International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 12, No. 10, October, 2025, pp. 3461-3467
Journal homepage: http://iieta.org/journals/mmep

Multi-Verse Optimizer Based Classical and Nonlinear PI Controller Design for TCP/AQM System: A Comparative Study

Dhiaa Z. Mohamed[1], Huthaifa Al-Khazraji[1*], Rawaa Al-Majeez[1], Alaq F. Hassan[2], Amjad J. Humaidi[1]

[1] College of Artificial Intelligence Engineering, University of Technology-Iraq, Baghdad 10066, Iraq
[2] Technical Engineering College, Middle Technical University, Baghdad 10074, Iraq

Corresponding Author Email: 60141@uotecnology.edu.iq

https://doi.org/10.18280/mmep.121012

Received: 4 June 2025
Revised: 18 July 2025
Accepted: 25 July 2025
Available online: 31 October 2025

Keywords:
TCP/AQM, controller design, PI controller, transmission control protocol

ABSTRACT

Due to the increase in the utilization of the internet, networks are subject to congestion. As a result, issues such as packet delay, packet loss, and buffer overflow may arise in intermediate routers. To mitigate these problems, the Transmission Control Protocol combined with Active Queue Management (TCP/AQM) has been introduced. The primary objective of AQM is to manage network congestion and enhance overall performance. In this paper, a comparative study between the classical and the nonlinear PI controllers based on the Multi-Verse Optimizer (MVO) is conducted to enhance the AQM system performance. The simulation study is conducted using the MATLAB program. Simulation results demonstrate that the TCP/AQM system utilizing the nonlinear PI controller successfully tracks the desired step input with zero steady-state error, whereas the classical PI controller results in a steady-state error of 0.748. Besides, the numerical results show that the settling time of the nonlinear PI controller is reduced by 28.6% compared to the classical PI controller. Moreover, the findings demonstrate that under a disturbance scenario, the nonlinear PI controller reduces the recovery time by 22.22% compared to the classical PI controller. These findings indicate that the nonlinear PI controller achieves stable system performance.

1. INTRODUCTION

The internet becomes a necessary tool for performing numerous operations using Web applications, such as data transmission, reception, etc. However, network congestion has increased due to the rise in circulating data and internet users [1, 2]. Time delays, packet loss in packet delivery, and buffer overflow in intermediate carriers are some of the fundamental issues with using Internet networks [1]. The need to enhance the network performance leads to establishing the Transmission Control Protocol combined with Active Queue Management (TCP/AQM). Floyd and Jacobson developed Random Early Detection (RED) in 1993, which then became the first scheme of the AQM [3]. In the context of computer networks, the term congestion control refers to a time-varying, nonlinear, difficult problem that requires a robust controller to achieve good, resilient, efficient network performance. Using control theory to examine and analyze the dynamic behavior of numerous systems, then design a suitable controller was the choice of many authors to improve the performance of the systems [4-10]. In this direction, considerable contributions have been developed for TCP/AQM systems to avoid congestion in data traffic of computer networks and to increase the network utilization by organizing queues at network bottlenecks.

A detailed review was conducted by Mahawish and Hassan [11], focusing on the classification of AQM techniques based on queue length, queue delay, or a combination of both. Their work introduced tailored TCP/AQM strategies aimed at enhancing system performance. In another study [11], a modified RED-exponential approach was proposed, integrating AQM with optimization-based congestion control through intelligent algorithms to improve network efficiency. By using this method, different types of service flows are managed successfully due to integrate the nonlinear packet dropping behavior [12]. To address the challenges of heterogeneous traffic and enable future queue length prediction, numerous intelligent controllers have been applied. For example, Bisoy and Pattnaik [13] introduced an AQM model utilizing a feed-forward neural network. The stability of the system is achieved by using an adaptive learning where the sign activation function is used to update weights. Reinforcement learning approach is presented by Gomez et al. [14]. The decision-making process was built on a Markov Decision Process framework, which accounted for various congestion states and corresponding actions to optimize performance.

