ISSN 2047-3338

Performance Evaluation and Comparison of EQM with RED, SFQ, and REM, TCP Congestion Control Algorithms Using NS-2

Nauman, A., and Arshad, M. J.,

Department of Computer Science and Engineering, University of Engineering and Technology Lahore, Pakistan

Abstract– **Network congestion deteriorates the quality of service in the Transmission Control Protocol (TCP) network model. A number of congestion control algorithms have been proposed to achieve the high link utilization with minimum queue latency. Queue management is a very active research area in real time networking traffic so as to meet the demands of real time internet based applications. A new AQM algorithm called Effective Queue Management (EQM) strategy has recently been proposed to manage the queue proactively. This queue management scheme proposes solution to overcome all the drawbacks experienced in all the earlier existing active queue management (AQM) schemes but effectiveness of this proposed scheme has not yet been analyzed in terms of average queue length, throughput, packet loss, utilization and delay with the variations in network load. The overall goal of this paper is to focus on analyzing the effectiveness of EQM and comparing its performance with the earlier existing AQM schemes Random Early Detection (RED), Stochastic Fair Queuing (SFQ), and Random Exponential Marking (REM), in terms of average queue length, throughput, utilization, packet loss and delay. Performance parameters (queue length, throughput, utilization, loss, delay) for the EQM, RED, SFQ and REM algorithms have measured with different bandwidths. The effectiveness of all these mechanisms is analyzed by using simulations in Network Simulator (NS-2). Each of the algorithms EQM, RED, SFQ and REM is categorized on the basis of simulation results using performance parameters.**

Index Terms– **AQM, Active Queue Management, Effective Queue Management, EQM, Random Early Detection, RED, Stochastic Fair Queuing, SFQ, Random Exponential Marking and REM**

I. INTRODUCTION

HEAVILY moving network traffic comprising of network packets competes for limited shared network resources, such as queue buffer size in the router lying in the way of network traffic and the outgoing bandwidth for packets movement. Congestion may happen due to limited shared resources in the data communication. Shared resources may be limited buffers, limited bandwidth of bottleneck links [7] etc. Due to congestion problem, large number of packets drop because of queue overflow or experience delay resulting poor quality of service. Frequently happening congestion may result in large packet loss rate hence causing degradation of throughput. Congestion also degrades efficiency and reliability of the overall network; moreover, performance downfalls completely in case of very high traffic failing delivery of packets. Being a very active research area, Researchers took interest and proposed many congestion control techniques [6] to handle this problem effectively and avoid the delay and loss of packets.

Congestion can be measured by two methods [1]: (i) Flow based, and (ii) Queue based. In the flow low based method, AQMs determine network congestion and take action on the basis of the incoming packet rate. For such techniques, backlog, and all its adversarial implications, is not necessary for the control process. In queue based, method AQMs congestion is observed by measuring the queue size. The shortcoming of this technique is that a backup of data segments is integrally needed by the congestion methodology, as a bottleneck is experienced at times when the backlog is already up. This method produces needless packet deferral. The overall Network delay is basically the sum of packets' queuing delay and its propagation delay. Presently queuing delay of a network packet dominates most of the round trip times (RTTs).

A new AQM scheme called effective queue management [1] is proposed. The aim of this scheme is to design an effective queue management for all types of networks that can monitor and avoid congestion traffic more effectively than that of earlier existing AQM mechanisms by overcoming their shortcomings and thereby improving the quality of service of network traffic.

The main objective of this paper is to focus on analyzing the effectiveness of EQM and comparing its performance with the earlier existing AQM schemes Random Early Detection (RED) [9], Stochastic Fair Queuing (SFQ) and Random Exponential Marking (REM) in terms of average queue length, throughput, utilization, packet loss and delay.

The rest of the paper is structured as follows: Section II describes a critical review of relevant work done in this research area; Section III elaborates on configured network and the analysis of EQM, RED, SFQ, REM schemes; In Section IV we present our conclusions and discuss future work .

