IJE TRANSACTIONS A: Basics Vol. 34, No. 04, (April 2021) 891-900

International Journal of Engineering

Journal Homepage: www.ije.ir

Bandwidth Management with Congestion Control Approach and Fuzzy Logic

G. Azami, H. Gholizade Narm*

Faculty of Electrical Engineering, Shahrood University of Technology, Shahrood, Iran

PAPER INFO

Paper history:
Received 12 November 202
Received in revised form 07 February 2021
Accepted 15 February 2021

Keywords:
Bandwidth Management
Congestion Control
Fuzzy PID Controller
TCP/IP Networks

ABSTRACT

One of the problems with today's TCP/IP networks is their transmission system. If the bandwidth of a network is full, human and physical factors must be used for a new transmission system with a higher capacity to provide its bandwidth, which is very time consuming and costly. In this article, we proposed a method that in addition to the optimal use of available bandwidth, if the network capacity is full, it will be automatically transferred to a higher bandwidth network. For this purpose first, by designing a fuzzy PID controller for the existing network, it is tried to congestion control them and make use of it. It can be seen that the proposed controller performs much better in terms of an output response, following the queue length, stability and uncertainly, compared to the classical controller. If the input data to the network is increased, more packets are lost and this reduces the quality of the network. To solve this problem by using bandwidth management, by considering the threshold for packets loss in each network, if exceeding this limit, the existing network is switched to a network with a higher capacity and the problem of bandwidth and network quality is solved and causes subscriber satisfaction.

***doi**: 10.5829/ije.2021.34.04a.15*

NOMENCLATURE

W	Average TCP window size (packet)	e	Error
q	Average queue length (packet)	$\dot{e}$	Derivative of error
R(t)	Packets return time (s)	μ	Membership function
C	Link capacity (packet/s)	Bw	Bandwidth
Tp	propagation delay (s)	NB	Negative big
N	Number of resources	NM	Negative medium
P(t)	probability of the packet being marked	NS	Negative small
K_p	Proportional gain	ZO	Zero
K_i	Integral gain	PB	Positive big
K_d	Derivative gain	PM	Positive medium
K_u	Gain (Nichols method)	PS	Positive small
T_u	Period (Nichols method)	RTT	Round trip time(s)

INTRODUCTION

The subject of data congestion in networks based on Transmission Control Protocol (TCP) has become a major research topic in recent years [1]. Floyd and Jacobson [2] offered two solutions to the problem of congestion. The first solution suggests the solution in the initial and final points of sending packets, i.e. sender and receiver, and the second solution provides the adoption of the policy in the middle points of the network, i.e. routers. The first solution is generally interpreted as congestion control through TCP and the second solution is interpreted as Active Queue Management (AQM). In 1993, the Random Early Detection (RED) method was

---

*Corresponding Author Institutional Email: *gholizade@shahroodut.ac.ir* (H. Gholizade Narm)

Please cite this article as: G. Azami, H. Gholizade narm, Bandwidth management with congestion control approach and fuzzy logic, International Journal of Engineering Transactions A: Basics Vol. 34, No. 04, (2021) 891-900

projected [2], according to which the middle points of the network are randomly deleted before the packing queue. Through this method, it was determined according to the studies that it does not have the required efficiency and cause of the fluctuation of the queue length [3]. Consequently, numerous methods have been projected to develop the projected method, including Stability RED, Fuzzy RED [4], Adaptive RED [5], PD-RED (Proportional-Integral) methods. Notably, some have also tried to offer newer methods for active queue control, including BLUE [6], Multi-GREEN [7]. The nonlinear RED model has also been applied through the TRED method to develop the RED method [8]. Smart methods are also used for congestion control. A new AQM controller called a feed-forward neural network [9] has been projected to effectively control network congestion through establishing queue lengths. This pattern exhibits nonlinear and dynamic network traffic and predicts the future value of queue length. Moreover, we have created a good interaction between network operational capacity and delay through a learning system with loss prediction (LP) and reinforcement learning (RL) [10]. In another study, the problem of density in Bandwidth Delay Product (BDP) networks was solved through presenting a new Elastic-TCP method, resulting in a higher power than other Media Transport Protocol (MTP) [11].

Through mathematical models, it is feasible to envisage the behaviour of the system in different modes and conditions and to adopt the essential policy in dealing with such behaviour. The TCP network has always been regarded by researchers as a complex nonlinear system whose dynamics are frequently changing. A good deal of effort has been completed to mathematically recognize computer network models [12]. This is while one of the best models in this field can be regarded as the Misra model, introduced in 2000 [13]. This model is based on the fluid flow model and provides a nonlinear model of the network. Through presenting this model, different individuals attempted to use control theory to provide a method for active queue control. The basic idea was to use a control rule to accidentally remove the packets in the queue before the queue fills out.

The first controller to be offered was a simple Proportional Integral (PI) controller [14]. The PID controller was introduced, which performed better than the PI and RED controllers, and the process of changing parameters has less oscillation, leading to better queue adjustment. In this method, it was seen that increasing resources and the number of routers is better than control. PI and RED actuators operate and its oscillation is much less than the above methods, but in response to turbulence and uncertainty, the response is slower. A nonlinear congestion control model was used to overcome this problem; an example of which is the use of a slip mode controller [15]. In this method and through finding a rule of feedback control, the model is such that the source is stable and its application to the system has significantly overcome the turbulence and uncertainty. In another study, it uses PID-derived feedback control to correct AQM limitations, monitoring dynamic network changes at any time, and controlling congestion in these environments. In this method, it is observed that dynamically, tracking the length of the queue, the rate of loss of the package, and the use of the link are significantly improved [16].

