International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 25 November 2023 **Revised:** 12 December 2023 **Accepted:** 30 January 2024*

Fuzzy Fractional-Order PID Congestion Control Based on African Buffalo Optimization in Computer Networks

Fatima H. Abd [1], Osama A. Awad [2]
[1] Systems Eng. Dept., Information Engineering College, Nahrain University, Baghdad, Iraq
[2] College of Engineering and Technical Engineering, Al Mustaqbal University, Babylon, Iraq
Email: fatimahameedfa62@gmail.com[1], osama.ali.awad@uomus.edu.iq[2]

***Abstract**:* Congestion is the primary factor that slows down data transfer in communication networks. Transmission Control Protocol and Active Queue Management (TCP/AQM) collaborated to resolve this issue. The fuzzy-Fractional-Order-PID (FFOPID) controller is developed in this paper to control the linearized TCP/AQM model. The strategy is founded on a combination of fractional-order PID and fuzzy logic controllers. The primary objective of the proposed controller is to maintain the queue length of the router within the appropriate queue threshold for a congestion model. The control parameters are tuned using African Buffalo optimisation (ABO). The suggested controller is compared to other controllers (PID, Fuzzy-PID, and Fractional-order PID) to demonstrate the controller's efficiency, and all of these controllers are optimised using African Buffalo Optimisation (ABO). In MATLAB (R2017b), the simulation of the linearized model is introduced. Comparing the results of the Fuzzy-Fractional-Order-PID controller with those of other controllers in the same network scenarios reveals that the Fuzzy-FOPID is robust for a wide variety of TCP flows.

***Keywords**: Active Queue Management (AQM), TCP, African Buffalo Optimizer (ABO), PID controller, Fuzzy-PID (FPID) Controller. Fuzzy-fractional-order-PID (FFOPID) controller.*

1.

INTRODUCTION

Congestion control is crucial for ensuring the quality of service for network users. Designing and utilizing computer networks must consider this issue. Congestion control is a recent study topic with an increasing number of publications at the same time [1-3]. As an example, highway congestion is defined as being overloaded or filled to the brim. The most effective method to avoid congestion is to avoid places and times where it is likely to occur, but this is not always possible. There is always a bottleneck in real-world networking configurations, such as a sluggish computer, a slow link, or an intermediate node with limited throughput [4]. It is crucial to prevent network congestion and ensure that data transmissions are received as quickly as feasible. TCP is a protocol used for Internet-based computer communication that includes a mechanism to prevent congestion in computer networks. TCP detects congestion by analyzing acknowledgments or time-out processing and adjusts sender TCP window sizes accordingly. This control strategy has the disadvantage of preventing congestion only after it has already occurred on computer networks. AQM techniques have been proposed to supplement TCP's role in network congestion control [5].

This paper is structured as follows: The second section presents related work. In Section 3, the TCP behavior paradigm is introduced. The Fuzzy-Fractional-Order PID controller is designed in Section 4. Section 5 describes the ABO algorithm, Section 6 describes the Network Topology Scenario, Section 7 describes the simulation results, and Section 8 describes the work's conclusions.

2. RELATED WORK

One of the key problems of network engineers in recent years has been how to control congestion in internet networks. As a result, a lot of papers have been published to help develop new algorithms for congestion controllers. A few studies are reviewed below.

The Fuzzy Proportional-Integral controller was designed as a (AQM) in an Internet router in 2011 [6] in order to improve the efficacy of PI controllers and decrease network congestion. As an optimisation technique, the genetic algorithm (GA) was used to modify the Fuzzy-PI parameters. 2013 Salim and his companions The "MATLAB/Nonlinear Control Design Blockset (NCD)" was used to fine-tune the (PI) controller's congestion avoidance settings. Amjad J. et al. introduced model predictive control and PID controllers in 2014[8]. This controller is based on the nonlinear dynamic TCP/AQM model. The system-based (PID) controller has less parameter change resistance than the MP controller. The Fuzzy-PID-like-Gain Scheduling controller was proposed by Isra'a L. and Osama A. Awad in 2015[9] to address time delays in the network control system (NCS). The proposed method utilises both fuzzy and gain scheduling control elements and is dependent on the network's online measured actual time delay. The performance result demonstrates that the controller is optimally adapted for time-varying delays and that it is resilient even when packet loss is possible. Based on Internal Model Control, K.Godweena.A. and K.Sundaravadivu (2015) [10] build and modify a fractional-order controller class (IMC). Comparing integer-order controllers with the same IMC rule enabled this. To ascertain the maximum delay time uncertainty, a robust stability study was conducted, resulting in a stable closed-loop control system. In 2019[11], a new congestion control approach for (TCP/AQM) scheme employing adaptive back-stepping, funnel control, and fuzzy systems is proposed to resolve a tracking challenge.The transient and steady-state performances of the tracking error could meet the design specifications. Haneen S. Abdulkareem and Osama A. Awad proposed in 2019 [12] an FFOPID controller optimised by antlion optimisation to address the congestion problem in TCP/AQM routers.Berbek and Oglah [13, 14] devised a hybrid congestion control method that integrates intelligence (PID) and a type 1 fuzzy logic controller. In this study, social spider optimisation (SSO) is employed to decrease queue size errors.