II. CRITICAL REVIEWS OF RELEVANT WORKS

In this recently published paper [1], a new algorithm called Effective Queue Management has been proposed that solves the backgrounds in all the existing queue management techniques. Its efficiency over the existing AQM algorithms is verified using the comparison graphs. The proposed scheme is very simple, robust, very low in computational complexity, easily configurable, and autonomous to a single router hence very easy to deploy.

In paper [10], the Random Early Detection (RED) gateways technique is proposed to avoid congestion in packet-switched networks. RED is a first generation Active Queue Management (AQM) technique. RED suffers from some severe limitations. The nodes' queue size is not a good method and indicator of the severity of the network congestion, and congestion warning levels issued might be too burst and great, leading to excessive and unnecessary packet loss rate. RED is liable to times of high loss rate followed by link underutilization.

Reference [4] addresses the issues with existing network congestion control methods and demonstrated various performance parameters of RED, Stochastic Fair Queuing (SFQ), and Random Exponential Marking (REM) for considered network configurations. Analysis of this paper shows that RED suffers from an extreme packet loss ratio and SFQ suffers from a smallest average packet loss ratio. It also concludes that performance parameters such as throughput, loss ratio, and utilization vary according to different AQM algorithms. RED attained a fruitful outcome in relationship with queue delay whereas REM gives the best results in terms of utilization, loss ratio and throughput. If equal weights are provided to each performance parameter, then REM outputs the better results among the three considered algorithms.

Through simulation and queue law analysis, [3] verifies that RED techniques that work on the principle of dropping a packet cannot increase the performance of network traffic compared to drop-tail based AQM techniques.

In paper [8], a comparison of various aspects of two extensively used queue management techniques, RED and Drop-tailing have been presented. Experiments are designed to simulate the queue management methods, and analyzed the fairness and throughput of network flow. Comparison presented that RED performed to some extent better with higher throughput and higher fairness Index than Drop-tail [5].

III. ANALYSIS OF EFFECTIVE QUEUE MANAGEMENT

This section covers the effectiveness of EQM and its performance comparison with the earlier existing AQM schemes Random Early Detection (RED), Stochastic Fair Queuing (SFQ) and Random Exponential Marking (REM) in terms of average queue length, throughput, utilization, packet loss and delay. Performance parameters (queue length, throughput, utilization, loss, delay) for EQM, RED, SFQ and REM algorithms have been measured with different bandwidths. The effectiveness of all these mechanisms is analyzed by using simulations in Network Simulator (NS-2). Each of the algorithms EQM, RED, SFQ, and REM, is categorized on the basis of simulations results using performance parameters.

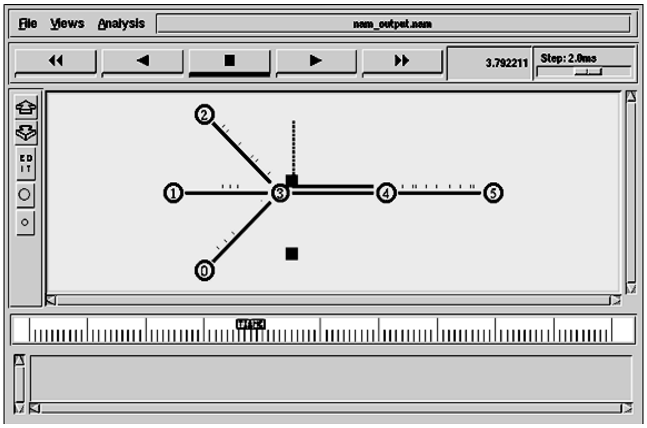

Fig. 1. Network Simulation Topology

A. Performance Parameters

Following is a brief introduction to the performance parameters.

Average Queue Length

The average queue length is the number of packets arriving and waiting in queue if they are not immediately processed at any node in the network that need some processing by that node. This parameter is very important to measure the activeness of any congestion control mechanism. The smaller the queue length better will be the performance of the congestion control algorithm.