Integrating classical and intelligent controllers was performed by Gomez et al. [14] where a fuzzy PI controller optimized via Genetic Algorithm (GA) was developed to improve PI controller efficiency in AQM systems, aiming to mitigate network congestion. Similarly, the Ant Colony Optimization (ACO) algorithm was employed by Sulttan et al. [15] for fine-tuned the parameters of a PID controller. A comparative study [16] is conducted including a traditional H∞ controller, a PSO-optimized PID controller, and an ACO-optimized PID controller. The outcomes revealed that the ACO-based PID controller outperformed the other controller approach. To further enhance the performance of congestion system, Ali et al. [17] examined the performance of both traditional PID and Fuzzy-PID controllers applied to the AQM system. Their results show that the Fuzzy-PID approach has superior performance that the traditional PID. A Fuzzy-PID congestion control system was introduced [1, 18, 19] for adjusting PID parameters based cuckoo algorithm to achieve the best results on congestion when using a different type of service flow over the network. To improve the tuning of AQM parameters, several studies have integrated multiple optimization algorithms. For instance, a fuzzy PI controller optimized via a GA was proposed by Oudah et al. [20], effectively enhancing queue length performance. Shneen et al. [2] conducted a comparison between interval type-2 and type-1 fuzzy PID controllers combined with Particle Swarm Optimization (PSO), as well as other algorithms such as Social Spider Optimization (SSO) and Ant Colony Optimization (ACO) to optimize PID gains. Among these, the interval type-2 fuzzy PID controller with SSO produced the most favorable results [21]. To achieve improved stabilization, reduced settling time, and minimal delay, Sabry and Nayl [22] introduced a linear quadratic servo controller tuned using the PSO algorithm within the AQM framework. The utilization of intelligent control methods including fuzzy logic and neural networks has grown as a tuning process to optimize PID parameters such that the performance of the congestion system is enhanced. In this context, a comparative analysis by Al-Majeed and Saud [23] was conducted to examine four different controllers to address network congestion issues, while predictive control techniques were introduced by Humaid et al. [24]. To end this, the main contribution of this work can be listed as follows:

- Evaluation of the performance of the nonlinear PI in compared to the classical PI controllers is presented, emphasizing improved performance under uncertainty.
- The Multi-Verse Optimizer (MVO) is introduced as tuning process to adjust design parameters of the proposed controllers in terms of enhancing system robustness, achieving fast settling time, and minimizing fluctuations during variable traffic conditions in the AQM system.

The structure of this paper is outlined as follows: Section 2 introduces the TCP/AQM model. Section 3 discusses the proposed controller designs. Section 4 details the MVO method. Section 5 presents the results obtained from simulation-based evaluations, and Section 6 concludes the study with key findings.

2. MATHEMATICAL MODEL OF TCP/AQM

The core concept of the congestion system consists of bidirectional relationship of a TCP/AQM system as depicted in Figure 1. The dynamic behavior of TCP/AQM has been modeled based on the fluid-flow theory and using two coupled stochastic differential equations, with the TCP timeout mechanism excluded from the analysis, as outlined in studies [24, 25]:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))}P\big(t-R(t)\big) \tag{1}$$

$$\dot{q}(t) = \frac{W(t)}{R(t)}N(t) - C(t) \tag{2}$$

In this context, W denotes the expected TCP window size in packets, q represents the queue length in packets, t stands for time in seconds, and R is the round-trip time in seconds. The variable N refers to the number of active TCP sessions, C indicates the link capacity in packets per second, and P corresponds to the packet marking or dropping probability. The marking probability P ranges between 0 and 1. Additionally, both the queue length q and the window size W are positive and constrained within upper bounds, (i.e., $q \in [0,\bar{q}]$ and $W \in [0,\bar{W}]$), $\bar{q}$ denotes the buffer's maximum capacity, while $\bar{W}$ indicates the peak window size. The nonlinear differential equations governing the AQM system can be linearized around a specific operating point (W_0, q_o, P_o) such that $\dot{W} = 0$ and $\dot{q} = 0$. It also assumes that the number of TCP sessions and the link capacity are constant (i.e., $N(t) \cong N$ and $C(t) \cong C$). Based on that, the resulting linearized transfer function as follows [25]:

$$P(s) = \frac{\delta q(s)}{\delta p(s)} = \frac{\frac{C^2}{2N}e^{-sR_o}}{\left(s+\frac{2N}{{R_o}^2C}\right)\left(s+\frac{1}{R_o}\right)} \tag{3}$$

where, $R_o = \frac{q_o}{C_o} + T_p$, $P_o = \frac{2N^2}{{R_o}^2C^2}$ and $W_0 = \frac{R_oC}{N}$, T_p is the propagation delay. The symbol δ signifies a minor deviation in the variables relative to their nominal values used during the linearization process.

