Volume 1 No.6, December 2011 ISSN 2224-3577
International Journal of Science and Technology

http://www.ejournalofsciences.org

Queue Management In Network Performance Analysis

[1]Olawoyin, L.A, [2]Faruk, N , [3]Akanbi, L.A
[1,2]Department of Telecommunication Science, University of Ilorin
[3]Department of Computer Science and Engineering, Obafemi Awolowo University Ile- Ife

ABSTRACT

One of the problems encountered in network design is about data carrying capacity of the network systems. This is mainly due to the state-of-the-art network links operating at very high speeds and can support advanced and high quality of service (QoS). In this paper we analyze the performance bottleneck of different queuing policies. We also study active queue management and scheduling as an approach to overcome this problem.

Keywords: *Quality of Service, Drop tail, Congestion Control, Random Early Detection, Queue Management, Stochastic Fair Queuing.*

I. INTRODUCTION

To meet the demand for higher performance, flexibility, low queue rate, low delay time and high economy in emerging multi-service broadband networking systems the output throughput of a data network must be of higher value [1]. In the design of active queue management and scheduling for broadband networks there is needs to take into account the unique features of broadband networks (such as end-to-end delay, packet losses, congestion and the distributed nature of broadband networks). A systematic approach to this problem is still elusive as currently proposed approaches are either too complex, or do not provide clear design guidelines for obtaining good system performance [2]. To prevent the network congestion and collapse, the current Internet uses end-to-end congestion control techniques so as to place a complex functions outside not inside the network [3]. At the end-host, responsive traffics tow like TCP treat network packet losses as implicit network congestion signals from routers and reduce their transmission rate. In the network, routers use outbound queues to accommodate traffic bursts and achieve high link utilization. Due to the simplicity of the FIFO (First in First Out) queuing mechanism, drop-tail queues that drop incoming packets when the queue is full are the most widely used on the Internet today. Unfortunately, when faced with persistent congestion, drop-tail queues all up resulting in higher delays. In addition, drop-tail queues can also result in bursty packet drops/packet loss that would degrade the system stability) and bandwidth fairness, but have a tendency to penalize bursty flows, and to cause global synchronization between flows, by dropping packets probabilistically. In this work Active Queue Management (AQM) is proposed [Ref] to replace drop-tail queue management in order to improve network performance in terms of delay, link utilization, and packet loss rate and system fairness. AQM enhances routers to detect and notify end-systems of impending congestion earlier, allowing responsive traffic sources to reduce their transmission rates before congested router queues overdo. AQM dramatically reduce network packet loss rate and improve throughput. Understanding AQM congestion control requires knowledge of the response of the characteristics of the traffic to control. The queue law is particularly useful for configuring the Random Early Detection (RED) family router queue management mechanisms that use the average queue length to compute drop probability. Yet, the queue law does not clearly illustrate how different TCP traffic parameters affect the traffic control aspect of AQM. AQM disciplines typically avoid both of these issues by providing endpoints with congestion indication before the queue is full; AQM disciplines are able to maintain a shorter queue length than drop-tail queues, which reduces network latency. But AQM disciplines require careful tuning of their parameters in order to provide good performance. Modern AQM disciplines are self-tuning, and can be run with their default parameters in most or all circumstances.

Stochastic Fair Queuing method of network management is a simple policy based on randomization of flow of traffics in the queen and hence it provides better fairness of the link compared with the Drop Tail policy

II. QUEUE MANAGEMENT CONTROL

Congestion collapse generally occurs at choke points in the network, where the total incoming packet to a node exceeds the outgoing packet. To avoid congestion collapse at choke point, congestion control concerns controlling traffic entry into a telecommunication network, so as to avoid congestive collapse by attempting to avoid oversubscription of any of the processing or link capabilities of the intermediate nodes and networks and taking resource reducing steps, such as reducing the rate of sending packets. It should not be confused with flow control, which prevents the sender from overwhelming the receiver.

Active queue disciplines drop or mark packets before the queue is full. Typically, they operate by maintaining one or more drop/mark probabilities, and probabilistically dropping or marking packets even when the queue is short.

Volume 1 No.6, December 2011 ISSN 2224-3577
International Journal of Science and Technology

http://www.ejournalofsciences.org

III. SIMULATION ENVIRONMENT

The simulation experiment is carried out in LINUX (FEDORA 6). The detailed simulation model is based on network simulator-2 (ver-2.31). The network simulator instructions can be used to define the topology structure of the network and the motion mode of the nodes, to configure the service source and the receiver, to create the statistical data track file and so on.

