Journal of Electronics, Computer Networking and Applied Mathematics
ISSN: 2799-1156
Vol: 04, No. 03, April-May 2024
http://journal.hmjournals.com/index.php/JECNAM
DOI: https://doi.org/10.55529/jecnam.43.8.17

Synergistic Integration of Advanced Control Strategies for Comprehensive PID Controller Tuning using MATLAB

Saiful Islam Tuhin[1*], Md. Arif Hasan Masum[2], Md. Al Araf[3], M. Mohi Uddin[4]

[1,2,3,4]Department of Electrical & Electronic Engineering, Faculty of Engineering and Applied Sciences, Bangladesh University of Business & Technology (BUBT), Dhaka, Bangladesh.*

Email: [2]arifhasanmasum4@gmail.com, [3]mdalaraf@icloud.com,
[4]eng.mohiuddin08@gmail.com
Corresponding Email: [1]saifulislamtuhin15@gmail.com*

Received: 28 November 2023 **Accepted:** 14 February 2024 **Published:** 01 April 2024

***Abstract:** This study investigates the impact of Proportional-Integral-Derivative (PID) tuning on system performance through comprehensive measurements before and after tuning. The parameters analyzed include rise-time, settling-time, overshoot, undershoot, and peak time. Prior to tuning, the system exhibited significant deficiencies, characterized by prolonged rise and settling times, substantial overshoot and undershoot, and extended peak time. Following PID tuning, notable improvements were observed across all parameters. The rise time and settling time were dramatically reduced, indicating a faster system response and improved stability. Moreover, overshoot and undershoot were effectively minimized, leading to greater accuracy and precision in the system's output. These findings underscore the efficacy of PID tuning in optimizing system performance and achieving desired control objectives.*

***Keywords:** PID Tuning, Control Systems, MATLAB-Simulink, Tuning Process, Tuning Methods.*

1. INTRODUCTION

A proportional-integral-derivative (PID) controller is frequently utilized as a feedback loop element in industrial control systems 1]. "Tuning" a controller involves adjusting or determining its parameters to attain optimal values for achieving the desired response[2]. The optimal behavior of a process or its response to set point changes varies across applications. In this framework, parameters KP, KI, and KD are derived from the gain (Ku) that induces marginal stability in a system with a P-only controller ("u" denotes "ultimate"). Depending on the scenario, tuning may involve either PID or PI adjustment. However, PD tuning poses a challenge as even minor noise can trigger substantial output variations[3].

Distinguished from simpler controllers, a PID possesses the capability to regulate process outputs based on historical data and the rate of change of its error signal, rendering it capable of more precise and stable control. PID controllers don't demand complex mathematics or intricate optimal control algorithms for design and can be readily adjusted or tuned to suit specific applications. Various software tools are employed for PID tuning, with MATLAB and SIMULINK being notable examples[4]. An illustrative example is provided to elucidate this process. A schematic block diagram of a control system featuring a PID controller is depicted in Figure 1.

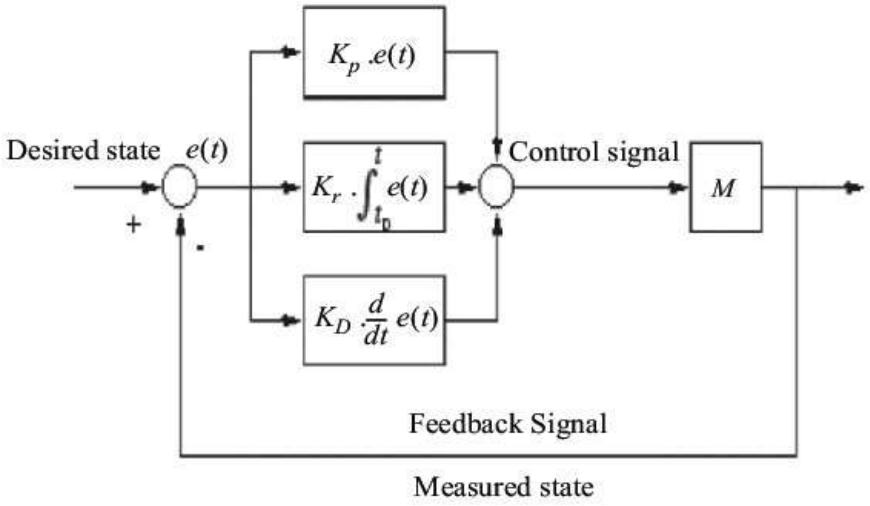


Figure 1: Block Diagram of a System with PID Controller.[5]

