FUDMA Journal of Sciences (FJS)
ISSN: 2616-1370
Vol. 2 No. 2, June, 2018, pp 73 - 79

PROPORTIONAL-INTEGRAL-DERIVATIVE (PID) CONTROLLER TUNING FOR AN INVERTED PENDULUM USING PARTICLE SWARM OPTIMISATION (PSO) ALGORITHM

Joseph, S. B. and Dada, E. G.
Department of Computer Engineering, University of Maiduguri
Maiduguri - Borno State, Nigeria
Correspondence: sjbassi74@unimaid.edu.ng

ABSTRACT

Linear control systems can be easily tuned using conventional tuning techniques such as the Ziegler- Nichols and Cohen-Coon tuning formulae. Empirical studies have found that these conventional tuning methods result in an unsatisfactory control performance when they are used for industrial processes. It is for this reason that control practitioners often prefer to tune most nonlinear systems using trial and error tuning, or intuitive tuning. A need therefore exists for the development of a suitable automatic tuning technique that is applicable for a wide range of control processes that do not respond satisfactorily to conventional tuning. The balancing of an inverted pendulum by moving a cart along a horizontal track is a classic problem in the area of control. The encouraging results obtained from the simulation of the PID Controller parameters-tuning using the PSO when compared with the performance of PID and Ziegler-Nichols (Z-N) makes PSO-PID a good addition to solving PID Controller tuning problems using metaheuristic techniques as will reduce the time and cost of tuning these parameters and improve the overall system performance.

Keywords: *Inverted Pendulum, PID Controller, Particle Swarm Optimization Algorithm, Ziegler- Nichols method, tuning*

INTRODUCTION

The inverted pendulum problem is a classic control systems problem (Lam, 2004 and Ooi, 2003). Maintaining an equilibrium position of the pendulum pointing up is a challenge as this equilibrium position is unstable. As the inverted pendulum system is nonlinear it is well-suited to be controlled by artificial intelligence technique (AI) of Particle Swarm optimization Algorithm technique (Anbumani, Malini and Pechinathan, 2017). The inverted pendulum system is a standard problem in the area of control systems which is often used as a bench mark for control systems. They are often useful to demonstrate concepts in linear control such as the stabilization of unstable systems. Since the system is inherently nonlinear, it has also been useful in illustrating some of the ideas in nonlinear control.

During the past decades, process control techniques in the industry have made great advances. Numerous control methods such as: adaptive control, predictive control, neural control, and fuzzy control have been studied. In despite of many efforts, the proportional–integral-derivative (PID) controller continues to be the main component in industrial control systems, included in the following forms: embedded controllers, programmable logic controllers, and distributed control systems. The reason is that it has a simple structure which is easy to be understood by the engineers and it presents robust performance within a wide range of operating conditions (Coelho, *et al.*, 2007). PID controller is a generic control loop feedback mechanism (controller) widely used in industrial control system - a PID is the most commonly used feedback controller (Manoj and Patra, 2014) and (Biplab *et al.*, 2014).

Overschee and Moor (2000), report that 80% of PID type controllers in the industry are poorly/less optimally tuned. They state that 30% of the PID loops operate in the manual mode and 25% of PID loops actually operate under default factory settings. Over the years, many techniques have been suggested for tuning of the PID parameters. In this context there are classical (Ziegler/Nichols, gain-phase margin method, Cohen/Coon and pole place-ment) (Ziegler and Nichols, 1942; Ho WK, *et al.*, 1995; Cohen and Coon, 1953; Cominos and Munro, 2002) and advanced techniques such as minimum variance, gain scheduling and predictive (Astrom and Hagglund, 2001) and (Abbasi and Naghavi, 2017). These tuning techniques are characterized with the following shortcomings: (i) excessive number of rules to set the gains, (ii) inadequate dynamics of closed loop responses, (iii) difficulty to deal with nonlinear processes, and (iv) mathematical complexity of the control design (Wojsznis and Blevins, 2002). However, since it is fairly difficult to determine the PID parameters suitably, lots of researches have been reported with respect to PID parameter tuning schemes. Recently, as an alternative to the classical mathematical approaches, modern heuristic optimization techniques such as simulated annealing (Ho, *et al.*, 2006), evolutionary algorithms (Vlachos, *et al.*, 2002), artificial neural networks (Chen and Mills, 1997), and fuzzy systems (Li, *et al.*, 2005) have been given much attention by many researchers due to their ability to find an almost global optimal solution in PID tuning. This paper presents an automatic tuning method for a PID controller using artificial intelligence (AI) technique of particle swarm optimization algorithm (PSO) for an inverted pendulum.