End to End Delay

Delay refers to average time taken by the packets moving from one node to the next node. The higher the delay time the lower will be the bandwidth of the network. This chapter presents the end to end delay. There are two types of end to end delay. The first type of delay is called fixed delay which combines packet transmission delay and propagation delay. The second type of delay is called variable delay which contains processing and queuing delay. The end to end delay, T_d is calculated as:

$$Td = \sum (Packet\ arrival\ time - Packet\ send\ time) / \sum Number\ of\ connections \qquad (1)$$

Packet Loss

Packets may be dropped during their movement over the network from source to destination due to either queue overflow or noise dependent on the medium on which a packet moves from source to destination. The higher the packet loss rate the lower will be the network bandwidth. This is another important characteristic to measure the effectiveness of any network. Packet loss can be calculated as

$$Packets\ lost = Total\ Packets\ Sent - Total\ Packets\ Received \quad (2)$$

The average loss ratio is the ratio of the total number of lost packets to the total number of sent packets.

Packet Delivery Ratio

This is the ratio of the total number of packets delivered to destination successfully to the total number of packets sent. The delivery ratio D_r can be calculated as

$$Dr = \sum Packets\ Received\ / \sum Packets\ Sent \quad (3)$$

A greater value of delivery ratio results in better performance of network.

Throughput

The rate of successfully delivered packets from source to destination per time interval is called throughput in networking. Usually it is measured in bps. Sometimes it is expressed as Kbps or Mbps. It can be calculated using the following formula

$$Throughput\ = \sum Packets\ Received * Packet\ Size * 8\ / \sum Time\ Interval \quad (4)$$

Utilization

Utilization can be defined as percentage value of network's bandwidth that is being used by network traffic at a specific interval of time. Utilization is a network parameter which can help us to identify the point of network slowdown or point of failure. Higher the value of bandwidth usage results in slow network traffic. Formula to calculate the utilization is as given below:

$$Utilization(\%)\ = Total\ Bits\ Measured\ / (Bandwidth * Bits\ Calculation\ Interval) \quad (5)$$

B. Experiments and Analysis

This section deals with the detailed analysis of EQM with RED, REM and SFQ with respect to network load. Variation in network load can be made either by changing the number of nodes in the underlying network or by changing the network traffic rate which is commonly called bit rate generated by the nodes in the network. This analysis deals with variations in bit rate while keeping a constant number of nodes which are 6 in number in this simulated network as illustrated in Fig. 1.

Simulations, Configuration and Results

FIG. 1 demonstrates the network topology used to measure the performance of these four active queue management techniques.

Here Node 0, Node 1 and Node 3 are the network UDP traffic generating sources. Node 5 is the destination node and acts as sink node. All the links are half duplex links which transport data only from source to destination in one direction. The link between Node 3 and Node 4 is the bottleneck link. Bandwidth and delay for this bottleneck node is 8Mb and 10ms respectively throughout the analysis. All the links from nodes ranging from 0 to 2 to nodes 3 have a bandwidth of 3Mb each and an end-to-end delay of 10ms. The link between node 4 and node 5 has bandwidth of 9Mb and 10ms delay. The packet size is fixed to 500 bytes and maximum queue size is also fixed to 50 packets for all links throughout the simulation.

Trace File Structure

When simulations are run, the network simulator (ns) keeps track of each occurred event of simulation in a file called trace file [2]. An event may either be a packet received, packet dropped packet enqueued or packet dequeued. This trace file is organized into the twelve fields as shown in Table I with some sample data.

Table II shows all the possible events likely to be happened with network traffic and each and every event is stored in the trace file for analysis of traffic.

C. Experiments and Simulations

The following sections present the experimental results based on network load for each of EQM, RED, SFQ and REM in graphical form. In the end all comparative results are compiled and presented in tabular form.

