

## Comparative analysis of the performance of various active queue management techniques to varying wireless network conditions

Kennedy Okokpujie<sup>1</sup>, Chukwu Emmanuel<sup>2</sup>, Olamilekan Shobayo<sup>3</sup>, Etinosa Noma-Osaghae<sup>4</sup>,  
Imhade Okokpujie<sup>5</sup>

<sup>1,2,3,4</sup>Department of Electrical and Information Engineering, Covenant University, Nigeria

<sup>5</sup>Department of Mechanical Engineering, Covenant University, Nigeria

---

### Article Info

#### Article history:

Received Sep 16, 2017

Revised Aug 29, 2018

Accepted Sep 13, 2018

---

#### Keywords:

Active queue management (AQM)

Vary link capacity

Varying network load

Varying propagation delay

---

### ABSTRACT

This paper demonstrates the robustness of active queue management techniques to varying load, link capacity and propagation delay in a wireless environment. The performances of four standard controllers used in Transmission Control Protocol/Active Queue Management (TCP/AQM) systems were compared. The active queue management controllers were the Fixed-Parameter Proportional Integral (PI), Random Early Detection (RED), Self-Tuning Regulator (STR) and the Model Predictive Control (MPC). The robustness of the congestion control algorithm of each technique was documented by simulating the varying conditions using MATLAB® and Simulink® software. From the results obtained, the MPC controller gives the best result in terms of response time and controllability in a wireless network with varying link capacity and propagation delay. Thus, the MPC controller is the best bet when adaptive algorithms are to be employed in a wireless network environment. The MPC controller can also be recommended for heterogeneous networks where the network load cannot be estimated.

Copyright © 2019 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Kennedy Okokpujie,  
Department of Electrical and Information Engineering,  
Covenant University,  
Ota, Ogun State, Nigeria.  
Email: kennedy.okokpujie@covenantuniversity.edu.ng

---

## 1. INTRODUCTION

The internet as a communication network has been found to exhibit different forms of complex dynamic behaviour that can be categorized based on the chaotic behaviour of user flows, flow rate oscillations caused by delay, dynamic flow synchronisation passing through a bottleneck link and the presence of queues in a router [1].

The sharing of a delay prone link between a tangible amount of host clients transport causes the protocol incumbent and the TCP variant that is designed to improve the link performance fall short of making best use of the communication channel. This shortcoming is neither caused by the control algorithm deployed nor the link performance. This limit however, can be attributed to the fundamentals of the TCP protocol [2].

The TCP remains the legacy protocol for the transport of packets in the internet backbone; the design of the protocol was made to help battle the issue of congestion in a connection-oriented, reliable end-end packet delivering IP network. However, this design was made for communication over wired architecture because that was the order of the day at that time [3].

Nowadays, there has been traffic in the internet due to the wider use of wireless networks (Wi-Fi and WLAN). When the legacy TCP is applied to this network, which might be heterogeneous in nature,

the congestion control mechanism available leads to reduction in the quality of communication and performance of the overall network [4].

The Internet Engineering Task Force (IETF) tried to find an alternative solution to the problem by proposing the deployment of Active Queue Management (AQM) strategies which combines perfectly with the TCP to ameliorate congestions problems. A lot of research is currently underway in the internet community regarding the development of a robust and scalable AQM system. The means by which the concept of AQM tackles the problem of congestion is to smartly report any congestion before the buffer overloads. This is done by sending a signal which can be interpreted in two ways, which are: a) reporting huge loss in packets b) increasing the delay, thereby causing packets to be repeatedly sent [5]. This property makes the AQM scheme a desirable choice for packet dropping/marking. The parameters used to mark or drop packets using active queue management techniques are the queue length and link occupancy. AQM schemes are also known for their robustness, and when in use, the network operator requires less tuning on the network. This makes the network operator immune to obnoxious network fluctuations [6]. The block diagram in Figure 1 shows the relationship between the TCP dynamics and the Queue dynamics.