SYSTEM MODELING

The inverted pendulum system is a classic control problem that is used in universities around the world. It is a suitable process to test prototype controllers due to its high non-linearities and lack of stability. The system consists of an inverted pole with mass, m, hinged by an angle ϴ from vertical axis on a cart with mass, M, which is free to move in the x direction. The goal of the study is to stabilize the pendulum (bar) on the top vertical position. This is possible by exerting on the carriage through the motor a force which tends to contrast the 'free' pendulum dynamics. The correct force has to be simulated measuring the values of the horizontal position and the pendulum angle. A schematic of the inverted pendulum is shown in figure 1.

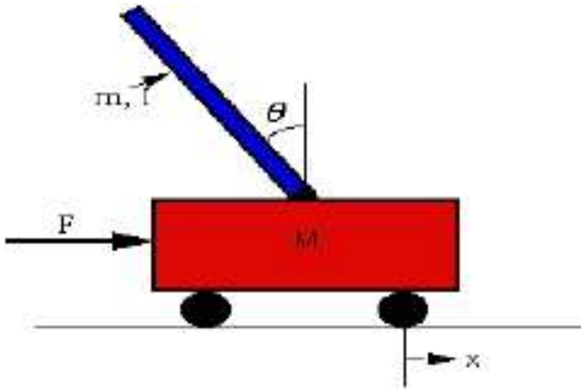

Fig. 1: A schematic of an inverted pendulum

By applying the law of dynamics on the inverted pendulum system the state space equations are obtained as equation (1).

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\varphi}(t) \\ \ddot{\varphi}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(I+ml^2)b}{I(M+m)+Mml^2} & \frac{m^2gl^2}{I(M+m)+Mml^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-mlb}{I(M+m)+Mml^2} & \frac{mgl(M+m)}{I(M+m)+Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \varphi(t) \\ \dot{\varphi}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{I+ml^2}{I(M+m)+Mml^2} \\ 0 \\ \frac{ml}{I(M+m)+Mml^2} \end{bmatrix} u(t) \quad (1)$$

$$y(t) = \begin{bmatrix} x \\ \varphi \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \phi(t) \\ \dot{\phi}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t)$$

Where x is the cart position, $\dot{x}$ is the cart position, φ is the pendulum angle and $\dot{\varphi}$ is the angular velocity of the pole.

METHODOLOGY

PID controller consists of Proportional, Integral and Derivative gains. The feedback control system is illustrated in Figure 2 where r, e, u, y are respectively the reference, error, controller output and controlled variables.

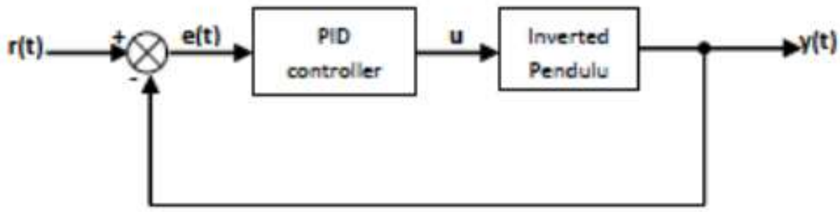

Fig. 2: A common feedback control system

The PID controller is described in equation (2) as:

$$u(t) = K_P e(t) + K_I \int e(t)dt + K_D \frac{d}{dt} e(t) \qquad (2)$$

The desired closed loop dynamics is obtained by adjusting the three parameters K_P, K_I and K_D. Where u_t is the controller output, et is the error, and t is the sampling instance (Muliadi and Kusumoputro, 2018). 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 inverted pendulum model is described in equation (3) as:

$$\frac{\varphi(s)}{U(s)} = \frac{4.5455s}{s^3 + 0.1818s^2 - 31.1818s - 4.4545} \qquad (3)$$

