DESIGN AND OPTIMIZATION OF PID CONTROLLER USING GENETIC ALGORITHM

Dr.P.Lakshmi Supriya
Assistant Professor, Department of Electrical & Electronics Engineering, Mahatma Gandhi Institute of Technology, Email: plaxmisupriya_eee@mgit.ac.in
Mr.Ch.Vinay Kumar
Assistant Professor, Department of Electrical & Electronics Engineering, Mahatma Gandhi Institute of Technology, Email: chvinaykumar_eee@mgit.ac.in

ABSTRACT:

Stochastic global search methods for optimization evolved from genetic algorithms due to the widespread development of a natural tendency. However, since it relies on a trial-and-error approach to selecting parameters, the Proportional Integral Derivative (PID) control has had difficulty reaching its full potential. Due to its foundation in selection and genetics, the Genetic Algorithm has evolved into a search algorithm for the study of human genetics. The more versions of GA, the more refined it becomes. It is assumed that each repetition would provide a better outcome. The accuracy of these findings has been double-checked. Based on the selection criteria, only the best roots or solutions are evaluated for the following generation. It does this by producing the GA steps. As a performance measure, the Mean Square Error (MSE) value is used. In this study, a genetic algorithm-based PID controller for missile altitude control is tested against the more traditional Zeigler-Nichols approach (Z-N). The Z-N approach is the most common way to adjust the settings of a PID controller. The best answer may be found by adjusting the settings. Maximum overshoot, settling time, rise time, and steady state error are only a few of the metrics that the system's minimum error measures. Comparing quicker reaction times between traditional and PID-based genetic algorithms and the suggested PID technique has also been shown.

DOI Number: 10.14704/nq.2022.20.4.NQ22294 **NeuroQuantology 2022; 20(4):705-714**

I. Introduction

To maximise system efficiency in today's world, a Proportional Integral Derivative (PID) controller is widely accepted. Use a controller to fine-tune the controller's settings. The settings of a PID controller may be adjusted in a variety of ways. The Ziegler Nichols oscillation technique, the Ziegler Nichols reaction curve method, and the Cohen-Coon reaction curve approach are traditional methods for fine-tuning the PID controller's settings. Using an evolutionary approach, however, has resulted in a significant increase in tuning in recent years. Each repetition of these algorithms improves the final outcome. Computer methods such as Ant Colony Optimization, Particle Swam Optimization, and Genetic Algorithm are often utilised nowadays. A live being's behaviour is the basis for these [5, 6, and 7]. The PID controller will be used to regulate the missile's height. Modern missile systems need a sturdy, sophisticated and accurate system to operate them. For comparison, the Ziegler-Nichols oscillation technique is utilised. Using genetic algorithms, the authors of this work want to demonstrate that a system may be made more efficient [1].

The system model of a missile control system is first outlined in this work. The Ziegler-Nichols method is used to tune the standard PID controller. An overview of the Genetic Algorithm Based PID Controller and its implementation will be discussed later in the session. Lastly, the matching system's simulation results are collected and compared.

II. LITERATURE REVIEW:

In recent years, there has been a significant rise in road accidents. Property losses and other costs might account for up to 3% of global GDP, according to current projections. Back-up help for drivers in longitudinal vehicle control to prevent crashes has been the focus of numerous research projects in this area. The first 'cruise control devices' appeared in the United States in the 1970s. When the device is turned on, you may keep a constant pace by accelerating or braking. However, other drivers are not considered.
Adaptive cruise control (ACC) devices employ sensor data to adjust the brakes and the engine throttle in order to maintain a safe distance from the vehicle in front of you. The Mercedes S-class, the Jaguar XF, and the Volvo XC90 are just few of the luxury vehicles that already feature it. It was also created by the U.S. Department of Transportation and the Japan-based ACAHSR in support of 'Intelligent Vehicles'. This investigation focuses on Adaptive Cruise and its improved version. According to a literature assessment, power quality and customised power devices play an important role in the power system. The Particle Swarm Optimization technique, utilised in the distribution system, benefits automatic voltage regulators (AVR). According to published research, many types of controllers may be used to compensate for different sorts of automated voltage regulators (AVR).