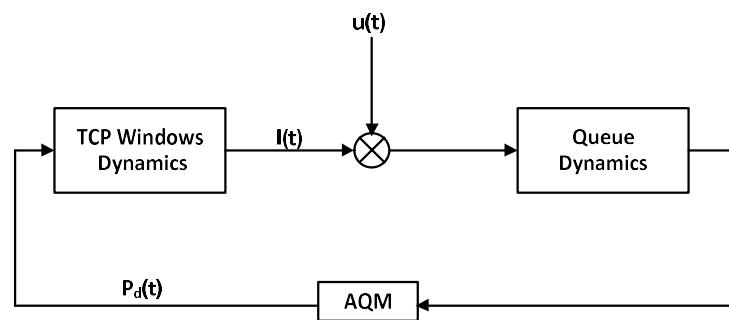


Figure 1. Block diagram of a TCP/AQM system [7]

### 1.1. The problem

A lot of algorithms have been proposed to provide optimum queue management in wireless networks. These techniques can be classified as classical and modern. The classical techniques include the Fixed-Parameter Proportional Integral (PI) and the Random Early Detection (RED). The modern techniques are the Self-Tuning Regulator (STR) and the Model Predictive Control (MPC) [8]. Most of the work done on adaptive network congestion control techniques only provided the algorithm, but they did not consider the effect of varying network conditions on congestion control parameters and the subsequent effect on their respective controllers. Most of the proposed algorithms were never tested for robustness. Hence, we cannot ascertain the response of their algorithms to changes in the network operating parameters.

### 1.2. The proposed solution

This work shows the robustness of various active queue management controllers to varying network parameters. The parameters varied are the network's Load, link capacity and propagation delay. The effects of these varying network parameters on the controllers of the active queue management techniques under study were monitored. The results of the simulation study show how the different AQM techniques respond to varying network conditions whilst solving the issue of network traffic congestion.

## 2. LITERATURE REVIEW

The Many authors over the years have worked on AQM (Adaptive or Predictive) techniques by improving some of the controllers. The authors in [5] proposed an improvement to the PI controller by making the response rate of their developed technique to be the function of the queue length error. This model saw an increase in the response rate along with increase in the queue length error. The authors in [9] proposed an AQM algorithm that works on the principle of combining both the average queue length and the estimates of the rate of packet arrival as the control parameter. It is an adaptive algorithm that calculates the mean length of the queue, maintaining it so as to keep it constant with a certain reference value. This gives rise to high link utilization with significantly lesser delays. The authors in [10] proposed a novel solution known as NARED. This is an improvement to the RED algorithm. It introduces the Bufsize parameter to implement the adaptive nature of the algorithm by dynamically adjusting the queue length of the incoming

packets with respect to the target queue length. This work however did not consider the effect of varying network parameters on the proposed solution. The authors in [11] developed a novel algorithm known as Model Predictive Active Queue Management (MPAQM). This adaptive technique is based on MPC. This algorithm also uses the queue length as the congestion control parameter. They could index the network performance to achieve an ideal control function. This work performed extensive simulations to show the robustness of their algorithm and could show that their results were better than RED and PI controllers. The work however, did not consider the effect of varying network link capacity and propagation delay. More recently, the authors in [12] developed a hybrid model which uses PPC technique to solve congestion problems in TCP/AQM compliant networks. It also combines backstepping and  $H_\infty$  control mechanisms [13]. Their simulation results, like [14], show the robustness of the hybrid algorithm and the adaptive nature was also confirmed by varying the network components but did not also consider the effect of varying link capacity and propagation delay [15-19].

### 3. RESEARCH METHOD

This section discusses the various parameters used for the study with the different experiments performed. Here, the calculations needed to generate our control parameters chosen for our proposed algorithm is determined.

#### a. Selection of Controller Parameters:

Below is the list of control parameters used for the simulation study. This simulation was performed using the MATLAB® and Simulink® package.

1. The router's buffer size;  $q_{max}=800$  packets.
2. Queue length (reference);  $q_{ref}=200$  packets.
3. Initial queue length;  $q_0=150$  packets.
4. The link capacity(bottleneck);  $c=3750$  packets
5. The link capacity in Mb/s; $c=15$ Mb/sec.
6. Packet size; 500 bytes.

#### b. Recursive Least Squares (RLS) Parameters

The parameters shown below are derived from empirical calculations and the values obtained are listed:

1. Initialized RLS parameter estimate;  $\theta_0 = 0$ .
2. Parameter estimation covariance matrix initialized as;  $P = 1000I$ . Where (I) is the identity matrix.
3. Forgetting factors are initialized as;  $\lambda_o = 0.92$ ;  $\lambda(0) = 0.9$ .
4. Formula used to update the forgetting factors;

$$\lambda(K) = \lambda_o \lambda(k-1) + (1 - \lambda_o) \quad (1)$$

#### c. MPC Parameters

The scalar metrics used in weighting the MPC are:

1.  $Q = 5$ ,  $S = 1$  (ensures that importance is placed more on minimization of error signal rather than penalizing the control effort made).
2. Prediction horizons:  $N_p = 5$   $N_u = 3$
3. Sampling time:  $T_3 = R_0$  (to synchronize the MPC to the TCP packet marking dynamics)