1. Scenario 1

Consist of four nodes (n0, n1, n2, n3), the duplex link between n0 and n2, and n1 and n2 have 1Mbps of bandwidth and 10ms of delay. The duplex link between n2 and n3 has 1Mbps of bandwidth and 10ms of delay, each node n0 and n2, n1 and n2 use a drop tail queue while n2 and n3 uses both queue and stochastic fair queuing policy techniques in queue management.

IV. SIMULATION RESULTS

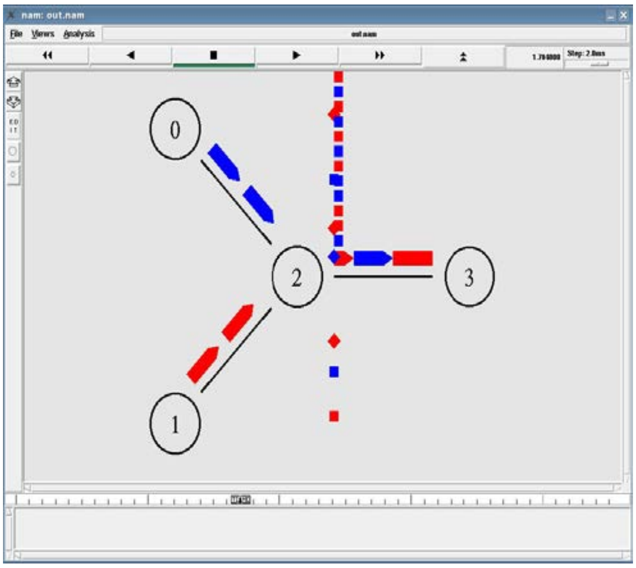

Fig 1 Nam File of Drop tail queuing method in network analysis

Table 1: Showing the link capacity

Node	Link Capacity (MB)	Delay (ms)	Packet Size (Byte/bit)	Loss
0-2	2	10	1000	
1-2	2	10	1000	
2-3	1.7	20	1000	Yes
0-2	2	10	1000	
1-2	2	10	1000	
2-3	2	20	1000	Yes
0-2	2	10	1000	
1-2	2	10	1000	
2-3	3	30	1000	No

V. EXPLANATION OF THE RESULT

From the table above showing link capacity, at the start of the simulation the duplex link between node 0 and 2 with one Constant Bit Rate (CBR) with packet size of 1000MB and the delay of 10seconds, the link capacity is 2MB, the link able to carry the packet successfully without any drop in the packet. This shows that the bandwidth use is enough to carry the packet. The transmission of second CBR packet, the link capacity between nodes 2 and 3 can equally carry the packet but as the packet size increases, the capacity of the link can no longer be able to carry the packet, this leads to dropping in the packet or causes a queue in the link which causes delay in the packet arrival rate. The queue can be avoided either by reducing the delay time, increasing the link capacity or reducing the packet sizes.

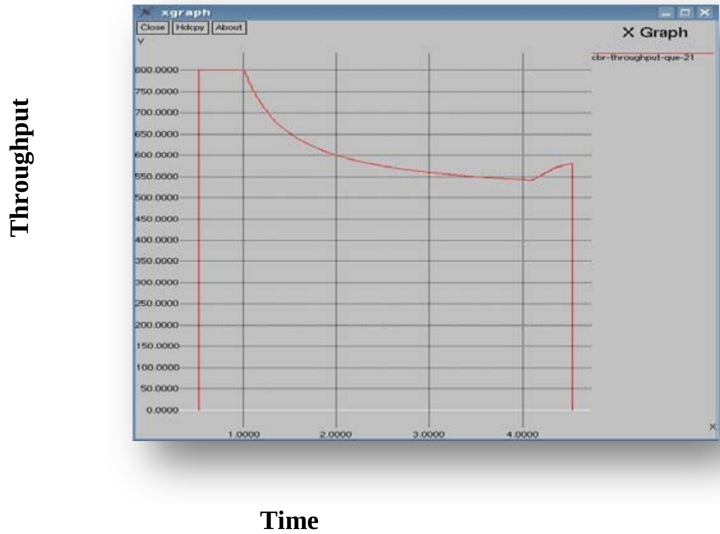

Fig 2 Throughput graph of the simulation

The throughput graph shown above shows that when only one packet is flowing in the link, the throughput is high but at 1.0sec when the second packet commence flowing, the queue increases on the link and that reduces the system throughput since there are many packets that need to be carried by the link and this leads to more packet loss, this reduce the link efficiency. The throughput graph above shows the performance of the network topology which depicts the capacity of the link between the nodes in the network topology. When a single packet is flowed in the link, the throughput, as expected, is very high but as the delay time set expires i.e. after 1.0second, the link buffer could not contain all the packets which led to increase in the queue on the link and reduction in system throughput and leads to packets dropped as a result in reduction in the link buffer and leads to reduction in link efficiency.