3. THE TCP BEHAVIOR MODEL

A dynamic model of the behaviour of transmission control protocol (TCP) was created by ignoring the TCP timeout mechanism and applying fluid-flow and stochastic differential equation analysis. This model can be represented by the following nonlinear differential equations. [15, 16]

$$W^{\bullet}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))} p(t - R(t)) \quad (1)$$

$$q^{\bullet}(t) = \frac{W(t)}{R(t)} N(t) - C \quad (2)$$

Where $W^{\bullet}(t)$ and $q^{\bullet}(t)$ denotes the time-derivative of W (t) and q (t) respectively, and t denotes time in (sec).
W = average TCP window size (packets);
q = average queue length (packets);
R (t) = round trip time (sec); $R(t) = \frac{q}{c} + T_p$
p = probability of packet drop/mark;
N = Number of TCP sessions (load factor);
C = link capacity (packets/sec.);
The marking probability must be only between [0, 1].
The AQM model's nonlinear differential equations could be linearized around (W_o, q_o, P_o), which represents an operating point for constructing the following linearized model [15]:

$$p(s) = \frac{\delta q(s)}{\delta p(s)} = \frac{\frac{C^2}{2N} e^{-sR_0}}{(s + \frac{2N}{R_0{}^2 C})(s + \frac{1}{R_0})} \quad (3)$$

Such that δp (s)" is the loss probability, δq (s) is the queue length

4. CONTROLLER DESIGN

The controller is built as follows:

Fuzzy-Fractional-Order-PID (FFOPID) Controller:

The Fractional-order-PID (FOPID) controller, proposed by Podlubny, is a type of PID controller in which the derivative gain and integral gain are fractional rather than integer as in PID. The use of fractional-order derivatives and integrals improves the performance of the PID controller because fractional-order controllers are less sensitive to changes in a controlled system's and controller's parameters. The equation of the FOPID controller is [17]:

$$u(t) = k_p e(t) + k_d D^{\mu} e(t) + k_i D^{-\lambda} \quad (4)$$

Such that K_p ,K_d and K_i represented a proportional, derivative, and integral gain respectively where μ and λ represented a fractional order of k_d and k_i. Fuzzy logic controller (FLC), as intelligent systems, can be used to represent human decision-making and behavior [18]. To express the input-output relationships in FLC, The use of a set of language norms or relational expressions. The FLC consists of four main parts: a fuzzifier, a defuzzifier, an inference engine, and a rule base. In many fuzzy control applications, the input data are typically discrete; consequently, fuzzification is required to transform the input discrete data into the set of linguistic values required by the inference engine. The proposed fuzzy-Fractional-Order-PID controller is illustrated in Figure.1.

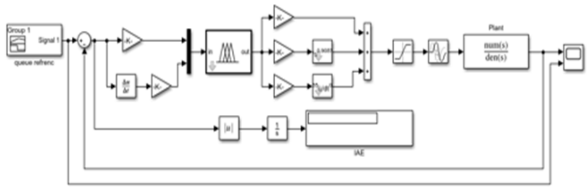
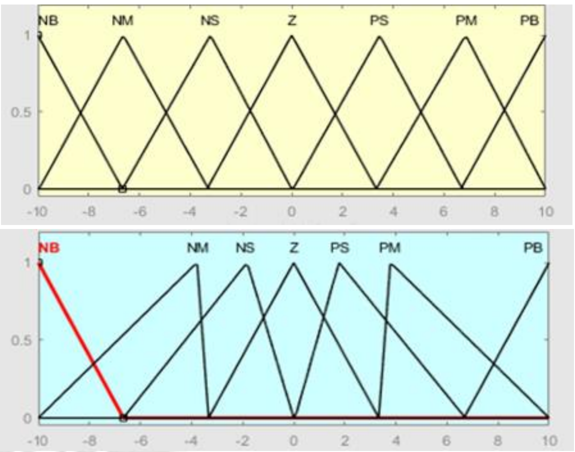