#### d. STR Parameters

1. Closed-loop discrete pole:  $t_1 = R_0$ (for closed-loop stability).
2. Time constant: about 9 samples.

#### e. PI Parameters

The PI controller used in the simulated study, was derived from wireless the established wired access network PI controllers. The equations employed for this scenario is stated:

$$C_{PI}(s) = K_{PI} \frac{s}{w_g + 1} \quad (2)$$

$$w_g = \frac{2N^+}{(R^+)^2 C} \quad \text{and} \quad K_{PI} = \left| \frac{jw_g}{P_{queue} + 1} \right| \frac{(R^+)^3}{(2N^-)^2} \quad (3)$$

The conditions necessary for the design of a PI controller which will stabilize the TCP/AQM system when employed on wireless network is listed.

For all  $N \geq N^-$  and  $R_0 \leq R^+$

$$w_g = \frac{2N^+}{(R^+)^2 C}; K_{PI} = \left| \frac{\frac{jw_g}{P_{queue}} + 1}{\frac{(R^+ C)^3}{(2N^-)^2}} \right|; C_{PI}(s) = K_{PI} \frac{s+1}{s} \quad (4)$$

Here, the worst-case scenario i.e. when  $N=30$  and  $R^+=0.24$  results in a value for  $w_g=0.278$  and  $K_{PI}=1.3748e-6$ . This will in turn provide the reference value for the phase margin and it is calculated from (5).

$$PM \approx 90^\circ - \frac{180^\circ}{\pi} w_g^2 \quad (5)$$

$$PM \approx 90^\circ - \frac{180^\circ}{\pi} w_g^2 = 85.6^\circ \quad (6)$$

Where;

c-Link capacity [Mb/s]; q-Buffer size [Packets]; I-Identity matrix; S-Weights (scalar); Q-Weights (scalar); R-Sampling time (Seconds); N-Horizon Nil; t-Pole; C-Transfer function; K-Scaling factor; w-frequency (Hertz);  $\lambda$ -Forgetting factor;

$\hat{\theta}_0$  – Least square parameter;

Subscripts

° -Initial; Ref-Reference; Max-Maximum; P-Prediction; U-Control; 1-First; PI-Parameter integral; g-Gain

Superscripts

(+)-Increment; (-)-Decrement

### 3.1. Experimental setup

#### 3.1.1. Simulations on wireless networks

The topology used for the simulation of the wireless network employed for this experiment is shown in the Figure 2. Here, a router is used to create a LAN with two laptop computers wirelessly. From the two nodes, it is assumed that several TCP connections are being established and are denoted by N. The Simulink model for the TCP/AQM dynamics is shown in Figure 3.

The wireless conditions that are varied for this experiment are:

1. Network load
2. Propagation delay
3. Link capacity

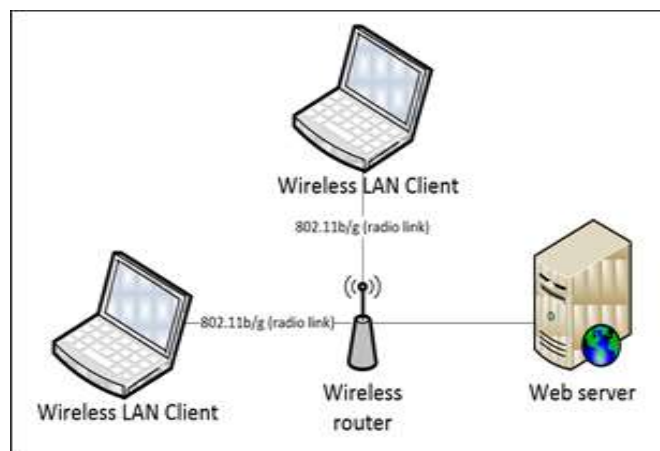


Figure 2. Wireless simulation topology

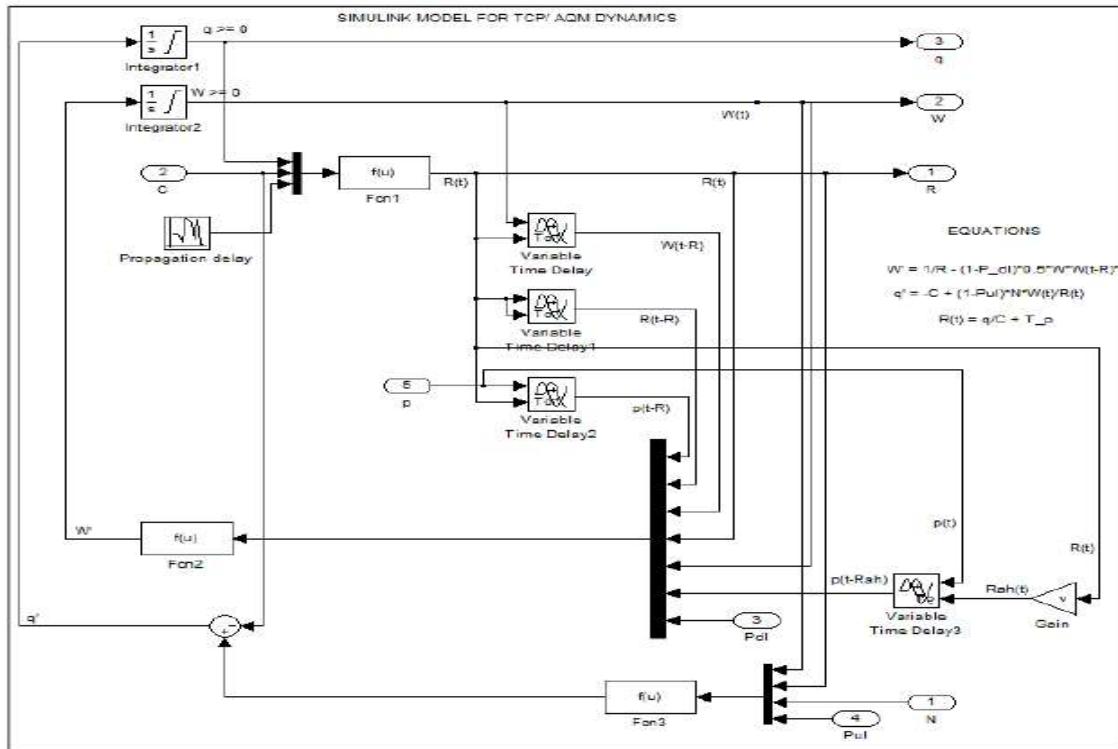


Figure 3. Wireless simulation topology

**3.2. Experiments, results and analysis**

The experiments were conducted by varying the under listed network parameters:

1. Network loads
2. Propagation delay
3. Link capacity because of unresponsive User Datagram Protocol (UDP) flows

To simulate network loads, File Transfer Protocol (FTP) packets with a longer Time to Live (TTL) values are sent through the network. A total of 3750 FTP packets are sent at every second. This is tantamount to a link with a bandwidth of 15Mbps and a packet size of 500 bytes.

**3.2.1. Experiments, results and analysis**

To commence the experiment, the load N is set at 500 FTP packets. At time t=30s, 100 more FTP packets were injected into the network. This was however stopped at time t=120s. The propagation delay was set at T<sub>p</sub>=120s. The simulation lasted for 220s. The link capacity of 3750 packets/seconds was maintained. All other parameters used are the same with the ones generated above. The result of the experiment is shown in Figure 4.

From the Matlab plot shown in Figure 4, it can be observed that as the load condition changes, MPC controllers show a positive response. There was an increase in the drop probability of packets so as to sustain the prescribed values of the queue length and window size. The STR controller responded with a change which can be said to be oscillatory in nature. The PI controller however shows the weakest response by maintaining a constant drop probability and the queue length experienced a rise to a value of 400. The response of the RED controller was also weak, but can still be categorized to be better than that of PI.

