Web of Scholars: Multidimensional Research Journal (MRJ)
Volume: 02 Issue: 10 | 2023 ISSN: (2751-7543)
http://innosci.org

Fuzzy PID Congestion Control Based on African Buffalo Optimization in Computer Network

Fatima H. Abd, Osama A. Awad
Systems Eng. Department, Information Engineering College, Nahrain University, Baghdad, Iraq

Abstract: *The efficiency of computer networks is essential to modern information services technology, there are several problems that come with it, including the congestion issue. Some congestion issues in the intermediate routers lead to packet time delays, packet losses, and buffer overflows. In order to solve this issue Transmission Control Protocol and Active Queue Management (TCP/AQM) have been used. Bypassing or discarding packets at the intermediate routers, AQM is crucial for controlling the length of the queue. In this study, the Fuzzy-PID (FPID) controller is designed to desist the congestion in networks. The strategy is based on a hybridization between the Proportional-Integral-Derivative (PID) controller and the Fuzzy Logic Controller (FLC), optimized by African buffalo optimization (ABO).Simulation of a linearized model (TCP/AQM) is introduced in MATLAB (R2017b). The Fuzzy-PID controller's (FPID) results are compared with the PID controller based on African buffalo optimization algorithms in each controller, and the results of the comparisons indicate that FPID achieved the best results.*

Keywords: *Active Queue Management (AQM), TCP, African Buffalo Optimization (ABO), PID controller, Fuzzy-PID (FPID) controller.*

1. INTRODUCTION

Over the past few years, computer networks have grown rapidly, and this growth has resulted in serious congestion issues [1]. Large numbers of packets may be delayed or even discarded because of queue overflow during congestion. Throughput suffers as a result of severe congestion issues, as does the packet loss rate. Also, if there is a lot of traffic, performance completely collapses and essentially no packets are sent, which will reduce the efficiency and dependability of the entire network due to congestion [2].

It is essential to avoid congestion and to ensure that transmitted data is received as quickly as possible. It is challenging to fully satisfy this need, nevertheless, because of the enormous volume of data exchanged. In order to prevent data loss and intolerable delays, researchers look for the best solutions rather than the ideal ones. To solve this issue, active queue management and the transmission control protocol are used together. TCP is designed to deal with congestion after it occurs and offers secure data transfer. AQM anticipates congestion and works to find a solution before it happens [3].

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

2. RELATED WORK

In recent years, numerous studies have been published to aid in the development of novel algorithms for congestion controllers, one of the primary concerns of network engineers. This section provides an overview of several investigations.

In [4], Chrysostomou, C., and Pitsillides, A. used the fuzzy logic controller (FLC) as a control mechanism to reduce congestion in two dissimilar technologies, ATM and (TCP/IP) networks, and their findings demonstrated significant improvements in reducing congestion for ATM and TCP/IP networks. H. Ashtiani et al. proposed in 2010 [5] an active queue management concept based on fuzzy logic techniques as a hybrid (fuzzy-PID) controller to provide effective crowding management with minimal delay time, high utilisation, and minimal packet loss. The results indicated that the Hybrid-Fuzzy Control technique can significantly improve congestion control in TCP/IP networks. Compared to conventional adaptive controllers, such as RED and PID, it allows for a very rapid response. Yang proposed the (LQ-servo) controller for TCP/AQM networks with time-varying capacity in 2013[6]. The objective of the proposed controller is to prevent network congestion. Azadegan et al., 2015[7] For (TCP) network congestion control, a novel proportional-derivative state feedback controller is proposed. Hind. A and Osama A. Awad To reduce congestion, [8] developed a Multi-Wavenet PID controller (AMWPID) for TCP/AQM routers in 2016. The comparison test with other AQM algorithms demonstrates that the created AMWPID is superior to the other approaches in terms of TCP fluxes, link capacity, and round-trip time. Zeyad A. Karam, 2018 [9] proposed To regulate the queue length, round trip duration, and PSO Based Loss of Packets, various fuzzy hybrid controllers, such as controller PID like (FLC) based on Particle Swarm Optimisation (PSO), PD like (FLC) with classic IPSO based, and PID tuned by Fuzzy Logic with PSO based, are recommended. Sen suggested in 2018[10] combining a PID controller with an imprecise set-point weight structure. The foundation of the fuzzy rule base design is the sliding mode technique. Use of the proposed controller in a closed-loop system.