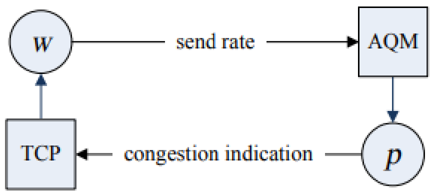

Figure 1. Bidirectional relationship of a TCP/AQM system

3. CONTROLLER DESIGN

The main goal of designing a PI controller is to ensure system stability and drive the output toward the desired setpoint. The controller's output (u) is formed by combining two components: the proportional and the integral terms, as illustrated in Figure 2(a). The process output (y) is measured and compared with the reference value (y_r) to compute the error (e). The proportional part modifies u In proportion to the error using the gain K_p, while the integral part adjusts u based on the accumulated error over time, scaled by the gain K_i. The PI controller's transfer function is expressed as follows [26]:

$$u(t) = K_pe(t) + K_i\int e(t)dt \tag{4}$$

To address the shortcomings of the traditional PI controller, various improved structures have been introduced. In this study, a NPI controller as shown in Figure 2(b) is proposed by substituting the standard error integration in Eq. (4) with the integration of the arctangent function of the error. The arctan function can aid in stabilizing systems with time delays. It also provides a smooth transition in control action, which can be advantageous in systems requiring gradual changes to avoid abrupt responses. The control law of the NPI controller is expressed as follows [27]:

$$u(t) = K_p e(t) + K_i \int tan^{-1}\big(\lambda e(t)\big)\, dt \tag{5}$$

where, the terms λ is a design coefficient.

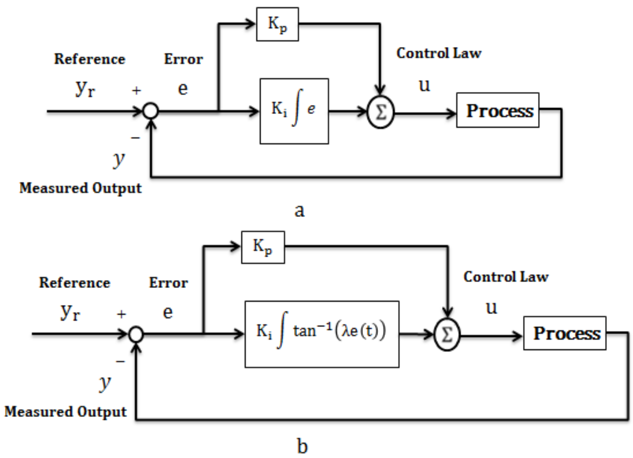

Figure 2. (a) System with the PI controller; (b) System with the NPI controller

4. MULTI-VERSE OPTIMIZER

Swarm optimization methods are a class of optimization algorithms capable of dealing with complex and challenging engineering problems. These algorithms are motivated by natural processes, social dynamics, and improve version of the heuristic strategies [28-30]. In the field of control systems, tuning controller parameters to produce effective control signals that achieve the desired system behavior is a non-trivial task. Many researchers use swarm-based optimization algorithms instead of the traditional trial-and-error techniques because these algorithms offer more efficient and reliable solutions for find the optimal controller settings [31-33].

The MVO is a swarm-based algorithm inspired by the multi-Verse theory in cosmology. Three key concepts are used to describe the algorithm, which are white holes, black holes, and wormholes. These three concepts used to establish the balance exploration, exploitation, and local search within the optimization process. MVO has shown promising results in addressing both standard benchmark functions and practical engineering optimization challenges [34].

Within the concept of the MVO algorithm, each agent search process represents a universe, where in terms of optimization concept represents a possible solution. The quality a universe is measured by its inflation rate. Under this assumption, higher inflation rates indicate better quality. Universes with higher inflation rates contain white holes, enabling them to transfer variables (objects) to others, whereas those with lower rates feature black holes, making them more likely to receive these variables. To mathematically describe this exchange mechanism, a roulette wheel selection is used. During each iteration, white hole is determined by the universes with the high inflation rate and is selected using the roulette wheel. The procedure continues as follows, assuming that:

$$U = \begin{bmatrix} x_1{}^1 & x_1{}^2 & \dots & x_1{}^d \\ x_1{}^1 & x_2{}^2 & \cdots & x_2{}^d \\ \vdots & \vdots & \vdots & \vdots \\ x_n{}^1 & x_n{}^2 & \cdots & x_n{}^d \end{bmatrix}$$

where, d represents the total number of design parameters (or decision variables), and n denotes the number of universes, which correspond to the candidate solutions:

$$x_j{}^i = \begin{cases} x_j{}^k & if\ r_1 < NI(U_i) \\ x_j{}^i & otherwise \end{cases} \tag{6}$$

In Eq. (6), $x_j{}^i$ refers to the j parameter of the i universe, U_i denotes the i universe itself, and $NI(U_i)$ is the normalized inflation rate of that universe. The term r_1 refers to a randomly generated value within the interval [0, 1], while $x_j{}^k$ signifies the j parameter of a universe k, which is chosen using a roulette wheel selection method. This selection strategy enables stronger solutions to guide the evolution of weaker ones, while preserving diversity among the candidate solutions.