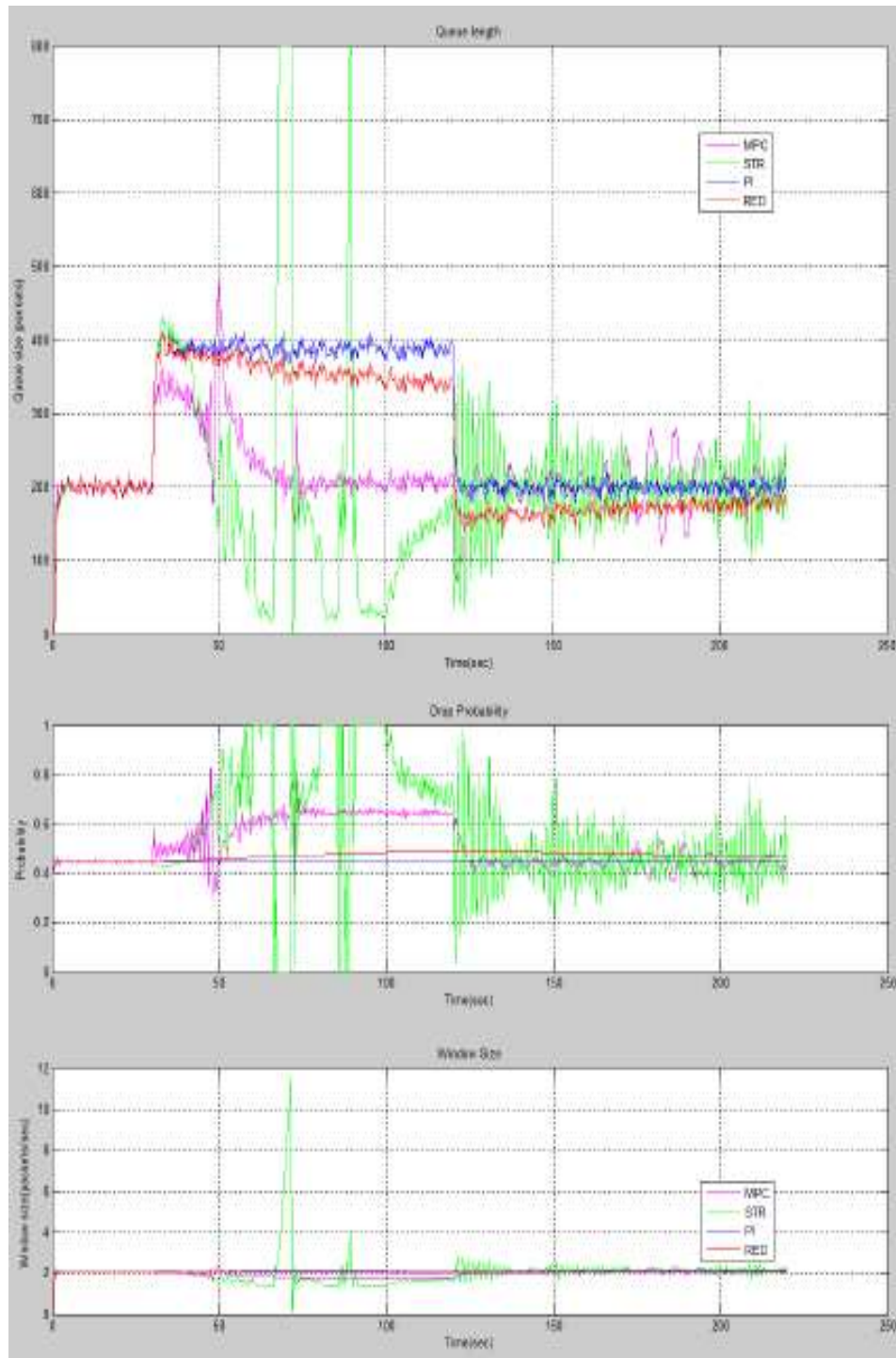


Figure 4. Matlab plots for varying load

### 3.2.2. Varying propagation delay

In order to introduce delay into the simulated network, the delay parameter was varied in a continuous fashion. The delays range from 0.04 and 0.2 seconds. The load in the network this time was set at 100 FTP/seconds. The link capacity remained the initial value of 3750 packets/s. All other parameters were held constant and the same with the values used in the previous experiments. The result for this experiment is shown in the Figure 5.