3. THE TCP BEHAVIOR MODEL

Fluid-flow and stochastic differential equation analyses were employed to develop a dynamic TCP model's conduct [11]. This model relates the average value of the most important network parameters to the following coupled nonlinear differential equations [12]:

$$W'(t) = \frac{1}{R(t)} - \frac{W(t)\,W(t-R(t))}{2R\,(t-R(t))}\,p(t - R\,(t)) \quad (1)$$

$$q'(t) = \frac{W(t)}{R(t)}N(t) - C \quad (2)$$

Such that $W'(t)$ and $q'(t)$ represents the time derivative of W (t) and q (t) respectively, and (t) represents the time in second.

W = Average of the TCP window size (packets);

q = Average of the queue length (packets);

p = Probability of the packet drop/mark;

R (t) = Round trip time (sec); $R(t) = \frac{q}{c} + T_p$

C = Link capacity (packets per sec);

N = Number of TCP sessions (load factor);

The marking probability should be only between [0, 1].

In order to make the model linear, the number of TCP sessions and link capacity are assumed to be constant. An approximation linearized model can be constructed by applying small-signal linearization about an operating point (Wo, po, qo). The linearized model is represented by the following equations:

$$\delta W'^{(t)} = -\frac{2N}{R_0^2 C}\delta W\,(t) - \frac{R_0 C^2}{2N^2}\delta p\,(t - R_0) \quad (3)$$

Web of Scholars: Multidimensional Research Journal (MRJ)
Volume: 02 Issue: 10 | 2023 ISSN: (2751-7543)
http://innosci.org

$$\delta q'^{(t)} = \frac{N}{R_0}\delta W\,(t) - \frac{1}{R_0}\delta q\,(t) \qquad (4)$$

4. CONTROLLER DESIGN

The controller is created as follows:

Fuzzy-PID (FPID) Controller:

Traditional PID controller is a common controller for practically every system. The PID controller determines the control signal using that the following formula [13].

$$U\,(t) = K_p\, e(t) + K_d\, \Delta\, e(t) + K_i \int e\,(t)\, dt \quad (5)$$

Such that K_p ,K_d and K_i represented a proportional, derivative, and integral gain respectively. Although the objective of classical control theory is frequently to utilize empirical methods to clarify system behavior and the controller's future design, fuzzy systems are useful for simulating undefined information, like a technical mechanism or a real (humans) controller. The input-output relationships in FLC are expressed using a sets of language rules or relational phrases. In essence, the FLC is divided into four parts: (a fuzzifier, defuzzifier, rule base, and an inference engine). In many fuzzy control applications, the input data are frequently crisp; hence, the fuzzification is necessary to transform input crisp data into a set of linguistic values required by the inference engine. Figure 1. Displays the proposed fuzzy-PID controller.

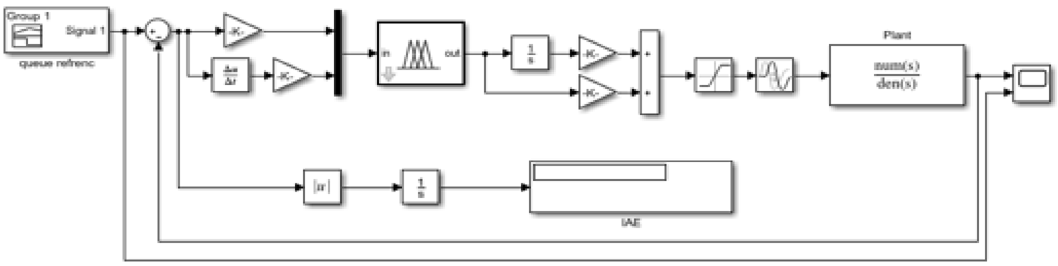
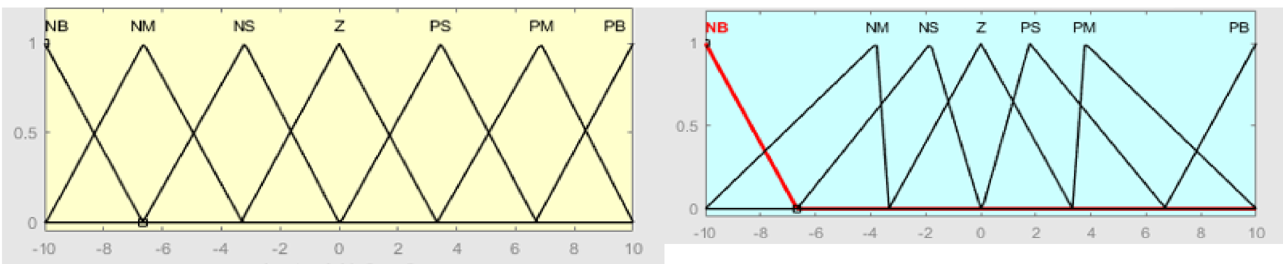