Much research has been done on fuzzy logic about congestion control. One of the methods is presented in literature [17], which uses fuzzy logic to solve the problem of linearization and parameterization of RED and has developed the RED method with this device. To solve the problem of congestion in wireless sensor networks used in medical, agricultural and military equipment, a method has been proposed [18] in which, using a combination of fuzzy logic and PID, congestion it is detected earlier, and the length of the buffer queue is adjusted by determining the probability of packet loss. In another study, a fuzzy sliding mode congestion control (FSMC) algorithm is presented in literature [19] which, the queue buffer length is adjusted in congestion nodes through a new model of congestion control between the transfer layer and the MAC layer, leading to a decrease in delay and the packet loss and also, to improvement in network power.

In this study, we first examine the issue of active queue AQM management as the basis for different control systems and then obtain the mathematical and dynamic model of the TCP network. In the following, we examine two issues simultaneously. First, the Fuzzy PID control method is presented automatically to adjust the PID parameters, so that in addition to setting the parameters, we can automatically improve the disturbance and uncertainty, and compare it with the classical control method, and point out its advantages. Then, we control the congestion in the network according to the design. If the congestion control is no longer effective, we need to design a mechanism that allows us to use a network with more bandwidth and better performance with Bandwidth management. We propose a new method to solve this problem. By designing three networks with different bandwidths and managing this network, in addition to the optimal use of each network through congestion control, if congestion in each network significantly increases, by taking into account the threshold and using switching system. We moved the existing network to a network with more bandwidth. In this way, the appropriate bandwidth is allocated to each network. The important advantage of this method is that it always allocates the appropriate bandwidth to the subscribers and makes them satisfied.

2. ACTIVE QUEUE MANAGEMENT

Active queue management is applied in the middle points and to control congestion. The initial encounters with the problem of congestion in the middle points were such that the router would insert the packets into the buffer for processing, and if the buffer was full, the newly received packets would be destroyed. In this case, the packets that reached the buffer were processed earlier, which is the same as the implementation of the First-In-First-Out (FIFO) method. This method is called Drop Tail. The instability of the queue in the middle points causes the difference between delays, and this is inappropriate for Real-Time users. In AQM, the queue length (number of packets left in a queue) is tried to remain at a reference point determined by the designer or operator toward preventing packet overflows. Any deviation of the queue length from the reference results in changing from the router to the network resources through sending commands so that the queue length returns to the reference point. AQM can be applied as a controller for the TCP network to keep the queue length constant, assuming that its output is the probability of destroying packets and its input is the amount of congestion. Therefore, the theory of control can be applied to form a control law, which is the same as the possibility of eliminating packets. Many methods based on this principle have been projected for AQM, such as RED, PI, PID, etc [20].

2. 1. TCP/IP Network Mathematical Model

A mathematical model of the network was introduced in 2000. Based on the studies, this model is closer to the real system. It describes the mathematical model of an AQM that supports TCP currents and designs an AQM-resistant controller. The controller is designed for the N data flow of TCP. The shutdown time (RTT) of each stream is defined as follows:

$$R(t) = \frac{q(t)}{c} + T_p \tag{1}$$

One dynamic model of TCP behaviour is the use of fluid flow and statistical differential equation analysis [13]. These dynamics include the same N number of TCP sources and a router to determine the average amount of changes and dynamics queue the router (Figure 1).

This model depends on key network changes and it is described through the following nonlinear differential equations.

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

The above equation consists of two increasing and decreasing sections. The increasing section indicates that the amount of compression window increases by one unit for the time changes of a time shift. This happens during the TCP compression prevention phase. The decreasing section indicates that the value of the window decreases to one half in the event of a loss. The second part of the AQM model is for the middle points, which should be written based on the dynamics of queue length changes. The dynamics of this section are expressed as Equation (3).

$$\dot{q} = \begin{cases} -C + \dfrac{\mathrm{N}(t)}{R(t)}W(t) & q \succ 0 \\ \max\left\{0, -\mathrm{C} + \dfrac{N(t)}{R(t)}W(t)\right\} & q = 0 \end{cases} \tag{3}$$

Equation (3) includes two increasing and decreasing sections. The decreasing section, as the capacity of the midpoints, represents the amount of packet processing per time unit, and the increasing section represents the number of packets reaching queue per time unit. In this equation, the queue length is modelled through the difference between NW/R and the C link capacity. In Equations (1), (2), and (3), the parameters are defined as follows:

W: Average TCP window size (packets); q: Average queue length (packets); R(t): packets return time (seconds); C: Link capacity (seconds/pack); q(t)/C: queue delay; Tp: propagation delay (seconds); P(t): the probability of the packet being marked in the router. By aligning the nonlinear Equations of (1), (2), and (3) around the working point and assuming that N(t)= N and C(t)= C and placing $\dot{w} = 0$ and $\dot{q} = 0$ we reach the following values.

$$\begin{aligned} &\dot{w} = 0 \Rightarrow w_0^2 p_0 = 2 \\ &\dot{q} = 0 \Rightarrow w_0 = \frac{R_0 C}{N} \\ &R_0 = \frac{\dot{q}}{C} + T_p \end{aligned} \tag{4}$$

In linearization, we do not consider the dependence of $t - R$ time delay parameters on the length of the queue (q) and assume that the time delay is replaced through a constant value $t - R_0$. In other words, we do not consider the dependence of the return time depending on the queue length in the dynamic parameters.