FIGURE 1. Proposed Fuzzy-Fractional-PID controller

The centre of gravity approach is chosen as the defuzzification mechanism.The input and output membership functions are in the shape of seven triangles and the universe of discourse for the input and output variables is equal [-10 10]. Figure 2 depicts the input and output MFs. The linguistic variables of Fuzzy logic controller MFs are depicted in Table (1). The imprecise rule base is depicted in Table 2.



a b

FIGURE 2**.** (a) Input (e and ec) MFs, (b) output MFs

TABLE 1. Linguistics variables of MFs

Linguistics description	**Linguistics abbreviation**
NB	Negative Big
NM	Negative Medium
NS	Negative Small
Z	Zero
PS	Positive Small
PM	Positive Medium
PB	Positive Big

TABLE 2.Fuzzy rule base

e\ec	**NB**	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
NB	NB	NB	NB	NB	NM	NS	Z
NM	NB	NB	NB	NM	NS	Z	PS
NS	NB	NB	NM	NS	Z	PS	PM
Z	NB	NM	NS	Z	PS	PM	PB
PS	NM	NS	Z	PS	PM	PB	PB
PM	NS	Z	PS	PM	PB	PB	PB
PB	Z	PS	PM	PB	PB	PB	PB

5. AFRICAN BUFFALO OPTIMIZATION (ABO)

The African buffalo optimisation was inspired by the migratory behavior of African buffalos, specifically the buffalo herd's organizational prowess as it travelled from one region of Africa to another in search of grazing grounds [19]. To organize themselves during their search for answers, buffalos primarily employ two vocalisations: the attraction sound /maaa/ for exploitation and the repulsion sound /waaa/ for exploration. The appropriate use of both sounds is crucial for herd mobility, provisioning, and the protection of the entire bison community. The ABO's design is an attempt to create a fast, robust, efficient, yet simple-to-implement and user-friendly algorithm endowed with the capacity to exploit and explore the solution space by modelling the democratic cum communicative abilities of cape buffalos (also known as African buffalos) in their search for solutions [20]. According to (1) and (2), ABO can be represented mathematically.

$$\mathrm{m_{k+1}} = \mathrm{m_k} + \mathrm{lp_1}(\mathrm{bg_{max}} - \mathrm{w_k}) + \mathrm{lp_2}(\mathrm{bg_{max.k}} - \mathrm{w_k}) \quad (1)$$

Where $\mathrm{m_k}$ denotes a “maaa” call with a specific reference to a buffalo k (k = 1, 2, …n), $\mathrm{bg_{max}}$ denotes the best buffalo within the herd, while $\mathrm{bg_{max.k}}$ denotes the best location which an individual buffalo k finds, $\mathrm{lp_1}$ and $\mathrm{lp_2}$ denote the learning parameters that take any value between 0 and 1, $\mathrm{m_{k+1}}$ denotes that the buffalo has moved from its current position $\mathrm{m_k}$ to a new place, which reflects the buffalo's vast memory capacity in the migration lifestyle. The mathematical representation (2) accomplishes real herd movement adjustment.

$$w_{k+1} = \frac{(w_k + m_k)}{\lambda} \quad (2)$$

Where w_k denotes a “waaa” call represents the values of the current exploration, w_{k+1} denotes the migration to a new location, m_k denotes the current exploitation values, and λ denotes a parameter that sets the unit of time for the buffalo movement interval and is frequently set to 1. Figure 3 depicts the ABO flowchart.