FIGURE 1. Proposed fuzzy-PID controller

The input and output membership functions are seven triangles shaped, and the universe of discourse for the input and output variables are equal [-10 10]. The center of gravity approach is used as the defuzzification technique [13].

Figure 2. displays Membership functions (MFs) for inputs (error (e) and error change (ec)) and the output. Table 1 displays linguistic variables for Fuzzy logic controller membership functions. Table 2 displays the fuzzy rule base.



(a) (b)

FIGURE 2. (a) Inputs (e and ec) MFs, (b) output of MFs

TABLE 1. Linguistics variables of (MFs) **TABLE 2.** Fuzzy rule base

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

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, a newly created metaheuristic algorithm, was inspired by the herd management and organisational abilities of the African buffalos as they migrate through the various African landscapes (including arid deserts, savannah, and forests) in search of lush grazing fields to satisfy their voracious appetites and maintain their species in the ecosystem. [14]. By utilising the calls /waaa/ (explore) and /maaa/ (exploit), African buffaloes are able to transform their search for optimal grazing pastures from a desperate situation to a highly rewarding one [15]. The African Buffalo Optimisation (ABO) algorithm is depicted in Figure 3.

Step 1: Initialize the buffaloes at random in various locations.
Step2: Update the buffaloes' exploitation fitness by using:

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

Step3: Update the location of buffalos using:

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

Step 4: Is bg_{max} updating? Yes, go to 6. If No, return to Step 1.
Step 5: Return to algorithm step 2 if the stopping criteria are not met; otherwise, move on to step 6.
Step 6: Output the best solution.

FIGURE 3. (ABO) Algorithm

w_(k) represents the /waaa/ call. With reference to the buffalo k, this cry mobilises the herd to continue exploring. The m_(k) indicates the exploitable /maaa/ call. Consequently, w_(k+1)represents the need for more exploration, m_(k+1)represents the need for more exploitation, lp_1and lp_(2)represent the learning parameters, and a random number that can take any value between 0 and 1 depending on the problem being solved: the higher the value, the more exploitation and the less exploration.

6. NETWORK TOPOLOGY SCENARIO

The linearized (TCP/AQM) model was simulated in MATLAB (2017b) in order to assess the efficacy of the designed FPID controller and compare it to the PID controller. The link capacity (C) has been set to 3750 packets/second, or 15 Mbps. With a desired queue size of 300 packets and a propagation delay of 40 milliseconds, the Round Trip Time (R0) is 0.25 seconds. The utmost queue length will be 800 packets with a maximum number of 60 TCP sessions per source and destination. The proposed controller is configured on Router A, while Router B employs drop Tail. To evaluate the robustness of the FPID controller, a number of studies involving the modification of network parameter values have been conducted. Figure 4 displays the case studies' network topology.