The transfer function of the most fundamental form of a PID controller is [6]:

$$C(s) = K_P + \frac{K_I}{s} + K_D \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(i)}$$

Where K_P represents the proportional gain, K_I denotes the integral gain, and K_D signifies the derivative gain; commonly known as controller parameters. [7].

The fundamental attributes of a closed-loop step response encompass rise-time, overshoot, settling-time, and steady-state error. The impact of elevating the controller parameters is delineated in Table 1.

Table 1: The Impact of Increasing the Controller Parameters[8].

Parameter	**Rise-Time**	**Overshoot**	**Settling-Time**	**S.S. Error**
K_P	Decrease	Increase	Small Change	Decrease
K_I	Decrease	Increase	Increase	Eliminate
K_D	Small Change	Decrease	Decrease	Small Change

2. RELATED WORKS

PID controllers are widely used in industrial applications due to their simplicity and effectiveness. The PID controller considers three parameters: proportional, integral, and derivative, which are combined to generate a control signal that regulates the output of a system. In the literature, there are numerous studies on the design, simulation, and analysis of PID controllers using MATLAB. Researchers have investigated various tuning methods for optimizing the control parameters of PID controllers, including trial and error, heuristic methods, andoptimization algorithms. Moreover, some researchers have explored the use of advanced control techniques, such as fuzzy logic and neural networks, to enhance the performance of PID controllers. Other studies have focused on the implementation of PID controllers in specific applications, such as temperature control, pressure control, and speed control. Overall, theliterature review highlights the importance and versatility of PID controllers in engineering applications, as well as the usefulness of MATLAB in designing and simulating control systems.The literature also provides insights into various methods for optimizing the performance of PID controllers and the potential for advanced control techniques to improve control system performance.

Background: PID control is a widely used method for regulating systems in industries such as manufacturing and robotics. It relies on three key factors: proportional, integral, and derivative terms, which are adjusted to minimize the difference between desired and actual values. Proper tuning of these factors is essential for optimal system performance, balancing response time, stability, and disturbance rejection. Various tuning methods exist, from simple heuristics to advanced algorithms, aiming to refine control strategies for increased efficiency and precision in diverse applications.

Motivation: Choosing a topic on PID controller using MATLAB is an excellent decision for several reasons. Firstly, PID controllers are widely used in industry for controlling a range of physical systems, including temperature, pressure, and speed. Therefore, gaining an understanding of how to design and implement a PID controller is highly relevant and practical for a wide range of engineering applications. Secondly, MATLAB is a powerful tool that provides a user-friendly interface for designing and simulating control systems.It is widely used in industry and academia for system-level design, simulation, and analysis. Opting to delve into a study on PID controller utilizing MATLAB not only offers practical exposure to this tool but also cultivates a valuable skill set highly coveted by employers. Finally, the study on PID controllers using MATLAB offers an opportunity to develop problem-solving skills and explore the theoretical concepts of control systems. The study will involve working with mathematical models, analyzing system behavior, and designing a control strategy to meet specific performance requirements. This paper will not only provide a solid foundation in control theory but also help to develop practical skills that are essential in the field of Engineering. Overall, by choosing a study on PID controller using MATLAB, you will gain practical skills, theoretical knowledge, and problem-solving abilities that are highly valuable in the field of engineering.

Objectives: The objectives of a study on PID controller using MATLAB may include:

Understanding the Theoretical Concepts of PID Control: The study may aim to provide a comprehensive understanding of the basic principles of PID control, including the proportional, integral, and derivative components, as well as how they are combined to generate a control signal.

