Journal of Engineering Research and Reports

Volume 25, Issue 5, Page 17-30, 2023; Article no.JERR.101821
ISSN: 2582-2926

Performance Evaluation of Networks Using Gain Scheduling PID Networked Control System for Nonlinear Systems

Mmasom Iruoma Ndefo [a], Solomon Chibuzo Nwafor [b*], Chidiebele C. Udeze [a], Oghenevwaire Akpeghagha [c] and Oluchi Christiana Ugbe [d]

[a] Department of Electronic and Computer Engineering, Faculty of Engineering, University of Nigeria, Nsukka, Enugu State, Nigeria.
[b] Department of Mechatronic Engineering, Faculty of Engineering, University of Nigeria, Nsukka, Enugu State, Nigeria.
[c] Africa Center of Excellence for Sustainable Power and Energy Development, University of Nigeria, Nsukka, Enugu State, Nigeria.
[d] Department of Electrical Engineering, Faculty of Engineering, University of Nigeria, Nsukka, Enugu State, Nigeria.

Authors' contributions

This work was carried out in collaboration among all authors. Authors MIN, OA and SCN developed the methodology and carried out the result analysis. Authors MIN, CCU and OCU managed the literature analyses. Authors MIN and SCN wrote the protocol and wrote the first draft of the manuscript. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/JERR/2023/v25i5908

Open Peer Review History:
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/101821

Original Research Article

Received: 12/04/2023
Accepted: 18/06/2023
Published: 03/07/2023

ABSTRACT

Aims: Though Networked Control Systems (NCS) have great benefits, such as remote control, low costs, and better flexibility, it also has several drawbacks such as network-induced time delay and packet loss, which may result in performance degradation and system instability in nonlinear

**Corresponding author: Email: Solomon.nwafor@unn.edu.ng;*

systems. This paper proposes a gain scheduling Proportional-Integral-Differential (GS-PID) controller design to address the performance degradation issues associated with NCS.
Study Design: TrueTime toolbox simulator in MATLAB is used for the NCS simulation and network comparison.
Place and Duration of Study: Department of Electronic and Computer Engineering, Faculty of Nigeria, University of Nigeria, Nsukka Enugu State, Nigeria, between March 2022 and March 2023.
Methodology: The performance of the proposed GS-PID and conventional PID controllers is compared through simulations on three different networks (Ethernet, Controlled Area Network (CAN), and switched ethernet).
Results: The GS-PID controller outperforms the traditional PID controller with improvements in settling time and overshoot ranging from 15% to 22%.
Conclusion: The results of the GS-PID over the conventional PID on these three networks highlight the effectiveness of the proposed approach and show potential in improving system performance.

Keywords: Networked control systems; gain scheduling; proportional integral derivative; ethernet; controlled area network; switched ethernet.

1. INTRODUCTION

The Networked Control System (NCS) has been a subject of significant academic and industrial research for many years due to its ability to provide remote control and enhance efficiency [1,2]. Traditional control systems, which are plagued by several drawbacks [1], have been replaced by NCS. However, NCS performance has its own set of problems, such as delays caused by the network, transmission delays, and lost packets [3]. Various issues arise when control systems are implemented over networks, ranging from the analysis and design of control networks [5] to the design of controllers capable of handling the effects of communication [6]. An overview of NCS, including system settings, network delay features, and networked delay outcomes, can be found in [7-12].

Research on NCS control systems has mainly focused on PID and optimized PID controllers [13-19]. For instance, [13] evaluated the performance of a nonlinear level control system using PID fuzzy supervision and PID gain scheduling. Meanwhile, [14] presented a performance evaluation of a PID controller for NCS using Ant Colony Optimization (ACO) and fuzzy logic. In [15], fuzzy logic (FL) with a PID controller was recommended for the design of Wireless NCS (WNCS), along with the use of the PSO technique and a ZigBee network for obtaining the best rules. [16] developed an NCS model with an Ethernet network for a third-order DC motor as the plant, and [17] suggested a fuzzy PID-like GS control method based on traditional PID, FL, and Gain scheduling. In [18], a controller technique for NCS was proposed that makes a GS-based state feedback integral controller with an integral action to handle disturbances that are not zero. Finally, [19] developed a way to schedule the limited number of connections in the NCS communication network using a probabilistic algorithm for designing scheduling logic and static state feedback controllers that meet the stability requirements of scheduling logic.

In this work, we looked into the behaviour of a nonlinear model controlled by a gain-scheduling PID controller in MATLAB, with data transmission over a communication network in TrueTime Simulator, to address the issue of network-induced delay in control systems. We compared three different communication networks - Ethernet, CAN, and Switched Ethernet - to observe how well the system works compared to a conventional PID controller. Section 2 describes the nonlinear system model and the system model formulation, followed by the networked control system for the work, which includes the induced delay, the TrueTime simulator, the PID controller design, and the gain scheduling scheme. Section 3 presents the simulation results and comparison for the different networks, and finally, section 4 summarizes the research work by discussing the model performance with respect to the three different networks [20].

2. METHODOLOGY

2.1 Nonlinear System Model

The schematic block diagram of the proposed NCS for a nonlinear model with induced network delay is illustrated in Fig. 1. The NCS effectively schedules the transmission of Input control from GS-PID and Output measurement (Amplitude) from the nonlinear model to meet the desired Amplitude, Ar. The weights of the PID controller are optimally tuned using gain scheduling technique to minimize the error difference between the measured output (amplitude) and the desired amplitude.

2.1.1 System model formation

Consider a continuous time-invariant nonlinear system that produces an amplitude, x(t) with a given control input, u(t) as defined in Eq. (1). The nonlinear system is assumed to operate without disturbance.

$$u(t) = \dddot{x}(t) + 3\ddot{x}(t) + 3\dot{x}(t) + x(t) \qquad (1)$$

The nonlinear model is reformulated into state-space [21] as:

$$\dot{x}(t) = A\vec{x}(t) + Bu(t) \qquad (2)$$

$$y(t) = Cx(t) \qquad (3)$$

The continuous time-invariant of Eq. (2) and Eq. (3) must be approximated by a discrete-time system because the nonlinear system is computer-based. This discrete-time system is described as follows [21]:

$$x(k+1) = A_d x(k) + B_d u(k) \qquad (4)$$

$$y(k) = Cx(k) \qquad (5)$$

Where

$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -3 & -3 \end{bmatrix}$,
$B = [0 \quad 0 \quad 1]^T$,
$C = [1 \quad 0 \quad 0]$, $\vec{x}(t) = [x_1 \quad x_2 \quad x_3]^T$,
$A_d = e^{A.Ts} \approx (I + A.Ts)$, $B_d = B.Ts$,
and Ts is the sampling time.

To evaluate the NCS of the nonlinear model shown in Fig. 1 through a communication network, we developed the nonlinear model, PID controller with a gain scheduler, and NCS model in MATLAB/SIMULINK, as shown in Fig. 2. The Truetime kernel and Truetime blocks developed by [22] were utilized by the NCS. The interference node, Controller node, and System node are the communication schedules for the SIMULINK model. The schedule uses Truetime Send and Truetime Receive to transmit and receive signals between the three nodes. The Interference node schedules interferences of data transmission between the three communication networks (Ethernet, CAN, and Switch Ethernet) and the Controller and the System model. As shown in Fig. 3, the system node receives an input signal via the Truetime Receive (Actuator) and outputs a signal via the Truetime Send (Sensor).