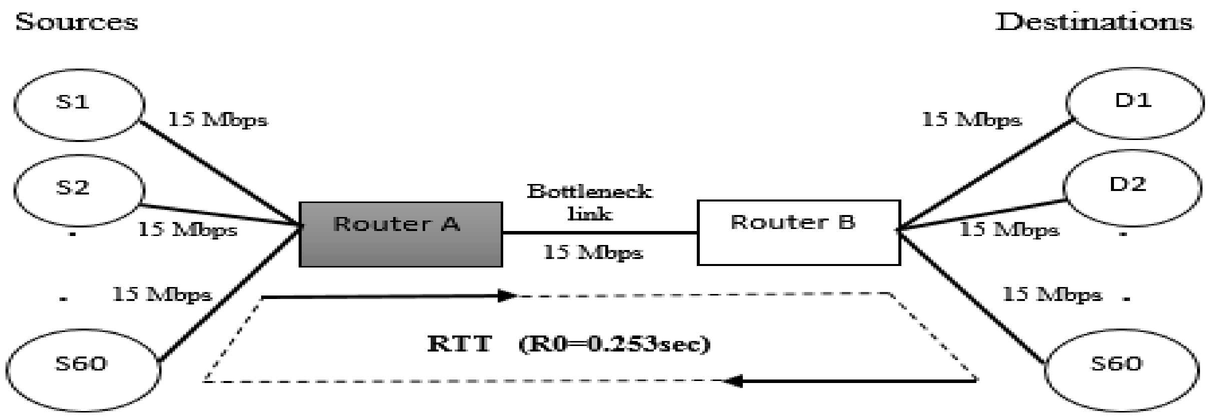


FIGURE 4. Network Topology Case Study.

7. SIMULATION RESULTS

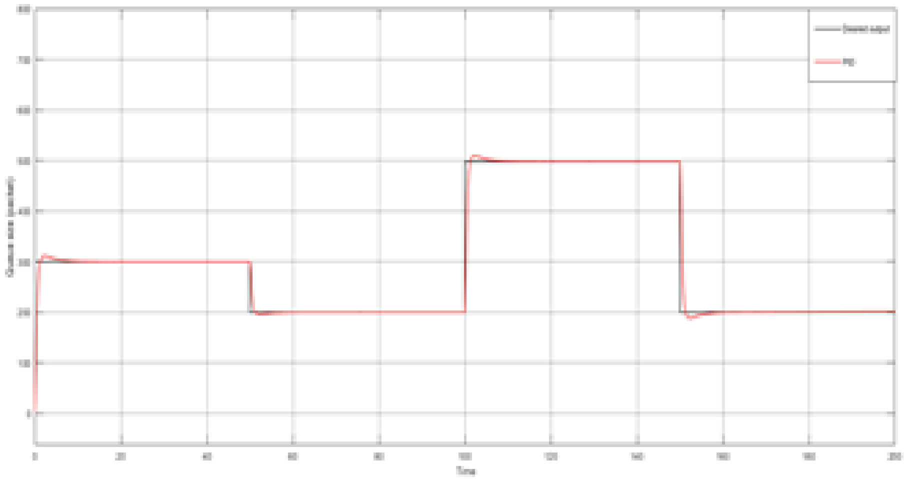
Consider the (TCP/AQM) with the network parameters specified in Section 6, as well as the reference input, which represents the queue size and varies every 50 seconds. This is how the simulation is carried out:

1. Test 1

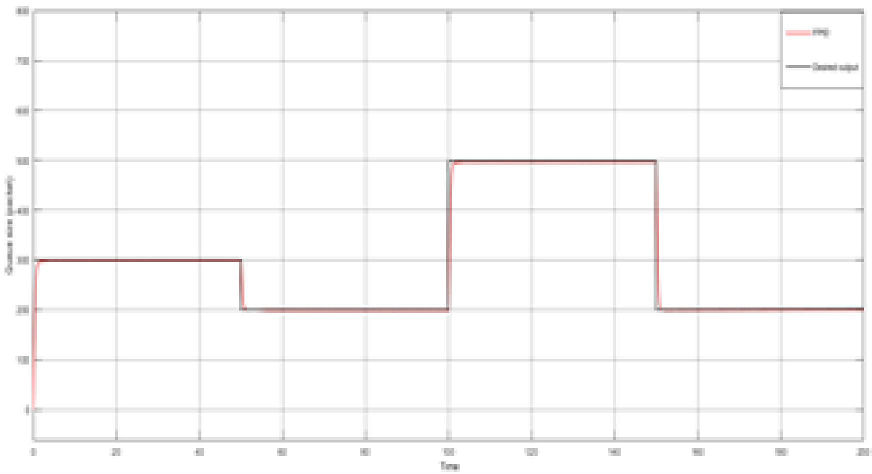
For the standard case study with the value of N is 60 and C is 15 Mbps, the response of the system for two controllers (PID, FPID) display in Figure 5. Where (PID, FPID) controllers can track desired queue length. The queue length performance of controllers is displayed in Table 4. The optimal values founded by ABO for (PID, FPID) displays in the Table 3 below.