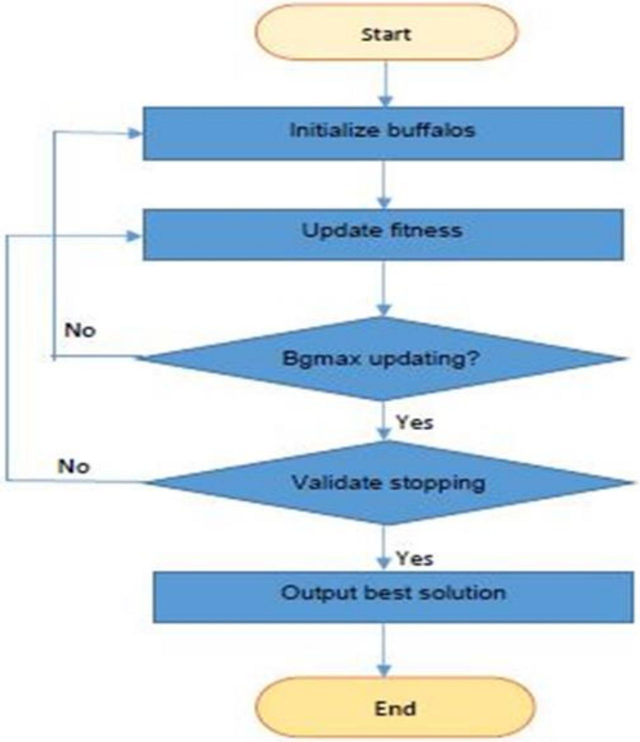


FIGURE 3. ABO algorithm.

6. NETWORK TOPOLOY SCENARIO

MATLAB (2017b) was used to simulate the linearized TCP/AQM model in order to evaluate the efficacy of the designed FFOPID controller and compare it to other controllers. The link capacity has been set to C=3750 packets per second, or 15 Mbps. When the target queue size is 300 packets and the propagation delay is 40 milliseconds, R0 is 0.253 seconds. N for source and destination TCP sessions is 60, and the utmost queue length is 800 packets. While Router A is configured with the proposed controller, Router B employs a drop Tail. Several studies have been conducted to assess the robustness of the FFOPID controller by modifying the values of various network parameters. Figure 4. depicts the utilised network case study.

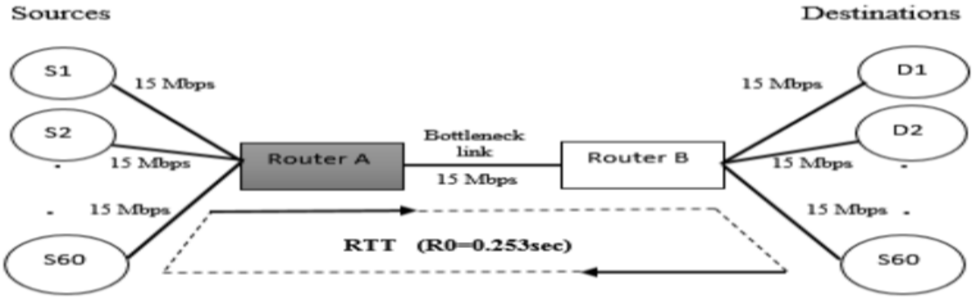


FIGURE 4. A Case Study of Network Topology.

7. SIMULATION RESULT

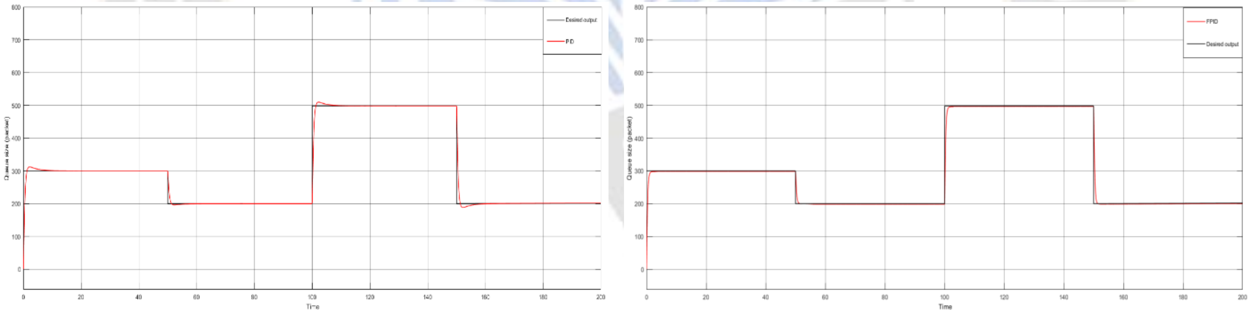
Consider TCP/AQM with the network parameters given in section 6, and reference input, which indicates queue size and varies every 50 seconds. The simulation is performed as follows:

1. **Case study of network**

For the standard case of a network with the value of N is 60 and C is 15 Mbps, the response of the system for four controllers (PID, FPID, FOPID, FFOPID) depicts in Figure 5. PID and FPID had good results, while FOPID outperforms them in term of dynamic performance but with overshoot reach to 20.9% these drawbacks solved by using FFOPID controller. The Queue length performance of the controllers is depicted in Table (4). The optimal values found by ABO for (PID, FPID, FOPID, and FFOPID) displays in the Table (3) below.