2.2 Networked Control System

2.2.1 Networked induced delay

A communication network facilitates the data transmission between the nonlinear system output of Eq. (1) and the controller and the transfer of data between the nonlinear system model and the controller. Therefore, the nonlinear system of Eq. (1) will encounter a network-induced delay, as stated in Eq. (6), because networks are involved in data transmission [23,24]. This delay could seriously impair the performance of the NCS and even cause systemic instability [25].

$$x(k+1) = A_d x(k) + B_d u(k - \hat{\tau}) \qquad (6)$$

Where $\hat{\tau}$ is the total delay, τ per sampling time, Ts. According to definitions in Eq. (7) [25,26], the total delay is calculated as the product of the controller to the nonlinear system (or actuator) delay, τ_{ca}, and the controller to nonlinear system output measurement (or sensor) delay, τ_{sc}.

$$\tau = \tau_{ca} + \tau_{sc} \qquad (7)$$

2.2.2 TrueTime simulator

The nonlinear system defined in Eq. (1) is controlled in real-time using the TrueTime [22] MATLAB/Simulink simulator, as demonstrated in Fig. 4. This powerful simulator allows for the simulation of network transmissions, model dynamics, and execution of controller tasks on real-time kernels. The block library includes the Network block, which enables nodes to communicate over a simulated network; the TrueTime Kernel block, which models a real-time kernel and performs user-defined tasks and interrupt handlers; and several standalone interface blocks. The kernel block, which represents a real-time kernel, performs a user-defined task and an interrupt handler. Nodes, also known as kernel blocks, can communicate over simulated networks using various network blocks, including wireless and wired options. Fig. 5 shows that the network scheduler selects three different networks, Ethernet, Control Area Network (CAN), and Switched Ethernet, each with unique parameters such as processing time delays, data transmission rate, frame overhead, and loss probability, which can be specified through the masked scheduler [20].

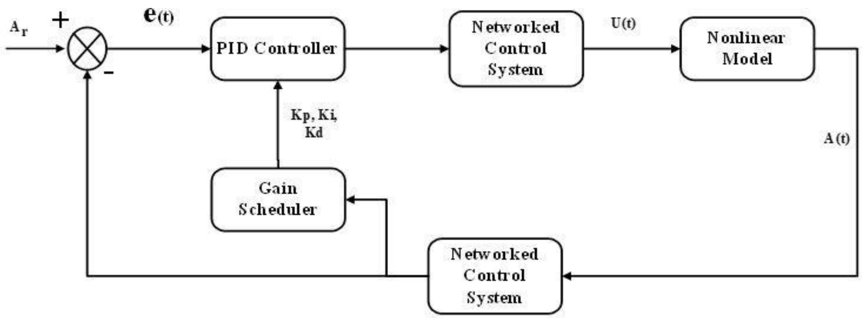

Fig. 1. The networked control system with GS-PID schematic

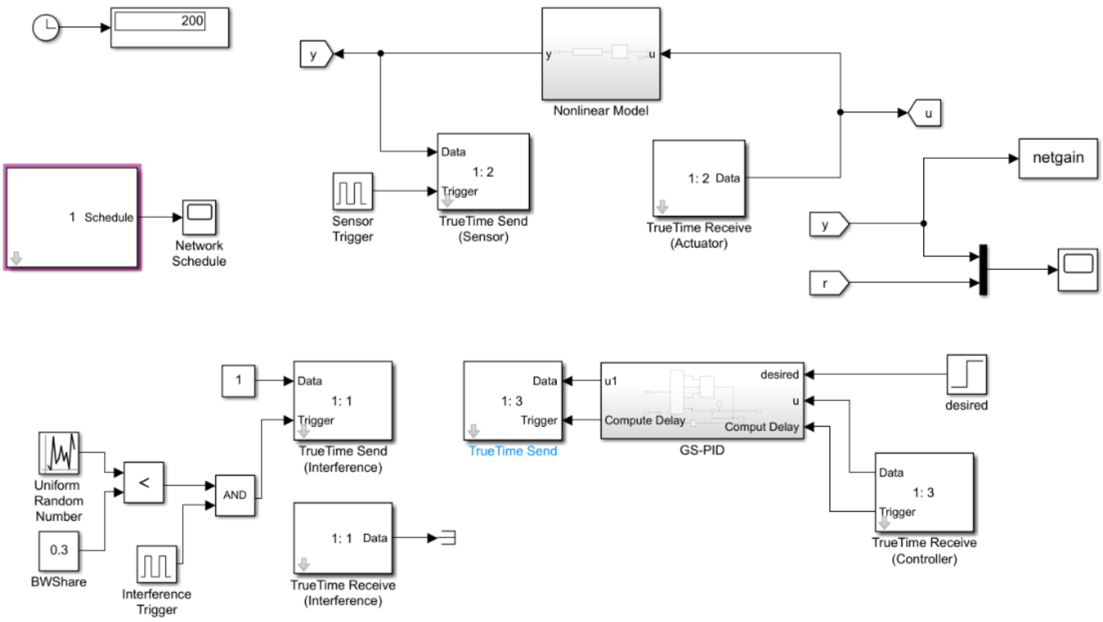

Fig. 2. Simulink model of the nonlinear system with GS-PID Networked control system