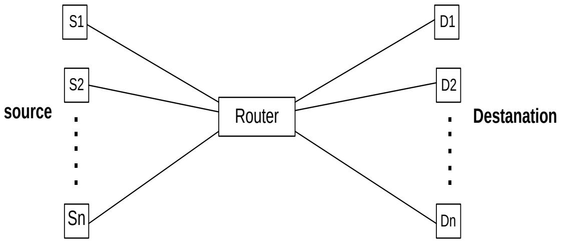

Figure 1. Network topology with N similar sources TCP and a router

$$\dot{w}(t) = \frac{1}{R_0} - \frac{w(t)}{2}\,\frac{w(t-R_0)}{\frac{q(t-R_0)}{C} + T_p}\,p(t-R_0)$$

$$\dot{q}(t) = \begin{cases} -c + \dfrac{N}{\frac{q(t-R_0)}{C}}\,w(t) & q \succ 0 \\ \max\left\{0, -c + \dfrac{N}{\frac{q(t-R_0)}{C}}\,w(t)\right\} & q = 0 \end{cases} \tag{5}$$

Therefore, linearization at the equilibrium point leads to the following equations:

$$\delta\dot{w}(t) = -\frac{2N}{R_0{}^2 C}\,\delta w(t) - \frac{R_0 C^2}{2N^2}\,\delta p(t-R_0)$$
$$\delta\dot{q}(t) = \frac{N}{R_0}\,\delta w(t) - \frac{1}{R_0}\,\delta q(t) \tag{6}$$

where,

$$\delta w = w - w_0\,,\ \delta q = q - q_0\,,\ \delta p = p - p_0 \tag{7}$$

From the Laplace transform of the Equations (6) and (7), the system conversion function is obtained as follows. the nonlinear Equations (1), (2), and (3) around the operating point (w_0, q_0, p_0) and assuming that N(t)=N, C(t)=C, and by putting $\dot{q} = 0$, $\dot{w} = 0$, we get the dynamic Equation (4).

$$\frac{\frac{(\mathrm{RC})^3}{4N^2}e^{-Rs}}{(\mathrm{Rs}+1)(\frac{R^2C}{2N}s+1)} \tag{8}$$

This dynamic equation of congestion control is in the TCP/IP network, as the basis for our further discussion.

3. CONGESTION CONTROL USING FUZZY PID

Due to the change of network parameters, the need for a controller that can respond to changes properly and keep the system stable seems necessary. One of the suitable methods for this work is fuzzy methods that can adapt the system to changes and uncertainties and also adjust the parameters. In congestion control, fuzzy methods have been able to receive changes in the network and adjust the controller parameters based on it and send the appropriate control signal to achieve the desired output [21]. In this research, the fuzzy controller method has been used to control congestion. In this method, using the swarm dynamics, we design a PID controller so that the controller details are adjusted by the above algorithm and then a suitable control signal is generated for the process. In the fuzzy control method, using the error and error derivative and applying the appropriate rules on these parameters, the PID controller coefficients are adjusted so that a suitable signal is generated for the process. This is shown in Figure 2 [22].

3. 1. Mathematical Topics To implement the fuzzy PID control, we define the control signal U (t) based on the PID as follows.

$$G_C(S) = K_p + \frac{K_i}{S} + K_D S = K_p(1 + \frac{1}{T_i S} + T_d S)$$
$$K_i = \frac{K_p}{T_i}, K_d = K_p T_d \tag{9}$$

To determine the coefficients of the PID controller, we can use the Ziegler and Nichols method. For this purpose, we put the controller in P mode, and by increasing the controller gain, we act so that the system output is on the verge of fluctuation. In this case, the gain is K_u and Consider the oscillation period T_u. The following values of PID controller values are obtained as follows [23]:

$$K_p = 0.6K_u, K_i = 2k_p / k_u, K_d = k_p T_u / 8 \tag{10}$$

After obtaining the controller finances, which we also use in the fuzzy controller, we describe the fuzzy system in this research. To begin with, instead of K_p, K_d and K_i, we get K_p', K_d' and α, where $\alpha = T_i/T_d$ is. As a result, the value of K_i is obtained using Equation (9) and the value of α is obtained as follows.

$$K_i = \frac{K_p}{T_i}, T_i = \alpha T_d, T_d = \frac{K_d}{K_p} \rightarrow K_i = \frac{K_p{}^2}{\alpha K_d} \tag{11}$$

In the following, instead of obtaining the absolute values of the controlling fines, we obtain their normalized values by the following method.

$$k_p' = \frac{k_p - k_p{}^{\min}}{k_p{}^{\max} - k_p{}^{\min}} \in [0,1]$$
$$k_d' = \frac{k_d - k_d{}^{\min}}{k_d{}^{\max} - k_d{}^{\min}} \in [0,1] \tag{12}$$

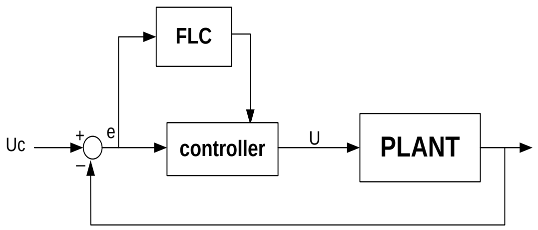

Figure 2. Fuzzy controller block diagram network topology with N similar sources TCP and a router

According to Figure 3 with error input and error derivative, fuzzy system outputs are defined as follows. The values of K_p, K_d and K_i are calculated for the values of K'_p, K'_d and α To obtain fuzzy PID coefficients, we must carefully define fuzzy rules. For this purpose, we define the rules as follows:

$$\begin{aligned} &\text{If } e \text{ is } A_i \text{ and } \dot{e} \text{ is } B_i \text{ , then } k'_p \text{ is } \begin{Bmatrix} Big \\ Small \end{Bmatrix} \\ &\qquad k'_d \text{ is } \begin{Bmatrix} Big \\ Small \end{Bmatrix} \\ &\qquad \alpha = 2,3,4,5 \end{aligned} \tag{13}$$

For this purpose, we specify the membership function for e and $\dot{e}$ in Figure 4 [21, 24].

For get K'_p and K'_d we consider the membership functions graphically as shown in Figure 5. According to Figure 5, we consider the membership function as follows [25]:

$$\begin{aligned} &\mu_{Small}(x) = \min(-\frac{1}{4}\ln x, 1) \\ &\mu_{Big}(x) = \min(-\frac{1}{4}\ln(1-x), 1) \end{aligned} \tag{14}$$

We also use a fuzzy singleton for α.

