IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 5, July 2014.

www.ijiset.com

ISSN 2348 – 7968

Intelligent PID Controller Tuning Using PSO for Linear System

Mrs. A. Asuntha, M.E [1], Dr Andy Srinivasan [2]

[1] Assistant Professor, Department of Electronics and Instrumentation Engineering, SRM University, Tamil Nadu, India

[2] Professor & Head , Department of Electronics and Instrumentation Engineering, Valliammai Engineering College, Tamil Nadu, India

Abstract

In chemical industries in mid 90's, process engineers were using the classical controller tuning methods to find the controller parameter values like $K_{p,}$ K_i ,and $K_{d.}$ The classical controller tuning methods may require complex computations to identify the controller parameters. With the evolution of soft computing methods these classical control methods were eliminated. This paper focus on design method for determining the optimal proportional-integral-derivative (PID) controller parameters of linear system using the particle swarm optimization (PSO) algorithm. The proposed approach had superior features, including easy implementation, stable convergence characteristic, and good computational efficiency. At first genetic algorithm (GA) of automatic voltage regulator(AVR) setting is designed and compared with PSO based PID controller settings. The proposed method has more robust stability and efficiency, and can solve the searching and tuning problems of PID controller parameters more easily and quickly than the GA method.

Keywords: GA, PSO, PID controller, AVR, MATLAB.

1.Introduction:

During the past decades, the process control techniques in the industry have made great advances. Numerous control methods such as adaptive control, neural control, and fuzzy control have been studied. Among them, the best known is the proportional-integral-derivative (PID) controller, which has been widely used in the industry because of its simple structure and robust performance in a wide range of operating conditions. Unfortunately, it has been quite difficult to tune properly the gains of PID controllers because many industrial plants are often burdened with problems such as high order, time delays, and non- linearity's. Over the years, several heuristic methods have been proposed for the tuning of PID controllers. The first method used the classical tuning rules proposed by Ziegler and Nichols, in general, it is often hard to determine optimal or near optimal PID parameters with the

Ziegler-Nichols formula in many industrial plants.For these reasons, it is highly desirable to Increase the capabilities of PID controllers by adding new features. Many Artificial Intelligence (AI) techniques have been employed to improve the controller performances for a wide range of plants while retaining their basic characteristics. AI techniques such as Neural Network, Fuzzy System, and Neural-Fuzzy Logic have been widely applied to proper tuning of 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)).

In this paper, a simple performance criterion in time domain is proposed for evaluating the performance of a PSO-PID controller that was applied to the complex control system. The generator excitation system maintains generator voltage and controls the reactive power flow using an automatic voltage regulator (AVR). The role of an AVR is to hold the terminal voltage magnitude of a synchronous generator at a specified level. Hence, the stability of the AVR system would seriously affect the security of the power system. In this paper, a practical high-order AVR system with a PID controller is adopted to test the performance of the proposed PSO-PID controller. In this paper, besides demonstrating how to employ the PSO method to obtain the optimal PID controller parameters of an AVR system, many performance estimation schemes are performed to examine whether the proposed method has better performance than the real-value GA method in solving the optimal.

2.Linearised Model Of An AVR System:

Main aim of AVR system is to control and maintain the voltage between limits by adjusting the excitation of the machines. The entire excitation system maintains voltage and controls the reactive power flow using an automatic voltage regulator (AVR). The role of an AVR is to hold the terminal voltage magnitude of a generator at a specified level. Hence, the stability of the AVR system would seriously affect the security of the power system. A practical high-order AVR system with a PID controller is been used to measure the performance.

3. Block Diagram of an AVR System:

Since the components of a power system are non linear, a linearized model around an operating point is used in the design process of AVR controllers. However, controllers based on linearized models are not capable of supporting parameter variations for stability. When the process becomes too complex to be described by analytical models, it is unlikely to be efficiently controlled by conventional fixed gain controllers. Conventional proportional–integral–derivative (PID) controllers have been well developed and are extensively used for industrial automation and process control. The role of an AVR is to hold the terminal voltage magnitude of a generator at a specified level. A simple AVR system comprises four main components, namely amplifier, exciter, generator, and sensor.

For mathematical modeling and transfer function of the four components, these components must be linearized, which takes into account the major time constant and ignores the saturation or other nonlinearities. The reasonable transfer function of these components may be shown in Fig.5.1.

