

# DESIGN AND OPTIMIZATION OF PID CONTROLLER USING GENETIC ALGORITHM

Gauri Mantri<sup>1</sup>, N. R. Kulkarni<sup>2</sup>

<sup>1</sup>Assistant Professor, E & TC Department, Symbiosis Institute of Technology, India, <sup>2</sup>HOD & Professor, Electrical Department, P.E. Society's Modern College of Engineering, India, [gaurim@sitpune.edu.in](mailto:gaurim@sitpune.edu.in), [nrkmcoe@gmail.com](mailto:nrkmcoe@gmail.com)

## Abstract

Natural evolution is mimicked by Genetic Algorithms (GAs) which is a stochastic global search method used for optimization. . In missile control systems Proportional Integral Derivative (PID) control is widely used, but due to empirically selected parameters  $K_p$ ,  $K_i$ ,  $K_d$  it is difficult to achieve parameter optimization. Genetic algorithm is a search algorithm that is based on natural selection and genetics principles. GA is a computational algorithm which deals with genetics of the human body. It evolves with the number of iterations. After every iteration a better result is expected. These results are checked for the error. The fittest roots or solution are considered for the next generation based on the selection criterion. GA randomly generates the initial population of the PID control parameters according to the calculation of selection (Normalized Geometric Selection), crossover (Arithmetic Crossover) and mutation (Uniform Mutation), thus optimizing the control parameters. Mean Square Error (MSE) value is chosen as the performance assessment index. For a missile altitude control Proportional Integral Derivative (PID) controller using genetic algorithm is implemented & compared with the classical method Zeigler-Nichols (Z-N) in the paper. Z-N method is classical method which tunes the parameters of PID. The parameters of PID are difficult to tune. Tuned parameters give the optimum solution. Optimum solution generally converges to a solution having minimum error. Minimum error gives a response of the system in terms of maximum overshoot, Settling time, Rise time & Steady State Error. The designed PID with the Genetic Algorithm has much faster response than the classical method.

**Index Terms:** Genetic Algorithm, Optimization, PID controller Zeigler-Nichols

\*\*\*

## 1. INTRODUCTION

In today's world Proportional Integral Derivative (PID) controller is widely used for an optimum solution. Optimum solution gives a better efficiency. For obtaining the better efficiency the actual output should match the set output. For this purpose there is a need of a controller. Difficulty of using a controller is tuning of the parameters of the controller. These days there are various methods which are used for the tuning the parameters of PID controller. The classical methods for tuning the parameters of the PID controller are Ziegler Nichols oscillation method, Ziegler Nichols reaction curve method, Cohen-Coon reaction curve method. But recent trends show that there has been a drastic improvement in the tuning by using the evolutionary algorithm. These are computational algorithms which gives better result after every iteration. Few computational algorithms frequently used these days are Ant Colony Optimization, Particle Swarm Optimization, and Genetic Algorithm etc. These are based on behavioural pattern of a living being [5, 6, 7].

An altitude of missile is to be controlled using the PID controller. The high-tech missiles these days require a robust, intelligent & precise system. Ziegler Nichols oscillation method is used & compared with genetically tuned

parameters. The objective of this paper is to show that a system can be optimized using genetic algorithm [1].

The following section II formulates the system model of a Missile control system. The focus of section III is on conventional PID Controller, it's tuning by Ziegler Nichols Method. A brief review of Genetic Algorithm Based PID Controller along with its implementation is brought up in section IV. In section V, simulation results of the corresponding system are obtained and compared.

## 2. SYSTEM MODEL

The Missile control system along with a PID controller is shown in the Fig 1. The PID controller used here is the continuous PID. The altitude of the missile is to be controlled so the actual position is manipulated in such a way that the missile is exactly on the required position. PID controller is used in series with the Steering Engine with a unity feedback system. The Unity feedback system which gives a feedback of the actual position sensed & can manipulate the position. The transfer function of the continuous PID controller is given by equation (1). The transfer function of the steering engine is given by equation (2).