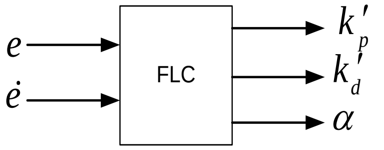

Figure 3. Fuzzy system input and output

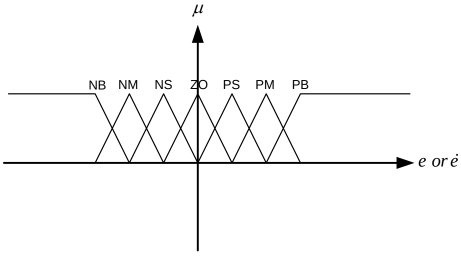

Figure 4. Membership functions for e and $\dot{e}$

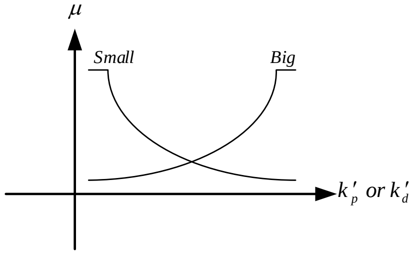

Figure 5. Membership functions for K'_p and K'_d

In the following, we consider the following conditions to determine the fuzzy rules.

$$\begin{aligned} &Rule\ i: \\ &\text{If } e \text{ is } A_i, \dot{e} \text{ is } B_i, \text{Then } K'_p \text{ is } C_i, K'_d \text{ is } D_i, \alpha \text{ is } \alpha_i \\ &\mu_i \propto \mu_{A_i}(e)\mu_{B_i}(\dot{e}) \quad \sum \mu_i = 1 \end{aligned} \tag{15}$$

As a result, the output values of the fuzzy system are equal to:

$$K'_p = \sum_i \mu_i k'_{p_i}, K'_d = \sum_i \mu_i k'_{d_i}, \alpha = \sum_i \mu_i \alpha_i \tag{16}$$

To obtain the fuzzy system rules table, we use the appropriate answer. In this case, for different points of this answer, we must analyze the performance of the fuzzy system to obtain its rules. Suppose the response of a system is in Figure 6.

The purpose of fuzzy system design is to achieve the appropriate response of Figure 6. For this purpose, we follow the answer path. If we want to get from point a_1 to b_1, we have to increase K_p K_i and decrease K_d. Also, if we want to change the answer around the reference value, to follow the path from b_1 to c_1, we have to decrease K_p K_i and increase K_d. Then, in order to reach from c_1 to d_1, K_p and K_i have to increase and K_d decrease. Also, similar to the previous case, to reach from point a_1 to b_1, must be increased K'_p, K'_d and α must be reduced by using relations $K_i = \frac{K_p}{T_i}$ and $\alpha = T_i/T_d$. Other changes are obtained according to the answer.

For this purpose, using the above method and the membership function of Figures 4 and 5 we obtain the fuzzy rules, which you can see in the Tables 1, 2 and 3 [21, 22, 26].

In Table 1, we consider the gain changes K'_p as follows. If the error ***e(k)*** is a large value (NB), regardless of the value of the error derivative ***ė(k)*** , we choose a large gain of K'_p (B). If the error value decreases (NM), and also the error derivative value decreases (NM or NS), we consider the value of gain K'_p to be large(B). If the error value is zero (ZO) and the error derivative value is

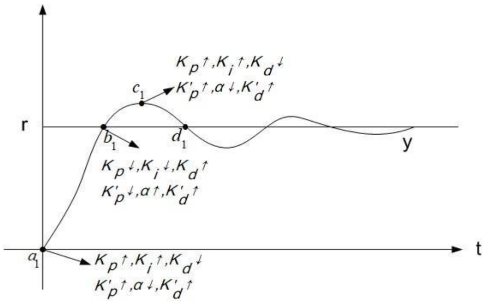

Figure 6. The response of the second-order system for determining fuzzy rules

low, we consider the value gain K_p' to be small(S), unless the derivative of the error is zero, in this case, we choose a large gain of K_p' (B). In the same way, different amounts of gain K_d' and α obtained based on the error and the error derivative.

From the rules of Tables 1 to 3, the values of K_p', K_p' obtained from Equations (11) and (12), and PID control coefficients are obtained as follows:

$$\begin{aligned} K_P &= K_p^{\min} + (k_p^{\max} - k_p^{\min})k_p' \\ K_d &= K_d^{\min} + (k_d^{\max} - k_d^{\min})k_d' \\ K_i &= \frac{K_p^2}{\alpha K_d} \end{aligned} \qquad (17)$$

where in:

$$\begin{cases} k_p^{\min} = 0.32k_u \\ k_p^{\max} = 0.6k_u \end{cases} \quad \begin{cases} k_d^{\min} = 0.08k_u T_u \\ k_d^{\max} = 0.15k_u T \end{cases} \qquad (18)$$

Using the Zicklor and Nichols method [27, 28], we can first adjust the PID parameters and use the values k and t obtained from it to design the rules of the fuzzy system and extract the final PID values to produce the fuzzy controller control signal. We will explain this in the simulation section of this article.