- The Darwinian notion of "survival of the fittest" is at the heart of genetic algorithms (GAs). The genetic algorithm's most important algorithmic ideas are presented here.
- Create a random population of chromosomes, which is to say, a solution to the issue that is appropriate.
- Evaluate the fitness of each chromosome in the population in Step II [Fitness].
- A new population is introduced in Step III. Repeat the following stages until the new population is complete.
- [Selection] In a population, choose two chromosomes from each parent based on their ability to produce children. The more fit you are, the more likely you are to be chosen as a parent.
- [Crossover] Create new offspring by crossing the parents with a high possibility of success. Offspring are a carbon replica of their parents if no crossover occurred.
- New progeny should be mutated at each locus with a mutation frequency of one in three.
- New offspring should be added to the new population.
- In the fourth and last step, we replace Rerun the algorithm with the newly created population.
- In Step V [Test], if all conditions are met, halt and return the current population's best answer.

III. System Model

A PID controller is seen in Fig. 1 as part of the missile control system. Here, a continuous PID controller is employed. Using a PID controller in conjunction with the Steering Engine, the missile's height may be maintained at a pre-determined level The Unity feedback system is regarded to deliver a feedback of the real position felt and to be able to alter the position. Equation represents the continuous PID controller transfer function (1). The equation for the steering engine's transfer function may be found here (2).

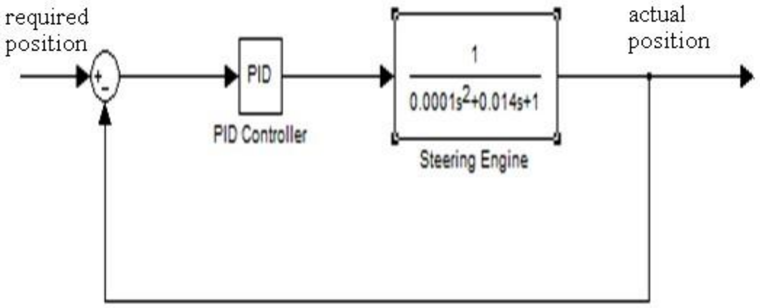

Figure 1. Block diagram of Entire system

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

$$Gc(s) = K\left(1 + {}^{1}/_{sT_i} + sT_d\right) \qquad (1)$$

Where,

K Is a constant of proportionality, and T_i is Integral time; it must be a real and positive number. It must be a real, finite, and nonnegative number known asT_d.

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

$$Trasfer\ Function = \frac{1}{0.0001s^2 + 0.014s + 1} \qquad (2)$$

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

$K_P = 6.3843, K_i = 281.25,\ K_d = 0.01004$

Transfer function of the compensator is given by

$$Trasfer\ Function\ Gc(s) = \frac{0.01004s^2 + 6.38435s + 281.25}{s} \qquad (3)$$

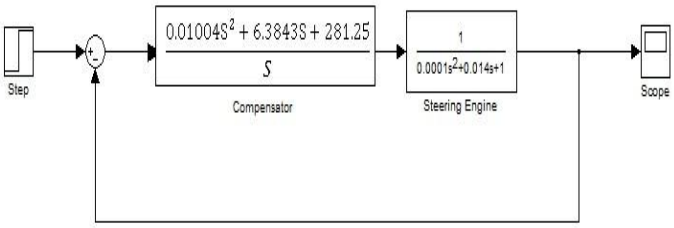

Figure 2. System with Compensator transfer function

In order to get unity feedback, both transfer functions in series are multiplied and subsequently solved. Figure 4 depicts the system's transfer function, with a scope serving as an output.

