. The index of the best particle among all particles in the population is represented by the symbol *g*, called as *gbest*. The velocity for the particle *i*, is represented as $V_I = (V_{i!}, V_{i2}, ..., V_{iD})$. The particles are updated according to equations (1) and (2).

$$v_{i,m}^{(t+1)} = w \cdot v_{i,m}^{(t)} + c_1 * rand() * (pbest_{i,m} - x_{i,m}^{(t)} + c_2 * rand() * (gbest_m - x_{i,m}^{(t)}) \tag{1}$$

$$x_{i,m}^{(t+1)} = x_{i,m}^{(t)} + v_{i,m}^{(t+1)} \tag{2}$$

where: c_1 and c_2 are two positive constants, while *rand* () is random function between 0 and 1, and *n* represents iteration. Equation (1) is used to calculate particle's new velocity according to its previous velocity and the distances of its current position from its own best experience (position) and the group's best experience. Then the particle flies toward a new position according to Equation (2). The performance of each particle is measured according to a pre-defined fitness function (performance index), which is related to the problem to be solved. Inertia weight, *w* is brought into the equation to balance between the global search and local search capability. It can be a positive constant or even positive linear or nonlinear function of time. It has been also shown that PSO with different number of particles (swarm size) has reasonably similar performance.

3. Materials and methods

3.1 Problem formulation

PID controller consists of Proportional, Integral and Derivative gains. The feedback control system is illustrated in Fig. 1 where *r*, *e*, u, *y* are respectively the reference, error, controller output and controlled variables.

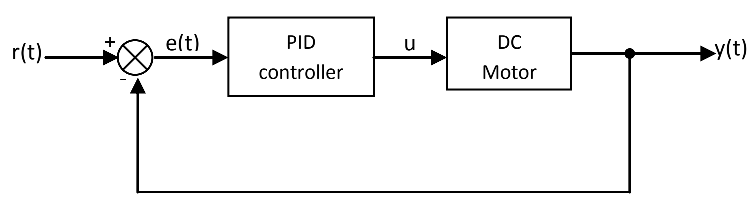


Figure 1: A common feedback control system

The PID controller is described in equation (3) as:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de}{dt} \tag{3}$$

Where: u_t is the controller output, e_t is the error, and *t* is the sampling instance. The factors k_p, k_i and k_d are the proportional, integral and derivatives gains (or parameters) respectively that are to be tuned. The DC motor model is described in equation (4) as:

$$G_c(s) = \frac{6.3223s^2 + 18s + 12.812}{s} \tag{4}$$

Furthermore, performance index is defined as a quantitative measure to depict the system performance of the designed PID controller. Using this technique an 'optimum system' can often be designed and a set of PID parameters in the system can be adjusted to meet the required specification. For a PID- controlled system, there are often four indices to depict the system performance: ISE, IAE, ITAE and ITSE. Therefore, for the PSO-based PID tuning, the ITAE performance index given in equation (5) will be used as the objective function. In other word, the objective in the PSO-based optimization is to seek a set of PID parameters such that the feedback control system has minimum performance index.

$$ITAE = \int_0^\infty t|e(t)|dt \tag{5}$$

3.2 Tuning of PID controller using Ziegler Nichols Method

The first method of Z-N tuning is based on the open-loop step response of the system. The open-loop system's S shaped response is characterized by the parameters, namely the process time constant T and L. These parameters are used to determine the controller's tuning parameters. The second method of Z-N tuning is closed-loop tuning method that requires the determination of the ultimate gain and ultimate period. The method can be interpreted as a technique of positioning one point on the Nyquist curve (Astrom and Hagglund, 1995). This can be achieved by adjusting the controller gain (Ku) till the system undergoes sustained oscillations (at the ultimate gain or critical gain), whilst maintaining the integral time constant (Ti) at infinity and the derivative time constant (Td) at zero. This paper uses the second method as shown in Table 1.

