Genetic Algorithms Based PID controller Design

Charanjeet Singh
Department of Electronics &Communication Engineering
Giani Zail Singh PTU Campus Bathinda, India

***Abstract* - A Proportional-Integral-Derivative (PID) control is the most commonly used algorithm in industrial control systems. With its three term functionality providing regulation to both transient and steady state responses, PID control offers the simplest and most efficient solution to many industrial control problems. Tuning of PID gain parameters continuous to be important as these parameters has great influence on the stability and the performance of the control system. The objective of this paper is to tune and analyze the performance of PID controller using Genetic Algorithms (GA). The performance of Genetic Algorithm based PID controller is compared with that of traditional methods e.g. Zeigler Nicholas method. The result obtained reflect that use of GA based PID controller improves the performance of system.**

***Index Terms* - Tuning of PID controller, Zeigler-Nicholas Method, Genetic Algorithm.**

I. INTRODUCTION

PID controller is a control loop feedback mechanism widely used in process control systems. This is because of its simple design and very few parameters need to be tuned in it. The three term functionality of PID controller provides treatment to both transient and steady state responses. Since the invention of PID controller in 1910 largely owning to Elmer Sperry's ship auto pilot and the Ziegler–Nicholas' tuning methods in 1942, the popularity of PID controller has grown tremendously. The PID controller calculation involves three separate parameter proportional, integral and derivative gains. The proportional value determines the reaction of current errors, the integral value determines the reaction based on the sum of recent errors, and derivative value determines the reaction based on the rate at which the error has been changing and the weighted sum of these three terms is used to regulate the process via the final control element. PID controller is mostly tuned by Zeigler-Nicholas method. The Genetic Algorithms based tuning method for a PID controller considerably reduce the overshoot and rise time as compared to Ziegler Nicholas tuning method.

II. PID CONTROLLER ALGORITHM

PID controller is combination of proportional, integral and derivative control actions, also called three-mode controller.
Proportional (P) Control: In this control action the output of controller is proportional to the error. A proportional controller continuously changes the manipulated variable according to error. The main disadvantage of proportional only control action is that it cannot keep the process variable on set point. The difference between controlled variable and desired set point is called steady state error or offset.
Integral (I)/Reset Control Action: The steady state error of proportional controller is removed (reset) in an integral controller. In integral control action the controller output is changed at a rate proportional to error signal. The integral control action continuously looks at the total past history of error by continuously integrating the area under the error curve.
Derivative (D) Control Action: In this control action, the controller output is function of the rate at which the error is changing. An advantage of using derivative control action is that it responds to the rate of change of actuating error and can produce a significant correction before the magnitude of actuating error becomes too large. The proportional mode considers the present state of the error, and integral control looks at its past history, while the derivative mode anticipates its future state and acts on that prediction. Derivative control anticipates process error before they have evolved and takes corrective action in advance.
Proportional-Integral-Derivative (PID) Controller: A standard PID controller is represented as:

$$m = K_p\left(e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{de(t)}{dt}\right) \quad (1)$$

Transfer function of PID controller is given by

$$G(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) = K_p + K_i\frac{1}{s} + K_d s \quad (2)$$

Where,
K_p = Proportional gain,
K_i = Integral gain,
K_d = Derivative gain,
T_i = Integral time constant and

© 2015 IJEDR | Volume 3, Issue 3 | ISSN: 2321-9939

T_d = Derivative time constant

III. PID CONTROLLER TUNING

The process of computing and setting the optimal values of K_p, K_i and K_d to get desired response from a control system, called tuning. PID controllers are mostly tuned by Ziegler-Nicholas method. Many model based controller techniques such as internal model control are used in conjunction with PID controller to improve the dynamic response of the process. Apart from conventional tuning methods there are many soft computing based intelligent tuning rules like Particle Swarm Optimization (PSO), Genetic Algorithms (GA) etc. The soft computing techniques for a PID controller considerably reduce the overshoot and rise time as compare to any other tuning method.