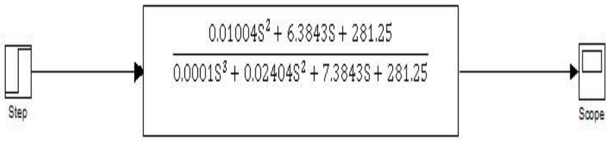

Figure 3. Block diagram with unity feedback transfer function

Analysis of Conventionally Tuned PID Controller

Fig 5 shows how the output of the scope may be examined for the transfer function acquired. The following is the step response derived from the aforementioned system:

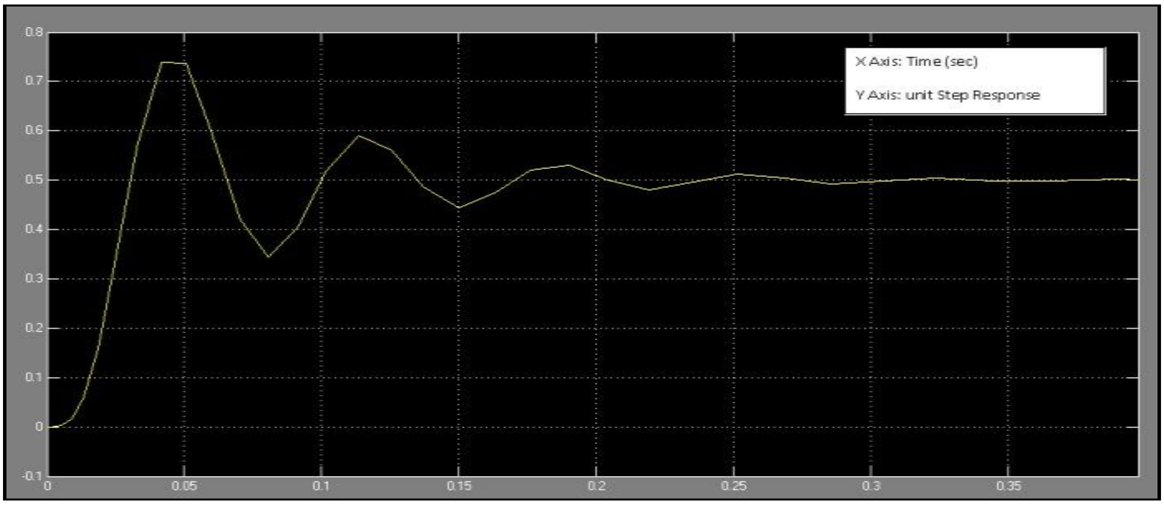

Figure 4. Simulation of transfer function

Table 1. Simulation Result

Parameters	Value
Maximum Overshoot	0.74
Rise Time	0.045
Settling Time	.355

An initial rising time of around 0.045 seconds may be expected. The system's maximum overshoot,M_P is around 0.74. In end, t_s is around 0.355 seconds?
According to our findings, the system's settings aren't optimal. The genetic algorithm technique is used to attain the following parameters. Table 1 shows the simulation results.

IV. Standard PID Tuning Methods

In 1942, J.G. Ziegler and N.B. Nichols developed the Open Loop Tuning Method, which is defined by two parameters and is based on the plant's responsiveness to a step input. Known as Open Loop or Step Response tuning, it is widely used. The processes a and L values are calculated using a unit step function.
It was a model of the way things worked.

$$G(S) = \frac{b}{s} e^{-sL}$$

$b = \frac{a}{L}$ Is a time-delayed integral operation.

Ziegler and Nichols also created the Frequency Response method, a closed loop tuning technique.
Process non-features linearity's are predicated on understanding the moment at which $G(S)$(Nyquist)'s curve crosses the negative real line. A Gain,K_u and a time, T_u were defined.
Using the Chien, Hrones, and Reswick Tuning Method The original Ziegler-Nichols Open Loop approach for the fastest reaction with and without overshoot was derived from this algorithm.
In other words, the "Rule of Thumb" The Rule of Thumb approach is the correct way to adjust a PID loop. Tuning rules are difficult to understand since there is no clear explanation or illustration of how the arithmetic underlying them works.