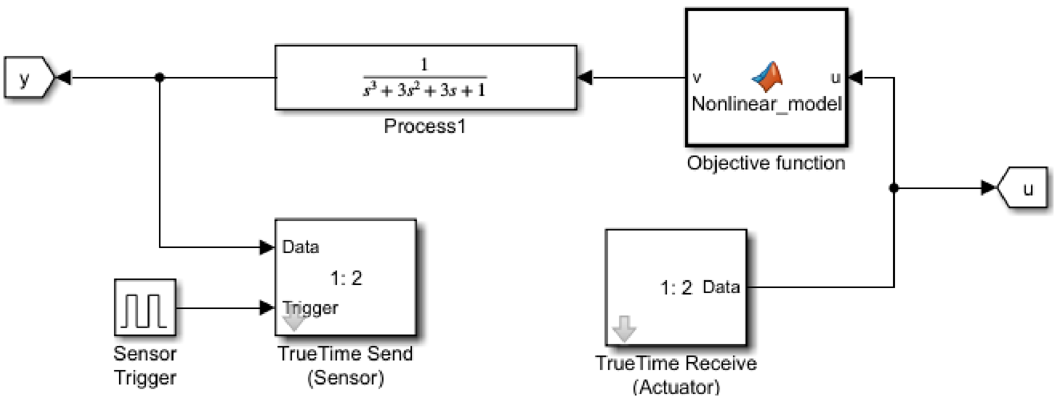

Fig. 3. The nonlinear system Simulink model

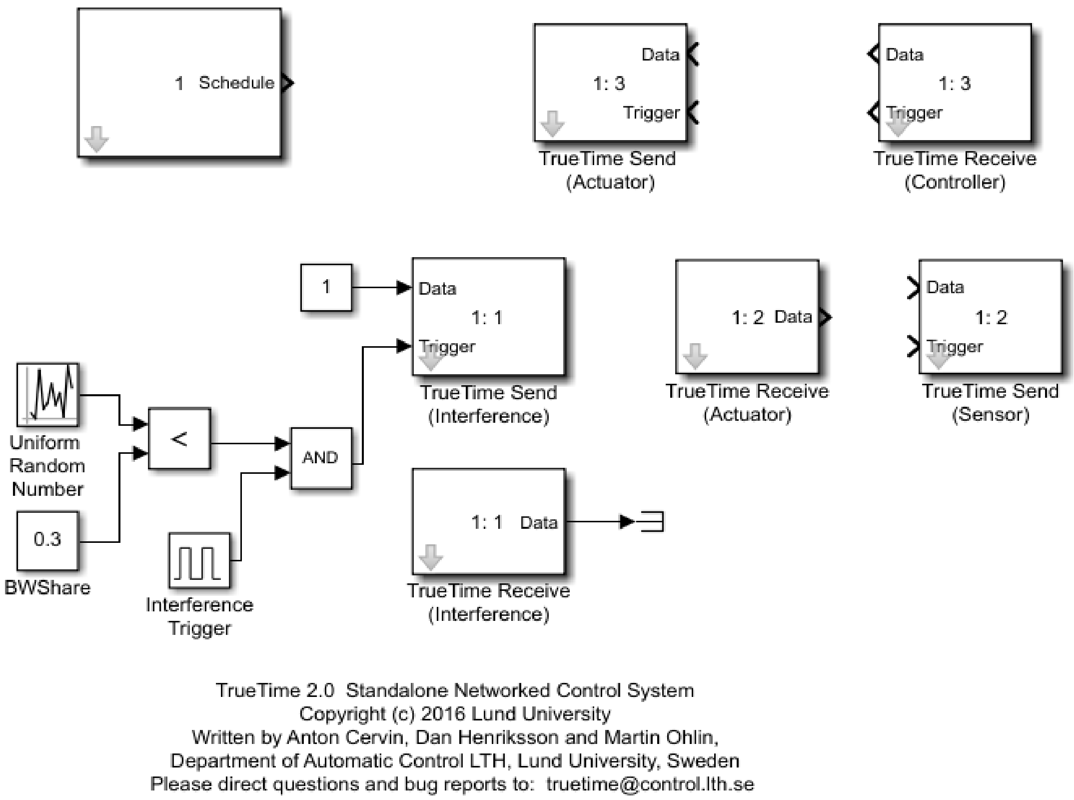

Fig. 4. Truetime standalone networked control system [23]

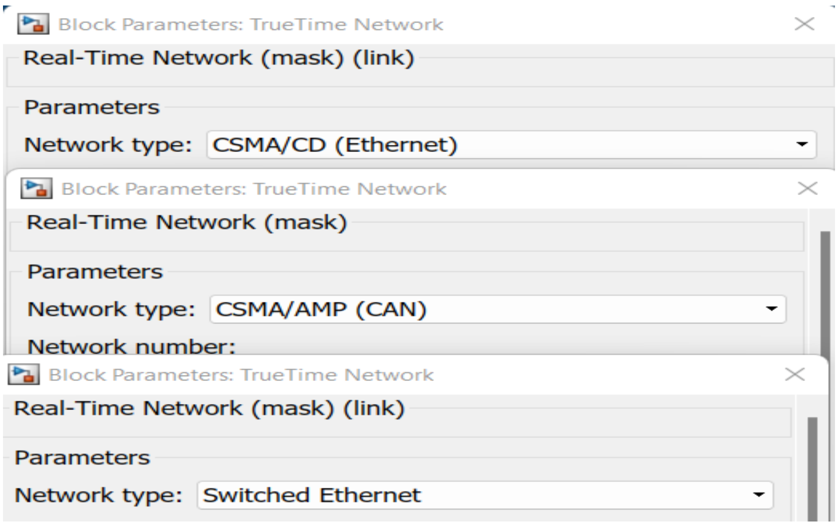

Fig. 5. Truetime block library

2.2.3 PID controller design

In Fig. 6, it can be observed that the PID controller, modeled using MATLAB/SIMULINK, produces the necessary input parameter for the nonlinear system model. This is achieved by minimizing the error between the measured amplitude from the nonlinear system and the desired amplitude, resulting in a value close to zero. The PID is not affected by delays [20] and is obtained from the discrete model equation (Eq. 6) as defined in Eq. (8).

$$u(k) = K_P e(t_K) + K_i \int e(t_K) + K_D \frac{d(e(t_K))}{dt_K} \tag{8}$$

The proportional gain, integral gain, and derivative gain are denoted by K_P, K_i, and K_D, respectively. The gain scheduler is used to optimize the tuning of these PID gains. The error term $e(t)$ is defined in Eq. (9).

$$e(t) = A_r - A \tag{9}$$

Where A_r is the desired amplitude and A is the measured amplitude from the nonlinear model.

2.2.4 Gain scheduling design

Fig. 6 demonstrates the use of Gain Scheduling (GS), which is a useful technique for addressing non-linear systems that have control requirements that vary with time or operating conditions. This approach is particularly effective in mitigating the negative effects of parameter changes that can impact the process dynamics. One key benefit of implementing GS is the ability to quickly adjust controller parameters in response to process changes [27]. However, a potential drawback of GS is that secondary measurements may not react to process changes quickly enough, which can be a limiting factor [28]. Furthermore, transitioning from one set of controller parameters to another may not always be smooth, potentially resulting in oscillatory behavior or even instability in the control system. In practice, the controller can use the parameter set closest to the intermediate set-point when the model operates between two points. As a result, GS may have a negative impact on the overall control effectiveness of the system [29]. For optimal performance of the proposed NCS in MATLAB, Table 1 presents the gain scheduler that should be employed.

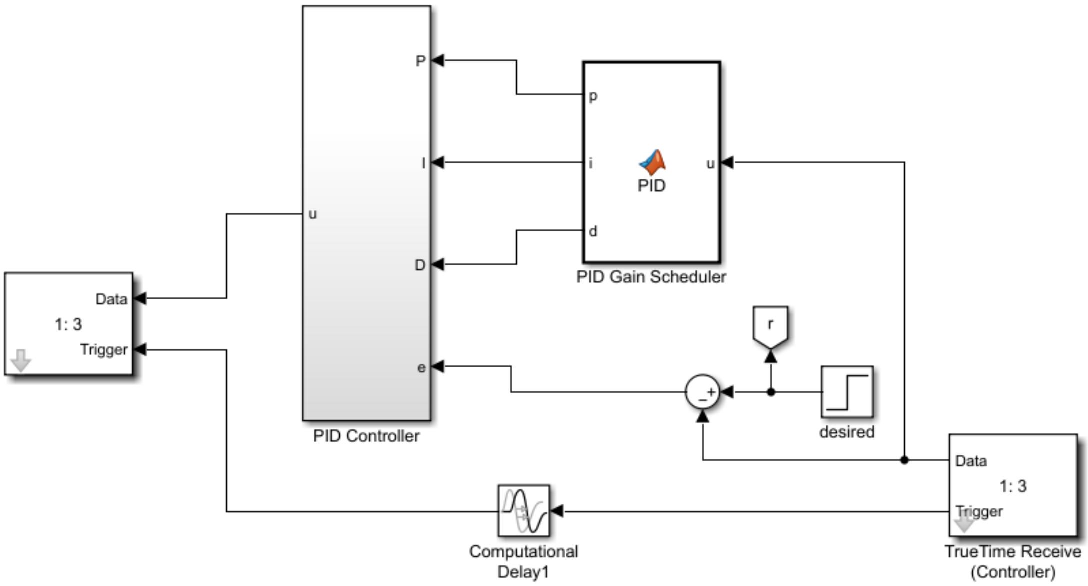

Fig. 6. Proposed Simulink Gain Scheduling PID control technique

Table 1. GS-PID MATLAB script algorithm

%Algorithm: PID Gain scheduling

```
function [Kp, Ki] = PI(u)
if u < 1
   Kp=u;
   Ki=0.15;
   Kd=0;
elseif u==1
   Kp=1/u;
   Ki=1/u/2.5;
   Kd=0;
else
   Kp=1/u;
   Ki=1/u/5.5;
   Kd=0;
end
```