TABLE 1. Fuzzy system rules for K_p'

		ė(k)						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	B	B	B	B	B	B	B
	NM	S	B	B	B	B	B	S
e(k)	**NS**	S	S	B	B	B	S	S
	ZO	S	S	S	B	S	S	S
	PS	S	S	B	B	B	S	S
	PM	S	B	B	B	B	B	S
	PB	B	B	B	B	B	B	B

TABLE 2. Fuzzy system rules for K_d'

		ė(k)						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	S	S	S	S	S	S	S
	NM	B	B	S	S	S	B	B
e(k)	**NS**	B	B	B	S	B	B	B
	ZO	B	B	B	B	B	B	B
	PS	B	B	B	S	B	B	B
	PM	B	B	S	S	S	B	B
	PB	S	S	S	S	S	S	S

TABLE 3. Fuzzy system rules for α

		ė(k)						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	2	2	2	2	2	2	2
	NM	3	3	2	2	2	3	3
e(k)	**NS**	4	3	3	2	3	3	4
	ZO	5	4	3	3	3	4	5
	PS	4	3	3	2	3	3	4
	PM	3	3	2	2	2	3	3
	PB	2	2	2	2	2	2	2

4. BANDWIDTH MANAGEMENT WITH CONGESTION CONTROL APPROACH AND FUZZY LOGIC

In previous discussions, it was explained how to control network congestion with a fuzzy method so that the network can remain stable; but in some cases, the network inputs are so high that the existing network no longer responds to network data despite the congestion control system. In this case, the data packets are largely deleted and the congestion control response to the system is not available. An appropriate solution must be found for this issue. In this paper, the proposed solution is to set a threshold for deleting packages. As long as the removal of packets is less than this threshold, the existing system will work without change. If the amount of packet deletion exceeds the threshold, use a switching system to transfer the network to a network with more bandwidth so that it can have more capacity to transfer data. With this method, in addition to optimizing the use of the existing network, if the network is full, we can transfer to the network with more capacity using the designed system and keep the network stable and cause more satisfaction of data subscribers. We display this in Figure 7.

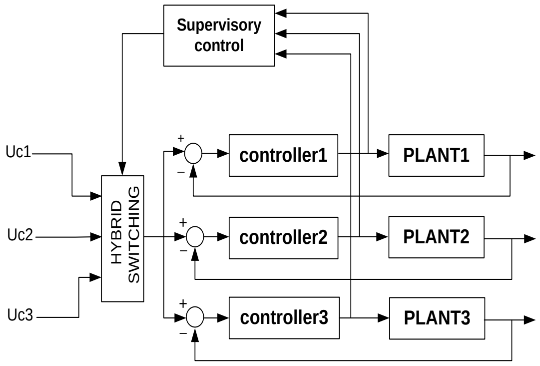

Figure 7. Bandwidth management

5. SIMULATION

To simulate bandwidth management with congestion control and fuzzy logic approach, we consider three networks with bandwidths of 10Mb/s, 50Mb/s, and 100Mb/s. We first test the network with the specifications of N=60, RTT=200ms, and C=152paket/s or bandwidth of 10mb/s. With these features, the network's congestion dynamics are equal to:

$$\frac{194.1e^{-0.2t}}{S^2+24.67S+98.3} \tag{19}$$

In this mode, we simulate congestion control for both classical control modes and fuzzy logic for the buffer with the desired queue of 100 using MATLAB software.

As you can see in Figure 8, the fuzzy PID controller outperforms the classical PID controller in terms of output response and system speed in terms of congestion control.

In Figure 9, you can see that the control signal U shows a value of 50, which is the same amount of packet loss in this network, and its value is acceptable. Next, we increase the input data to the network from N=60 to N=100, and we want to see how congestion control works with existing network features. In this case, the network dynamics with N=100, RTT=200ms, and C=152packet/s or bandwidth of 10mb/s is obtained as follows:

$$\frac{116e^{-0.2t}}{S^2+37.77S+163.8} \tag{20}$$

You can see the simulation of the above network with the changes made with the desired queue of 100 for the output response and the control signal in Figures 10 and 11.

According to Figure 10, it can be seen that by increasing the input data to the network from N = 60 to N = 100, you follow the desired value of 100 well, but the control signal, which is the packets loss, increases from 50 to 141, which packets loss is much higher than the standard value and is not acceptable for the existing network.

To solve this problem, we consider a network with a higher capacity and a threshold value of 80 to remove packets. If the control signal, which is the removal of packets, exceeds the threshold value of 80, the existing network does not respond to the input data and must be switched to a higher capacity network. In this case, a network with N=100 and RTT=200ms and capacity

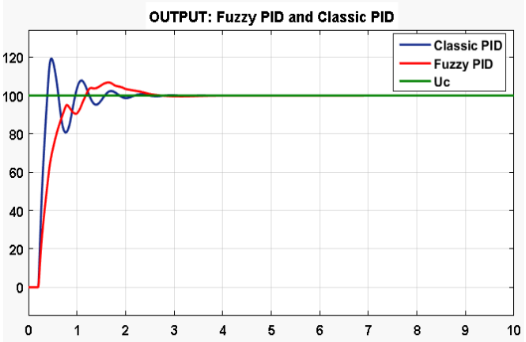

Figure 8. Congestion Control Output Based on Controller Fuzzy PID and Classic PID with BW=10mb/s, N=60, Queue Length=100

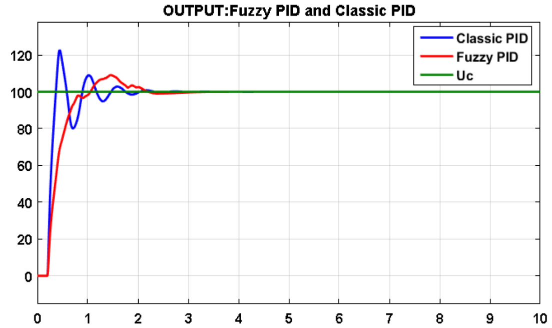

Figure 10. Congestion Control Output Based on Controller Fuzzy PID and Classic PID with BW=10mb/s, N=100, Queue Length=100

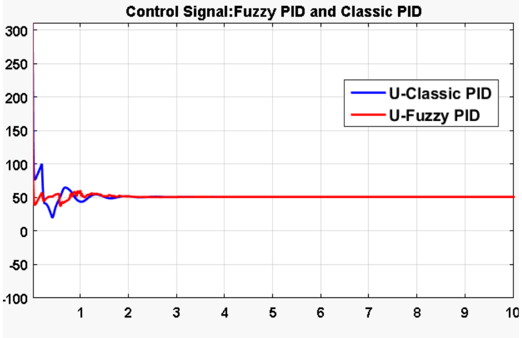

Figure 9. Control Signal Based on Controller Fuzzy PID and Classic PID with BW=10mb/s, N=60, Queue Length=100