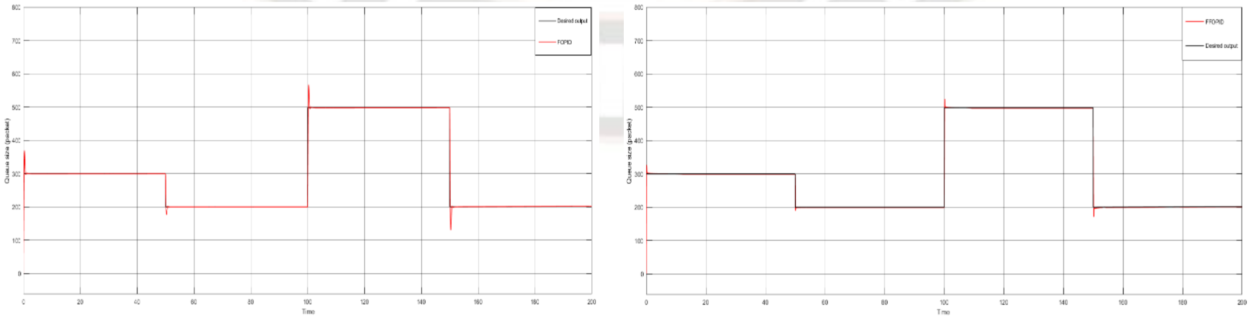
TABLE 3: The values for (PID, FPID, FOPID, and FFOPID) controllers

Parameters	**Kp**	**Ki**	**Kd**	**Ko**	**μ**	**λ**
PID	1.76e-4	5.98e-5	4.67e-5	-	-	-
FPID	0.019	0.0058	0.048	0.025	-	-
FOPID	1.78e-4	1.03e-3	1.42e-4	-	0.508	0.691
FFOPID	0.076	0.058	0.011	-	0.707	0.692



(a)PID controller

(b) FPID controller



(c)FOPID controller

(d) FFOPID controller

FIGURE 5. The queue length for different controllers with N=60 and C=15Mbps

TABLE 4. Queue length Performance using the four controllers for N=60, C=15Mbps, R=0.253s

N=60, C=15, R=0.253	PID	FPID	FOPID	FFOPID	Enhancement(PID-FFOPID)
Rise time (s)	0.7743	0.6159	0.1759	0.0856	88.9%
Settling time (s)	4.6161	1.1378	0.9938	0.2727	94.1%
Overshoot %	4.0261	0.0121	20.8942	9.3667	99.9%
IAE	178.13	169.05	72.46	60.86	65.8%

2. Round trip time (R0)

To study the effect of round trip time on the network, the value of RTT is changed to 0.12 s and the values of N and C are the same. Figure 6 depicts the response of the system of the four controllers (PID, FPID, FOPID, FFOPID), and Table (5) depicts the Queue length performance of the four controllers.

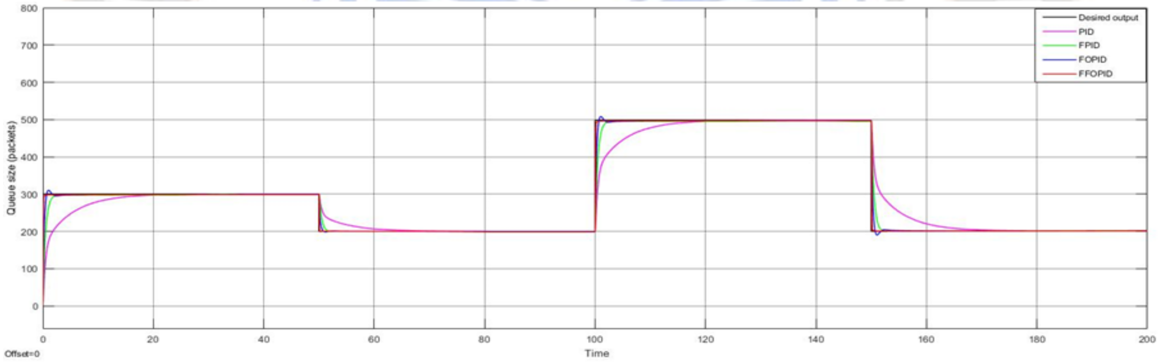


FIGURE 6. The queue length for different controllers when N=60, C=15Mbps, and R=0.12 s

TABLE 5. Queue length Performance using the four controllers for N=60, C=15Mbps, R=0.12s

N=60, C=15, R=0.12	PID	FPID	FOPID	FFOPID	Enhancement(PID-FFOPID)
Rise time (s)	7.7904	1.0191	0.4307	0.1141	98.5%
Settling time (s)	16.1523	1.7909	1.3797	0.1878	98.8%
Overshoot %	0.0068	0.0902	3.6434	-	-

IAE	791.50	280.98	120.78	95.48	87.9%

As clear from Figure 6 and Table (5). The rise time and settling time for the (FPID, FOPID, and FFOPID) controllers keep in acceptable values, but the rise time and settling time in the PID controller have increased which may lead to packet loosed and delay.