3. RESULTS AND DISCUSSION

In the earlier section, the NCS was developed and tested using the parameters from Table 2 in MATLAB 2021a's SIMULINK library on an MSI CROSSHAIR 15 system with an i7 11th Gen processor, 16GB RAM, and RTX 3050 GPU. The NCS was simulated for three communication networks, namely Ethernet, CAN, and Switched Ethernet. Furthermore, the performance data of these networks was exported to Python libraries, MATPLOTLIB and SEABORN, for better analysis and visualization.

3.1 Ethernet Network

Fig. 7 presents the simulation results of a nonlinear model developed in (1) to track a reference amplitude using PID controller and GS-PID controller. The communication network used was ethernet and the simulation time was 200 seconds. At zero initial conditions, both controllers started tracking the reference of 1 amplitude, with the PID controller as a red dashed line and the GS-PID controller as a green dashed line. The PID-controlled nonlinear model showed a 70% overshoot, a peak time of 10 seconds, and a settling time of 113 seconds. On the other hand, the GS-PID model presented a 45% overshoot, a peak time of 8 seconds, and a settling time of 56 seconds. From the simulation results, it was observed that the PID with GS presented superior tracking of the reference amplitude compared to the conventional PID.

Table 2. NCS Setup parameters

Parameters	Values
Initial state, $x(0)$	0
Sample time, Ts	0.001 sec
Time delay, $\tau_{ca} = \tau_{sc}$	0.002 sec
Conventional Proportional gain, Kp	0.15
Conventional Integral gain, Ki	1
Conventional Derivative gain, Kd	0
GS-PID gains (Kp, Ki, Kd)	Automatically computed by the gain scheduling algorithm
Minimum frame size	80 bits
Switch memory	80,000 bits
Data rate	8,000 bits/sec

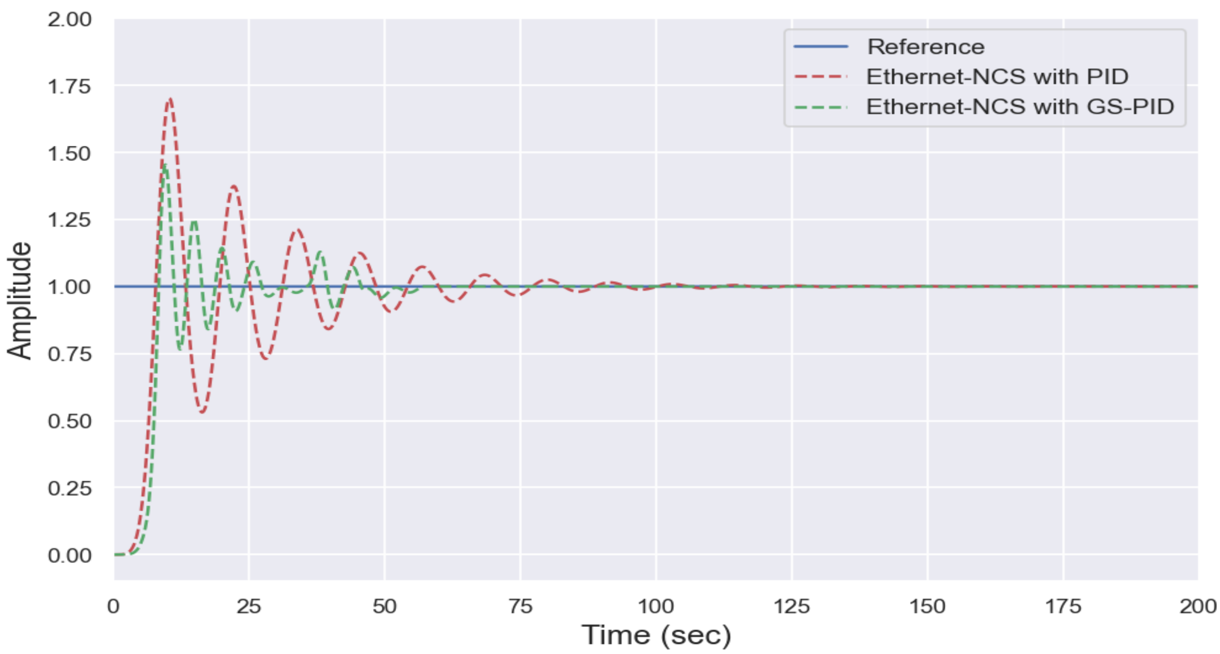

Fig. 7. Simulation output of NCS amplitude using ethernet communication network

In Fig. 8, two control inputs are presented; a conventional PID represented by a red dashed line, and a PID control with GS, depicted as a green dashed line. These are shown in relation to a simulation time of 200 seconds. The GS-PID control input oscillates less and settles at 80 seconds, while the PID control input oscillates more and finally stabilizes at an amplitude of 1 at 100 seconds. Moving on to Fig. 9, the Ethernet communication network schedule of a three-node system (Interference, controller, and system) is displayed. The left side of the result shows the Ethernet schedule with PID, while the right side shows the schedule with GS-PID. Based on simulation analysis, the schedule with PID has a broad range of scheduling intervals for the controller and system nodes, whereas the schedule with GS-PID has a smaller range of scheduling intervals.