Wormholes facilitate exploitation by allowing objects to randomly teleport across universes, independent of inflation rates. This ensures that the best solutions guide the search without restricting the exploration capability. The wormhole update rule is given by [35]:

$$x_j{}^i = \begin{cases} \begin{cases} x_j + TRD * (ub_j - lb_j) * r_4 + lb_j & \text{if } r_3 < 0.5 \\ x_j - TRD * (ub_j - lb_j) * r_4 + lb_j & \text{otherwise} \end{cases} & r_2 < WEP \\ x_j{}^i & \text{otherwise} \end{cases} \tag{7}$$

where, x_j denotes the current solution, and Traveling Distance Rate (TRD) compute the magnitude of the step size. The terms ub_j and lb_j represent the upper and lower bounds of the variable j respectively. Moreover, r_2,r_3, r_4 are random values distributed uniformly between [0, 1]. To balance the searching process between the exploration search and exploitation search, MVO integrates two adaptive parameters. One of these is the Wormhole Existence Probability (WEP), which defines the chance of performing a wormhole-based teleportation:

$$WEP = min + \frac{l * (max - min)}{L} \tag{8}$$

where, l is the current iteration and L is the total number of iterations. The TRD controls the movement precision towards the best solution [36]:

$$TRD = 1 - \left(\frac{l^{\frac{1}{p}}}{L^{\frac{1}{p}}}\right) \tag{9}$$

where, p determines how quickly the focus shifts from exploration to exploitation. The effectiveness of MVO is derived from exploration via white and black holes, ensuring global search, exploitation via wormholes, guiding local search around promising solutions, and adaptive parameter tuning, ensuring a smooth transition from exploration to exploitation.

The MVO is employed to optimize the tuning parameters of PI and NPI controllers based on minimizing the Integral of Absolute Error (IAE) between the controller's output and the desired reference signal.

5. SIMULATION RESULTS AND DISCUSSION

The value of the physical element of the TCP/AQM system is reported in Table 1. The implementation of the TCP/AQM system with proposed controllers was simulated in MATLAB/Simulink program, utilizing the "ode45" solver for numerical integration. The initial value of the queue size was set to 0 packets, and the desired size was set to 300 packets. The results of the optimization process based on the MVO algorithm for the design variables of each controller are given in Table 2. The tracking performance of the TCP/AQM system in reaching the target packet size is shown in Figure 3. Figure 4 and Figure 5 present the control signals and error signals generated by the respective controllers. A comparison of the system's performance using the NPI and conventional PI controllers is provided in Table 3.

Table 1. The parameters of TCP/AQM system

Parameter	Value	Unit
N	60	-
C	3750	packet/s
R_o	0.253	-

Table 2. The proposed values of the design parameters for NPI and PI by the MVO algorithm

Controller	k_p	k_i	λ
NPI	3e-05	3.2598e-04	40
PI	3e-05	1.77e-05	-

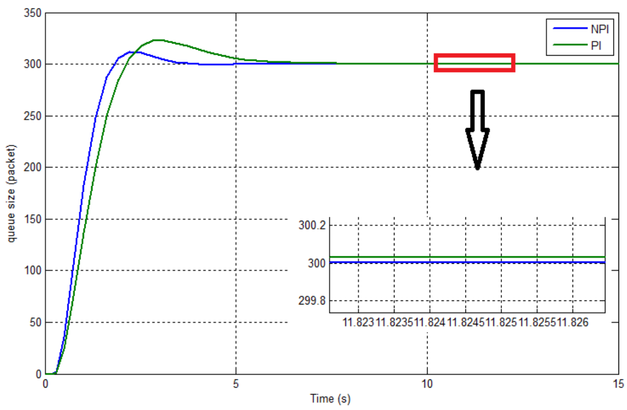

Figure 3. AQM system responses for the proposed controllers

The performance assessment relies on key metrics including settling time (t_s), maximum overshoot ($M_p\%$), steady-state error (Ess) and the value of the IAE index. Form Table 3, it can be observed that the TCP/AQM system based on the NPI is able to follow the desired input step with zero Ess as compared to the PI where the Ess is 0.748. Besides, the value of t_s for the NPI controller (3.14 s) is less than PI controller (5.4 s). Moreover, the PI controller has a larger $M_P\%$ value (7.96%) as compared to NPI controller which has a $M_P\%$ value (4.1%). Furthermore, the PI controller has a larger IAE value (386.5) as compared to NPI controller which has a IAE value (296.3).

