

Tuning of PID Controller Using Particle Swarm Optimization (PSO)

Mahmud Iwan Solihin, Lee Fook Tack and Moey Leap Kean

*School of Engineering, UCSI University
No.1, Jalan Menara Gading, UCSI Heights, 56000,
Cheras, Kuala Lumpur, Malaysia*

Abstract— The aim of this research is to design a PID Controller using PSO algorithm. The model of a DC motor is used as a plant in this paper. The conventional gain tuning of PID controller (such as Ziegler-Nichols (ZN) method) usually produces a big overshoot, and therefore modern heuristics approach such as genetic algorithm (GA) and particle swarm optimization (PSO) are employed to enhance the capability of traditional techniques. However, due to the computational efficiency, only PSO will be used in this paper. The comparison between PSO-based PID (PSO-PID) performance and the ZN-PID is presented. The results show the advantage of the PID tuning using PSO-based optimization approach.

Keywords— PID tuning, PSO algorithm, optimization, Ziegler-Nichols Method

I. INTRODUCTION

Proportional-Integral Derivative (PID) controllers have been widely used for speed and position control of various applications. Among the conventional PID tuning methods, the Ziegler-Nichols method [1] 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) [2-5] and the particle swarm optimization (PSO) [6-7]. 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 [6-7].

This paper attempts to develop a PID tuning method using PSO algorithm. The result is expected to show the effectiveness of the modern optimization such as PSO in control engineering applications especially for university student's level.

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.

II. PROBLEM FORMULATION

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

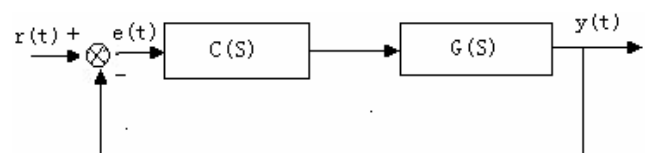


Fig.1 A common feedback control system

In the diagram of Fig.1, $G(s)$ is the plant transfer function and $C(s)$ is the PID controller transfer function that is given as:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (1)$$

Where K_p, K_i, K_d are respectively the proportional, integral, derivative gains/parameters of the PID controllers that are going to be tuned. The plant used here is a DC motor model which is a third order system written as:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s^3 + 9s^2 + 23s + 15} \quad (2)$$

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. They are defined as follows:

$$ISE = \int_0^{\infty} e^2(t) dt \quad (3)$$

$$IAE = \int_0^{\infty} |e(t)| dt \quad (4)$$

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

$$ITSE = \int_0^{\infty} te^2(t) dt \quad (6)$$

Therefore, for the PSO-based PID tuning, these performance indexes (Eqs. 3-6) 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.

III. TUNING OF PID USING Z-N 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 (see Table 1).

Table 1: Ziegler-Nichols open-loop tuning parameter

Controller	K_p	$T_i = K_p/K_i$	$T_d = K_d/K_p$
P	T/L	-	0
PI	$0.9(T/L)$	$L/0.3$	0
PID	$1.2(T/L)$	$2L$	$0.5L$

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 [8]. This can be achieved by adjusting the controller gain (K_u) till the system undergoes sustained oscillations (at the ultimate gain or critical gain), whilst maintaining the integral

time constant (T_i) at infinity and the derivative time constant (T_d) at zero (see table 2).

Table 2: Ziegler-Nichols closed-loop tuning parameter

Controller	K_p	T_i	T_d
P	$0.5K_u$	-	0
PI	$0.45K_u$	$1.2K_p/P_u$	0
PID	$0.6K_u$	$2K_p/P_u$	$K_p P_u/8$