TABLE 3. The values for (PID, FPID) controllers

Parameters	Kp	Ki	Kd	Ko
PID	1.76e-4	5.98e-5	4.67e-5	-
FPID	0.019	0.0058	0.048	0.025



(a) PID controller



(b) FPID controller

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

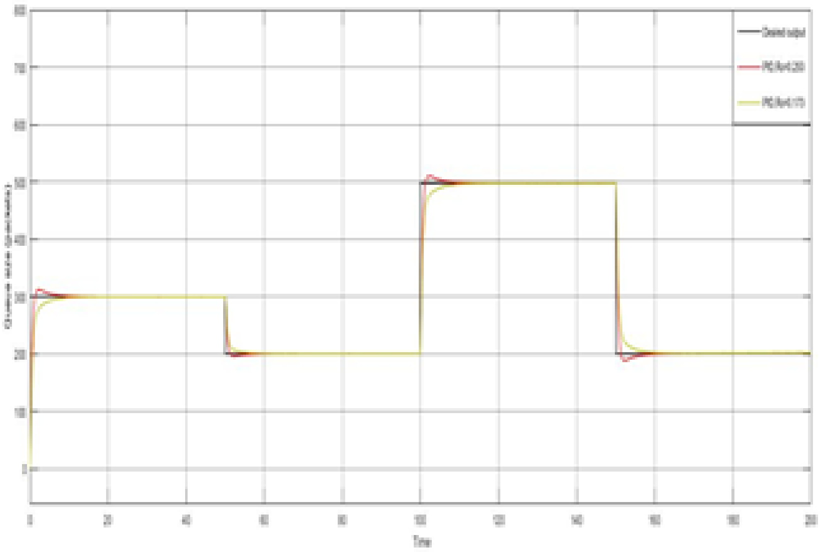
TABLE 4. Queue length Performance using the two controllers for N=60, C=15Mbps, R=0.253s

N=60, C=15 R=0.253	**PID**	**FPID**	**Enhancement rate(PID-FPID)**
Rise time (s)	0.7743	0.6159	20%
Settling time (s)	4.6161	1.1378	75%
Overshoot %	4.0261	0.0121	99%
IAE	178.13	169.05	5%

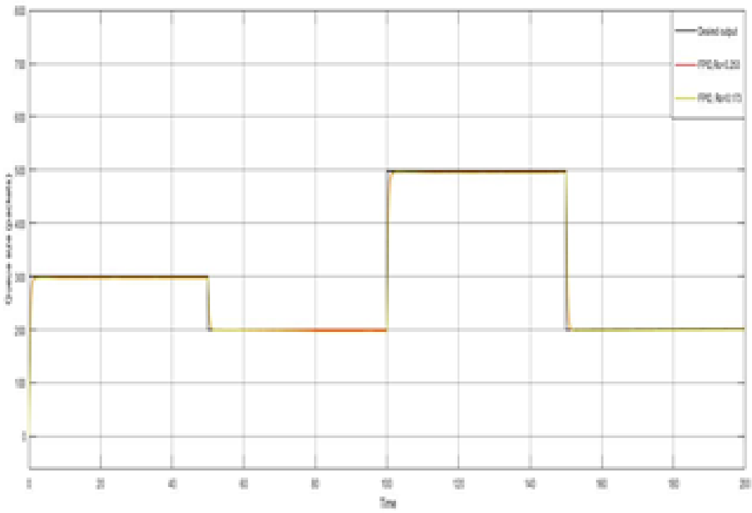
2. Test 2

To study the effect of decreasing round trip time (RTT) on queue length and behavior of the (PID, FPID) controllers in such case. The value of RTT is changed to be 0.173 s and the values of N and C are the same.