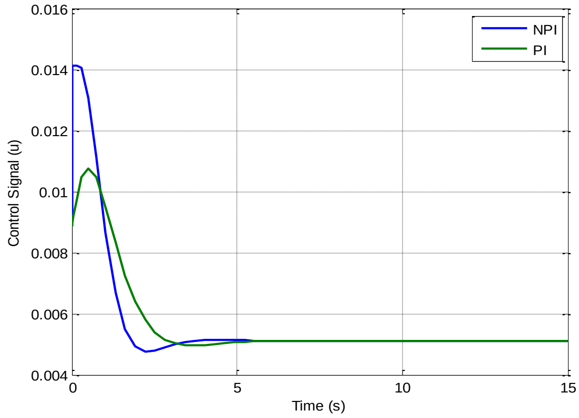

Figure 4. The behavior of control action

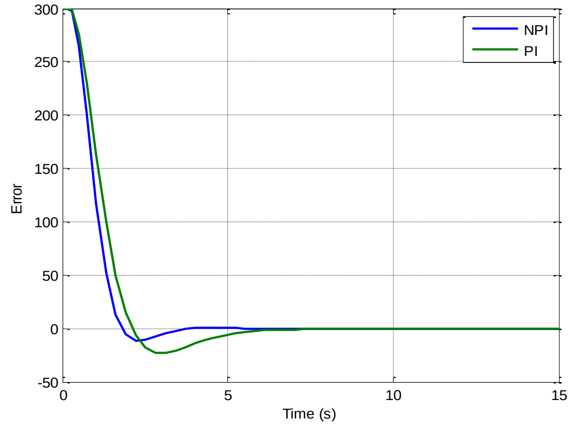

Figure 5. The behavior of error signal

Table 3. Evaluation of controlled TCP/AQM system using NPI and PI based on constant packet size

Performance	Controller	
	NPI	PI
t_s(s)	3.14	5.4
$M_P\%$	4.1	7.96
Ess	0	0.748
IAE	312.3	386.5

To test the controller robustness, analysis study based on simulation was adopted by subjected the system to a disturbance signal after 10 s of the simulation with an amplitude of 10% of the set point input. External disturbance examination is crucial for evaluating the proposed control system performance because they reveal how the controlled system can maintain its desired behavior in the face of unpredictable environmental factors. The performance evaluation is conducted based on the recovery time (t_r), the output response range (δ), defined as the difference between the maximum and minimum values, and the value of the IAE index. These results are summarized in Table 4.

The response for disturbance rejection test is shown in Figure 6. The control signal is shown in Figure 7 and the error signal is shown in Figure 8. Table 4 shows the system performance under disturbance scenario using both NPI and PI controllers. Based on Figure 6 and Table 4, when a disturbance occurs, the PI controller exhibits a δ of 16 packets and takes approximately 4.5 seconds to return to its desired state. In contrast, the NPI controller experiences a smaller deviation of 12.5 packets and recovers within 3.5 seconds. Furthermore, the PI controller has a larger IAE value (403.7) as compared to NPI controller which has a IAE value (296.3).

These findings clearly demonstrate that the NPI controller offers enhanced performance and greater robustness in optimizing the TCP/AQM system.