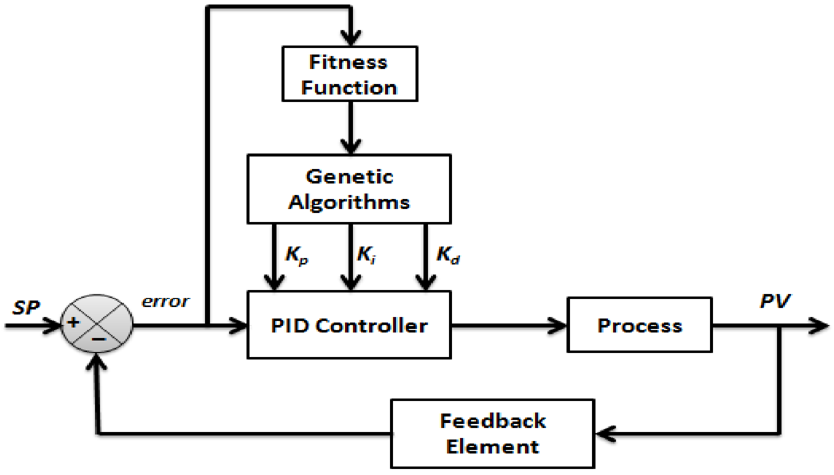

Fig. 1 Block Diagram of Genetic Algorithms Based PID Controller

IV. GENETIC ALGORITHMS (GA)

The Genetic Algorithm is method for solving optimization problem that is based on natural selection, the process that drives biological evolution. The GAs repeatedly modifies a population of individual solutions. At each step the GAs select individual at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generation, the population evolves toward an optimal solution. Three main steps of GAs are :
Selection selects the individuals, called parents that contribute to the next generation.
Crossover combines two parents to form children for the next generation.
Mutation apply random changes to individual children.

V. GENETIC ALGORITHMS BASED PID CONTROLLER

For the design of the GA-based PID controller, first an initial population of the GA is generated by random, this contains binary strings, where a string represents the proportional, integral and derivative gains (K_p, K_i, K_d). The objective function is required to evaluate the best PID controller for the system. An objective function could be created to find a PID controller that gives the smallest overshoot, fastest rise time or quickest settling time. However, in order to combine all of these objectives it is needed to design an objective function that will minimize the performance indices of the control system instead. Each chromosome in the population is passed into the objective function one at a time. The chromosome is then evaluated and assigned a number to represent its fitness. The genetic algorithm uses the chromosome's fitness value to create a new population consisting of the fittest members. Each chromosome consists of three separate strings constituting a P, I, and D term. When the chromosome enters the evaluation function, it is split up into its three terms. The newly formed PID controller is placed in a feedback loop with the system transfer function. This will result in a reduction of the compilation time of the program. The system transfer function is defined in another file and imported as a global variable. The controlled system is then given a step input and the error is assessed using an error performance criterion such as Integral Square Error (ISE). The chromosome is assigned an overall fitness value according to the magnitude of the error.