Web of Scholars: Multidimensional Research Journal (MRJ)
Volume: 02 Issue: 10 | 2023 ISSN: (2751-7543)
http://innosci.org



(a) PID controller



(b) FPID controller

FIGURE 6. The queue length for two controllers when N=60, C=15Mbps and R=0.173 s

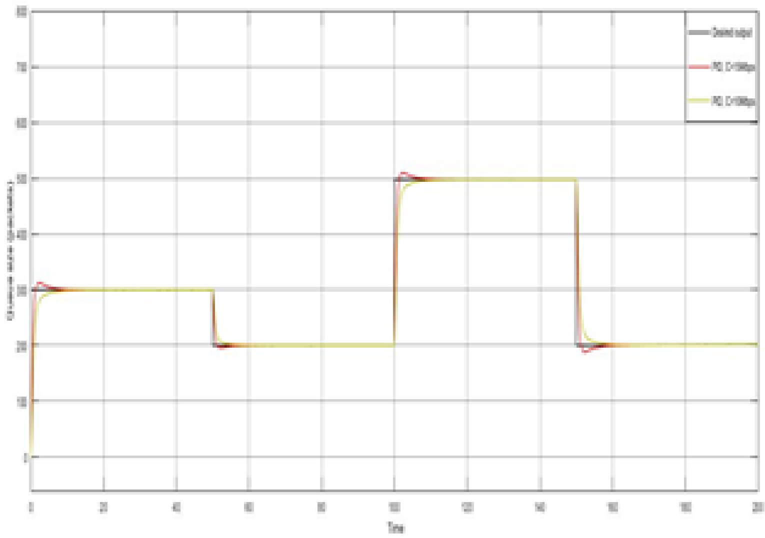
TABLE 5. Queue length Performance using the two controllers for N=60, C=15Mbps, R=0.173s

N=60, C=15 R=0.173	**PID**	**FPID**	**Enhancement rate(PID-FPID)**
Rise time (s)	1.5813	0.7107	55%
Settling time (s)	6.0291	1.2342	79%
Overshoot %	0.0262	0.2510	89%
IAE	264.43	245.54	7%

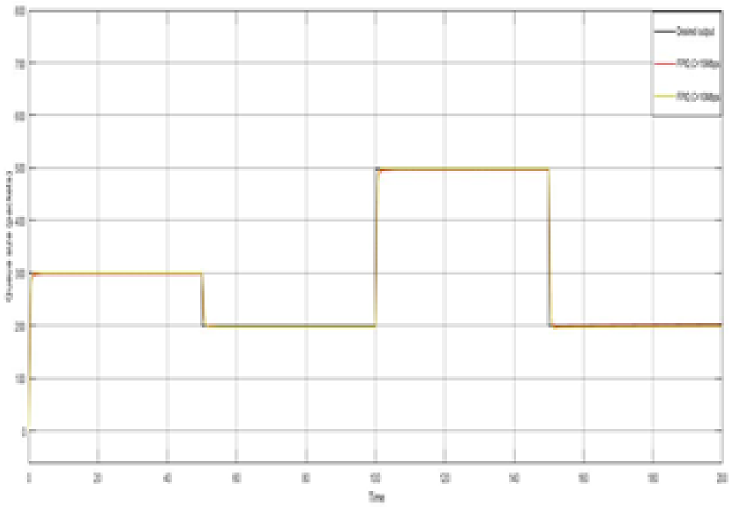
As clear from Figure 6. and Table 5. The rise time and settling time for the (PID-FPID) controllers keep in acceptable values.

3. Test 3

To study the effect of decreasing link capacity (C) on queue length and behavior of the (PID, FPID) controllers in such case to evaluate the robustness of the proposed controller (FPID). The value of C=10 Mbps and the value of N=60.



(a) PID controller



(b) FPID controller

FIGURE 7. The queue length for two controllers when N=60 and C=10Mbps, R=0.253 s

TABLE 6. Queue length Performance using the two controllers for N=60, C=10Mbps, R=0.253s

N=60, C=10 R=0.253	**PID**	**FPID**	**Enhancement rate(PID-FPID)**
Rise time (s)	1.6504	0.6299	61%
Settling time (s)	5.1075	1.0808	78%
Overshoot %	0.0430	0.1465	70%
IAE	285.34	116.79	59%

As clear from Figure 7. and Table 6. In the PID controller, the rise time and settling time increased which may be leading to packet loss and delay. While the FPID controller tries to keep acceptable values that means the system under control.