Designing a PID Controller: The study may aim to design a PID controller using MATLAB, including tuning the controller to meet specific performance requirements.

Simulating a Control System: The study may aim to simulate a control system using MATLAB, including evaluating the performance of the PID controller in regulating the output of the system.

Analyzing System Behavior: The study may aim to analyze the behavior of a physical system, including identifying the sources of error and determining the impact of different control parameters on system performance.

Optimizing Control Parameters: The study may aim to optimize the control parameters of the PID controller, including evaluating the impact of different tuning methods and parameter values on system performance.

Comparing Control Strategies: Compare the performance of different control strategies, including PID control, other feedback control methods, or open-loop control.

3. METHODOLOGY

The methodology for a study on PID controller using MATLAB that involves using code to verify simulation results may include the following steps:

Design the System using PID: Design a system with PID controller using MATLAB, including selecting appropriate control parameters and transfer functions to meet the desired performance requirements[8].

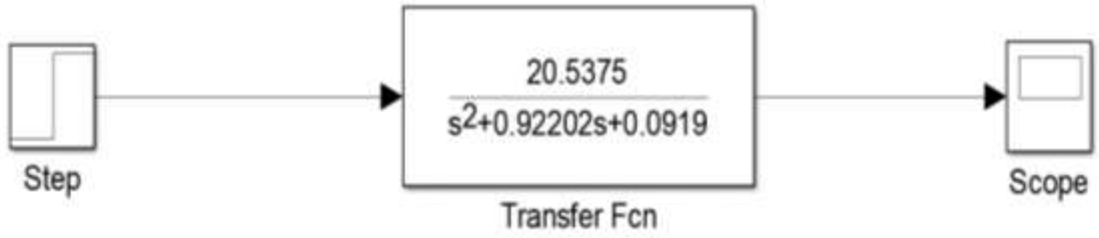


Figure 2: Open Loop System

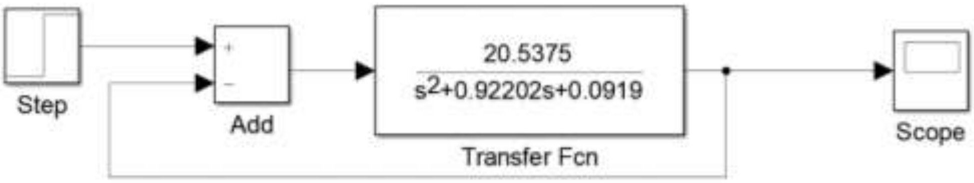


Figure 3: Closed Loop System

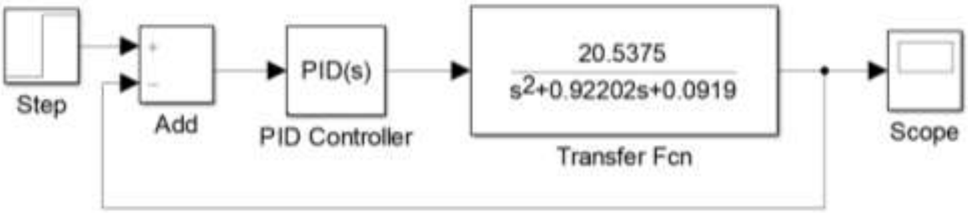


Figure 3: System Diagram with PID Controller

Tuning of PID Controller in Simulink: Simulate the control system using MATLAB to evaluate the performance of the PID controller. By using the Matlab pid controller tuning tool it can be obtained[9].