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 (4) 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.

$$J_{ITAE} = \int_0^\infty t|e(t)|dt \qquad (4)$$

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	$1.2\frac{T_p}{L_p K_p}$	$2 L_p$	$0.5 L_p$

Source: Ziegler and Nichols, 1942

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 (Ou, & Lin,2006). 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 (Shi and Eberhart, 1998). 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 is reported by Clerc (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 $X_i = (X_{i1}, X_{i2}, \ldots X_{iD})$. The best previous position (giving the minimum fitness value) of any particle is recorded and represented as $P_i = (P_{i1}, P_{i2}, \ldots P_{iD})$, 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 $V_i = (V_{i1}, V_{i2}, \ldots V_{iD})$. The particles are updated according to equations (4) and (5).

$$v_{i,m}^{(t+1)} = w \cdot v_{i,m}^{(t)} + c_1 * rand() * (pbest_{i,m} - x_{i,m}^{(t)} + c_2 * rand() * (gbest_m - x_{i,m}^{(t)}) \quad (4)$$

$$x_{i,m}^{(t+1)} = x_{i,m}^{(t)} + v_{i,m}^{(t+1)} \quad (5)$$

where, c_1 and c_2 are two positive constant. While *rand* () is random function between 0 and 1, and *n* represents iteration. Equation (5) 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 (6). 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 (Shi & Eberhart, 1998). 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 (Shi and Eberhart, 2001)

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 (K_p, K_i, K_d) using the model in Equation (3). 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 (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, Fig. 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

METHOD	Rise time (s)	Settling time (s)	P	I	D
Z-N	0.307	3.44	100	1	20
PSO	0.418	3.17	110	0	22

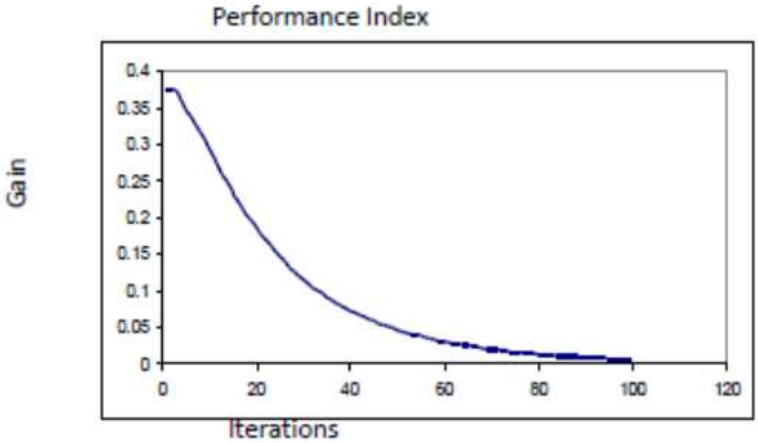

Fig. 3: The parameters and the performance index trajectory (PSO-PID)

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 Fig. 3. Finally, this result is only preliminary research. 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 Moth Flame Optimisation (MFO) Algorithm and 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.

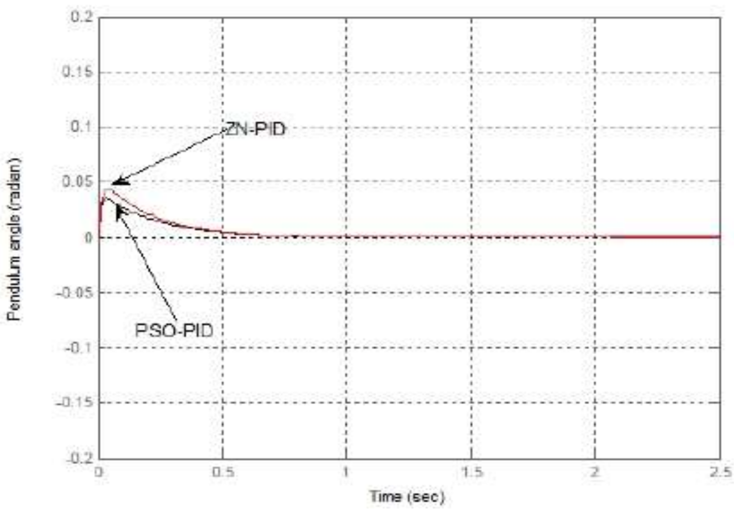

Fig. 4: Comparison of the step response for PID controllers

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

CONCLUSION

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

Abbasi E, Naghavi N. (2017). Offline Auto-Tuning of a PID Controller Using Extended Classifier System (XCS) Algorithm. Journal of Advances in Computer Engineering and Technology, 3(1), pp. 39-44.