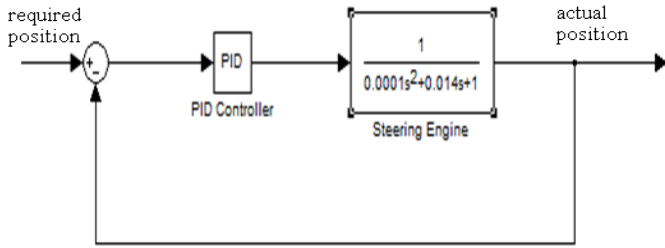


Fig -1: Block Diagram of the Entire System

The transfer function of the PID controller is given as follows:

$$G_c(s) = K (1 + 1/sT_i + sT_d) \tag{1}$$

K is constant of proportionality

T<sub>i</sub> is Integral time; it must be a real and positive value.

T<sub>d</sub> is Derivative time; it must be a real, finite, and nonnegative value.

The transfer function of the Steering engine is given as follows [1]:

$$\text{Transfer function} = \frac{1}{0.0001s^2 + 0.014s + 1} \tag{2}$$

### 3. TUNING OF PID CONTROLLER USING CONVENTIONAL APPROACH

#### 3.1 Conventional Approach - Ziegler Nichols Method

A Conventional approach is used to tune the parameters of PID. As shown in Fig 2, the parameters are obtained by using SISO tool in MATLAB. For a system shown in Fig 1, the PID is to be tuned .By giving a input to transfer function of the whole system the step response is obtained in SIMULINK.

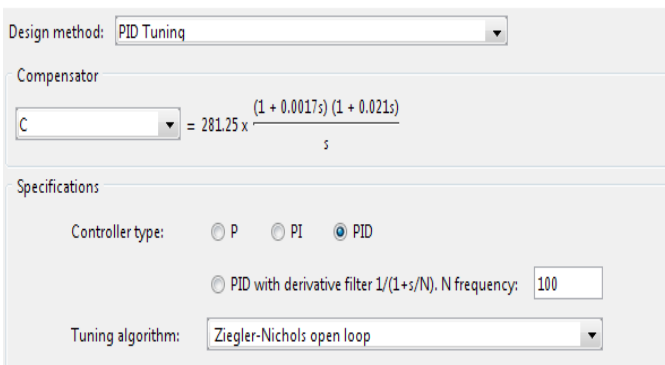


Fig -2: SISO Tool Used for Tuning of Parameters Using ZN method

Following are the values of the controller obtained from SISO tool.

K<sub>p</sub>=6.3843  
K<sub>i</sub>=281.25  
K<sub>d</sub>=0.01004

Transfer function of the compensator is given by

$$G_c(s) = \frac{0.01004s^2 + 6.3843s + 281.25}{s} \tag{3}$$

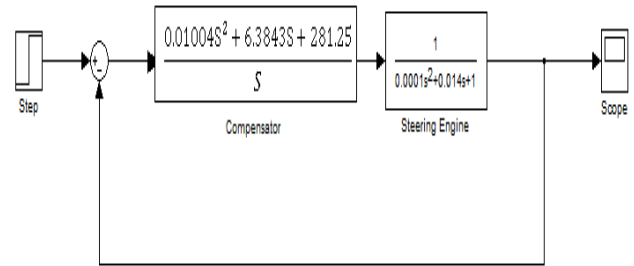


Fig-3: System with the compensator transfer function

Both the transfer functions which are in series are multiplied & further solved for unity feedback system. Fig 4 shows the transfer function of the system with a scope as an output.

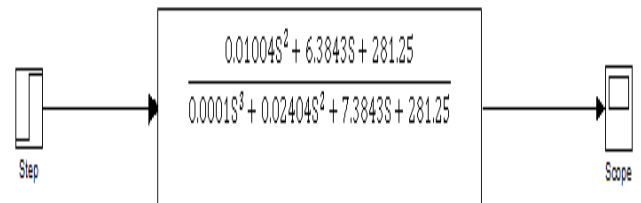
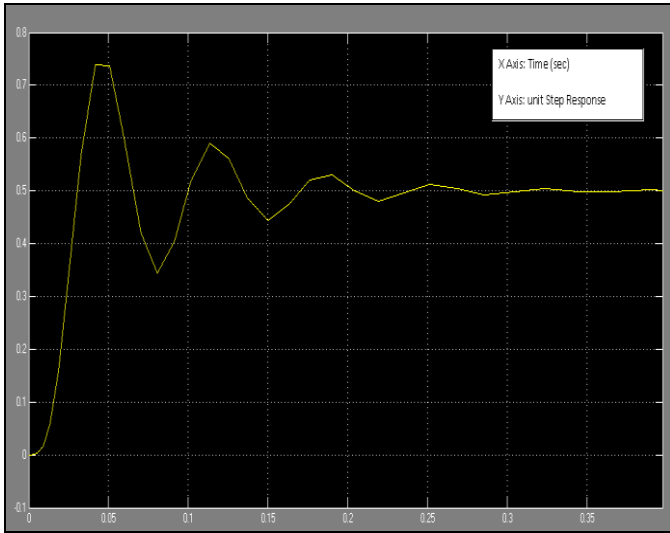