Controller parameters

Source:	internal
Proportional (P):	1
Integral (I):	1
Derivative (D):	0
Filter coefficient (N):	100

Figure 4: PID Controller's Parameters before Tuning

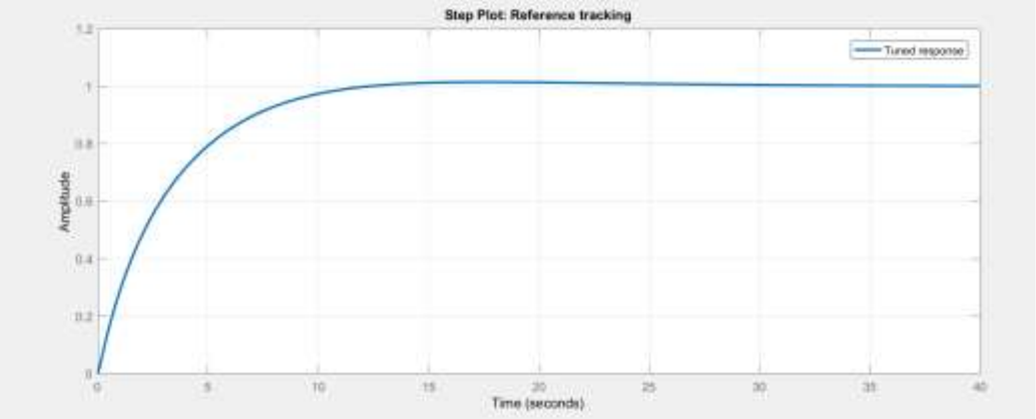


Figure 5: On Process Tuning

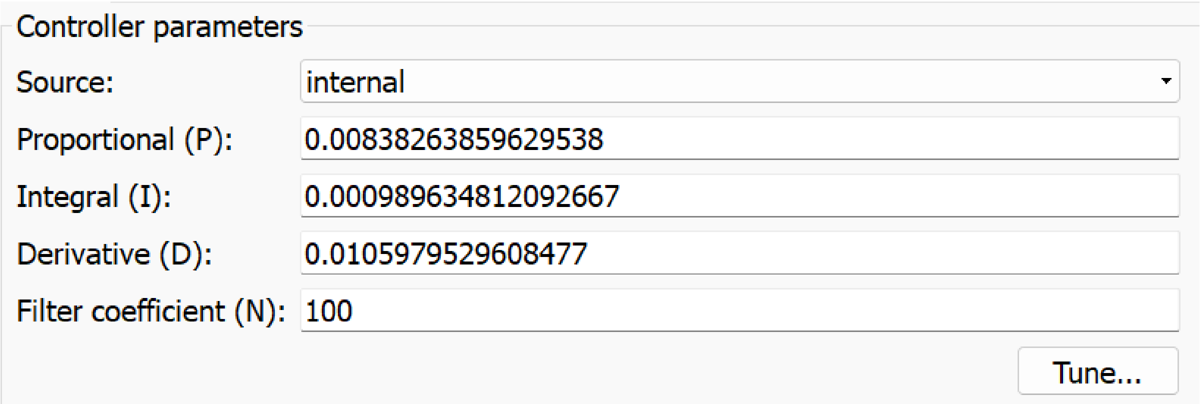
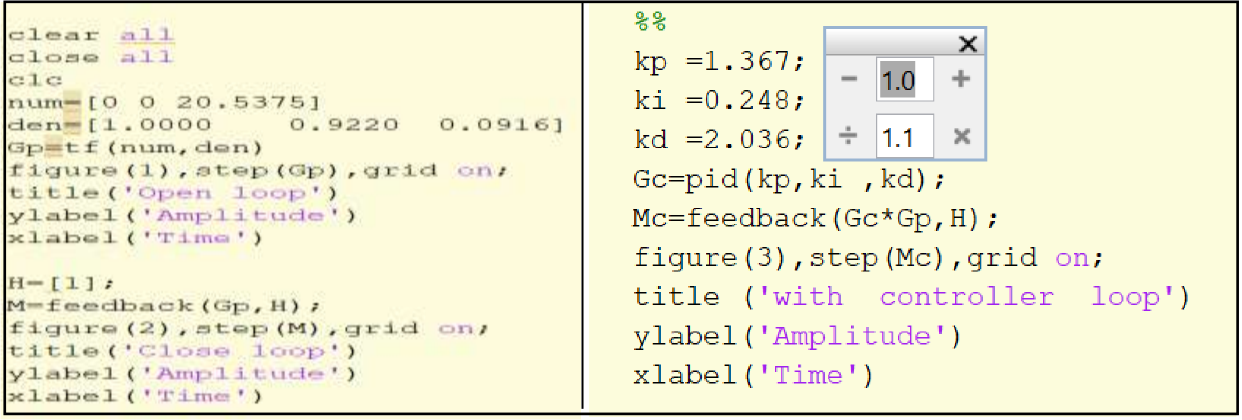