3. Link capacity (C)

To study the effect of capacity on the network, the following tests are conducted.

3.1 test 1

To study the effect of decreasing link capacity on queue length and behavior of the proposed controller in such status the value of N taken to be 80, and the link capacity C taken to be 10 Mbps. Figure7 depicts the response of the system of the four controllers, and Table (6) depicts the Queue length performance of the four controllers.

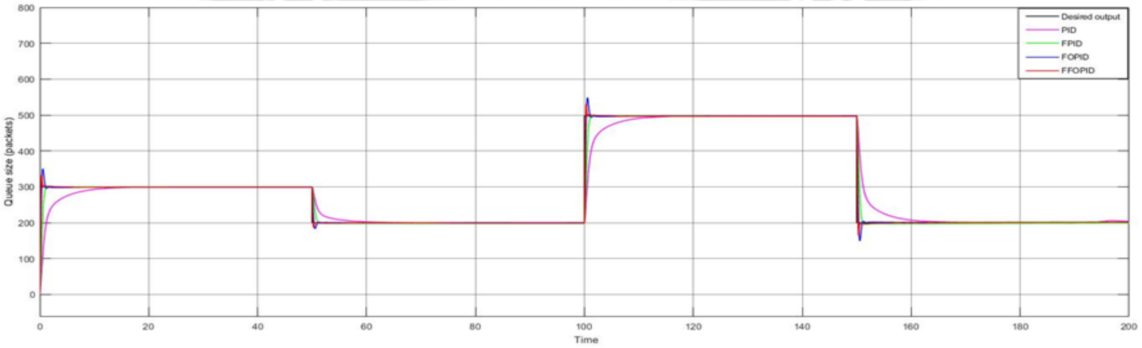


FIGURE 7. The queue length for different controllers when N=80, C=10Mbps, and R=0.253 s

TABLE 6. Queue length Performance using the four controllers for N=80, C=10Mbps, R=0.253s

N=80, C=10, R=0.253	**PID**	**FPID**	**FOPID**	**FFOPID**	**Enhancement(PID-FFOPID)**
Rise time (s)	4.1142	0.6903	0.2465	0.1571	96.1%
Settling time (s)	10.6503	1.1682	0.9493	0.5029	95.2%
Overshoot %	0.0105	0.2673	17.1949	11.6423	99.9%
IAE	506.89	129.97	110.38	106.66	78.9%

3.2 test 2

The value of N is changed from 80 to 100, and the capacity C is still 10 Mbps. Figure 8 depicts the response of the system of the four controllers, and Table (7) depicts the Queue length performance of the four controllers.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 2
***Article Received:** 25 November 2023 **Revised:** 12 December 2023 **Accepted:** 30 January 2024*

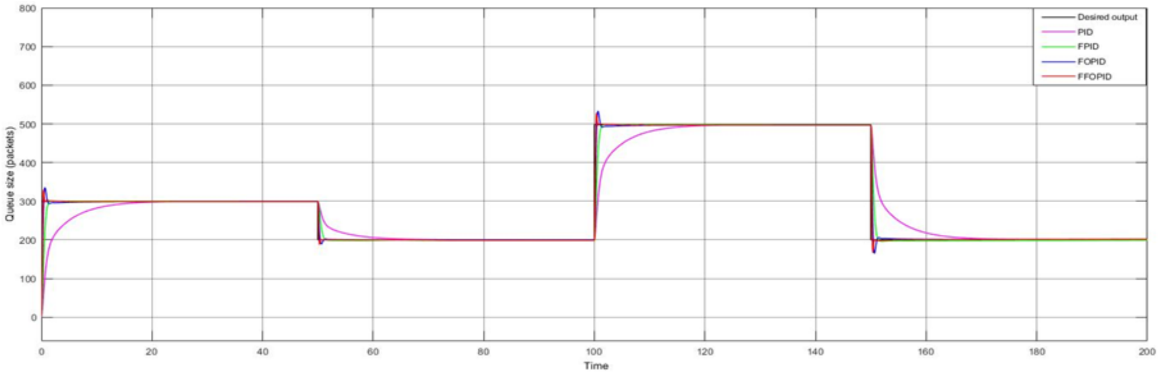


FIGURE 8. The queue length for different controllers when N=100, C=10Mbps, and R=0.253 s