V. Proposed Method

Tuning Of PID Controller Using Genetic Algorithm Approach:
It goes like this [8]: Given a clearly stated issue to be addressed and a basic GA:
Start with a population of chromosomes that has been randomly produced, i.e. potential solutions to a problem. Iteration after iteration, the Candidate Solution (the values of K_P, K_d & Ki) is the solution. Achieve the desired population density (i.e. No. of roots closer to the required root those are(K_P, K_d &K_i). For example, K_P K_d & K_i are defined as the root limits (i.e., the boundaries). There are three initial values of the roots (K_P, K_d &K_i) that need to be chosen. In order to choose any random value, a technique known as Normalized Geometric Selection is used. The root's fitness value is taken into account while making a selection. Optimize your solution by reproducing the chosen roots (i.e. K_P K_d &K_i). There are two types of crossover used to optimise the roots: Arithmetic crossover and Uniform mutation (i.e.K_P K_d &K_i) each root (K_P K_d &K_i) in the population has a fitness (x) (using a fitness function/performance index). Follow instructions as many times as necessary to produce infinite number of descendants. Generational error may be determined using a fitness function (performance index MSE) that measures the value of that error (iteration). There is a preference for the roots with the greatest fitness value. Thus, the final values ofK_P K_d & K_i are obtained. If the results aren't what you expected (the fitness level isn't up to snuff), go to step 2.

Each time this process is repeated, it is referred to as a generation. GA may be iterated for as little as 0 generations or as much as 500 or more. A run refers to a collection of generations as a whole. At the conclusion of a race, the population typically contains one or two very fit chromosomes. Because of the importance of randomization, two runs with different random-number seeds are likely to provide different results. Over a large number of GA runs on the same topic, researchers generally publish averaged data [5, 7].

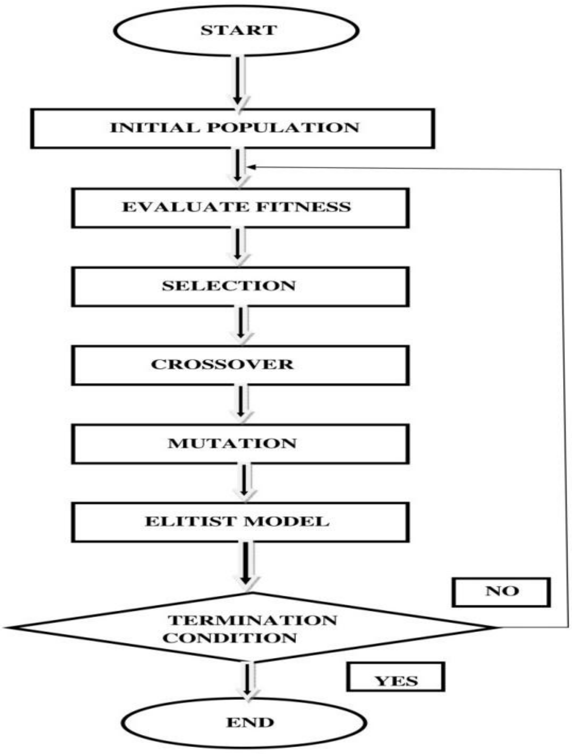

Figure 5. Genetic Algorithm Process

Implementation of GA Used PID Controller:

K_P K_d &K_i values provided in Table 2 in the genetic method are used to calculate the K_P values.

Table 2. GA Parameters

GA Parameters	Value/ Method
Population Size	60
Variable bounds[K_P K_d &K_i]	[0 400; 0 400: 0 400]
Maximum no.of generations	100
Performance Index/ Fitness function	Mean Square Error
Selection Method	Normalized Geometric Selection
Cross over Method	Arithmetic Cross over
Mutation Method	Uniform Mutation

By substituting the values in the genetic algorithm we get the value of(K_P, K_d &K_i over 100 generation as shown in Fig 5.