Controller parameters
Source: internal
Proportional (P): 0.00838263859629538
Integral (I): 0.000989634812092667
Derivative (D): 0.0105979529608477
Filter coefficient (N): 100

Figure 6: PID Controller’s Parameters after Tuning

Verify Simulation Results using Code: Develop MATLAB code to verify the simulation results obtained from Simulink. This could involve using MATLAB's numerical integration functions to simulate the system behavior and compare the results with those obtained from Simulink.[10]

Figure 7: Code for PID Tuning


```
clear all
close all
clc
num=[0 0 20.5375]
den=[1.0000    0.9220   0.0916]
Gp=tf(num,den)
figure(1),step(Gp),grid on;
title('Open loop')
ylabel('Amplitude')
xlabel('Time')

H=[1];
M=feedback(Gp,H);
figure(2),step(M),grid on;
title('Close loop')
ylabel('Amplitude')
xlabel('Time')
```

```
%%
kp =1.367;
ki =0.248;
kd =2.036;
Gc=pid(kp,ki ,kd);
Mc=feedback(Gc*Gp,H);
figure(3),step(Mc),grid on;
title ('with controller loop')
ylabel('Amplitude')
xlabel('Time')
```

Analyze the System Behavior: Analyze the behavior of the system to identify sources of error and evaluate the performance of the PID controller. This could involve analyzing key system variables, such as steady-state error, rise time, and overshoot[11].

Compare Control Strategies: Compare the performance of different control strategies, including PID control, other feedback control methods, or open-loop control. This could involve evaluating the impact of different control strategies on system performance and identifying the most effective approach[3].