TABLE 7. Queue length Performance using the four controllers for N=80, C=10Mbps, R=0.253s

N=100, C=10, R=0.253	PID	FPID	FOPID	FFOPID	Enhancement(PID-FFOPID)
Rise time (s)	7.2178	0.8145	0.2999	0.1813	97.4%
Settling time (s)	15.2701	1.4177	1.4499	0.5533	96.3%
Overshoot %	0.0173	0.0446	11.8963	10.4378	99.8%
IAE	791.82	149.86	139.35	114.66	85.5%

As clear from the results of test 1 and test 2, The PID controller starts to lose control of the system, but the performance of the other controllers is good.

4. Robustness of the system

In this test, the value of N is 100, and the link capacity C is 5 Mbps. Figure 9 depicts the response of the system of the four controllers, and Table (8) depicts the Queue length performance of the four controllers.

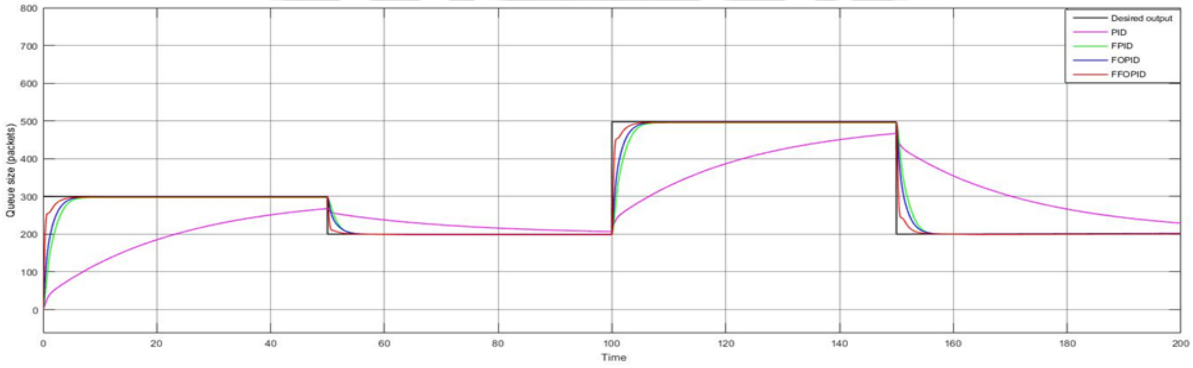


FIGURE 9. The queue length for different controllers when N=100, C=5Mbps, and R=0.253 s

TABLE 8. Queue length Performance using the four controllers for N=100, C=5Mbps, R=0.253s

N=100, C=5, R=0.253	PID	FPID	FOPID	FFOPID	Enhancement(PID-FFOPID)
Rise time (s)	35.0257	3.4057	2.7538	0.8541	97.5%
Settling time (s)	46.3642	5.5267	4.9970	2.9838	93.5%
Overshoot %	-	0.0014	0.0024	0.0017	-
IAE	5586.03	636.52	386.87	241.77	95.6%

As clear from Figure 9 and Table (8). The PID controller has completely unable to track the required queue length. On the other hand, the (FPID, FOPID) controller displayed an ability to deal with network changes, with good tracking to the required queue and acceptable values of rise time and settling time, as shown in (Table 8). However, if FFOPID is used, the IAE enhancement can reach 95.6%, as shown in the same table.

8. CONCLUSION

This paper proposed a Fuzzy-Fractional-Order-PID (FFOPID) controller for the network congestion problem, which was optimized using African Buffalo Optimization. In comparison to PID, FPID, and FFOPID controllers, the proposed controller has superior tracking performance regarding the desirable queue size required in a router's buffer size for dealing with congestion. The FFOPID controller was capable of verifying dynamic network changes such as TCP session, Round trip time, and link capacity.

In terms of rise time, settling time, overshoot, and IAE, the performance of FFOPID was compared to that of FOPID.utilising the African buffalo optimization, this controller is optimized. In all previous experiments, performance was improved by 68.9% in terms of rise time, 40.2% in terms of settling time, 29.0% in terms of overshoot, and 37.5% in terms of IAE.