Table 1: Ziegler-Nichols open-loop tuning rule

Controller	K_P	T_I	T_D
P	$\frac{T_p}{L_p K_p}$	∞	0
PI	$0.9\frac{T_p}{L_p K_p}$	$3.33\,L_p$	0
PID	$1.2\frac{T_p}{L_p K_p}$	$2\,L_p$	$0.5\,L_p$

3.3 Implementation of PSO-based PID Tuning

Stochastic Algorithm can be applied to the tuning of PID controller gains to ensure optimal control performance at nominal operating conditions. PSO algorithm is employed to tune PID gains/parameters (Kp, Ki, Kd) using the model in Equation (2). PSO algorithm firstly produces initial swarm of particles in search space represented by matrix. Each particle represents a candidate solution for PID parameters where their values are set in the range of 0 to 100. For this 3-dimentional problem, position and velocity are represented by matrices with dimension of 3xSwarm size. The swarm size is the number of particle where 100 are considered a lot enough.

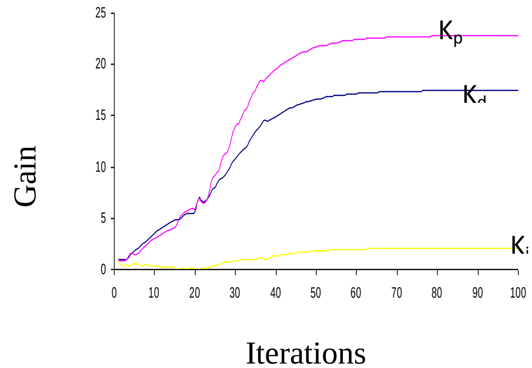
A good set of PID controller parameters can yield a good system response and result in minimization of performance index in Equation (3).

4. Simulation Results

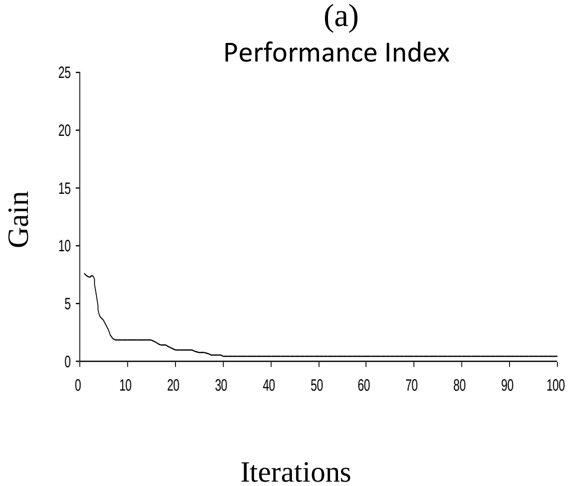
In the conventionally Z-N tuned PID controller, the plant response produces high overshoot and long settling time, but a better performance obtained with the implementation of PSO-based PID controller tuning. These are shown in Table 2. Furthermore, Figure 2 shows the curve of the PID parameters during optimization to see the convergence of the performance index optimized solution. The PID parameters are obtained for 100 iterations.

Table 2: Optimized PID Parameters

Tuning Method	K_p	K_i	K_d
Z-N PID	30.318	39.42	12.812
PSO-PID (ITAE)	22.8070	2.0734	17.4628



(a)



(b)

Figure 2: (a) PSO PID parameters and (b) performance index trajectory

Comparative results for the PID controllers are given in Table 3 where the step response performance is evaluated based on the rise time, settling time and overshoot. The corresponding plot for the step responses are shown in Figure 3. Finally, this result is the outcome of the preliminary investigation. To further investigate the effectiveness of the proposed method, some work may be done such as:

- Comparison of the PSO-PID with other artificial intelligence (AI) optimization techniques, like Genetic Algorithm (GA).
- Instead of PSO algorithm, others optimizer such as Differential Optimization can be used.
- Different objective functions other than ITAE performance index that is already used.