The throughput graph shown above shows that when only one packet is flowing in the link, the throughput is high but at 1.0sec when the second packet starts flowing, the queue increases on the link and that

Volume 1 No.6, December 2011 ISSN 2224-3577
International Journal of Science and Technology

http://www.ejournalofsciences.org

reduces the system throughput since there are many packets that need to be carried by the link and this leads to more packet loss, this reduce the link efficiency. At 4.089526sec, the second packet stop flowing in the link at it is timed to stop, it can be seen from the throughput graph that the throughput of the system starts increasing until the whole packet stop flowing.

VI. OBSERVATION

It is generally observed that when the first CBR packet flows it starts at 0.5sec, there are no drops in the link (colour blue) until the 2nd CBR from node1 (colour red) flows at 1.0sec in the link, at the router, node2 buffer size is able to store the number of packets from both nodes 0 and 1 until the buffer size is filled and packet start dropping in the link at 1.142752sec and stops at 4.089526sec. The link capacity cannot carry both UDP (User Diaphragm Protocol) sources when the two packet flowing at the same time. The link of interest is from node0 to node2 and using flow id of 0. The graph of SFQ (stochastic fair queuing) throughput show better performance than drop tail. The number of packets sent is 1402 while 219 were lost using SFQ while 403 were lost using drop tail.

In general it was observed that the two queuing policy techniques, stochastic fair queuing techniques has a better performance with less packet loss in the link as compared to drop tail techniques

VII. RANDOM EARLY DETECTION

Random early detection (RED), also known as **random early discard** or **random early drop** is an active queue management technique.

In the conventional drop tail techniques, a network router or other network components buffers as many packets as it can, and simply drops the ones it cannot buffer. If buffers are constantly full, the network is congested. Tail drop distributes buffer space unfairly among traffic flows.

RED monitors the average queue size and drops packets based on statistical probabilities. If the buffer is almost empty, all incoming packets are accepted. As the queue grows, the probability for dropping an incoming packet grows too. When the buffer is full, the probability has reached 1 and all incoming packets are dropped.

RED is more fair than tail drop, in the sense that it does not possess a bias against bursty traffic that uses only a small portion of the bandwidth. The more a host transmits, the more likely it is that its packets are dropped as the probability of a host's packet being dropped is proportional to the amount of data it has in a queue.

Scenario 2

Consist of six nodes (n0, n1, n2, n3, n4, and n5), the duplex link between n0 and n2, 10Mbps of bandwidth and 2ms of delay and n1 and n2 have 10Mbps of bandwidth and 3ms of delay. The duplex link between n2 and n3 has 1.5Mbps of bandwidth and 20ms of delay, each node n0 and n2, n1 and n2 use a drop tail queue while n2 and n3 uses RED policy queue management.