4. RESULTS AND DISCUSSION

Simulink Results

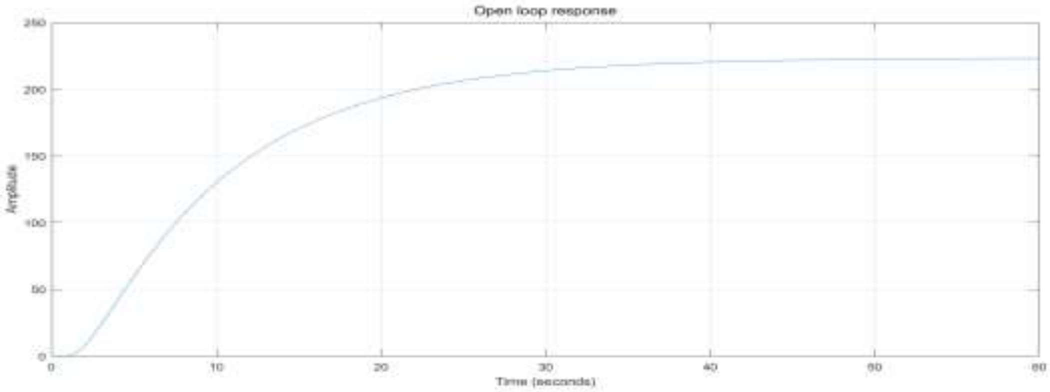


Figure 8: Open Loop System Response

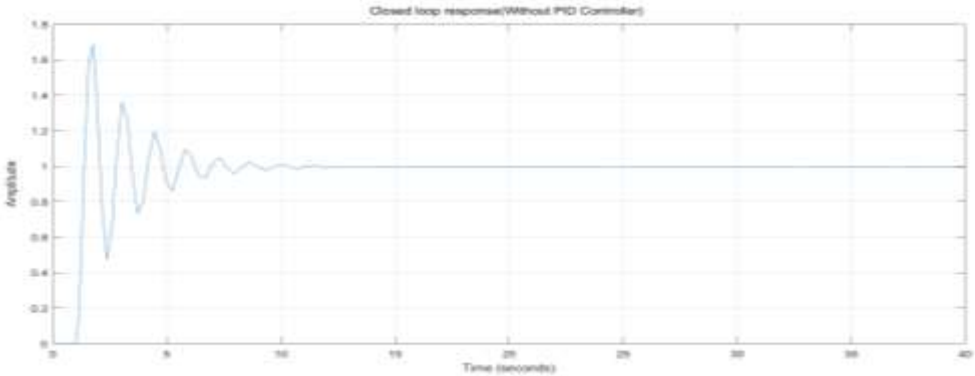


Figure 9: Closed Loop System Response (Without PID Controller)

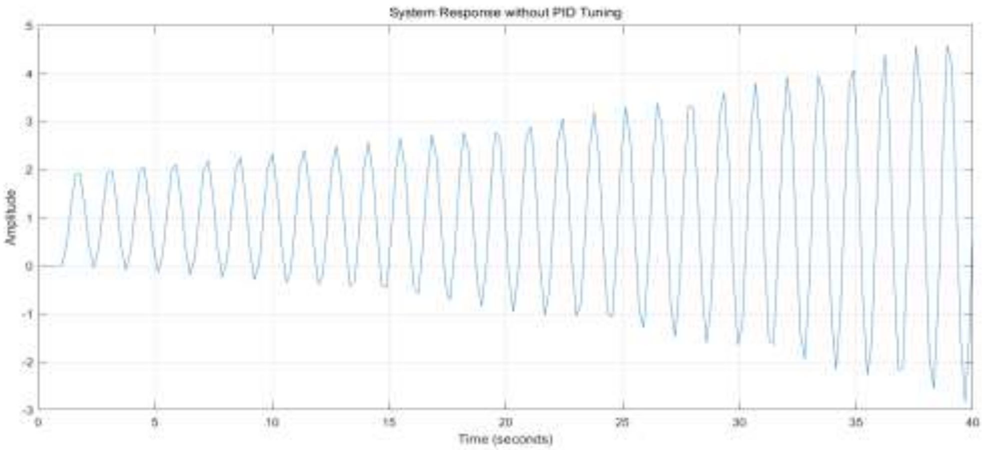


Figure 10: System Response without PID Tuning

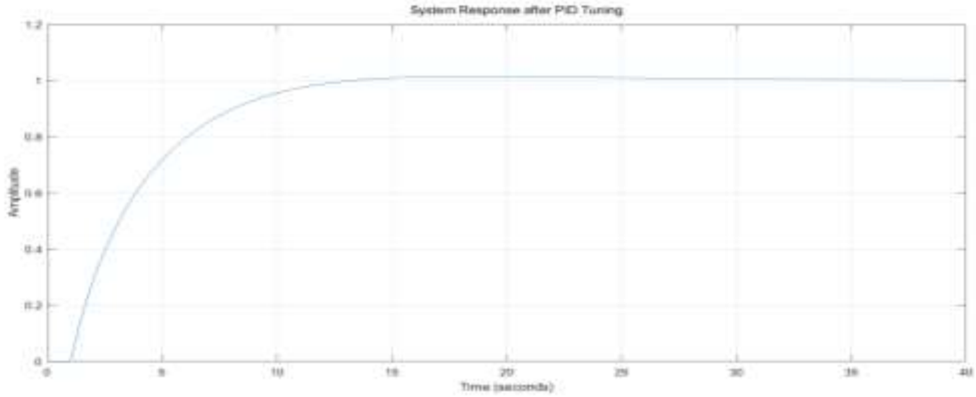


Figure 11: System Response after PID Tuning

Code Results:

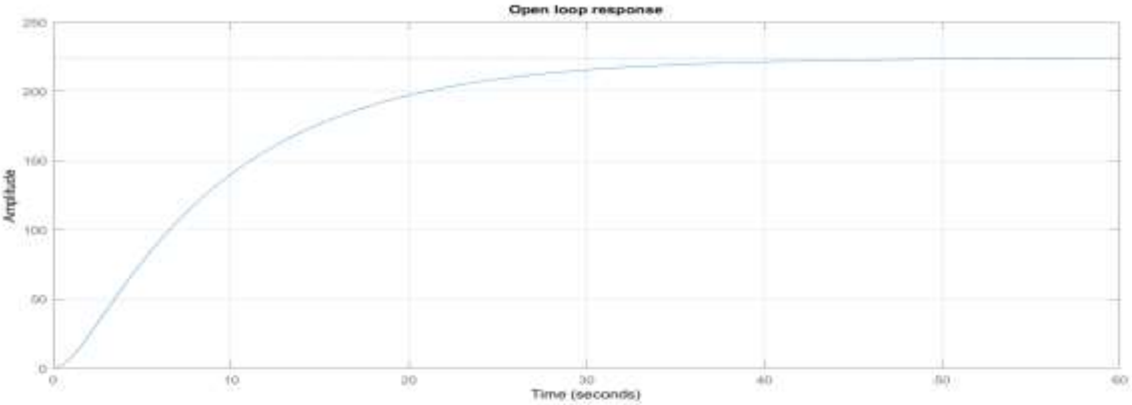


Figure 12: Open Loop System Response

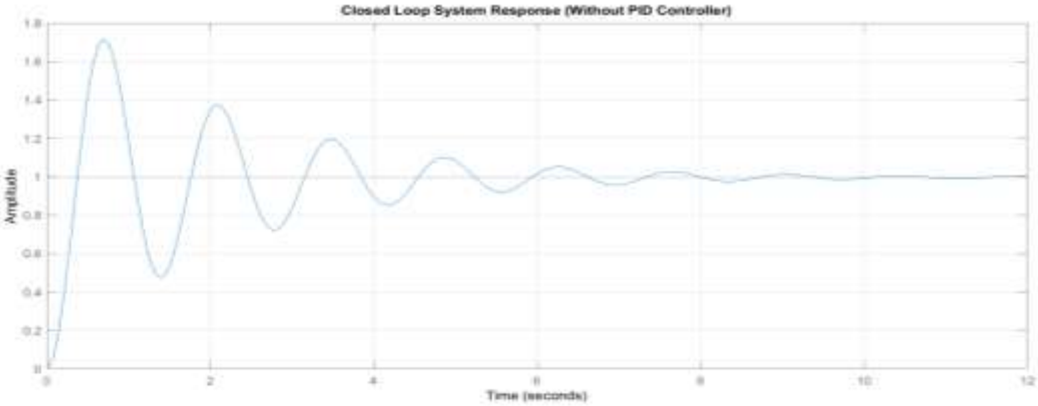


Figure 13: Closed Loop System Response (Without PID Controller)

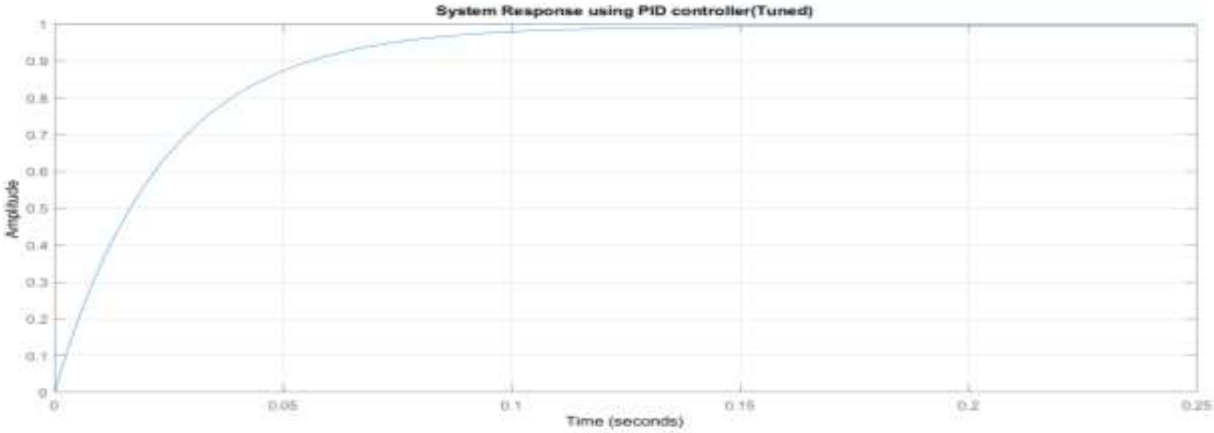


Figure 14: System Response using PID controller (Tuned)