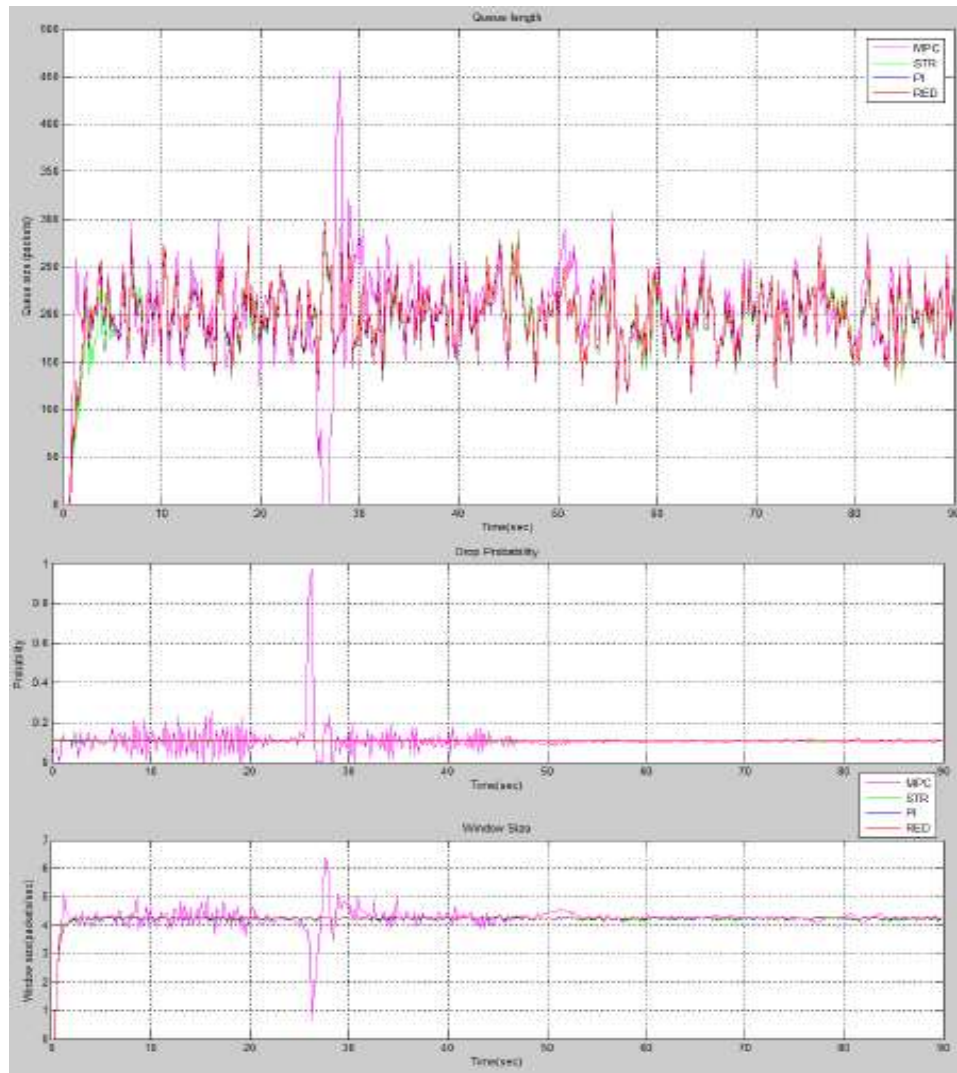


Figure 5. Varying propagation delay

From the Matlab plot above, only the MPC has shown a hike in its response and all other controllers have been stable. The queue length oscillation for all the controllers is a value of about 50 and it was also observed that the regulation for the window size was constant.

### 3.2.3. Varying link capacity

With reference to the initial link capacity which was set at 3750 packets/seconds i.e. 15Mbps, the link capacity was varied by using unresponsive UDP packets to reduce the link capacity to 2000 packets/seconds i.e. 8Mbps at time  $t=50s$ . The propagation delay time is set at  $T_p=0.2s$ . The simulation lasted for a period of 200 seconds and all other parameters remains the same as that of the previous experiments. The results of varying the network's link capacity are shown in the Figure 6.

From the Matlab plot shown in Figure 5, it can be observed that the PI controller response to the sudden change in the link capacity is weak. The drop probability however remains fairly constant which extends the queue length to 500. The RED controller shows a better response but its response slowly builds up. Out of all the controllers, the MPC shows the fastest convergence in response to the change in the link capacity by a quick reduction in the queue length. The MPC instantaneously dropped about three hundred 300 packets in response to the sudden change in the network link capacity. With the passage of more time, the queue length fluctuates between 200 and 500. The STR controller was the most stable of the controllers to the sudden change in link capacity with the passage of time. Although its initial response to the sudden change in link capacity was quite fast and very unstable.