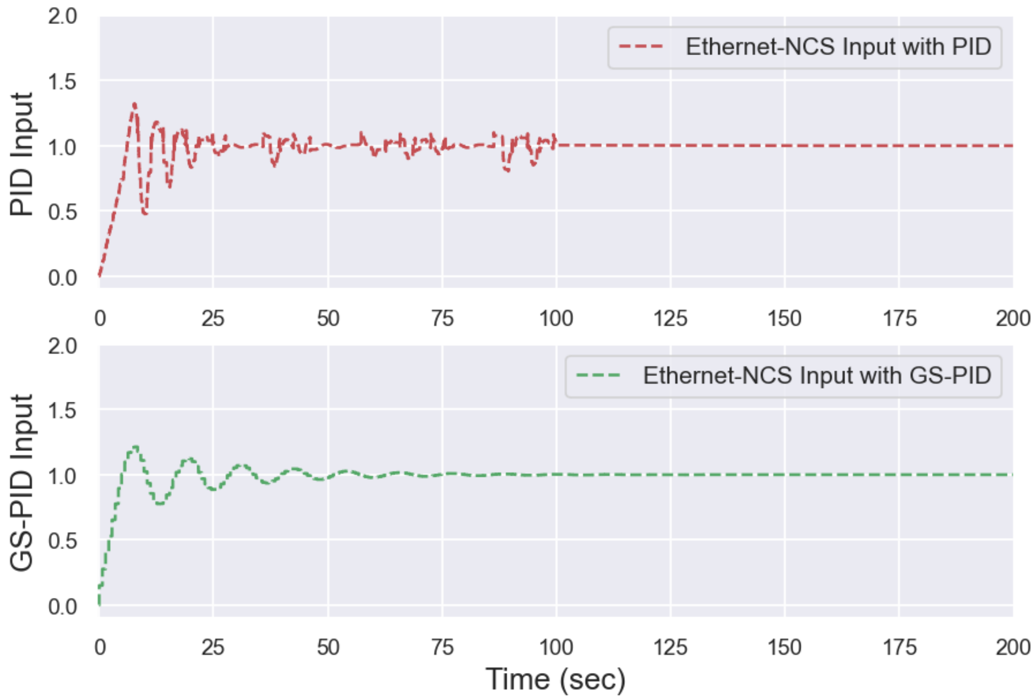

Fig. 8. Simulation of NCS control input using ethernet with respect to time

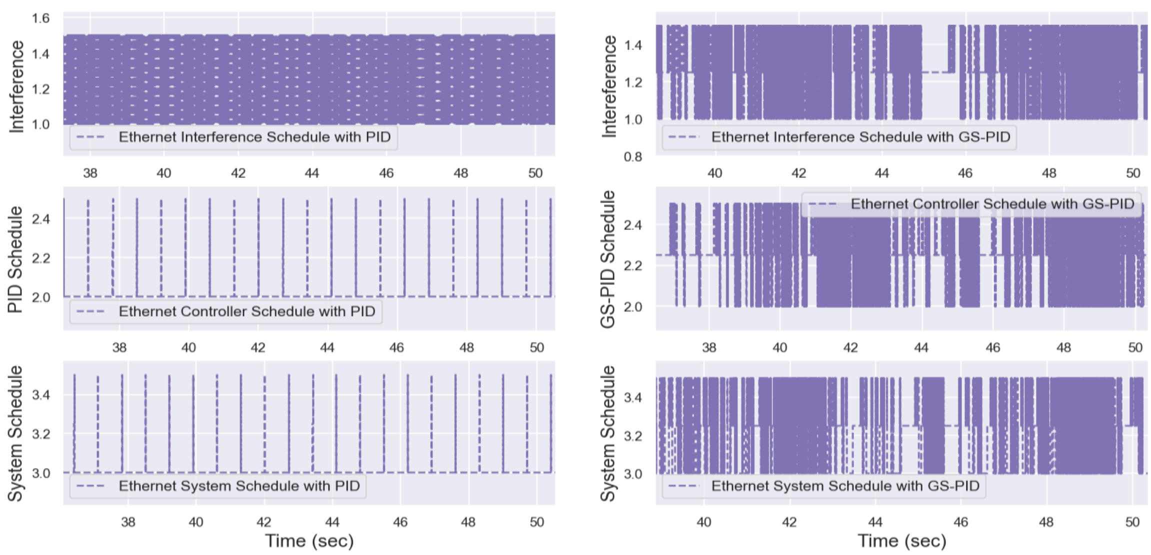

Fig. 9. The ethernet communication network schedule

3.2 Controlled Area Network (CAN)

In Fig. 10, we can observe the simulated reference tracking performance of two controllers - Amplitude with PID (indicated by the red dashed line) and GS-PID (indicated by the green dashed line) - using the CAN network for 200 seconds. As per the analysis, the PID controller had a 60% overshoot, a peak time of 11 seconds, and a settling time of 80 seconds. On the other hand, the GS-PID controller had a 45% overshoot, a peak time of 30 seconds, and a settling time of 8 seconds. It was concluded that the GS-PID controller performed better in tracking the reference amplitude compared to the traditional PID controller.

Fig. 10 clearly illustrates the reference tracking performance of the Amplitude model using PID (red dashed line) and GS-PID (green dashed line) controllers, simulated via the CAN network for 200 seconds. Our analysis unequivocally shows that the PID displays a 60% overshoot, a peak time of 11 seconds, and a settling time of 80 seconds, whereas the GS-PID shows a 45% overshoot, a peak time of 30 seconds, and a settling time of 8 seconds. Thus, it is evident that the GS-PID outperforms the conventional PID in terms of tracking the reference amplitude.

3.3 Switched Ethernet

In Fig. 13, the simulation shows the performance of two controllers, Amplitude with PID (red dashed line) and GS-PID (green dashed line), in reference tracking over a 200-second simulation time using a switched ethernet network. According to the results, the PID controller experienced a 65% overshoot, with a peak time of 10 seconds, and a settling time of 100 seconds. Meanwhile, the GS-PID controller had a 43% overshoot, with a peak time of 25 seconds, and a settling time of 8 seconds. In comparison, the GS-PID outperformed the conventional PID for switched Ethernet by 22% with an earlier settling time.

Fig. 14 displays the simulation results for the PID (red dash) and GS-PID (green dash) control input performances using a Switched Ethernet communication network at 200 seconds of simulation time. The PID controller achieved stable oscillations in 65 seconds, while the GS-PID controller had irregular oscillations and reached stabilization in about 25 seconds. Fig. 15 reveals that the switched Ethernet communication network schedule varied for the three nodes (Interference, controller, and system) concerning simulation time. For the PID with Switched Ethernet, the Interference has a packed schedule, while the Controller and Systems nodes have wide scheduling intervals. On the other hand, the Switch Ethernet network nodes for GS-PID have a more compact scheduling interval.