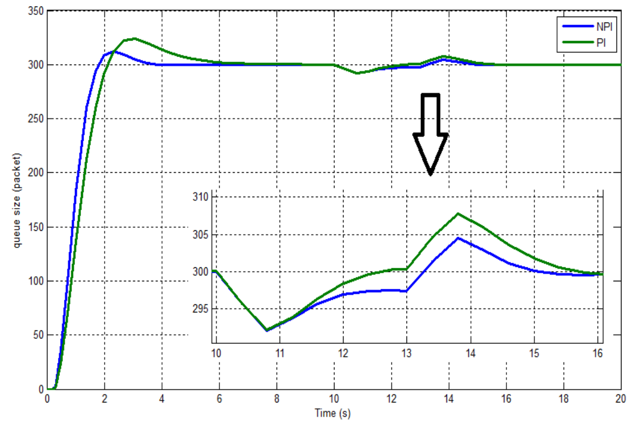

Figure 6. Responses of the AQM system under disturbance using the proposed controllers

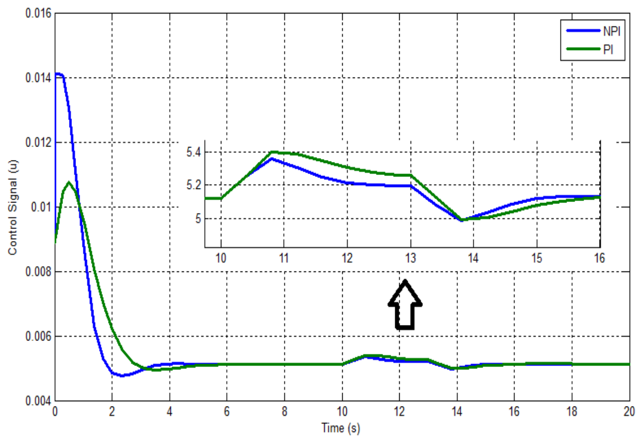

Figure 7. The behavior of control action with disturbance

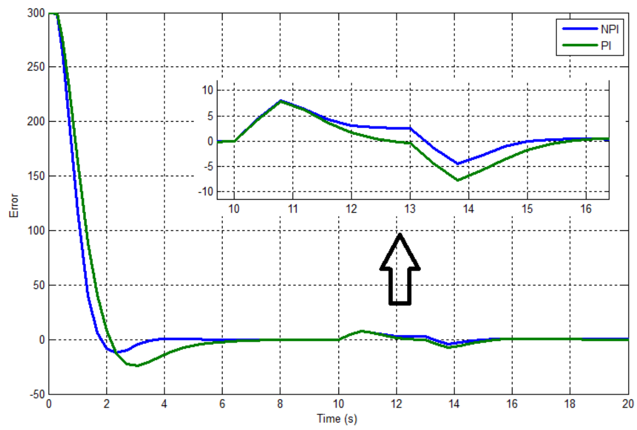

Figure 8. The behavior of error signal with disturbance

Table 4. Evaluation of controlled TCP/AQM system based on proposed controllers with +20% model uncertainty

Performance	Controller	
	NPI	**PI**
t_r	3.5	4.5
δ	12.5	16
IAE	312.3	403.7

6. CONCLUSION

The paper examines the performance of nonlinear PI controller for improving the performance of the AQM system. On the basis of comparison, the classical PI controller is introduced. A swarm-based optimization technique named Multi-Verse Optimizer (MVO) is employed to fine-tune the design parameters of the two controllers. MATLAB is used to perform the performance assessment of the two controllers. The simulation experiment under standard operating conditions is evaluated the controller effectiveness with respect to settling time, steady-state error, and overshoot. The results of this simulation reveal that the nonlinear PI controller outperforms the classical PI controller in terms of efficiency. Furthermore, robustness evaluation demonstrates that the nonlinear PI controller is better in terms of handling the external disturbances, offering a more stable and reliable system response. The outcomes of this paper in terms of optimized the NPI controller can be further evaluated by another dynamic system.

REFERENCES

[1] Umbricht, G.F., Tarzia, D.A., Rubio, D. (2022). Determination of two homogeneous materials in a bar with solid-solid interface. Mathematical Modelling of Engineering Problems, 9(3): 568-576. https://doi.org/10.18280/mmep.090302

[2] Shneen, S.W., Sulttan, M.Q., Oudah, M.K. (2022). Design and implementation of a stability control system for TCP/AQM network. Indonesian Journal of Electrical Engineering and Computer Science, 22(1): 129-136. https://doi.org/10.11591/ijeecs.v22.i1.pp129-136

[3] Kadhim, H.M., Oglah, A.A. (2020). Interval type-2 and type-1 fuzzy logic controllers for congestion avoidance in internet routers. IOP Conference Series: Materials Science and Engineering, 881(1): 012135. https://doi.org/10.1088/1757-899X/881/1/012135

[4] Floyd, S., Jacobson, V. (1993). Random early detection gateway for congestion avoidance. IEEE/ACM Transactions on Networking, 1(4): 397-413. https://doi.org/10.1109/90.251892