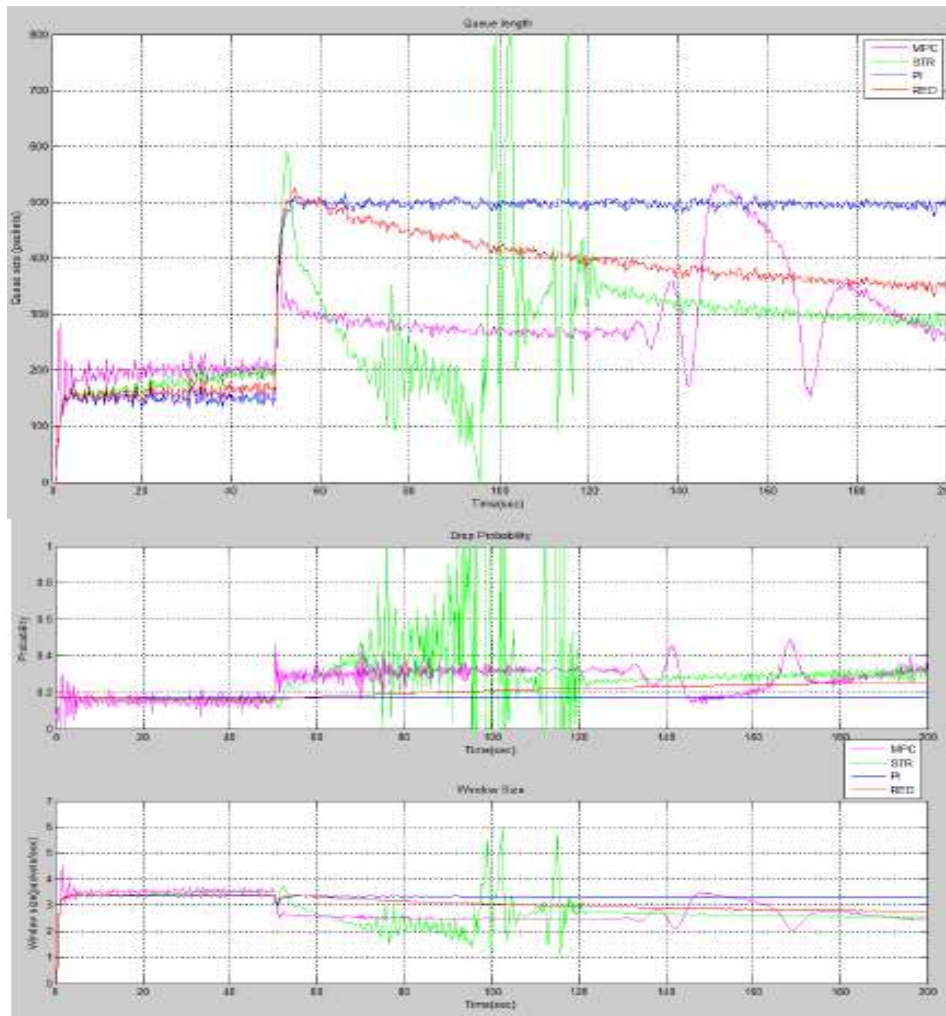


Figure 6. Varying Link Capacity

#### 4. RESULTS AND ANALYSIS

From the first experiment i.e. varying network loads, the PI and RED controllers have poor response to changes in the amount of FTP packets injected into the network (varying load). This can be attributed to the fact that they have fixed values of gains which have hindered their flexibility to generate the required control propensity when the load conditions are changed. MPC and STR show better response to the varying load due to the adaptive nature of their gain. This flexibility allows for a good control of the changing network conditions. MPC also provided the better response of the two because of its capability to predict on the horizon. Its computations of control actions are based on optimizing some steps into the future. The STR controller's response was oscillatory at the start of the experiment due to its non-predictive nature.

- a. In the second experiment i.e. varying propagation delay, the MPC shows a very weak adaptability to varying delays in the network. Several oscillations are seen at the early stages of the MPC controller, especially when the duration of the delays was made uncertain. PI and RED also show similar responses.
- b. From the third experiment i.e. varying link capacity, The MPC and STR had a pretty good adaptability to changes in the link bandwidth which is also due to the flexibility in their gain factor. The PI and RED were not very scalable when it comes to changing the capacity of the link.

##### 4.1. Comparison of the controllers

Starting with the lowest in terms of response of the controller to varying network conditions, the performance portrayed by each of the controllers is shown in Table 1. This table scales the controllers as good, fair or poor.



Table 1. Summary of the Performance of the Controllers

EXP	Network Condition	RED	PI	STR	MPC
1	Varying Network Load	Poor	Poor	Poor	Good
2	Varying Propagation Delay	Fair	Fair	Fair	Fair
3	Varying Link Capacity	Fair	Fair	Good	Good

## 5. CONCLUSION

This paper shows the effect of varying network conditions on four active queue management techniques namely; the Fixed-Parameter Proportional Integral (PI), Random Early Detection (RED), Self-Tuning Regulator (STR) and the Model Predictive Control (MPC). The simulation results from the Matlab plots show that the MPC technique for congestion control demonstrates superior performance under the varying network conditions of load, propagation delay and link capacity when compared to Fixed-Parameter PI, RED and STR. It is also seen that MPC has the capability to converge faster in all conditions. The STR gives the next best result. Therefore, MPC adaptive strategies will be the best algorithm for congestion control considering the nature of modern networks.