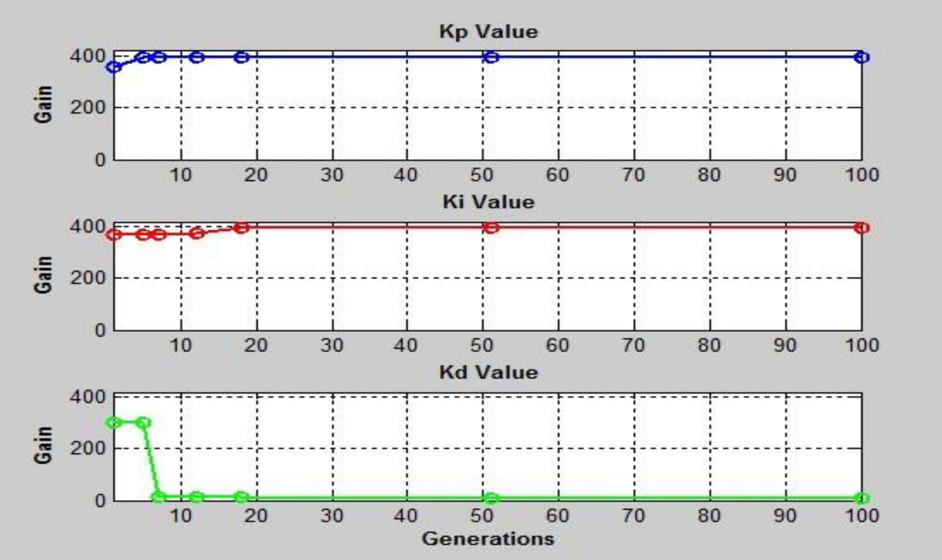

Figure 6. Values of K_P K_d &K_i over the generation

PID controller parameters are as follows:
For example, K_P Is 9.5156, K_d is 394.9682, K_i=395.9763.
The control system's temporal domain and stabilising behaviours are examined in [8] using Root locus. Figure 5 shows the root locus curve for the PID-controlled system. The system is stable because the closed loop poles are all located in the left side of the plane.

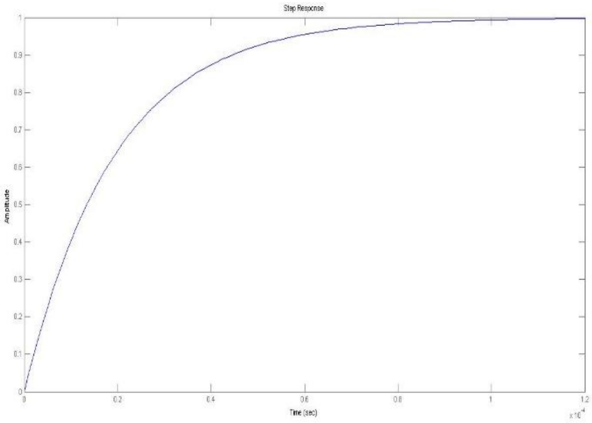

Figure 7. Root locus curve of system with PID controller

A bode plot [8] may be used to examine a system's frequency response information. PID cruise control with GA-optimized PID controller amplitude and phase plots are displayed.