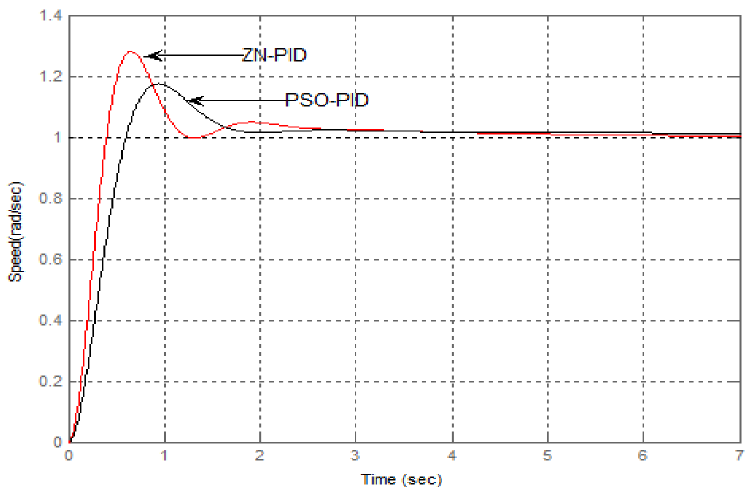


Figure 3: Comparison of the step response for PID controllers

Table 3: Comparison of ZN-PID and PSO-PID for Brushless DC Motor

Method	Rise time (s)	Settling time (s)	Overshoot (%)	P	I	D
Z-N	0.307	3.44	28.1	30.3	39.4	12.8
PSO	0.418	3.17	17.4	22.8	2.1	17.5

5. Conclusions

From the results, the designed PID controller using PSO algorithm shows superior performance over the traditional method of Ziegler-Nichols, in terms of the system overshoot, settling time and rise time. However, the traditional method provides us with the initial PID gain values for optimal tuning. Therefore the benefit of using a modern artificial intelligence optimization approach is observed as a complement solution to improve the performance of the PID controller designed by conventional method. Of course there are many techniques can be used as the optimization tools and PSO is one of the recent and efficient optimization tools.

References

Varsek, A., Urbacic, T. and Filipic, B. 1993. Genetic Algorithms in controller design and tuning. IEEE Transactions of Systems Management and Cyber, 23(5): 1330-1339.

Astrom, KJ. and Hagglund, T. 1995. PID Controllers: Theory, Design and Tuning, ISA, Research Triangle, Par, NC.

Clerc, M. 1999. The Swarm and the queen: towards a deterministic and adaptive particle swarm optimization. Proceedings of the Conference on Evolutionary Computation, 1951-1957.

Eberhart, RC. and Shi, YH. 2000. Comparing inertia weights and constriction factors in particle swarm optimization. Proceedings of The 2000 Congress on Evolutionary Computation, 1: 84-88.

Gaing, ZL. 2004. A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Transaction on Energy Conversion, 19(2): 384-391.

Ogata, K. 1987. Modern Control Systems, University of Minnesota, Prentice Hall, MN.

Ogata, K. 2005. Modern Engineering (fifth edition), University of Minnesota, Prentice Hall, MN.

Krishnakumar, K. and Goldberg, DE. 1992. Control system optimization using genetic algorithms. Journal of Guidance, Control and Dynamics, 15(3): 735-740.

Kennedy, J. and Eberhart, RC. 1995. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks (Perth,Australia), IEEE Service Center, Piscataway, NJ, 4: 1942- 1948.

Krohling RA. and Rey, JP. 2001. Design of optimal disturbance rejection PID controllers using genetic algorithm. IEEE Transactions of Evolutionary Computation, 5:78–82.

Kwok, DP., Leung, TP. and Sheng, F. 1993. Genetic algorithm for optimal dynamic control of robot arms. Proceedings of the International Conference on Industrial Electronics,Control and Instrumentation.15-19 November, 1993, San Francisco, CA, pp:380-385.

Pillay, N. and Govender, P. 2007. A Particle Swarm Optimization Approach for Model Independent Tuning of PID Control Loop., IEEE Africon 2007, IEEE Catalog: 04CH37590C, ISBN: 0-7803-8606-X, 2007.

Mahony, TO., Downing, CJ. and Fatla, K. 2000. Genetic algorithm for PID parameter optimization: Minimizing error criteria. Process Control and Instrumentation, University of Stracthclyde, pg 148- 153.

Visioli, A. 2001. Tuning of PID controllers with fuzzy logic. Proceedings of the IEEE Conference on Control Theory Applications, 148(1):1–8.