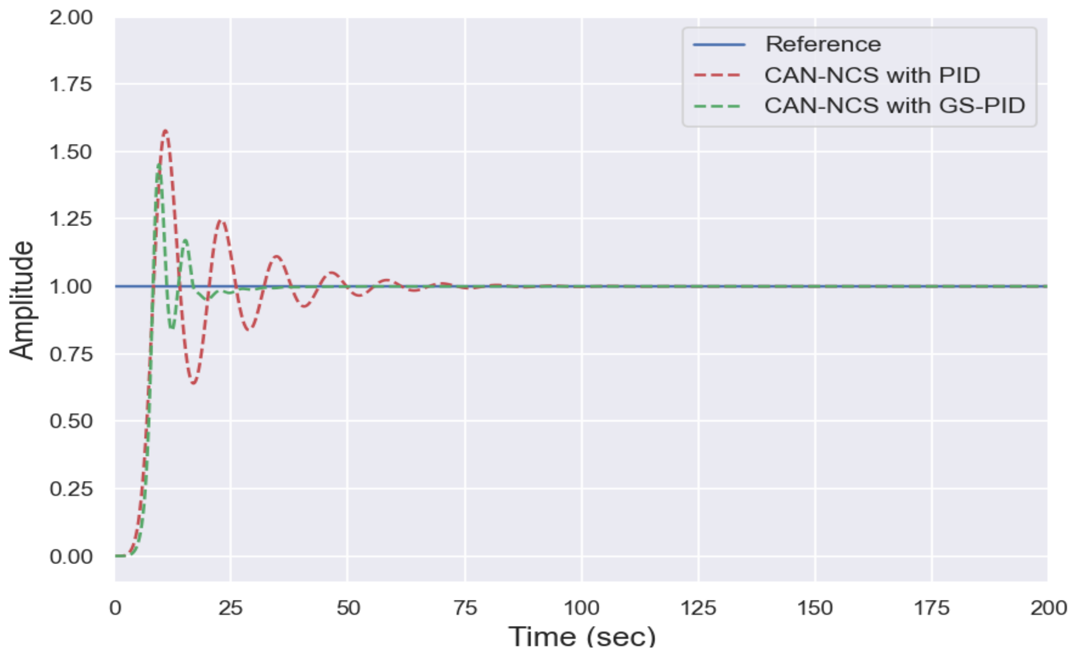

Fig. 10. Simulation output of NCS amplitude using CAN communication network

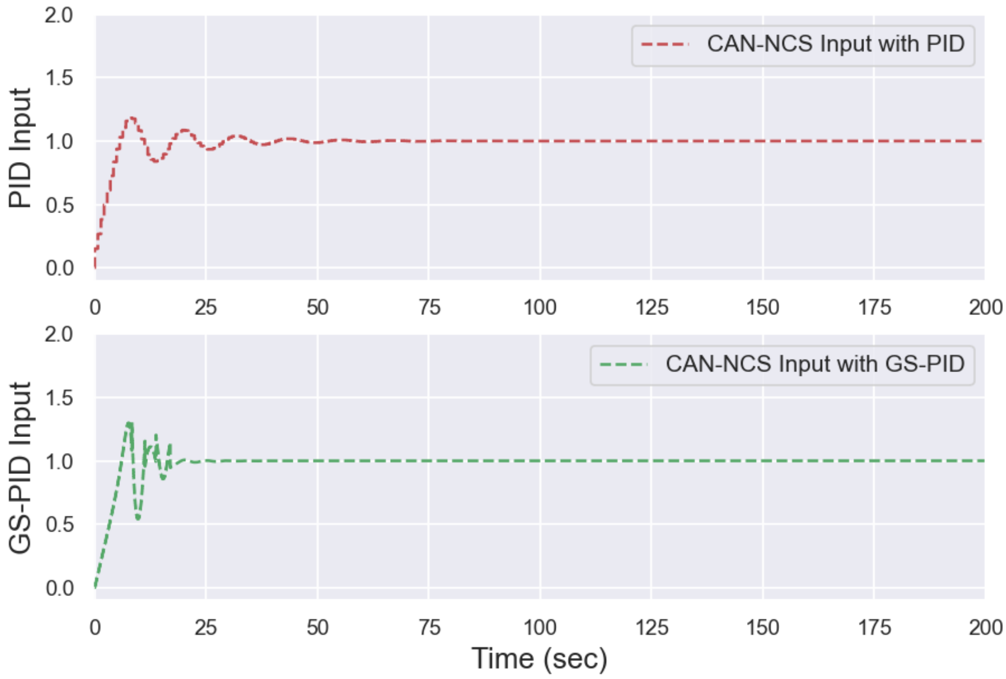

Fig. 11. Simulation of NCS control input using CAN with respect to time

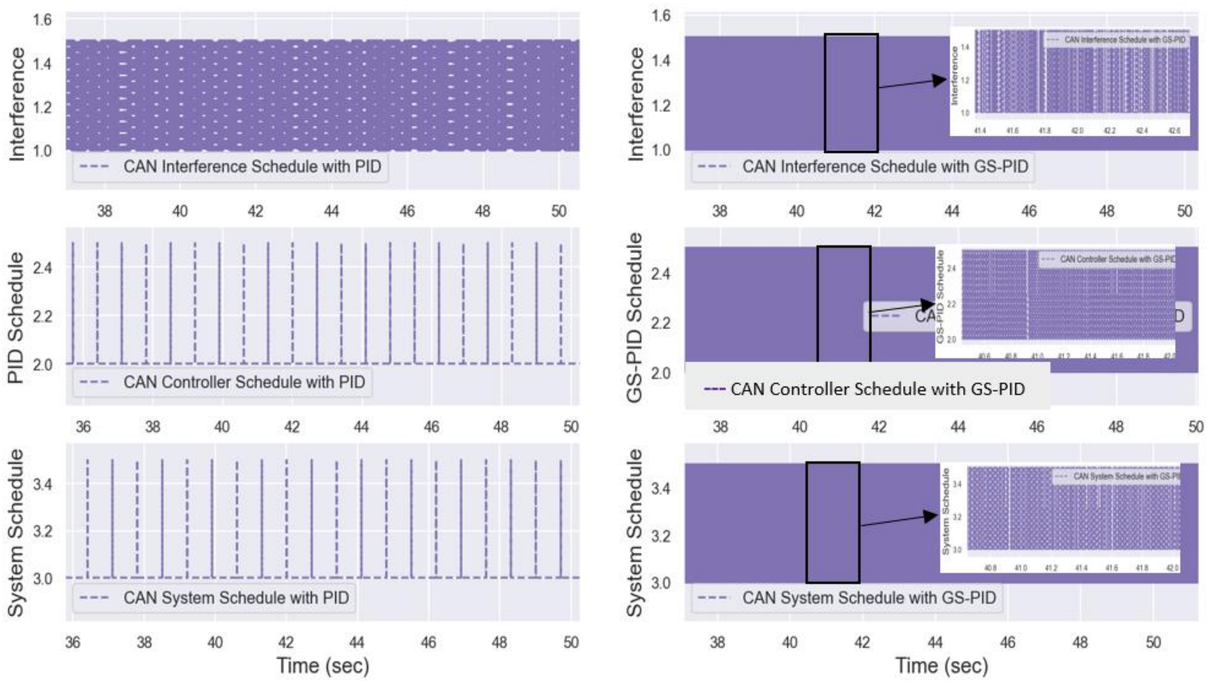

Fig. 12. The ethernet communication network schedule

3.4 Comparison

A comparison of the time delays induced by Ethernet, CAN, and Switched Ethernet networks in a networked controlled system with a PID controller and a GS-PID controller using TrueTime in MATLAB is presented in Table 3.

The PID controller exhibits high peak time, settling time, and overshoot values across all three network channels. However, the GS-PID controller substantially improves peak time and settling time while also reducing the overshoot percentage significantly.