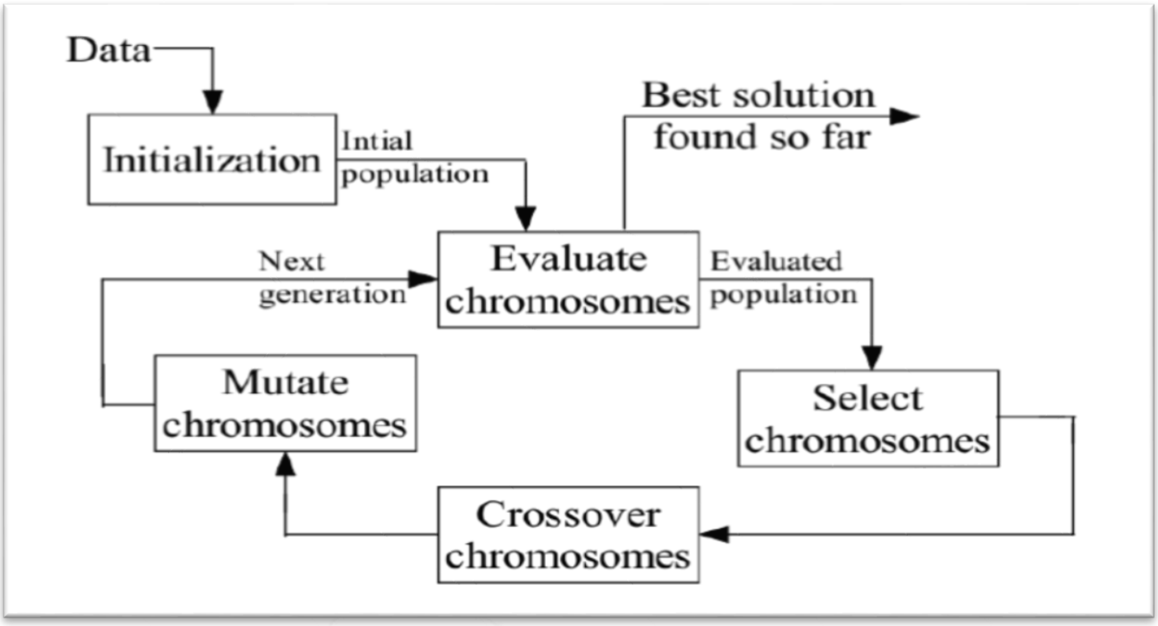

Fig. 2 Optimization Using Gas

VI. Results

Two different industrial process model have been taken to implement the GA based PID controller

Model 1: $$G_1(s) = \frac{0.002318}{s^2 + 0.201s + 0.00389}$$

Model 2: $$G_2(s) = \frac{6}{s^2 - 0.5s + 6}$$

For Model 1.

Tuning Method	K_p	K_i	K_d	Rise Time	Settling time	Overshoot %
Zeigler-Nicholas	28.2	4.16	47.9	5.003	62.8337	46.4433
Genetic Algorithms	79.9820	1.2042	83.4626	3.1069	18.8700	23.3011

For Model 2.

Tuning Method	K_p	K_i	K_d	Rise Time	Settling time	Overshoot %
Zeigler-Nicholas	5.3240	11.2521	8.3112	0.415	0.0665	1.1776
Genetic Algorithms	12.1858	15.9448	178182	0.199	0.0332	0.757

VII. Conclusion

PID controller based on Zeigler-Nicholas Method and Genetic Algorithm is simulated in MATLAB 2013a. Rise time, Settling time and Overshoot are calculated using 'stepinfo' command of MATLAB. Simulation results reflect that the Genetic Algorithm tuning method has a better control performance than Zeigler-Nicholas method. Genetic Algorithms based PID controller gives the smaller overshoot, faster rise time, quicker settling time.

VIII. References

[1] Kiam Heong Ang and Gregory Chong "PID control system analysis, design, and technology", *IEEE Transactions on Control Systems Technology*, Vol. 13, No. 4, 2005.

[2] M. K. Tan, Y. K. Chin, H. J. Tham and K. T. K. Teo, "Genetic Algorithm Based PID Optimization in Batch Process Control", *IEEE International Conference on Computer Applications and Industrial Electronics,* 2011.

[3] Mehmet Korkmaz, Omer Aydogdu and Huseyin Dogan, "Design and Performance Comparison of Variable Parameter Nonlinear PID Controller and Genetic Algorithm Based PID Controller", *IEEE Innovations in Intelligent Systems And Applications*, 2012

[4] Abhishek Singh and Dr. Veena Sharma, "Concentration Control of CSTR Through Fractional Order PID Controller by Using Soft Techniques", *IEEE Computing, Communication and Networking* 2013.

[5] A. Jayachitra and R. Vinodha, "Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor" *Hindawi Publishing Corporation, Advances in Artificial Intelligence* Vol. 2014, Article ID 791230.
[6] Jaison John, C. Sathish Kumar, "A Comparative Study of Genetic Algorithm and Particle Swarm Optimization based Optimization of PID Controller Parameters" *International Journal of Soft Computing and Engineering*, ISSN: 2231-2307, Vol. 4, Issue-5, November 2014.
[7] T. K. Sethuramalingam and B. Nagaraj, "PID Controller Tuning using Soft Computing Methodologies for Industrial Process- A Comparative Approach", *Indian Journal of Science and Technology*, Vol 7, pp. 140–145, November 2014