REFERENCE

[1] Imer, O., Basar, T., “Control of congestion in high speed networks,” Eur. J. Contr., 7, pp. 132-144, 2001.

[2] Misra, V., Gong, W., Towsley, D., “Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED,” Proceedings of ACM SIGCOMM, pp. 151-160, 2000.

[3] Turowska, M., “Application of uncertain variables to stability analysis and stabilization for ATM ABR congestion control systems,” Proceedings of the International Conference on Enterprise Information Systems. INSTICC Press, Porto, 2, pp. 523-526, 2004.

[4] D. E. Mcdysan and D. Paw, "ATM & MPLS Theory & Application: Foundations of Multi-Service Networking", McGraw-Hill, 2002.

[5] T. Azuma, T. Fujita, and M. Fujita, “Congestion control for TCP/AQM networks using state predictive control.” IEEE Transactions on Electronics, Information and Systems, Vol. 125, No.9, pp. 1491–1496, 2005.

[6] M. Z. Al-Faiz and A. L. A. M. Mahmood, "Fuzzy-Genetic controller for congestion avoidance in computer networks," Iraqi Journal of Computers, communication and Control & systems engineering, vol. 11, pp. 20-27, 2011.

[7] S. T. Salim and A. M. Mahmood, “Design of On-Line Tuned Controller for Congestion Avoidance in Computer Networks.” Eng. &Tech. Journal, Vol. 31, No. 3, 2013.J. Humaid, H.M. Hasan and F. A. Raheem, “Development of Model Predictive Controller for Congestion Control Problem, “IJCCCE 14 (3), pp. 42-51, 2014.

[8] L. Salim, "A Fuzzy Gain Scheduling Based Controller for Network Control Systems," MS.C Thesis Department of Networks Engineering and Internet Technologies, Nahrain University, 2015.

[9] K. G. A and K. Sundaravadivu, “IMC Based Tuning of Fractional Order Controller (PIλDμ Controller),” Int.Conf. Circuit, Power Comput. Technol. [ICCPCT], vol. 978-1–4799, no. ©2015 IEEE, 2015.

[10] K. Wang, Y. Liu, X. Liu, Y. Jing and S. Zhang, “Adaptive fuzzy funnel congestion control for TCP/AQM network,” Elsevier, ISA Transactions,2019.

---

[11] Haneen S. Abdulkareem and Osama A. Awad," FUZZY SET-POINT WEIGHT FOR PID CONTROLLER BASED ON ANTLION OPTIMIZER TO CONGESTION AVOIDANCE IN TCP/AQM ROUTERS," Iraqi Journal of Information and Communications Technology, Vol. 2, 2019.

[12] Berbek, M.I., Oglah, A.A. "Fuzzy like PID controller based on Sso design for congestion avoidance in Internet router," Iraqi Journal of Computers, Communication, Control & Systems Engineering, 21(2): 85-101, 2021.

[13] Berbek, M.I, Oglah, A.A." Adaptive neuro-fuzzy controller trained by genetic-particle swarm for active queue management in internet congestion," Indonesian Journal of Electrical Engineering and Computer Science, 6(1): 229-242, 2022.

[14] V. Misra, W.-B. Gong, and D. Towsley," Fluid-Based Analysis of a Network of AQM routers

[15] Supporting TCP Flows with an Application to RED," in Proc. ACM SIGCOMM, pp. 151-160, 2000.

[16] C.V. Hollot, V. Misra, D. Towsley and W. Gong," A Control Theoretic Analysis of RED, "in Conf. Anc. IEEE Int. Conf. Computer and Communications Societies, pp. 1510 – 1519.

[17] Domański, and et al, "The AQM dropping packet probability function based on non-integer order PIαDβ controller," Springer Int. Publ. AG, part Springer Nat. 2019, vol. 496, pp. 36–48, 2019.

[18] Wang, Bo Li, K. Sohraby, and Y. Peng, "AFRED: An Adaptive Fuzzy-based Control

[19] Algorithm for Active Queue Management, "IEEE LCN, 2003.

[20] Lorenzen, E., Heller, R. & Siegismund, H. R. "Comparative phylogeography of African savannah ungulates," Mol. Ecol. 21, 3656–3670, 2012.

[21] Odili, J. B., Kahar, M. N. & Noraziah, A. Solving traveling salesman's problem using African buffalo optimization, honey bee mating optimization & Lin-Kerninghan algorithms. World Appl. Sci. J. 34, 911–916, 2016.