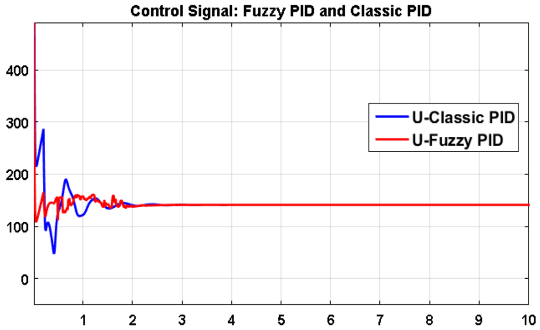

Figure 11. Control Signal Based on Controller Fuzzy PID and Classic PID with BW=10mb/s, N=100, Queue Length=100

consider C=50mb/s. In this case, the network dynamics are as follows:

$$\frac{88e^{-0.2t}}{S^2+11.553S+32.765} \tag{21}$$

We performed the simulation for this network with the desired queue value of 100.

As observed in Figures 12 and 13, in the new network, in addition to controlling congestion and making optimal use of it, the queue length of 100 is well followed and the packets' loss is greatly decreased (41 packets). If we would apply the previous network with this number of resources, the number of packets loss would reach 141, which is not acceptable.

5. 1. Effects of Uncertainty in the Fuzzy PID Controller

To study the effects of uncertainty in the fuzzy PID controller designed in Section 3.1, we first change one of the plant coefficients and observe the system response to this change. To change, we assume that the f coefficient b changes as follows.

$$b+[\alpha^-,\alpha^+] \tag{22}$$

We considered the value of α equal to 0.5. In this case, according to the value of b in the plant (19), which is 24.67, we apply the values of b as follows.

$$[24.17,\ 25.17] \tag{23}$$

First, we consider the value b to be 24.17 and apply it to plant (19). The plant is obtained as follows.

$$\frac{194.1e^{-0.2t}}{S^2+24.17S+98.3} \tag{24}$$

We perform a simulation with these specifications.

Then, considering b equal to 25.17, the plant it is obtained as follows.

$$\frac{194.1e^{-0.2t}}{S^2+25.17S+98.3} \tag{25}$$

The above plant simulation is obtained as follows.

We simulated the same conditions for the plant (21) with a coefficient (b=11.553) with a value of α equal to 0.5. The value of b is obtained as follows:

$$[11.053,\ 12.053] \tag{26}$$

By applying the value b to the plant, the following values are obtained and we perform a simulation for each.

$$\frac{88e^{-0.2t}}{S^2+11.053S+32.765},\ \frac{88e^{-0.2t}}{S^2+12.053S+32.765} \tag{27}$$

As shown in Figures 14 to 17, when changing the coefficient b, the fuzzy PID controller does not change,

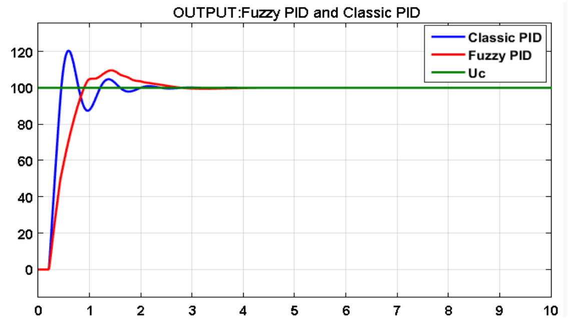

Figure 12. Congestion Control Output Based on Controller Fuzzy PID and Classic PID with BW=50mb/s, N=100, Queue Length=100

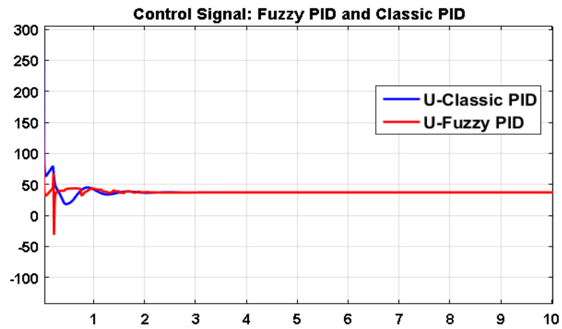

Figure 13. Control Signal Based on Controller Fuzzy PID and Classic PID with BW=50mb/s, N=100, Queue Length=100

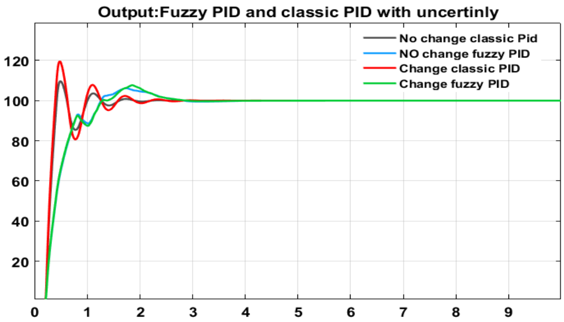

Figure 14. The output of classical PID and fuzzy PID control systems by changing the coefficient b (b=24.17)

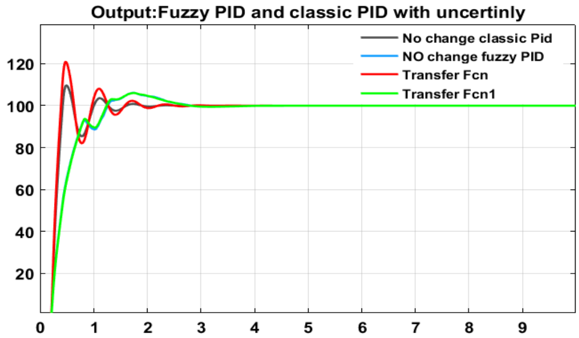

Figure 15. The output of classical PID and fuzzy PID control systems by changing the coefficient b (b=25.17)