Fig-4: Block diagram with the unity feedback transfer function

#### 3.2 Analysis Of Conventionally Tuned PID Controller

For the transfer function obtained a step response is applied & the output is checked on the scope as shown in Fig 5. From the above system the step response obtained is as follows:



**Fig-5:** Simulation of the transfer function

The rise time,  $t_r$  is the time taken to reach 10 to 90 % of the final value is about 0.047 sec. The Maximum Overshoot,  $M_p$  of the system is approximately 0.75. Finally the settling time,  $t_s$  is about 0.357 sec.

**Table -1:** Result of the Simulation

Parameters	Value
Maximum Overshoot	0.75
Rise Time	0.047
Settling Time	0.357

From the analysis above, the system has not been tuned to its optimum. In order to achieve the following parameters we go for genetic algorithm approach. Table 1 illustrates the result of the simulation

#### 4. TUNING OF PID CONTROLLER USING GENETIC ALGORITHM APPROACH

##### 4.1 Overview of Genetic Algorithm

Genetic Algorithms (GAs) are a stochastic global search method that mimics the process of natural evolution. It is one of the methods used for optimization. The computational systems keep on improving & this makes GA useful for optimization. The genetic algorithm starts with no knowledge of the correct solution and depends entirely on responses from its environment and evolution operators such as reproduction, crossover and mutation to arrive at the best solution. By starting at several independent points and searching in parallel, the algorithm avoids local minima and converging to sub optimal solutions. GA starts with an initial chromosome & check for the fitness value. The fittest chromosomes are taken as parents further they are reproduced, crossed over & mutated. The offspring is checked for the value of the fitness

& depending on the value it is either taken or neglected from the population [1, 2, 3].

Given a clearly defined problem to be solved and a simple GA works as follows [8]:

1. Start with a randomly generated population of chromosomes i.e. candidate solutions to a problem. Candidate Solution is the solution (values of  $K_p$ ,  $K_d$  &  $K_i$ ) after every iteration
2. Create the population size (i.e. No. of roots closer to the required root those are  $K_p$ ,  $K_d$  &  $K_i$ ). The limits of the roots (i.e.  $K_p$ ,  $K_d$  &  $K_i$ ) or the bounds are specified.
3. The roots (i.e.  $K_p$ ,  $K_d$  &  $K_i$ ) which are selected are the initial values of the roots required.
4. A selection method called Normalized Geometric Selection is applied so that any random value can be selected. Selection is based on the fitness value of the root.
5. Reproduce the selected roots (i.e.  $K_p$ ,  $K_d$  &  $K_i$ ) so as to get optimized solution.
6. Crossover called Arithmetic crossover & Uniform mutation are performed so as to alter the roots to get a optimized root (i.e.  $K_p$ ,  $K_d$  &  $K_i$ )
7. Calculate the fitness  $f(x)$  (using fitness function/ performance index) of each root (i.e.  $K_p$ ,  $K_d$  &  $K_i$ ) in the population.
8. Repeat the following steps until 'n' offspring have been created.
9. Using fitness function (performance index MSE) finds the value of the error in the generation (iteration).
10. The roots having the highest fitness value are chosen.
11. Thus the roots obtained are the final values of  $K_p$ ,  $K_d$  &  $K_i$ .
12. If the obtained value are not according to the required ones (fitness value is not up to the mark) then go to step 2

Each iteration of this process is called a generation. GA is typically iterated for anywhere from 0 to 500 or more generations. The entire set of generations is called a run. At the end of a run there are often one or more highly fit chromosomes in the population. Since randomness plays a large role in each run, two runs with different random-number seeds will generally produce different detailed behaviours. GA researchers often report statistics averaged over many different runs of the GA on the same problem [5, 7].