Anbumani, K., Malini, R., Pechinathan G. (2017). GWO based tuning of PID controller for a Heat Exchanger process. IEEE 3rd International Conference on Sensing, Signal Processing and Security (ICSSS), pp. 417-421.

Astrom KJ, Hagglund T. (2001). The future of PID control. Contr Eng Pract; 9(11):1163–75.

Biplab S, Chiranjib K., Subhashis D. (2014). Robust PID controller design using particle swarm optimization-enabled automated quantitative feedback theory approach for a first-order lag system with minimal dead time, Systems Science & Control Engineering: An Open Access Journal, 2(1): 502-511, DOI: 10.1080/21642583.2014.912570.

Chen PC, Mills JK. (1997). Synthesis of neural networks and PID control for performance improvement of industrial robots. J Intell Robot Syst 20(2–4):157–80.

Cohen GH, Coon GA. (1953) Theoretical consideration of retarded control. Trans ASME 75:827–34.

Cominos P, Munro N. (2002). PID controllers: recent tuning methods and design to specification. IEE Proc-Contr Theory Application 149(1):46–53.

Dos Santos Coelho L, (2007). Tuning of PID controller for an automatic regulator voltage ..., Chaos, Solitons & Fractals, doi:10.1016/j.chaos.2007.06.018

Ho SJ, Shu LS, Ho SY. (2006). Optimizing fuzzy neural networks for tuning PID controllers using an orthogonal simulated annealing algorithm OSA. IEEE Trans Fuzzy Syst; 14(3):421–34.

Ho WK, Hang CC, CaoLS. (1995). Tuning of PID controllers based on gain and phase margin specifications. Automatica 31(3):497–502.

Lam, J. (2004). "Control of an Inverted Pendulum", University of California, Santa Barbara, 10 June 2004, http://www.ece.ucsb.edu/~roy/student_projects/Johnny_Lam_report_238.pdf.

Li HX, Zhang L, Chen G. (2005). An improved robust fuzzy-PID controller with optimal fuzzy reasoning. IEEE Trans Syst, Man, Cybernet – Part B: Cybernet; 35(6):1283–94.

Manoj K and Ashis P. (2014). Tuning PID Controller for Speed Control of DC Motor Using Soft Computing Techniques-A Review. Advance in Electronic and Electric Engineering, Volume 4, Number 2, pp. 141-148.

Muliadi J and Kusumoputro B. (2018). Neural Network Control System of UAV Altitude Dynamics and Its Comparison with the PID Control System. Hindawi Journal of Advanced Transportation Volume 2018, Article ID 3823201, 18 pages. https://doi.org/10.1155/2018/3823201.

Ooi, R. (2003). "Balancing a Two-Wheeled Autonomous Robot", University of Western Australia, 3 Nov. 2003, http://www.cs.cmu.edu/~mmcnaugh/kdc/as7/2003-Balance-Ooi.pdf.

Van Overschee, De Moor B. (2000). RAPID: the end of heuristic PID tuning. In: Proceedings of the IFAC workshop on digital control:past, present and future of PID control, Terrassa, Spain; pp. 595–600.

Vlachos C, Williams D, Gomm JB. (2002). Solution to the Shell standard control problem using genetically tuned PID controllers. Contr. Eng Pract 10(2):151–63.

Wojsznis WK, Blevins TL. (2002). Evaluating PID adaptive techniques for industrial implementation. In: Proceedings of American control conference, Anchorage, AK, USA, p. 1151–5.

Ziegler JG, Nichols NB. (1942). Optimum settings for automatic controllers. Trans ASME; 64:759–68.