## ACKNOWLEDGEMENTS

This paper was sponsored by Covenant University, Ota, Ogun State, Nigeria

## REFERENCES

- [1] S. Pal and P. K. Pattnaik, "A Simulation-based Approach to Optimize the Execution Time and Minimization of Average Waiting Time Using Queuing Model in Cloud Computing Environment," *International Journal of Electrical and Computer Engineering*, vol. 6, p. 743, 2016.
- [2] S. Sharma, D. Jindal, and R. Agarwal, "Analysing Mobile Random Early Detection for Congestion Control in Mobile Ad-hoc Network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, 2018.
- [3] N. R. Reddy, C. R. Pakanati, and M. Padmavathamma, "An Enhanced Queue Management Scheme for Eradicating Congestion of TFRC over Wired Environment," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, pp. 1347-1354, 2017.
- [4] Y. Tian, K. Xu, and N. Ansari, "TCP in Wireless Environments: Problems and Solutions," *IEEE Commun. Mag.*, vol. 43, no. 3, 2005.
- [5] Q. Xu and J. Sun, "A Simple Active Queue Management based on the Prediction of the Packet Arrival Rate," *J. Netw. Comput. Appl.*, vol. 42, pp. 12–20, Jun. 2014.
- [6] J. Sun, S. Chan, and M. Zukerman, "I-API: An Intelligent Adaptive PI Active Queue Management Scheme," *Comput. Commun.*, vol. 35, no. 18, pp. 2281–2293, Nov. 2012.
- [7] N. Bigdeli and M. Haeri, "Predictive Functional Control for Active Queue Management in Congested TCP/IP Networks," *ISA Trans.*, vol. 48, no. 1, pp. 107–121, 2009.
- [8] A. K. Sharma, "A Survey on Active Queue Management Techniques," *Int. J. Eng. Comput. Sci.*, November 2016.
- [9] J. Hong, C. Joo, and S. Bahk, "Active Queue Management Algorithm Considering Queue and Load States," *Comput. Commun.*, vol. 30, no. 4, pp. 886–892, 2007.
- [10] J. Zhang, W. Xu, and L. Wang, "An Improved Adaptive Active Queue Management Algorithm based on Nonlinear Smoothing," *Procedia Eng.*, vol. 15, pp. 2369–2373, 2011.
- [11] P. Wang, H. Chen, X. Yang, and Y. Ma, "Design and Analysis of a Model Predictive Controller for Active Queue Management," *ISA Trans.*, vol. 51, no. 1, pp. 120–131, Jan. 2012.
- [12] Y. Liu, X. Liu, Y. Jing, and S. Zhou, "Adaptive Backstepping  $H_\infty$  Tracking Control with Prescribed Performance for Internet Congestion," *ISA Trans.*, vol. 72, pp. 92–99, 2018.
- [13] K. O. Okokpujie, E. C. Chukwu, E. Noma-Osaghae, and I. P. Okokpujie, "Novel Active Queue Management Scheme for Routers in Wireless Networks," *International Journal on Communications Antenna and Propagation (I. Re. CAP)*, vol. 8, pp. 53-61, 2018.
- [14] K. O. Okokpujie, O. Okoyeigbo, J. Okhaifoh, O. Osemwegie, and N. Nkordeh, "Performance Analysis and Modeling of MIMO Systems," *International Journal of Applied Engineering Research*, vol. 11, pp. 11537-11541, 2016.
- [15] O. Obinna, O. Kennedy, O. Osemwegie, and N. Nsikan, "Comparative Analysis of Channel Estimation Techniques in SISO, MISO and MIMO Systems," *International Journal of Electronics and Telecommunications*, vol. 63, pp. 299-304, 2017.
- [16] Kennedy, Okokpujie. "Realization of MPLS-Based VPN Network for Improved Qos Metrics." *Telecommunication Computing Electronics and Control (TELKOMNIKA)*, vol. 16, no. 5, 2018.
- [17] Okokpujie, Kennedy O., Emmanuel C. Chukwu, Etinosa Noma-Osaghae, and Imhade P. Okokpujie. "Novel Active Queue Management Scheme for Routers in Wireless Networks." *International Journal on Communications Antenna and Propagation (I. Re. CAP)*, vol.8, no. 1, pp. 53-61, 2018.

- 
- [18] Obinna, Okoyeigbo, Okokpujie Kennedy, Omoruyi Osemwegie, and Nkordeh Nsikan. "Comparative Analysis of Channel Estimation Techniques in SISO, MISO and MIMO Systems." *International Journal of Electronics and Telecommunications*, 63, no. 3, 299-304, 2017.
- [19] Obiazi, A. M. O., F. I. Anyasi, Kennedy O. Okokpujie, J. B. Erua, and O. A. Osahenvenwen. "An Innovative Technique in Switching Electrical Appliances: An Omni-Directional RF Remote Control Switch Mechanism." *Journal of Engineering and Applied Sciences*, vol. 4, no. 4, 268-271, 2009.