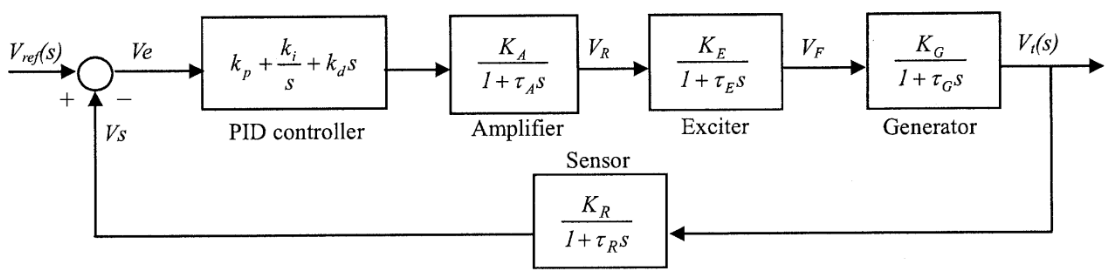

Fig 3.1. Block diagram of an AVR system

3.1. Controller settings:

The PID controller is simple and easy to implement. It is used to improve the dynamic response as well as to reduce or eliminate the steady-state error. The derivative controller adds a finite zero to the open-loop plant transfer function and improves the transient response. The integral controller adds a pole at the origin, thus increasing system type by one and reducing the steady-state error due to a step function to zero. The PID controller transfer function is

$$\frac{E(s)}{U(s)} = K_p + \frac{K_i}{S} + K_d S.$$

Where,

U(s) and E(s) - Is the control (controller output) and tracking error signals in S-domain, respectively

K_p - Proportional gain,

K_i - Integration gain,

K_d - Derivative gain,

3.2. Amplifier model:

The amplifier model is represented by a gain K_A and a time constant τ_A; the transfer function is

$$\frac{V_R}{V_E} = \frac{K_A}{1+\tau_A S}$$

Typical values of gain are in the range of 10 to 400. The amplifier time constant is very small ranging from 0.02 to 0.1 s.

3.3. Exciter model:

The transfer function of a modern exciter may be rep- resented by a gain K_E and a single time constant, τ_E;

$$\frac{V_F}{V_R} = \frac{K_E}{1+\tau_E S}$$

Typical values of gain are in the range of 10 to 400. The time constant is in the range of 0.5 to 1.0 s.

3.4. Generator model:

In the literalized model, the transfer function relating the generator terminal voltage to its field voltage can be represented by a gain K_G and a time constant τ_G ,

$$\frac{V_t}{V_F} = \frac{K_G}{1+\tau_G S}$$

These constants are load dependent, gain may vary between 0.7 to 1.0, and time constant between 1.0 and 2.0 s from full load to no load.

3.5. Sensor model:

The sensor is modeled by a simple first-order transfer function, given by,

$$\frac{V_S}{V_t} = \frac{K_R}{1+\tau_R S}$$

Where, time constant τ_R, is very small, ranging from of 0.001 to 0.06 s.

4. Key Features of AVR System include:

- Voltage Boost & Buck
- Common Mode noise filtering: <0.5 Volts
- Normal Mode noise filtering: <10 Volts
- Over Voltage Protection

5. PARTICLE SWARM ALGORITHM

Particle Swarm Optimization (PSO) algorithm is a population based evolutionary computation technique developed by the inspiration of the social behavior in bird flocking or fish schooling. It attempts to mimic the natural process of group communication of individual knowledge, to achieve some optimum property. In this method, a population of swarm is initialized with random positions S_i and velocities V_i. At the beginning, each particle of the population is scattered randomly throughout the entire search space and with the guidance of the performance criterion, the flying particles dynamically adjust their velocities according to their own flying experience and their companions flying experience. Each particle remembers its best position obtained so far, which is denoted pbest (P_i^t). It also receives the globally best position achieved by any particle in the population, which is denoted as gbest (G_i^t). The updated velocity of each particle can be calculated using the present velocity and the distances from pbest and gbest. The updated velocity and the position are given in Eqn.1 and 2 respectively.

$$V_i^{t+1} = W^t.V_i^t + C_1.R_1.(P_i^t - S_i^t) + C_2.R_2.(G_i^t - S_i^t) \quad (1)$$

$$S_i^{t+1} = S_i^t + V_i^{t+1} \quad (2)$$

$$W^t = (W_{max} - Iter) \, x \left[\frac{(W_{max} - W_{min})}{Iter_{max}} \right] \quad (3)$$

Where, C_1, C_2 are positive constants. C_1 is the cognitive learning rate and C_2 is the global learning rate. R_1, R_2 are random numbers in the range 0-1. The parameter W is inertia weight that increases the overall performance of PSO. The larger value of W (W_{max}) can favor the global wide-range search and lower value of W (W_{min}) implies a higher ability for local nearby search.