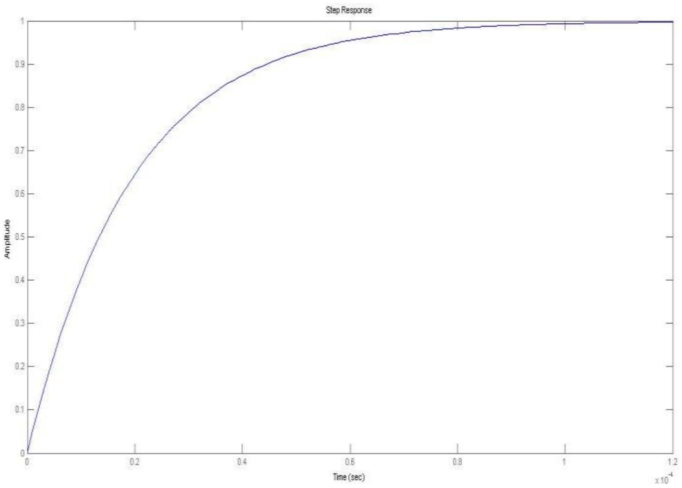

Figure 7. Bode plots of system with PID controller

VI. Results:

Unit step signal is used as the study's input signal. The following is a diagram depicting the system's simulation using a PID controller.

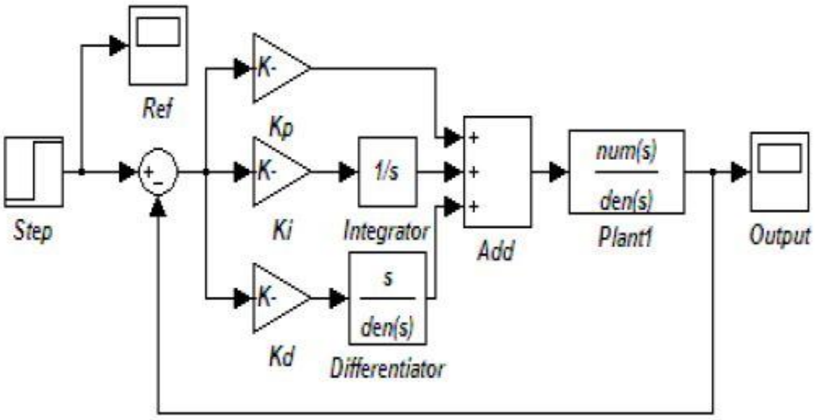

Figure 8. Cruise control system with PID controller

A 1.14 percent overrun, a 2.15 second peak time, a 0.945 second rising time, and a 1.46 second settling time are the maximums for this design's parameters. Table 3 summarises the performance of the GA-based PID controller.

Table 3. Parameters of Cruise control

GA Parameters	Cruise control GA method
K_P	3.845
K_d	0.086
K_i	3.484
Rise Time	1.0088

Settling Time	1.3928
Settling min	0.9729
Settling max	0.9977
Overshoot Time	1.0099
Under shoot	0
Peak	1.0099
Peak Time	1.7457

VII. Conclusion

Genetic Algorithm-designed PIDs are substantially more responsive than traditional PIDs. The classical technique is useful for establishing a range of possible PID values. When it comes to rising and settling time, the GA PID is superior to Z-N approach. With a GA-based PID, the steady-state error is 0.0001. GA tuning of PID parameters in this work yields a better outcome than the conventional Z-N technique.

For the cruise control system, a PID controller employing GA has been suggested. PID, state space and fuzzy logic controllers have been used to evaluate the performance of the controller. In terms of maximum overshoot, peak time, rising time, settling time, and steady state error, the GA-based PID controller outperforms the other controllers. High-frequency noise rejection is shown by the system's resilient behaviour, as is its ability to reject output disturbances. Eventually, a fractional order controller will be developed and the system's performance will be compared to this design.

Extending the study into fractional order and two-DOF PID controller design is possible, and the findings may then be compared to the present work.

References:

[1]. Zhu Supeng & Fu Wenxing Yang Jun Luo Jianjun "Applying Genetic Algorithm to Optimization Parameters of Missile Control System" 978-0-76953745-0/092009 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 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: 978-81909042-2-3 ©2012 IEEE-International Conference on Advances in Engineering, Science And Management (ICAESM -2012) March 30, 31, 2012

[6]. Rashmi A. Mahale, Prof. .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 1 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. Pettaway, 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.