IV. TUNING OF PID USING PSO-BASED OPTIMIZATION

A. Overview of PSO Algorithm

PSO is optimization algorithm based on evolutionary computation technique. The basic PSO is developed from research on swarm such as fish schooling and bird flocking [9]. After it was firstly introduced in 1995 [10], a modified PSO was then introduced in 1998 to improve the performance of the original PSO. A new parameter called inertia weight is added [11]. This is a commonly used PSO where inertia weight is linearly decreasing during iteration in addition to another common type of PSO which is reported by Clerc [12-13]. 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 $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$. The best previous position (giving the minimum fitness value) of any particle is recorded and represented as $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, 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 $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. The particles are updated according to the following equations:

$$v_{id}^{n+1} = w \cdot v_{id}^n + c_1 \cdot \text{rand}() \cdot (p_{id}^n - x_{id}^n) + c_2 \cdot \text{rand}() \cdot (p_{gd}^n - x_{id}^n) \quad (7)$$

$$x_{id}^{n+1} = x_{id}^n + v_{id}^{n+1} \quad (8)$$

where c_1 and c_2 are two positive constant. As recommended in Clerc's PSO, the constants are $c_1 = c_2 = 1.494$. While $\text{rand}()$ is random function between 0 and 1, and n represents iteration. Eq.7 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 Eq.8. 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 [11]. It can be a positive constant or even positive linear or nonlinear function of time. A guaranteed convergence of PSO proposed by Clerc set $w = 0.729$. It has

been also shown that PSO with different number of particles (swarm size) has reasonably similar performance [14]. Swarm size of 10-50 is usually selected. Here, we set 40.

B. 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 is employed to tune PID gains/parameters (K_p , K_i , K_d) in offline using the model in Eq.2. PSO 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-dimensional problem, position and velocity are represented by matrices with dimension of $3 \times \text{Swarm size}$. The swarm size is the number of particle where 40 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 Eqs.3-6 above.

V. SIMULATION RESULTS

In the conventionally Z-N tuned PID controller, the plant response produces high overshoot, but a better performance obtained with the implementation of PSO-based PID controller tuning. In the PSO-based PID controllers (PSO-PID), different performance index gives different results. These are shown in Table 3.

Furthermore, Fig. 2 shows a sample of the trajectory of the PID parameters during optimization to see the convergence of the optimized solution. The PID parameters are obtained for 50 iterations.

Table 3: Optimized PID parameters

Tuning Method	K_p	K_i	K_d
Z-N PID	115.2	175.9	18.9
PSO-PID1 (ISE)	95.6	14.6	10.9
PSO-PID2 (IAE)	72.9	12.0	16.5
PSO-PID3 (ITSE)	93.0	23.5	13.8
PSO-PID4 (ITAE)	35.2	14.8	12.2

Table 4: Step response performance for PID controllers

Tuning Method	Over-shoot (%)	Settling Time (s)	SSE
Z-N PID	67	4.2	72.3
PSO-PID1 (ISE)	3.8	2.6	48.0
PSO-PID2 (IAE)	0	4.9	77.3
PSO-PID3 (ITSE)	10.8	8.6	78.9
PSO-PID4 (ITAE)	1.4	2.3	77.4

Comparative results for the PID controllers are given below in Table 4 where the step response performance is

evaluated based on the overshoot, settling time and SSE (sum of squared error). 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 tuning method other than Z-N method.
- Instead of PSO, others optimizer such as Differential Optimization can be used.
- Different objective functions other than error performances that are already used.

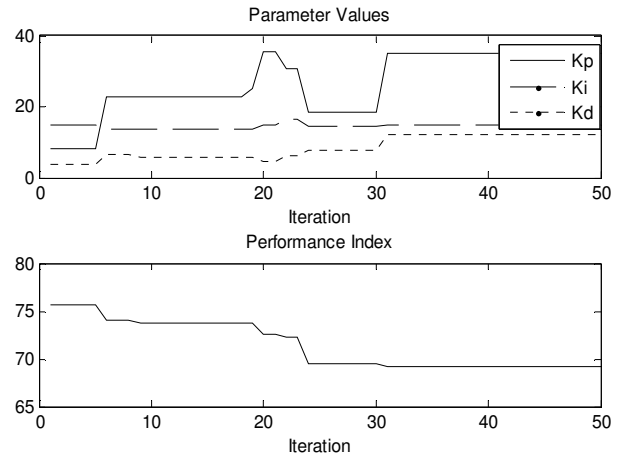


Fig.2 the parameters and the performance index trajectory (PSO-PID4)