In all of the following simulation experiments, the first three nodes numbered from 0 to 2 transmit the data to the destination node numbered 4. Initially the bit rate of each of the three transmitting nodes is set to 0.05Mbps then it is gradually increased for each of the three nodes from

TABLE I

TRACE FILE STRUCTURE

Event	Time	From Node	To Node	Pkt Type	Pkt Size	Flags	Fid	Src Addr	Dst Addr	Seq No.	Pkt ID
r	8.476	4	5	exp	500	------	2	1.0	5.1	1285	4531
+	8.477	2	3	exp	500	------	3	2.0	5.2	2083	4633
-	8.477	2	3	exp	500	------	3	2.0	5.2	2083	4633
d	8.479	3	4	exp	500	------	3	2.0	5.2	2069	4600

0.05Mbps to 3Mbps in increments of 0.05Mbps. Hence the overall network load increases from 0.15Mbps to 9Mbps with aggregated increment steps of 0.15Mbps.

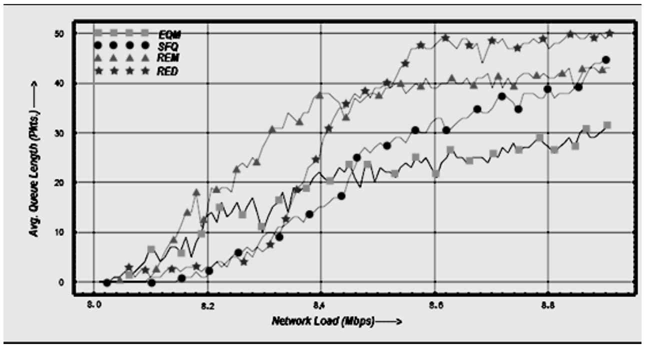

Fig. 2. Average Queue Length vs. Network Load

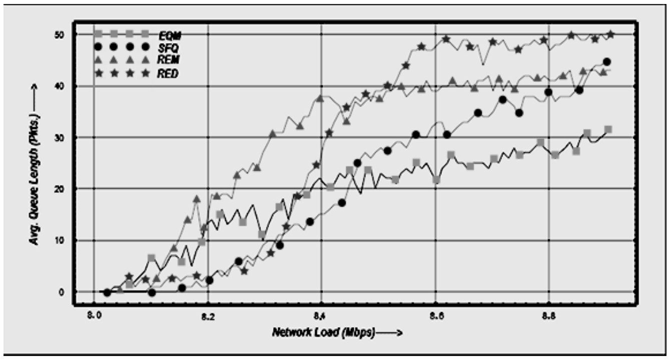

Fig. 3. Average Delay vs. Network Load

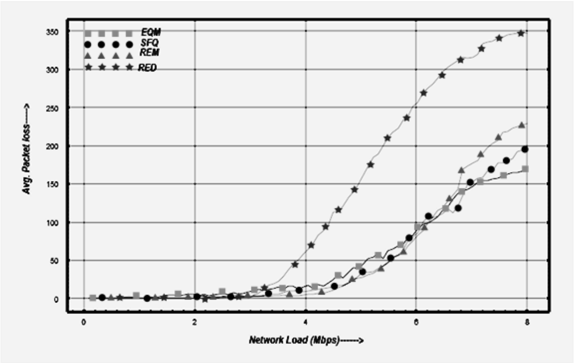

Fig. 4. Average Packet Loss vs. Network Load

TABLE II

TRACE FILE EVENTS

Event	Explanation
r	Packet received at “to node”
+	Packet enqueued at queue
-	Packet dequeued at queue
d	Packet dropped at queue

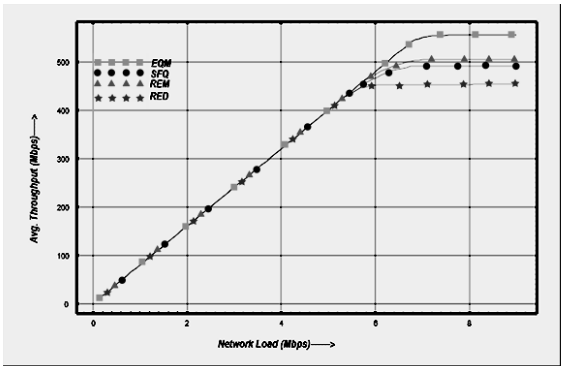

Fig. 5. Average Throughput vs. Network Load

Queue Length Vs. Network Load

Fig. 2 presents a graphical representation of queue length variations with respect to network load for EQM, REM, RED and SFQ.