8. CONOCLUTION

Based on simulation results, the objective of this work to solve the congestion problem in computer networks was accomplished. The proposed controller (FPID) can manage congestion problems with good tracking performance in the desired queue level and faster system response than routers. The African buffalo optimisation algorithm assisted in choosing the optimal FPID parameters. Comparing the two controllers (PID, FPID), the results indicate that the FPID controller's system response is superior to the PID controller, with a 20% improvement in rise time and a 75% improvement in settling time to prevent delay and packet loss.

The proposed controller outperforms (FPID-PSO, FPID-ACO) controllers mention in [16] for the same case. See Table 7.

TABLE 7. Comparison of the proposed controller with other controllers

N=60, C=15MBPS	**Rise time (s)**	**Settling time (s)**	**Overshoot % packets**
PROPOSED CONTROLLER	0.6159	1.1378	0.0121
FPID-PSO	4.046	10.23	-
FPID-ACO	2.081	6.546	0.505

REFERENCES

1. V. Jacobson and M. J. Karels, “Congestion Avoidance and Control.” in Proc.ACM SIGCOMM, pp. 314-329, (1988).
2. G. F. Ahammed and R. Banu, “Analyzing the Performance of Active Queue Management Algorithms.” in (IJCNC) International Journal of Computer Networks & Communications, Vol.2, No.2, March (2010).
3. M. Z. Al-Faiz and S. A. Sadeq, “Particle Swarm Optimization Based Fuzzy-Neural Like PID Controller for TCP/AQM Router,” Intell. pp. 71–77, (2012).
4. Chrysostomou, C., & Pitsillides, A. “Congestion control in computer networks using fuzzy logic,” In Proceedings of the 10th WSEAS International Conference on Communications, ICCOM, vol. 6, pp. 551-556, Jul (2006).
5. H. Ashtiani, H. M. Pour, and M. Nikpour, "Active Queue Management in TCP Networks Based on Fuzzy-Pid Controller," Journal of Applied Computer Science & Mathematics, (2012).
6. J. Yang, etc., "LQ-servo congestion control for TCP/AQM system in a wireless network environment," Int. J. control Autom , vol. 6, no. 3, pp. 281–290, (2013).
7. M. Azadegan, M. T. H. Beheshti, and B. Tavassoli, “Design of proportional-derivative-type state feedback controllers for congestion control of transmission control protocol networks,” pp. 1774–1779, (2015).
8. Hind Ali, Osama A. Awad, "A Multi-Resolution Wavelet PID Controller for Internet AQM Routers," Int. J. Innov. Res. Comput. Commun. Eng. (An ISO Certif. Organ., vol. 3297, no. 6, pp. Vol. 4, Issue 5, May (2016).
9. Z. A. Karam, “Hybrid Fuzzy Congestion Controllers for Computer Networks Tuned by Modified Particle Swarm Optimization,” p. 17, (2018).
10. K. Sen, B. Chakraborty, A. Gayen, and C. Dey, “Fuzzy Rule-Based Set Point Weighting for PID Controller Kausik,” Springer Singapore, vol. 537, no. January (2018).
11. V. Misra, W.-B. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," in Proceedings of the Conference on

Applications, Technologies, Architectures, and Protocols for Computer Communication, pp. 151-160, (2000).

12. C. V. Hollot, V. Misra, D. Towsley, and W.-B. Gong, "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 (Cat. No. 01CH37213), pp. 1510-1519, (2001).
13. L. Reznik, Fuzzy Controllers, Australia, Newnes, Ch. 4, (1997).
14. J. B. Odili, M. N. M. Kahar, and S. Anwar, "African Buffalo Optimization: A Swarm-Intelligence Technique," Procedia Computer Science, vol. 76, pp. 443-448, (2015).
15. J. B. Odili and M. N. M. Kahar, "Numerical Function Optimization Solutions Using the African Buffalo Optimization Algorithm (ABO)," British Journal of Mathematics & Computer Science, vol. 10, pp. 1-12, (2015).
16. Hanan M Kadhim and Ahmed A Oglah," Interval type-2 and type-1 Fuzzy Logic Controllers for congestion avoidance in internet routers," IOP Conf. Series: Materials Science and Engineering 881 (2020) 012135.