[5] Yaseen, F.R., Kadhim, M.Q., Al-Khazraji, H., Humaidi, A.J. (2024). Decentralized control design for heating system in multi-zone buildings based on whale optimization algorithm. Journal Européen des Systèmes Automatisés, 57(4): 981-989. https://doi.org/10.18280/jesa.570406

[6] Al-Khazraji, H., Al-Badri, K., Al-Majeez, R., Humaidi, A.J. (2024). Synergetic control design based sparrow search optimization for tracking control of driven-pendulum system. Journal of Robotics and Control, 5(5): 1549-1556. https://doi.org/10.18196/jrc.v5i5.22893

[7] Mahmood, Z.N., Al-Khazraji, H., Mahdi, S.M. (2023).

Adaptive control and enhanced algorithm for efficient drilling in composite materials. Journal Européen des Systèmes Automatisés, 56(3): 507-512. https://doi.org/10.18280/jesa.560319

[8] Al-Ani, F.R., Lutfy, O.F., Al-Khazraji, H. (2024). Optimal backstepping and feedback linearization controllers design for tracking control of magnetic levitation system: A comparative study. Journal of Robotics and Control (JRC), 5(6): 1888-1896. https://doi.org/10.18196/jrc.v5i6.24073

[9] Naji, R.M., Dulaimi, H., Al-Khazraji, H. (2024). An optimized PID controller using enhanced bat algorithm in drilling processes. Journal Européen des Systèmes Automatisés, 57(3): 767-772. https://doi.org/10.18280/jesa.570314

[10] Kadhim, M.Q., Yaseen, F.R., Al-Khazraji, H., Humaidi, A.J. (2024) Application of terminal synergetic control based water strider optimizer for magnetic bearing systems. Journal of Robotics and Control, 5(6): 1973-1979. https://doi.org/10.18196/jrc.v5i6.23867

[11] Mahawish, A.A., Hassan, H.J. (2021) Survey on: A variety of AQM algorithm schemas and intelligent techniques developed for congestion control. Indonesian Journal of Electrical Engineering and Computer Science, 23(3): 1419-1431. https://doi.org/10.11591/ijeecs.v23.i3.pp14191431

[12] Abdel-Jaber, H. (2020). An exponential active queue management method based on random early detection. Journal of Computer Networks and Communications, 2020: 11. https://doi.org/10.1155/2020/8090468

[13] Bisoy, S.K., Pattnaik, P.K. (2018). An AQM controller based on feed-forward neural networks for stable internet. Arabian Journal for Science and Engineering, 43: 3993-4004. https://doi.org/10.1007/s13369-017-2767-9

[14] Gomez, C.A., Wang, X., Shami, A. (2019). Intelligent active queue management using explicit congestion notification. In 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, pp. 1-6. https://doi.org/10.1109/GLOBECOM38437.2019.9013475

[15] Sulttan, M.Q., Jaber, M.H., Shneen, S.W. (2020). Proportional-integral genetic algorithm controller for stability of TCP network. International Journal of Electrical and Computer Engineering, 10(6): 6225-6232. https://doi.org/10.11591/ijece.v10i6.pp6225-6232

[16] Ali, H.I., Khalid, K.S. (2014). H-infinity based active queue management design for congestion control in computer networks. International Journal of Computers, Communications and Control Engineering, 14(3): 1-9.

[17] Ali, H.I, Khalid, K.S. (2016). Swarm intelligence based robust active queue management design for congestion control in TCP network. IEEJ Transactions on Electrical and Electronic Engineering, 11(3): 308-324. https://doi.org/10.1002/tee.22220

[18] Kadhim, H.M., Oglah, A.A. (2021). Congestion avoidance and control in internet router based on fuzzy AQM. Engineering and Technology Journal, 39(2): 233-247. https://doi.org/10.30684/etj.v39i2A.1799

[19] Lin, L., Shi, Y., Chen, J., Ali, S. (2020). A novel fuzzy PID congestion control model based on cuckoo search in WSNs. Sensors, 20(7): 1816. https://doi.org/10.3390/s20071862

[20] Oudah, M.K, Sulttan, M.Q., Shneen S.W. (2021). Fuzzy type 1 PID controllers design for TCP/AQM wireless networks. Indonesian Journal of Electrical Engineering and Computer Science, 21(1): 118-127. https://doi.org/10.11591/ijeecs.v21.i1.pp118-127

[21] Al-Faiz, M.Z., Mahmood, A.M. (2011). Fuzzy-genetic controller for congestion avoidance in computer networks. International Journal of Computers, Communications and Control Engineering, 11(2): 22-30.