Based on the steps flow chart for GA is as shown in Fig 6:

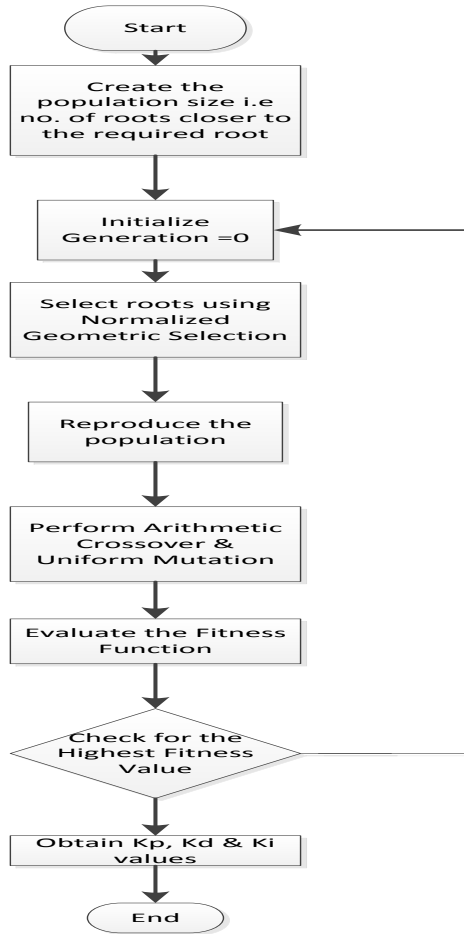


Fig-6: Flow chart for GA

#### 4.2 Implementation of GA Based PID Controller

Considering the following parameters given in Table 2 in the genetic algorithm the values of Kp, Ki & Kd are obtained.

Table-2: Parameters of GA

GA Parameters	Value/ Method
Population Size	60
Variable bounds [Kp Ki Kd]	[0 400; 0 400; 0 400]
Maximum number of generations	100
Performance Index/ Fitness function	Mean Square Error
Selection method	Normalized Geometric Selection
Crossover Method	Arithmetic Crossover
Mutation method	Uniform Mutation

By substituting the values in the genetic algorithm we get the value of Kp, Ki & Kd over 100 generation as shown in Fig 7.

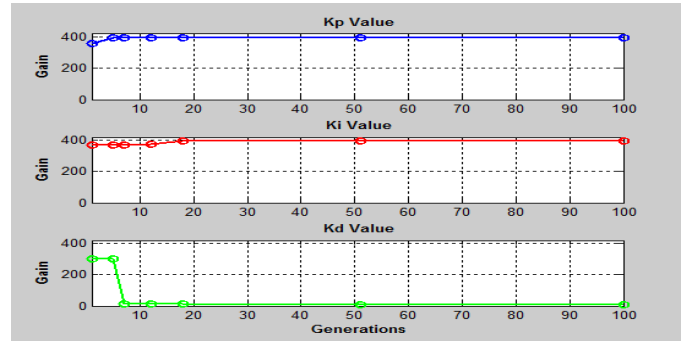


Fig-7: Values of Kp, Ki & Kd over the generation

The values of the parameters for PID controller are:

Kd = 9.5156  
 Kp =394.9682  
 Ki = 395.9763

#### 5. ANALYSIS OF RESULT

The Fig 8 shows minimum error in each generation. The error goes on reducing we get a steady state error of 0.003224 after 50 generations which remains constant further.

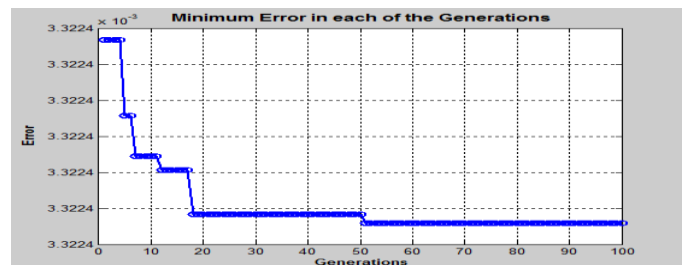


Fig-8: Minimum Error in the generation

Fig 9.shows the step response of the system with GA & Z-N tuned PID controller. Further the values are compared & analyzed.