6. PSO based PID Controller:

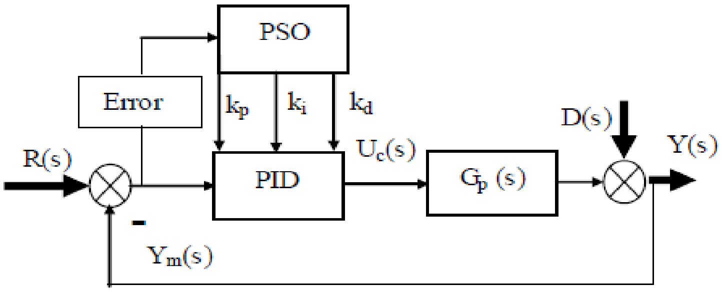

Fig. 6.1 PSO based PID controller

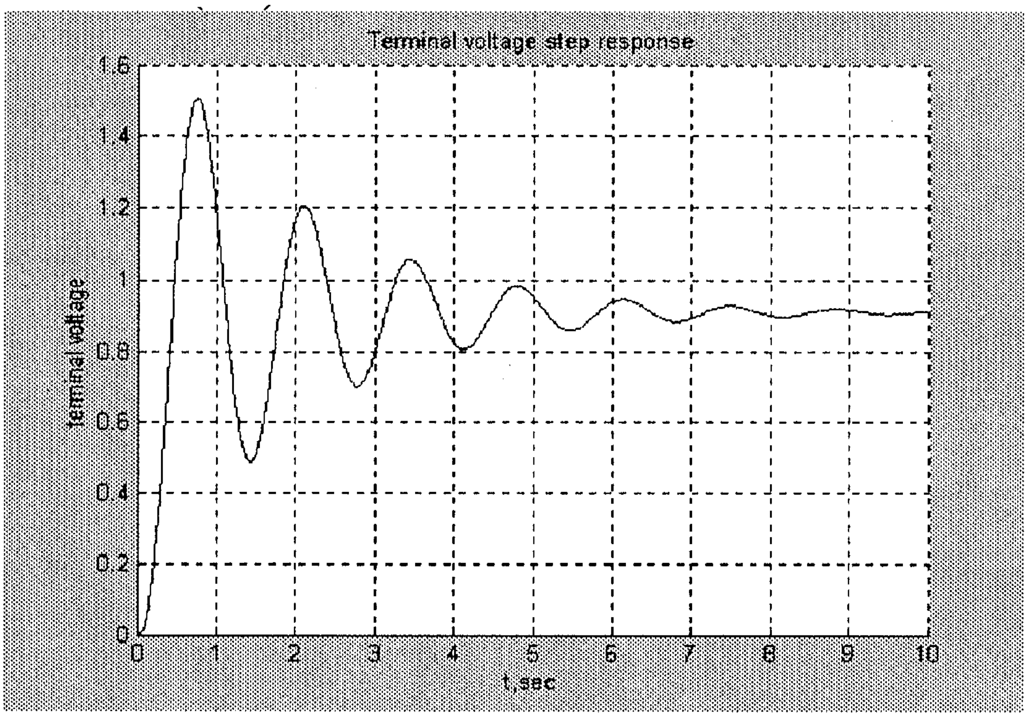

Fig. 6.2 . Step Response of AVR system using GA

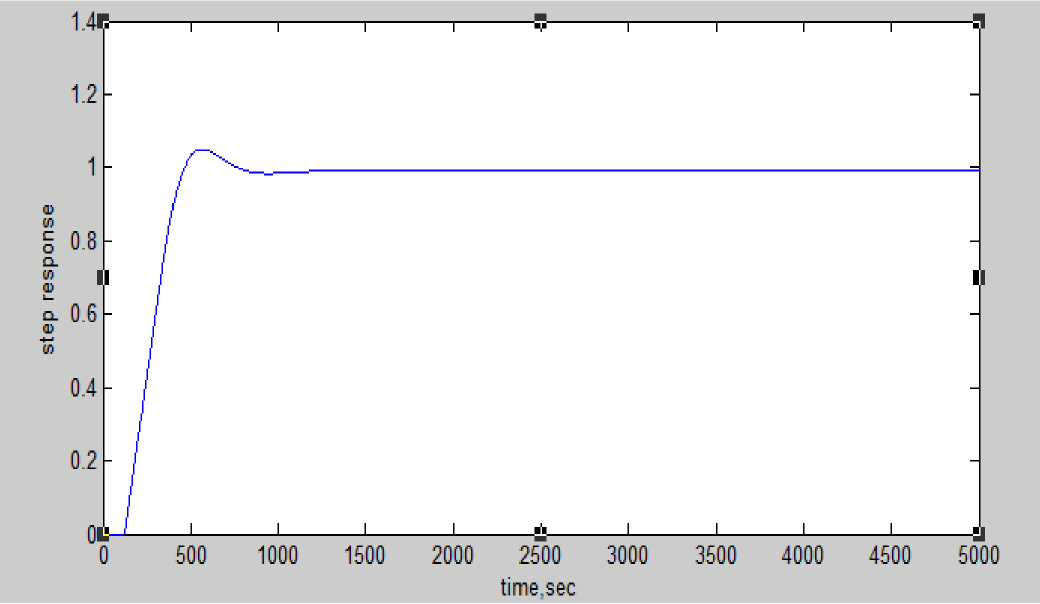

Fig. 6.3. Step Response of AVR system using PSO based PID controller

Conclusion:

In this work a novel design method for determining the PID controller parameters using the PSO method. Proposed method integrates the PSO algorithm with time-domain performance criterion into a PSO-PID controller. Through the simulation of a practical AVR system, the show that the proposed controller can perform an search for the optimal PID controller parameters. In addition, in order to verify it being superior to method, many performance estimation schemes are performed, such as

- Multiple simulation examples for their terminal voltage step responses;
- Convergence characteristic of the best evaluation value;
- Dynamic convergence behaviour of all individuals in population during the evolutionary processing;
- Computation efficiency.

Most of the industrial it is clear from the results that the proposed PSO method can void the shortcoming of premature convergence of GA method and can obtain higher quality solution with better computation efficiency. Therefore, the proposed method has more robust stability and efficiency, and can solve the searching and tuning problems of PID controller parameters more easily and quickly than the GA method.

ISSN 2348 – 7968

Different PSO optimization parameters are required for solving different problems in practical application, such as the number of agents (individuals), weight factors and, acceleration factors and, and the limit of change in velocity. Hence, how to select suit parameters for the target problem, such as the sensitivity analysis of optimization parameters for finding the best parameters, is one of our future process loops use conventional or modified structure PID controllers. Tuning the controller parameter is a challenging work if the system model is other than a first order plus dead time. In this work, design of optimization based model independent controller tuning for an AVR model have been attempted.

References

[1] A. Visioli, "Tuning of PID controllers with fuzzy logic," *Proc. Inst.Elect. Eng. Contr. Theory Applicat.*, vol. 148, no. 1, pp. 1–8, Jan. 2001.

[2] T. L. Seng, M. B. Khalid, and R. Yusof, "Tuning of a neuro-fuzzy controller by genetic algorithm," *IEEE Trans. Syst., Man, Cybern. B*, vol. 29, pp. 226–236, Apr. 1999.

[3] R. A. Krohling and J. P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithm," *IEEE Trans. Evol. Comput.*, vol. 5, pp. 78–82, Feb. 2001.

[4] Y. Mitsukura, T. Yamamoto, and M. Kaneda, "A design of self-tuning PID controllers using a genetic algorithm," in *Proc. Amer. Contr. Conf.*, San Diego, CA, June 1999, pp. 1361–1365.

[5] T. Kawabe and T. Tagami, "A real coded genetic algorithm for matrix inequality design approach of robust PID controller with two degrees of freedom," in *Proc. 12th IEEE Int. Symp. Intell. Contr.*, Istanbul, Turkey, July 1997, pp. 119–124.

[6] R. A. Krohling, H. Jaschek, and J. P. Rey, "Designing PI/PID controller for a motion control system based on genetic algorithm," in *Proc. 12th IEEE Int. Symp. Intell. Contr.*, Istanbul, Turkey, July 1997, pp. 125–130.

[7] D. P. Kwok and F. Sheng, "Genetic algorithm and simulated annealing for optimal robot arm PID control," in *Proc IEEE Conf. Evol. Comput.*, Orlando, FL, 1994, pp. 707–713.

[8] T. Ota and S. Omatu, "Tuning of the PID control gains by GA," in *Proc. IEEE Conf. Emerging Technol. Factory Automation*, Kauai, HI, Nov. 1996, pp. 272–274.

[9] A. H. Jones and P. B. D. Oliveira, "Genetic auto-tuning of PID controllers," in *Proc. Inst. Elect. Eng. Conf. Genetic Algorithm Eng. Syst.: Innovations Applicat.*, Sept. 1995, pp. 141–145.

[10] D. B. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, 2 ed. New York: IEEE Press, 2000.

[11] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proc. IEEE Int. Conf. Neural Networks*, vol. IV, Perth, Australia, 1995, pp. 1942–1948.

[12] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” in *Proc. IEEE Int. Conf. Evol. Comput.*, Anchorage, AK, May 1998, pp. 69–73.

[13] Y. Shi and R. C. Eberhart, “Empirical study of particle swarm optimization,” in *Proc. IEEE Int. Conf. Evol. Comput.*, Washington, DC, July 1999, pp. 1945–1950.

[14] R. C. Eberhart and Y. Shi, “Comparison between genetic algorithms and particle swarm optimization,” in *Proc. IEEE Int. Conf. Evol. Comput.*, Anchorage, AK, May 1998, pp. 611–616.

[15] P. J. Angeline, “Using selection to improve particle swarm optimization,” in *Proc. IEEE Int. Conf. Evol. Comput.*, Anchorage, AK, May 1998, pp. 84–89.