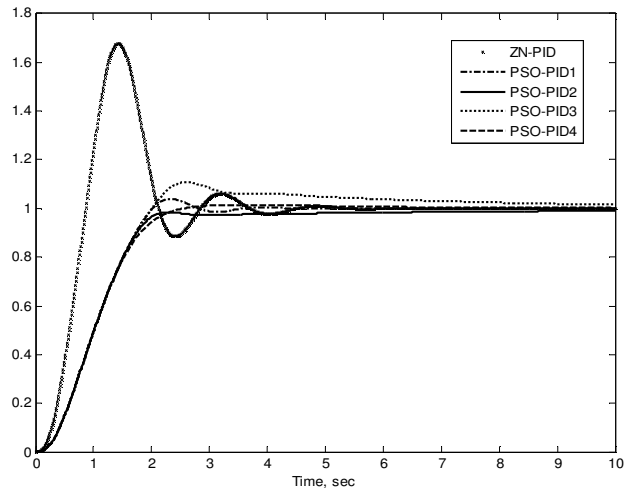


Fig.3 Comparison of the step response for PID controllers

VI. CONCLUSIONS

From the results, the designed PID controllers using PSO-based optimization have less overshoot compared to that of the classical method (Z-N). Furthermore, the PSO-based PID controllers which are optimized with different performance index have similar performances, except that is optimized by ITSE where long settling time is seen.

However, the classical method is good for giving us as the starting point of what are the PID values. Therefore the benefit of using a modern 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

- [1] K Ogata, Modern Control Systems, University of Minnesota, Prentice Hall, 1987
- [2] T. O. Mahony, C J Downing and K Fatla. (2000). "Genetic Algorithm for PID Parameter Optimization: Minimizing Error Criteria", Process Control and Instrumentation, University of Strathclyde, pg 148- 153.
- [3] Q.Wang, P Spronck and R Tracht. (2003). An Overview of Genetic Algorithms Applied to Control Engineering Problems. Proceedings of the Second International Conference on Machine Learning and Cybernetics.
- [4] K. Krishnakumar and D. E. Goldberg, Control System Optimization Using Genetic Algorithms, Journal of Guidance, Control and Dynamics, Vol. 15, No. 3, pp. 735-740, 1992.
- [5] A.Varsek, T. Urbacic and B. Filipic, Genetic Algorithms in Controller Design and Tuning, IEEE Trans. Sys. Man and Cyber, Vol. 23/5, pp1330-1339, 1993.
- [6] Gaing, Z.L. (2004). A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Transaction on Energy Conversion, Vol.19(2), pp.384-391.
- [7] Zhao, J., Li, T. and Qian, J. (2005). Application of particle swarm optimization algorithm on robust PID controller tuning. Advances in Natural Computation: Book Chapter. Springer Berlin / Heidelberg, pp. 948-957.
- [8] Astrom, K. J. and T., Hagglund, PID Controllers: Theory, Design and Tuning, ISA, Research Triangle, Par, NC, (1995)
- [9] Ou, C. and Lin, W. (2006). Comparison between PSO and GA for parameters optimization of PID controller. Proc. IEEE International Conference on Mechatronics and Automation. Luoyang, China.
- [10] Kennedy, J. and Eberhart, R.C. (1995). Particle swarm optimization. Proc. IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, pp. IV: 1942- 1948.
- [11] Shi, Y.H. and Eberhart, R.C. (1998). A modified particle swarm optimizer. IEEE International Conference on Evolutionary Computation, Anchorage, Alaska.
- [12] Eberhart, R.C. and Shi, Y.H. (2000).Comparing inertia weights and constriction factors in particle swarm optimization. Proc. The 2000 Congress on Evolutionary Computation. Vol.1, pp. 84-88.
- [13] Clerc, M. (1999). The Swarm and the queen: towards a deterministic and adaptive particle swarm optimization. Proceedings of the Conference on Evolutionary Computation, pp. 1951-1957.
- [14] Shi, Y.H. and Eberhart, R.C. (2001). Fuzzy Adaptive Particle Swarm Optimization. Proc. Congress on Evolutionary Computation. Seoul, Korea.