Fig. 2 shows that the average queue length of EQM attains smaller value than that of all the other three AQM techniques. Hence EQM is a suitable technique to maintain the minimum value of the queue. The maximum average queue length is achieved by RED for longer time.

Delay vs. Network Load.

Fig. 3 presents a comparative analysis of the average end-to-end delay with respect to gradually increased network load for the bottleneck, link n3-n4, which has a maximum delay of 10ms. Minimum values of delay achieved by each AQM are the same.

Fig. 3 shows that the average end to end delay for EQM is less than that from other AQM techniques under consideration. Initially its value is greater but this does not make any difference on the overall throughput of the network as this increase is for a short time interval. The End to end delay becomes prominent when network load crosses the bandwidth of the bottleneck. So Fig.3 is just a snapshot for the network load from 8Mb network load to 9Mb network load.

Packet loss vs. Network Load

Fig. 4 shows the comparison results of average packet losses of all of the AQM techniques under consideration. The diagram shows the results of a bandwidth up to 8Mbps. It shows that the maximum drop rate is achieved by RED and minimum average loss rate is achieved by EQM.

Throughput vs. Network Load

Fig. 5 shows comparative simulation results for all the throughputs of AQM techniques under study. The graph shows that maximum throughput is achieved by EQM. Hence EQM is also efficient in throughput compared with all these techniques under consideration.

TABLE III

PERFORMANCE MEASUREMENTS

Parameter	EQM	RED	REM	SFQ
Avg. Queue Length(Pkts.)	18	29	20	28
Avg. End to End Delay(ms)	3.07	4.21	5.47	4.02
Avg. Packet Loss(Pkts.)	49	51	47	124
Avg. Throughput(Mbps)	343.58	329.31	324.67	312.38
Avg. Utilization(Mbps)	43.03	43.13	43.43	37.28

Utilization vs. Network Load

Fig. 6 demonstrates the average utilization of all the AQM techniques under consideration. The utilization of EQM is almost equivalent to the other techniques at the initial stage. But with the increase of network load at later stages, its utilization increases which is evident of low data loss while moving from source node to destination node. Experimental values are taken for a maximum of 8Mb of network load as bottleneck link can afford a maximum of 8Mb data bandwidth. Fig. 6 also shows that SFQ gives comparatively low utilization throughout the experiment. Its peak values of network load EQM are giving good results.

At the earlier stages, AQM techniques other than SFQ show the same behavior. With higher demand of network load it would be beneficial to change the network infrastructure in accordance with this new EQM technique.

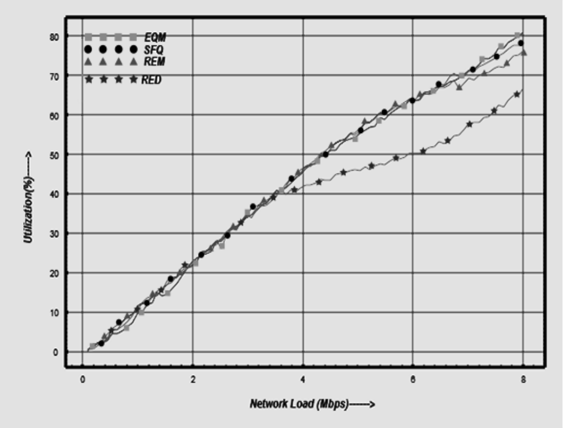

Fig. 6. Utilization vs. Network Load

D. Comparative Results

This section presents the performance of EQM, RED, REM and SFQ based on the different network performance parameters which are part of this network simulation process. The statistics shown in Table III are the average values of the above experiments performed from the configured network.