[22] Sabry, S.S., Nayl, T.M. (2019) Particle swarm optimization based LQ-servo controller for congestion avoidance. Iraqi Journal of Computers, Communications, Control, and Systems Engineering, 19(1): 63-70. https://doi.org/10.33103/uot.ijccce.19.1.8

[23] Al-Majeed, M.A., Saud, L. (2018). A comparative study among four controllers intended for congestion control in computer networks. American Scientific Research Journal for Engineering, Technology, and Sciences, 41(1): 333-355.

[24] Humaid, A.J., Hasan, H.M., Raheem, F.A. (2014). Development of model predictive controller for congestion control problem. Iraqi Journal of Computers, Communications, Control, and Systems Engineering, 14(3): 42-51.

[25] Misra, V., Gong W., Towsley, D. (2000). Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In SIGCOMM Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Stockholm, Sweden, pp. 151-160. https://doi.org/10.1145/347059.347421

[26] Hollot, C.V., Misra, V., Towsley, D., Gong, W.B. (2001) A control theoretic analysis of RED. In Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society, Anchorage, AK, USA, pp. 1510-1519. https://doi.org/10.1109/INFCOM.2001.916647

[27] Rasheed, L.T., Yousif, N.Q., Al-Wais, S. (2023). Performance of the optimal nonlinear PID controller for position control of antenna azimuth position system. Mathematical Modelling of Engineering Problems, 10(1): 366-375. https://doi.org/10.18280/mmep.100143

[28] Al-Khazraji, H. (2022). Comparative study of whale optimization algorithm and flower pollination algorithm to solve workers assignment problem. International Journal of Production Management and Engineering, 10(1): 91-98. https://doi.org/10.4995/ijpme.2022.16736

[29] Al-Khazraji, H., Khlil, S., Alabacy, Z. (2020). Industrial picking and packing problem: Logistic management for products expedition. Journal of Mechanical Engineering Research and Developments, 43(2): 74-80.

[30] Khlil, S., Al-Khazraji, H., Alabacy, Z. (2020). Solving assembly production line balancing problem using greedy heuristic method. IOP Conference Series: Materials Science and Engineering, 745(1): 1-7. https://doi.org/10.1088/1757-899X/745/1/012068

[31] Ahmed, A.K., Al-Khazraji, H., Raafat, S.M. (2024). Optimized PI-PD control for varying time delay systems based on modified smith predictor. International Journal of Intelligent Engineering & Systems, 17(1): 331-342. https://doi.org/10.22266/ijies2024.0229.30

[32] AL-Ali, M.A., Lutfy, O.F., Al-Khazraj, H. (2024). Comparative study of various controllers improved by

swarm optimization for nonlinear active suspension systems with actuator saturation. International Journal of Intelligent Engineering & Systems, 17(4): 870-881. https://doi.org/10.22266/ijies2024.0831.66

[33] Al-Ali, M.A., Lutfy, O.F., Al-Khazraj, H. (2024). Investigation of optimal controllers on dynamics performance of nonlinear active suspension systems with actuator saturation. Journal of Robotics and Control (JRC), 5(4): 1041-1049. https://doi.org/10.18196/jrc.v5i4.22139

[34] Mirjalili, S., Mirjalili, S.M., Hatamlou, A. (2016). Multi-verse optimizer: A nature-inspired algorithm for global optimization. Neural Computing and Applications, 27: 495-513. https://doi.org/10.1007/s00521-015-1870-7

[35] Abualigah, L. (2020). Multi-verse optimizer algorithm: A comprehensive survey of its results, variants, and applications. Neural Computing and Applications, 32(16): 12381-12401. https://doi.org/10.1007/s00521-020-04839-1

[36] Wang, X., Pan, J.S., Chu, S.C. (2020). A parallel multi-verse optimizer for application in multilevel image segmentation. IEEE Access, 8: 32018-32030. https://doi.org/10.1109/ACCESS.2020.2973411

NOMENCLATURE

W	expected TCP window size in packets
q	queue length in packets
R	round-trip time in seconds
N	number of active TCP sessions
C	link capacity in packets per second
P	packet marking or dropping probability
K_p	proportional gain
K_i	integral gain
u	control law
e	Error
X	potential solution
L	total number of iterations

Greek symbols

δ	minor deviation in the variables relative to their nominal values
λ	design parameter