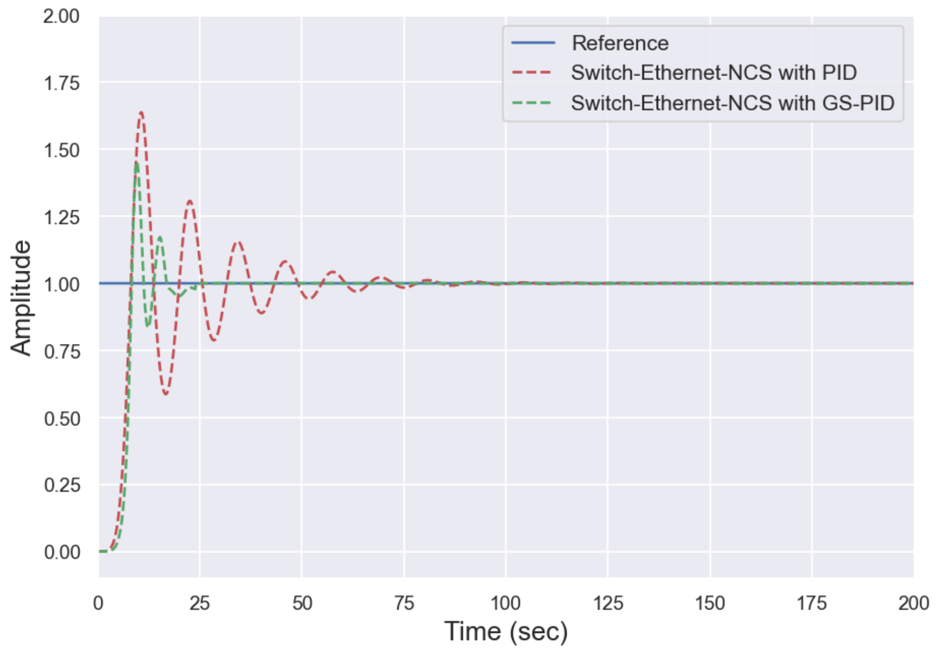

Fig. 13. Simulation output of NCS amplitude using switch ethernet communication network

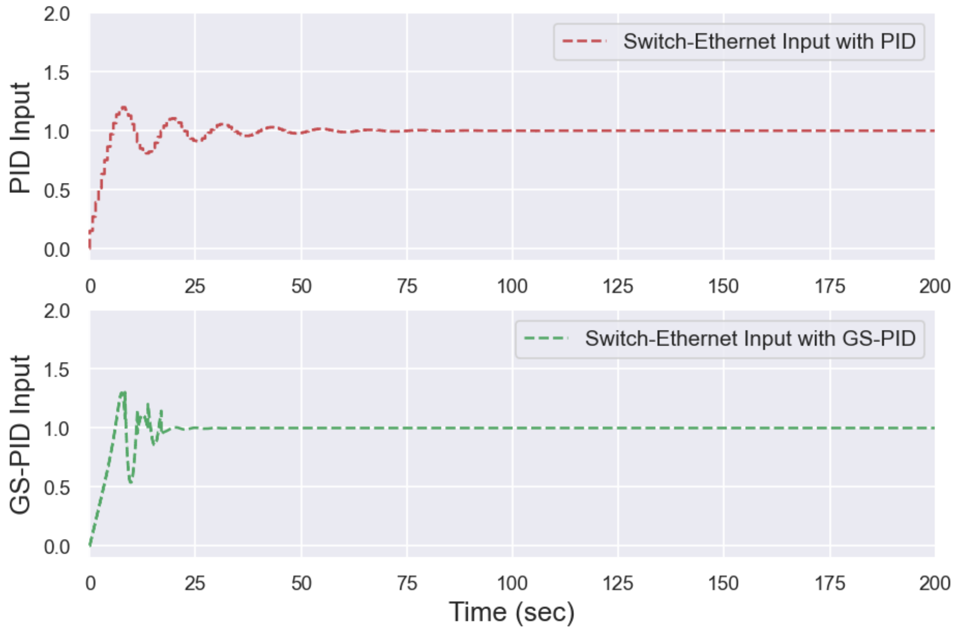

Fig. 14. Simulation of NCS control input using ethernet with respect to time

Table 3. Comparison of three different types of networks with GS-PID and PID controllers

Network	Controller	Peak Time	Settling Time	Overshoot
Ethernet	PID	10 seconds	113 seconds	70%
	GS-PID	8 seconds	56 seconds	45%
CAN	PID	11 seconds	80 seconds	60%
	GS-PID	8 seconds	30 seconds	45%
Switched Ethernet	PID	10 seconds	100 seconds	65%
	GS-PID	8 seconds	25 seconds	43%