E. Rankings of AQM Algorithms

All the four discussed algorithms can be ranked on the basis of the performance parameter statistics shown in Table IV This ranking can help in selecting the best suited algorithms for the network to be designed according to the environment demands. Table 4 presents the rankings of EQM, RED, SFQ, REM with respect to the above parameters measured, where A denotes the best and D the worst.

It is observed that all of these four algorithms have variations in their values for the same parameters with the same amount of network data. Table 4 shows that EQM achieves a maximum of the best suited values for 3 out of 5 performance parameters. Lack in packet loss and utilization is also not a permanent behavior of EQM it can give us good stats for higher values of network load. Assuming equal weights of all the performance parameters it can be concluded that EQM is better among all of these four AQM techniques.

IV. CONCLUSIONS AND FUTURE WORK

This paper presents the comparative analysis of the EQM, RED, SFQ and REM algorithms on the basis of various performance parameters (queue length, packet loss, throughput, utilization and delay) measured with varying bandwidths for a configured network. It is observed that none of these algorithms gives top ranking results for all performance parameters collectively but classification of these algorithms shows that EQM is the better technique from the other three earlier existing mechanisms. EQM achieves three top rankings out of five performance parameters. It can also be observed that performance parameters for which EQM has lower rankings for configured range of network load have tendency to give better results at higher network loads. Experiments performed and results show that overall performance of EQM is better technique compared with the other three considered algorithms.

In the future, analysis can be further extended for different numbers of nodes and with varying packet sizes. This analysis can bring up more pros & cons of these algorithms which in turn will open the doors for making improvements in these considered algorithms.

ACKNOWLEDGEMENT

The authors are thankful to the University of Engineering and Technology Lahore for its kind support to carry out this research work. The authors also acknowledge the expert opinions of their colleagues to complete this research work.

REFERENCES

[1] J. Venkatesan, S. Thirumal, Novel Approach for Queue Management and Improvisation of QOS for Communication Networks, International Journal of Scientific Engineering and Research (IJSER), 2 (2014), 51-56.

[2] D. Kalav, S. Gupta, Congestion Control in Communication Network Using RED, SFQ and REM Algorithm, International Refereed Journal of Engineering and Science (IRJES) , 1 (2012), 41-45.

[3] J. Chung, M. Claypool, Analysis of Active Queue Management, Network Computing and Applications, 2003. NCA 2003. Second IEEE International Symposium on, (2003), USA.

[4] S. Patel, P. K. Gupta, A. Garg, P. Mehrotra, Manish, Comparative Analysis of Congestion Control Algorithms Using ns-2, IJCSI, International Journal of Computer Science Issues, 8 (2011), 89-94.

[5] G.F.A. Ahammed, R. Banu, Analyzing the Performance of active queue Management Algorithms, International journal of Computer Networks & Communications (IJCNC), 2 (2010), 1-19.

[6] H. Chandra, A. Agarwal, T. Velmurugan, Analysis of Active Queue Management Algorithms and there Implementation for TCP/IP Networks Using OPNET Simulation Tool, International Journal of Computer Applications 6 (2010), 12-15.

[7] S. Miskovic, G. Petrovic, L. Trajkovic, Implementation and Performance Analysis of Active Queue Management Mechanisms, Telecommunications in Modern Satellite, Cable and Broadcasting Services, 2005. IEEE 7th International Conference on, 2 (2005), 317-326, USA.

[8] T.B. Reddy, A.F.F. Ahammed, R. Banu, Performance Comparison of Active Queue Management Techniques, IJCSNS International Journal of Computer Science and Network Security, 9 (2009), 405-408.

[9] T. Bonald, Analytic Evaluation of RED Performance, INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, (2000), USA.

[10] S. Floyd, J. Jacobson, Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking, 1 (1993), 397-413, USA.