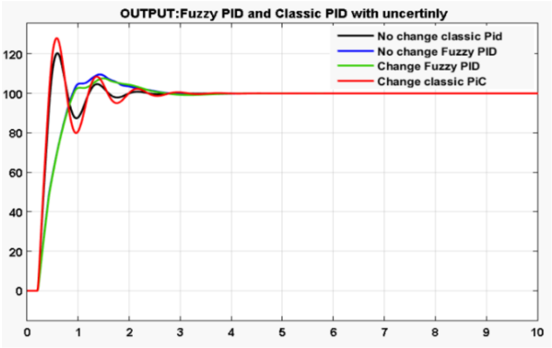

Figure 16. The output of classical PID and fuzzy PID control systems by changing the coefficient b (b=11.053)

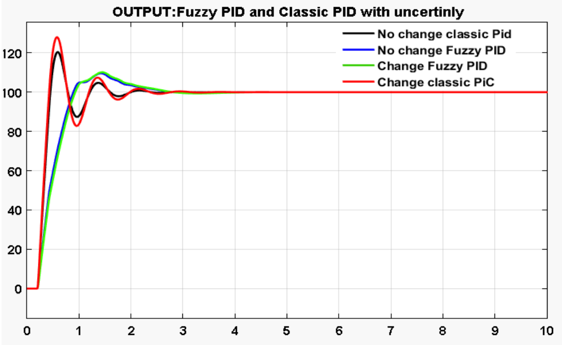

Figure 17. The output of classical PID and fuzzy PID control systems by changing the coefficient b (b=12.053)

but the classic PID controller significantly changes. In this case, we conclude that the fuzzy PID controller responds well to uncertainty.

6. CONCLUSION

In this study, we examined two issues, first by designing a fuzzy PID controller and considering the appropriate rules, we solved the problem of congestion control for the network and at the same time, we made optimal use of the existing network. The designed fuzzy PID controller showed how the desired queue length is well followed and by simulation, we showed that the output of the fuzzy PID controller is much better in terms of an output response, overshoot, speed and uncertainly than the classic controller. In the next step, which was the main topic of this research, it was suggested that the amount of packets loss is limited and is acceptable to an extent that does not exceed the threshold. If the packets loss increases too much, to overcome it, we designed a mechanism that includes different networks, each with its characteristics and designed with different capacities. If we use a network whose number of packets loss exceeds the allowable limit, with a management and supervisory system and using hybrid switching, it detects this issue and switches to a network with a higher capacity. By doing this, we always make sure that data subscribers never run out of bandwidth, and their satisfaction with the operators' increases.

7. REFERENCES

1. Clark, D., "The design philosophy of the DARPA internet protocols", ***ACM SIGCOMM Computer Communication Review***, Vol. 18, No. 4, (1988), 106–114. doi:10.1145/52325.52336
2. Floyd, S., and Jacobson, V., "Random early detection gateways for congestion avoidance", ***IEEE/ACM Transactions on Networking***, Vol. 1, No. 4, (1993), 397–413. doi:10.1109/90.251892
3. Christiansen, M., Jeffay, K., Ott, D., and Smith, F. D., "Tuning RED for Web traffic", ***IEEE/ACM Transactions on Networking***, Vol. 9, No. 3, (2001), 249–264. doi:10.1109/90.929849
4. Preye, U. H., and Nneka, O. L., "An Intelligent Fuzzy Logic System for Network Congestion Control", ***Circulation in Computer Science***, Vol. 2, No. 11, (2017), 23–30. doi:10.22632/ccs-2017-252-69
5. Domańska, J., and Domański, A., "Adaptive RED in AQM", In International Conference on Computer Networks, (2009), 174–183 Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-02671-3_21
6. Wu-chang Feng, Shin, K. G., Kandlur, D. D., and Saha, D., "The BLUE active queue management algorithms", ***IEEE/ACM Transactions on Networking***, Vol. 10, No. 4, (2002), 513–528. doi:10.1109/TNET.2002.801399
7. Hwang, L.-C., "M-GREEN: An active queue management mechanism for multi-QoS classes", ***Computer Standards & Interfaces***, Vol. 36, No. 1, (2013), 122–131. doi:10.1016/j.csi.2013.07.007
8. Feng, C.-W., Huang, L.-F., Xu, C., and Chang, Y.-C., "Congestion Control Scheme Performance Analysis Based on Nonlinear RED", ***IEEE Systems Journal***, Vol. 11, No. 4, (2017), 2247–2254. doi:10.1109/JSYST.2014.2375314
9. Bisoy, S. K., and Pattnaik, P. K., "An AQM Controller Based on Feed-Forward Neural Networks for Stable Internet", ***Arabian Journal for Science and Engineering***, Vol. 43, No. 8, (2018), 3993–4004. doi:10.1007/s13369-017-2767-9
10. Daigavhane, M. U., and Chawhan, M. D., "Congestion control algorithm for TCP in wireless network", In 2018 4th International Conference on Recent Advances in Information Technology (RAIT), (2018), 1–4. doi:10.1109/RAIT.2018.8388967
11. Alrshah, M. A., Al-Maqri, M. A., and Othman, M., "Elastic-TCP: Flexible Congestion Control Algorithm to Adapt for High-BDP Networks", ***IEEE Systems Journal***, Vol. 13, No. 2, (2019), 1336–1346. doi:10.1109/JSYST.2019.2896195
12. Tanenbaum, A. S., Computer Networks, (2003), Pearson Education.
13. Misra, V., Gong, W.-B., and Towsley, D., "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 - SIGCOMM '00, (2000), 151–160. doi:10.1145/347059.347421
14. Hollot, C. V., Misra, V., Towsley, D., and Wei-Bo Gong, "On designing improved controllers for AQM routers supporting TCP flows", Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat.