Table 2: Performance Measurements of PID Controller

Measurements Parameters	Before Tuning	After Tuning
Rise Time	0.2518	10.5718
Settling Time	39.9954	17.0549
Overshoot	422.6660	0.8284
Undershoot	323.9202	0
Peak Time	38.9074	26.1010

5. CONCLUSION

The before and after tuning measurements clearly demonstrate the significant improvements achieved through the PID tuning process. Prior to tuning, the system exhibited considerable performance issues, characterized by prolonged rise and settling times, substantial overshoot and undershoot, as well as extended peak time. However, after tuning, there is a remarkable enhancement across all parameters. Notably, the rise time and settling time have been dramatically reduced, indicating a quicker system response and faster stabilization. Moreover, the overshoot and undershoot have been effectively minimized, leading to a more accurate and stable output. The peak time has also been notably reduced, further indicating the improved dynamic behavior of the system. Overall, these results underscore the effectiveness of PID tuning in optimizing system performance and highlight its crucial role in achieving desired control objectives.

6. REFERENCES

1. “A simple procedure for improving performance of PID controllers | IEEE Conference Publication | IEEE Xplore.” Accessed: Mar. 12, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1223126

2. S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, “Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems,” Heliyon, vol. 8, no. 5, p. e09399, May 2022, doi: 10.1016/j.heliyon.2022.e09399.
3. A. Kundu, S. I. Tuhin, M. S. H. Sani, M. W. R. Easin, and M. A. H. Masum, “Analytical Comparisons of the PID, ANN, and ANFIS Controllers’ Performance in the AVR System,” Int. J. Innov. Sci. Res. Technol., vol. 8, p. 28, 2023, Accessed: Feb. 06, 2024. [Online]. Available: https://www.ijisrt.com/assets/upload/files/IJISRT23AUG699.pdf
4. “A review of PID control, tuning methods and applications | International Journal of Dynamics and Control.” Accessed: Mar. 12, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s40435-020-00665-4
5. H. Bansal, “Tuning of PID Controllers using Simulink,” Int. J. Math. Model. Simul. Appl., Jan. 2009.
6. O. Lequin, M. Gevers, M. Mossberg, E. Bosmans, and L. Triest, “Iterative feedback tuning of PID parameters: comparison with classical tuning rules,” Control Eng. Pract., vol. 11, no. 9, pp. 1023–1033, Sep. 2003, doi: 10.1016/S0967-0661(02)00303-9.
7. L. Wang, PID control system design and automatic tuning using MATLAB/Simulink. Hoboken, NJ Chichester, West Sussex: Wiley, 2020.
8. PID controller design using Matlab code and simulink model., (Jun. 07, 2020). Accessed: Mar. 12, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=B8B8I_aJD6Y
9. How To Tune PID Controller in MATLAB SIMULINK ? | Dr. J. A. Laghari, (Oct. 19, 2020). Accessed: Mar. 12, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=ARewp6SCwGs
10. S. I. Tuhin, M. A. Araf, M. A. H. Masum, M. N. A. Siddiki, and M. O. Faruk, “Integrative Fault Analysis of Transmission Lines using MATLAB and ETAP,” J. Electron. Comput. Netw. Appl. Math. JECNAM ISSN 2799-1156, vol. 4, no. 02, Art. no. 02, Mar. 2024, doi: 10.55529/jecnam.42.54.65.
11. D. H. Kim, “Hybrid GA–BF based intelligent PID controller tuning for AVR system,” Appl. Soft Comput., vol. 11, no. 1, pp. 11–22, Jan. 2011, doi: 10.1016/j.asoc.2009.01.004.