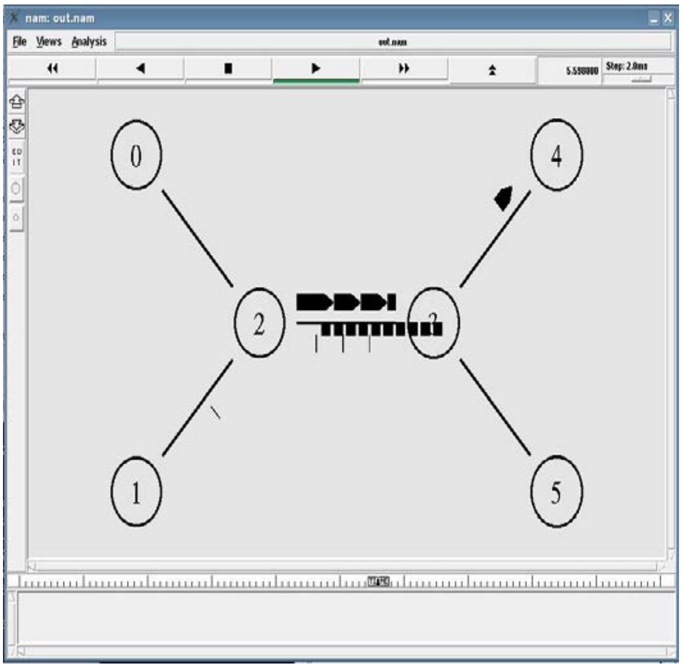

Fig 5.5a: Nam visualization of RED queuing method

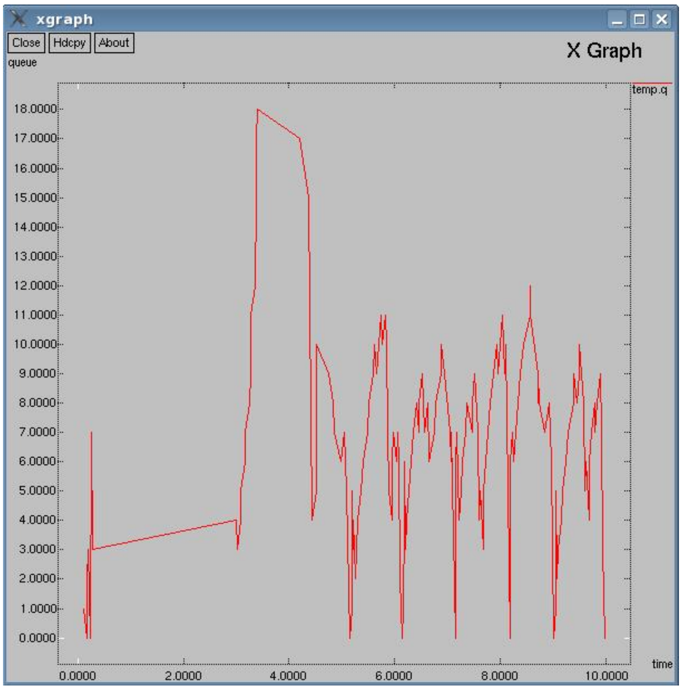

Fig 5.5b: performance graph random early detection

ISSN 2224-3577
International Journal of Science and Technology

http://www.ejournalofsciences.org

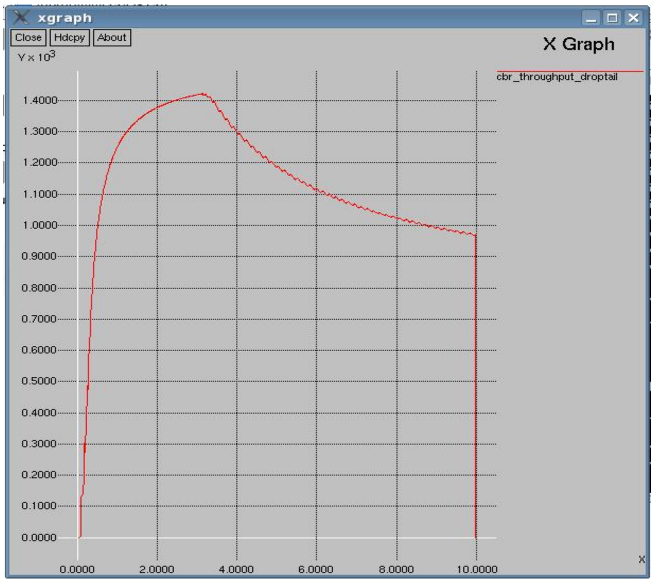

Fig 5.5c: throughput graph using drop tail techniques

VIII. OBSERVATION

It is observed that in all the policies, the performance measure is almost the same as regards to throughput, delay and jitter but there are performances in the area of loss measurement in RED more than other two queuing policy techniques.

IX. CONCLUSION

The network performance depends on how the network can be managed and the network performance parameters are adequately taken care of so high efficiency can be achieved.

REFERENCES

[1] I. Papaefstathiou1, T. Orphanoudakis, G. Kornaros, C. Kachris, I. Mavroidis, A. : Queue Management in Network Processors:

[2] G. Kornaros, I. Papaefstathiou, A. Nikologiannis, N. Zervos "A Fully-Programmable Memory Management System Supporting Queue Handling at Multi Gigabit rates", IEEE, ACM, Proceedings of the 40th DAC, Anaheim, California, U.S.A., June 2-6, 2003.

[3] Meleis, H.E., Serpanos, D.N., "Designing communication subsystems for high-speed networks", IEEE Network, Vol. 6, Issue 4.

[4] W. Zhou et. al., "Queue Management for QoS Provision Build on Network Processor" 9th IEEE Work-shop on Future Trends of Distributed Computing Systems (FTDCS'03), May 28 -30, 2003 San Juan, Puerto Rico.

[5] Halsall F (2005): Computer Networking and Internet Fifth Edition Addison Wesley

[6] http://en.wikipedia.org/wiki/Network_congestion

[7] Jeonghoon, M. and Walrand, J. (2010): Fair End-to-End Window-Based Congestion Control: IEEE/ACM transactions on networking, vol. 8, no. 5, October 2000

[8] Lathi B.P (1997): Modern Digital and Analog Communication Systems Third Edition ISBN-13: 978-0195110098

[9] D. Bertsekas and R. Gallager, Data Networks. Englewood Cliffs, NJ:Prentice-Hall, 1992.

[10] C. Eklund, R.B Marks, K.L Stanwood, and S.Wang (2002) "IEEE 802.16: a technical overview of the wireless MAN air interface for broadband wireless access" IEEE communication Magazine, 40, pp 98 102