No.01CH37213), Vol. 3, (2001), 1726–1734. doi:10.1109/INFCOM.2001.916670

15. Barzamini, R., and Shafiee, M., "A new sliding mode controller for TCP congestion control", In 2011 19thTelecommunications Forum (TELFOR) Proceedings of Papers, (2011), 214–217. doi:10.1109/TELFOR.2011.6143529
16. Bisoy, S. K., and Pattnaik, P. K., "Design of feedback controller for TCP/AQM networks", ***Engineering Science and Technology, an International Journal***, Vol. 20, No. 1, (2017), 116–132. doi:10.1016/j.jestch.2016.10.002
17. Abualhaj, M. M., Abu-Shareha, A. A., and Al-Tahrawi, M. M., "FLRED: an efficient fuzzy logic based network congestion control method", ***Neural Computing and Applications***, Vol. 30, No. 3, (2018), 925–935. doi:10.1007/s00521-016-2730-9
18. Rezaee, A. A., and Pasandideh, F., "A Fuzzy Congestion Control Protocol Based on Active Queue Management in Wireless Sensor Networks with Medical Applications", ***Wireless Personal Communications***, Vol. 98, No. 1, (2018), 815–842. doi:10.1007/s11277-017-4896-6
19. Qu, S., Zhao, L., and Xiong, Z., "Cross-layer congestion control of wireless sensor networks based on fuzzy sliding mode control", ***Neural Computing and Applications***, Vol. 32, No. 17, (2020), 13505–13520. doi:10.1007/s00521-020-04758-1
20. Fan Yanfie, Ren Fengyuan, and Lin Chuang, "Design a PID controller for active queue management", In Proceedings of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003, (2003), 985–990. doi:10.1109/ISCC.2003.1214244
21. Mohammadi, S., Pour, H. M., Jafari, M., and Javadi, A., "Fuzzy-based PID active queue manager for TCP/IP networks", In 10th International Conference on Information Science, Signal Processing and Their Applications (ISSPA 2010), (2010), 434–439, IEEE, 434–439. doi:10.1109/ISSPA.2010.5605462
22. Karam, Z. A., "Hybrid Fuzzy Congestion Controllers for Computer Networks Tuned by Modified Particle Swarm Optimization", ***International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems***, Vol. 7, No. 2, (2018), 17-26. doi:10.11601/ijates.v7i2.250
23. Taeib, A., Ltaeif, A., and Chaari, A., "A PSO Approach for Optimum Design of Multivariable PID Controller for nonlinear systems", In International Conference on Control, Engineering & Information Technology, Proceedings Engineering & Technology, (2013), 206–210.
24. Somwanshi, D., Bundele, M., Kumar, G., and Parashar, G., "Comparison of Fuzzy-PID and PID Controller for Speed Control of DC Motor using LabVIEW", ***Procedia Computer Science***, Vol. 152, (2019), 252–260. doi:10.1016/j.procs.2019.05.019
25. Zhen-Yu Zhao, Tomizuka, M., and Isaka, S., "Fuzzy gain scheduling of PID controllers", ***IEEE Transactions on Systems, Man, and Cybernetics***, Vol. 23, No. 5, (1993), 1392–1398. doi:10.1109/21.260670
26. Farshad Samadi, and Hamid Moghadam-Fard, "Active Suspension System Control Using Adaptive Neuro Fuzzy (ANFIS) Controller", ***International Journal of Engineering, Transactions C: Aspects***, Vol. 28, No. 3, (2015), 396–401. doi:10.5829/idosi.ije.2015.28.03c.08
27. Kumar, P., and Kumar Chaudhary, S., "Stability and Robust Performance Analysis of Fractional Order Controller over Conventional Controller Design", ***International Journal of Engineering, Transaction B: Applications***, Vol. 31, No. 2, (2018), 322–330. doi:10.5829/ije.2018.31.02b.17
28. Kumar, V., and Patra, A., "Application of Ziegler-Nichols method for tuning of PID controller", ***International Journal of Electrical and Electronics Engineering***, Vol. 8, No. 2, (2016), 559–570

Persian Abstract

چکیده

یکی از مشکلات شبکه‌های امروزی سیستم انتقال آنهاست. در صورت پرشدن پهنای باند یک شبکه باید عوامل انسانی و فیزیکی دست بکار شده و جهت تامین پهنای باند آن ازیک سیستم انتقال جدید با ظرفیت بالاتر استفاده نمایند، که خیلی زمانبر و هزینه‌براست. در این مقاله روشی را پیشنهاد می‌دهیم تا علاوه بر استفاده بهینه از پهنای باند موجود در صورت پر شدن ظرفیت شبکه بصورت اتوماتیک به شبکه با پهنای باند بالاتر منتقل گردد. برای این منظور ابتدا با طراحی یک کنترل کننده PID فازی برای شبکه موجود سعی می‌شود کنترل ازدحام را انجام داده و حداکثر استفاده را از آن نمود. مشاهده می‌گردد که کنترل‌کننده پیشنهادی از نظر پاسخ خروجی، دنبال نمودن طول صف مطلوب، پایداری و عدم قطعیت در مقایسه با کنترل‌کننده کلاسیک بسیار بهتر عمل می‌نماید. درصورت افزایش داده‌های ورودی به شبکه تعداد بسته‌های بیشتری حذف می‌شوند و این باعث پایین آمدن کیفیت شبکه می‌شود. برای رفع این مشکل با استفاده از مدیریت پهنای باند، با در نظرگرفتن حد آستانه برای حذف بسته ها در هر شبکه درصورت عبور از این حد شبکه موجود به یک شبکه با ظرفیت بالاتر سوئیچ گردیده و مشکل پهنای باند و کیفیت شبکه حل می‌گردد و باعث رضایتمندی مشترکین می‌شود.