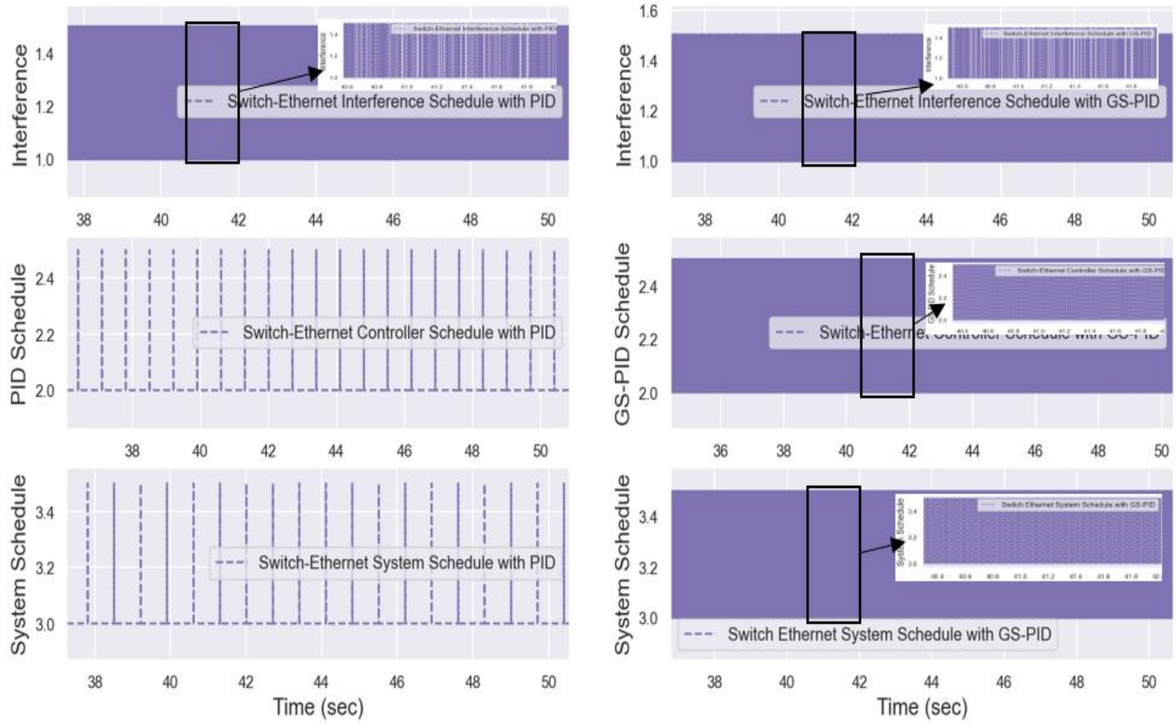

Fig. 15. The switched ethernet communication network schedule

4. CONCLUSION

In this study, a non-linear control system (NCS) model was developed using the TrueTime toolbox in MATLAB. The communication network consisted of three nodes: the interference node, the Controller node, and the System node. A gain-scheduling PID controller was used to control the non-linear model and compared to the conventional PID controller in three network communication channels: Ethernet, CAN, and Switched Ethernet. According to Table 3, the GS-PID controller outperformed the PID controller in settling time and overshoot by 57 seconds and 25%, respectively, in the Ethernet network. In the CAN network, the GS-PID controller had a settling time that was 50 seconds shorter and an overshoot that was 15% less than the PID controller. Finally, in the Switched Ethernet network, the GS-PID controller outperformed the PID controller with a settling time of 75 seconds and an overshoot of 22%. The Switched Ethernet GS PID controller had a smaller overshoot, while the CAN network's GS PID controller had a shorter settling time.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Gautam M K, Pati A, Mishra S K, Appasani B, Kabalci E, Bizon N, Thounthong P. A comprehensive review of the evolution of networked control system technology and its future potentials. Sustainability (Switzerland). 2021;13(5):1-39. Available:https://doi.org/10.3390/su13052962.
2. Huang S, Zhou C J, Yang S H, Qin Y Q. Cyber-physical system security for networked industrial processes. Int. J. Autom. Comput. 2015;12:567–578. Available:https://doi.org/10.1007/s11633-015-0923-9
3. Qi Q, Zhang H, Ji Z. Further results on stabilization for NCSs with packet losses and transmission delay: UDP case. J Franklin Inst. 2019;356(8):4601-4621. Available:https://doi.org/10.1016/j.jfranklin.2019.04.022
4. Bushnell L, Ye H. Networked Control Systems: Architecture and Stability Issues. In: Encyclopedia of Systems and Control. Springer London. 2015;835-842. Available:https://doi.org/10.1007/978-1-4471-5058-9-151
5. Park P, Ergen SC, Fischione C, Lu C, Johansson K H. Wireless Network Design

for Control Systems: A Survey. IEEE Communications Surveys and Tutorials. 2018;20(2):978-1013. Available:https://doi.org/10.1109/COMST.2017.2780114

6. Zhao Y B, Sun X M, Zhang J, Shi P. Networked Control Systems: The Communication Basics and Control Methodologies. Math Probl Eng. 2015; 2015:1-9. Available:https://doi.org/10.1155/2015/639793
7. Gupta R A, Mo-Yuen C. Networked Control System: Overview and Research Trends. IEEE Transactions on Industrial Electronics. 2010;57(7):2527-2535. Available:https://doi.org/10.1109/TIE.2009.2035462
8. Dallolio A, Øveraas H, Johansen T A. Gain-scheduled steering control for a wave-propelled unmanned surface vehicle. Ocean Engineering. 2022;257:111618. Available:https://doi.org/10.1016/j.oceaneng.2022.111618
9. Bett C J. Gain-Scheduled Controllers. In: The Electrical Engineering Handbook. Elsevier. 2005;1107-1114. Available:https://doi.org/10.1016/B978-012170960-0/50086-4
10. Gao J, Budman HM. Design of robust gain-scheduled PI controllers for nonlinear processes. J Process Control. 2005;15(7):807-817. Available:https://doi.org/10.1016/j.jprocont.2005.02.003
11. Vesel´y V, Ilka A. Gain-scheduled PID controller design. J Process Control. 2013;23(8):1141-1148. Available:https://doi.org/10.1016/j.jprocont.2013.07.002
12. Park B, Nah J, Choi JY, Yoon IJ, Park P. Robust Wireless Sensor and Actuator Networks for Networked Control Systems. Sensors. 2019;19(7):1535. Available:https://doi.org/10.3390/s19071535
13. Hamdani C N, Adi W W, Anggriawan D O. Comparative Study on Fuzzy Supervision and Gain Scheduling for Nonlinear Level Control System. In: Atlantis Press. 2021;493-497. Available:https://doi.org/10.2991/aer.k.210810.085
14. Eke-Nnodu CO, Udeze CC, Chukwudozie C, Ndefo MI, Ugwu HI. Performance Evaluation of an Optimized PID Controller for Networked Control System Using ANT Colony Optimization. Iconic Research And Engineering Journals. 2021;5(5):183-191.
15. Salman RS, Abdulsadda AT. Adaptive Gain Scheduling Based controller for Networking Control Systems. Al-Furat Journal of Innovations in Electronics and Computer Engineering. 2020;1(2):1. Available:https://doi.org/10.46649/110420-01
16. Popovi´c N, Naumovi´c M. Networked and cloud control systems - modern challenges in control engineering. IJEEC - International Journal of Electrical Engineering and Computing. 2019;2(2). DOI: 10.7251/IJEEC1802091P
17. Salim L, Awad OA. Gain Scheduling PID Controller for Networked Control System with Random Delay. 2015;4. Available:www.erpublications.com
18. Li H, Sun Z, Chow MY, Sun F. Gain-Scheduling-Based State Feedback Integral Control for Networked Control Systems. IEEE Transactions on Industrial Electronics. 2011;58(6):2465-2472. Available:https://doi.org/10.1109/TIE.2010.2066533
19. Singh M, Kundu A. A scheduling algorithm for networked control systems. IFAC-PapersOnLine. 2022;55(13):37-42. Available:https://doi.org/10.1016/j.ifacol.2022.07.232
20. Kalaivani CT, Kalaiarasi N. Earliest deadline first scheduling technique for different networks in network control system. Neural Comput Appl. 2019;31(S1):223-232. Available:https://doi.org/s00521-018-3729-1
21. Chaparro L F, Akan A. Discrete-Time Signals and Systems. In: Signals and Systems Using MATLAB. Elsevier; 2019:487-557. DOI: 10.1016/B978-0-12-814204-2.00020-X
22. Cervin A, Henriksson D, Ohlin M. TRUETIME 2.0 – Reference Manual. Available:http://archive.control.lth.se/media/Research/Tools/TrueTime/report_2016-02-10.pdf
23. Ghosh A B, Lal Priya P S. Adaptive Discrete Sliding Mode Controller for Networked Control Systems. In: 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET). IEEE. 2018;1-5.
24. Juanole G, Mouney G, Sauter D, Aubrun C, Calmettes C. Decision making

improvement for diagnosis in Networked Control Systems based on dynamic message scheduling. In: 18th Mediterranean Conference on Control and Automation, MED'10. IEEE. 2010;280-285. DOI: 10.1109/MED.2010.5547681

25. Phyo AN, Min NZ, Mo M WM, Myo TH. Simulation of wireless networked control system using Truetime and Matlab. International Journal of Scientific & Technology Research. 2016;5(06). Available:https://www.ijstr.org/final-print/june2016/Simulation-Of-Wireless-Networked-Control-System-Using-Truetime-And-Matlab.pdf
26. Shah D, Mehta A. Discrete-time sliding mode controller subject to real-time fractional delays and packet losses for networked control system. Int J Control Autom Syst. 2017;15(6):2690-2703. DOI: 10.1007/s12555-016-0761-8
27. Magalhães S C, Fontella G, Borges R F O, Vega M P, Calçada L A, Scheid C M. Construction of software using gain-schedule/auto-adaptive control strategy for automation in drilling fluid production plants. J Pet Explor Prod Technol. 2020;10(2):501-514. doi:10.1007/s13202-019-00746-x
28. Bagyaveereswaran V, Arulmozhivarman P. Gain Scheduling of a Robust setpoint tracking disturbance rejection and aggressiveness Controller for a Nonlinear Process. Processes. 2019;7(7):415. DOI: 10.3390/pr7070415
29. Tipsuwan Y, Chow MY. Gain Scheduler Middleware: A Methodology to enable existing controllers for networked control and teleoperation—Part I: Networked Control. IEEE Transactions on Industrial Electronics. 2004;51(6):1218-1227. DOI: 10.1109/TIE.2004.837866

Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle5.com/review-history/101821