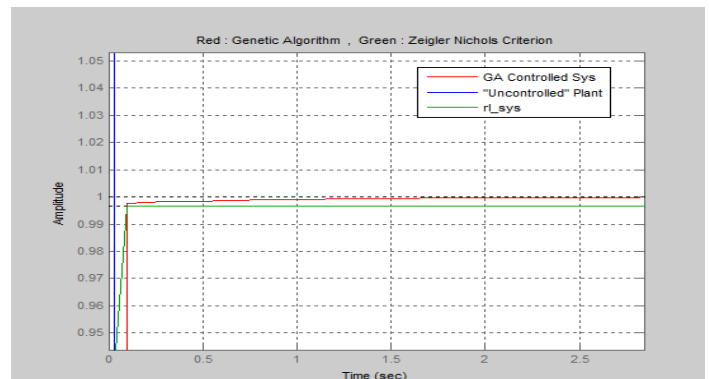


Fig-9: Step Response of the system with GA & Z-N tuned PID

Following Table 3 shows the comparison of the results obtained from the step response of the system. The parameters considered are the Maximum Overshoot, Rise time, Settling time & steady state error. Rise time, settling time & steady state error shows differences. Better results are obtained using GA than the classically tuned parameters. These results are obtained from Fig 9.

**Table-3:** Comparison of results

Tuning Parameters	Z-N Method	Genetic Algorithm
Maximum overshoot	0.997	0.990
Rise time (sec)	0.18	0.068
Settling time (sec)	0.3	0.13
Steady State Error	0.005	0.0001

## CONCLUSIONS

The designed PID with the Genetic Algorithm has much faster response than the classical method. Classical method is good for giving the starting points of what the PID values can be. However the GA designed PID is much better in terms of the rise time & the settling time than the Z-N method. The steady state error associated with the GA based PID is 0.0001. PID parameters tuned by GA in this paper gives the result which is better than the classical Z-N method.

## REFERENCES:

- [1]. Zhu Supeng & Fu Wenxing Yang Jun Luo Jianjun "Applying Genetic Algorithm to Optimization Parameters of Missile Control System" 978-0-7695-3745-0/092009 DOI 10.1109/HIS.2009.297 IEEE 2009 Ninth International Conference on Hybrid Intelligent Systems
- [2]. Neenu Thomas, Dr. P. Poongodi "Position Control of DC Motor Using Genetic Algorithm Based PID Controller" Proceedings of the World Congress on Engineering 2009 Vol II WCE 2009, July 1 - 3, 2009, London, U.K.
- [3]. Mahmud Iwan Solihin, Wahyudi, M.A.S. Kamal, Ari Legowo "Objective Function Selection of GA-Based PID Control Optimization for Automatic Gantry Crane" 978-1-4244-1692-9/08/\$25.00 ©2008 IEEE Proceedings of the International Conference on Computer and Communication Engineering 2008 May 13-15, 2008 Kuala Lumpur, Malaysia.
- [4]. Mohamed .M. Ismail "Adaptation of PID Controller using AI Technique for Speed Control of Isolated Steam Turbine" 978-1-4673-0484-9/12/\$31.00\_c 2012 IEEE
- [5]. Japan-Egypt Conference on Electronics, Communications and Computers
- [5]. R.Valarmathi, P.R.Theerthagiri, S.Rakeshkumar "Design and Analysis of Genetic Algorithm Based Controllers for Non Linear Liquid Tank System" ISBN: 978-81-909042-2-3 ©2012 IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012) March 30, 31, 2012
- [6]. Rashmi A. Mahale, Prof.S.D.Chavan " A Survey: Evolutionary and Swarm Based Bio-Inspired optimization Algorithm" International Journal of Scientific and Research Publications, Volume 2, Issue 12, December 2012 I ISSN 2250-3153.
- [7]. Mohammed El-Said El-Telbany "Employing Particle Swarm Optimizer and Genetic Algorithms for Optimal Tuning of PID Controllers: A Comparative Study" ICGST-ACSE Journal, Volume 7, Issue 2, November 2007
- [8]. M. B. Anandaraju, Dr. P.S. Puttaswamy , Jaswant Singh Rajpurohit "Genetic Algorithm: An approach to Velocity Control of an Electric DC Motor" International Journal of Computer Applications (0975 – 8887) Volume 26– No.1, July.

## BIOGRAPHIES:



**Gauri Mantri**, Assistant Professor, E & TC Department, Symbiosis Institute of Technology, India, gaurim@sitpune.edu.in



**N. R. Kulkarni**, Head Of the Department & Professor, Electrical Department, P.E. Society's Modern College of Engineering, India, nrkmcoe@gmail.com