Proc. of the IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)
26-27 November 2013, Kuala Lumpur, Malaysia

Performance Analysis of a Token Bucket Shaper for MPEG4 Video and Real Audio Signal

Faheem[#1], Fahd Humayun[#2], Mohammad Inayatullah Khan Babar[#3], Mohammad Haseeb Zafar[#4], Megat Farez Zuhairi[*5]

[#]*Department of Electrical Engineering, University of Engineering & Technology, Peshawar, Pakistan*
[1]fahim_1989@hotmail.com
[2]fahdhumayun@gmail.com
[3]babar@nwfpuet.edu.pk
[4]haseeb@nwfpuet.edu.pk
[*]*Systems and Networking Section, Universiti Kuala Lumpur, 50250, Malaysia*
[5]megatfarez@miit.unikl.edu.my

***Abstract*—The modern communication system has made the world a "Global Village" in true sense. However, as the number of users of this communication system is increasing, the challenges are mounting as well. The users send different types of traffic to the network, however it is necessary that the traffic must conform to the specified network parameters otherwise it will be troublesome. In case the sent traffic violates the agreement between the network and the user, then there is a need to regulate it. Traffic shaping plays an important part in the provision of Quality of Service (QoS) guarantee. Token Bucket Shaper, based on Token Bucket Algorithm is used to shape the bursty and random data in order to provide the QoS guarantee. This paper presents the performance analysis of a Token Bucket Shaper for real world data such as MPEG4 video and Real Audio signal. The effect of varying various Token Bucket Shaper parameters on its performance is also discussed.**

Keywords—Quality of Service; Token Bucket Shaper; MPEG4 video; Real Audio signal.

I. Introduction

Traffic policing can be used to regulate the user traffic to make sure that the sent data is in accordance with the network parameters. The leaky bucket algorithm is used to determine if the sent traffic is in accordance with the specified parameters [1].

Traffic Shaping is used to provide the Quality of Service (QoS) guarantee as QoS guarantee is the integral function of a network which can provide audio and video services. In order to submit conforming packets to the network, traffic shaper can vary the temporal characteristics of the sent traffic. This shaper is defined by the generation and consumption of tokens, and is termed as the Token bucket shaper or token bucket filter [2]. Traffic Shaping deals with making the data conform to the bandwidth while traffic policing is used to check if the data conforms to the network parameters. Traffic shaping is carried out at the user end while traffic policing takes place at the network end in order to make sure the user has not violated the service level agreement [2]. Traffic shaping is of great importance since it makes sure that no packet will be discarded at the network end due to traffic policing.

For various traffic classes, there are two approaches for using the Token Bucket shapers. One approach is to use independent token bucket shapers/filters for every traffic class, whereas the other approach is to use multiple correlated token bucket shapers for the various traffic classes. Though independent token bucket shapers are not flexible, but they are less complex compared to the correlated token bucket shapers [2].

Section II of the paper reviews some of the work done related to Token Bucket Shapers. The adopted methodology is presented in section III. Section IV shows the simulation results of the Token Bucket Shaper for various communication streams. Section V discusses the effects of varying the token bucket shaper parameters on its performance where as we conclude this paper in Section VI.

II. Related Work

T.H. Lee [2] presents the idea of using multiple correlated token bucket shapers/filters for shaping multiple traffic classes. The basic idea behind the use of correlated token bucket filters is that it provides the efficient use of the bandwidth. The unused network resources of one traffic class can be used by other traffic class, hence the user is provided with additional flexibility which does not exist when independent token bucket shapers are used. However, complexity is the major issue with the use of correlated token bucket shapers.

W.N. Gansterer et al. [3] have used the token bucket for preventing the spam emails at the outgoing Simple Mail Transfer Protocol (SMTP) server. The basic motive behind such idea is to reduce the outgoing unwanted emails that are sent through any network, since too many emails which are usually sent by any spammer can cause an extra load on the network. The token bucket is used as a central component in dynamically controlling the outgoing emails. The proper use of this technique will significantly reduce the spam emails, while it will not affect the regular e-mails. The mentioned method can eliminate the spam email messages hence reducing the network load as well as providing the customers with convenience.

I. Shames et al. [4] presents the idea of training an intelligent agent for learning about the token generation of the Token Bucket at various network states. This will result in efficient utilization of bandwidth as well preventing the overloading of the traffic in various other parts of the network. This will cause a decrease in the number of packets dropped in the whole network. The simulation results are quite satisfactory as packet dropping probability is low and the packet injection into the network is increased by decreasing the used buffer size in every router for keeping the delay (caused by large buffer/ queue size) as small as possible. The routers are divided into network routers and source routers. The source routers are those which are connected to the end nodes while the network routers are those which are not directly connected to the end nodes. Every source routers has a token bucket which has a specific bucket size and token generation rate.

P.P. Tang and T.C. Tai [5] highlight different problems which might arise in the derivation of the parameters of the token bucket. The parameters can be obtained by using the computer network's flow of traffic patterns in two scenarios. The scenarios are

- The token bucket parameter set is identified for the flow of data such that all the data are delivered immediately without introducing any delay or packet loss.
- A queue is added with the token bucket model so that the "non-conformant" packets are stored until there are sufficient tokens available in the bucket.

The addition of queue makes the traffic smoother which makes the resulting parameters of the token bucket less demanding and its fluctuations decrease as well with the passage of time. The relationship between the size of the queue and parameters of token bucket is thoroughly analyzed. The delay caused due to queuing of every packet is also presented which can be used to adjust the traffic pattern after the queuing. Algorithm for deriving the Measurement-Based Traffic Specification (MBTS) is also highlighted which gets rid of the laborious job of characterizing the traffic in advance for reservation.

III. METHODOLOGY ADOPTED

The methodology that is used in this paper is based on the use of the multiple independent token bucket shapers for every traffic class. The advantage of such token bucket shapers is that it is much simpler to implement compared to the multiple correlated token bucket shapers [2].

Each token bucket shaper has its own traffic generator, token bucket and packet queue. The user is able to vary the values of the bucket size, queue length and the token generation rate. This provides the flexibility of utilizing the token bucket shaper to its optimum potential.

Network Simulator 2 (NS-2) [6] is the utilized simulation software. The data sources which are usually used are Constant Bit Rate (CBR) and Variable Bit Rate (VBR) sources; however this paper utilizes the real world data sources such as MPEG4 video and real audio signal. The significance is that it gives an insight into the performance of the token bucket shaper with real world data rather than the CBR and VBR functions, which fails to replicate the randomness and burstiness in the real world data.

The effect of the token bucket shaper on the delay, the utilized bandwidth and the packet loss is used to analyze the performance of the token bucket shaper. The parameters of the token bucket shaper are varied and their effect on the throughput, delay and packet loss is analyzed.

Fig. 1 shows multiple independent token bucket shapers used for shaping individual traffic classes. The MPEG4 video and real audio signal data which is sent from the user end is independently shaped by the respective token bucket shaper. The data will be sent to the network only if there are sufficient tokens present in the token bucket. The presence of token in one bucket will not have any effect on the performance of the other traffic class.

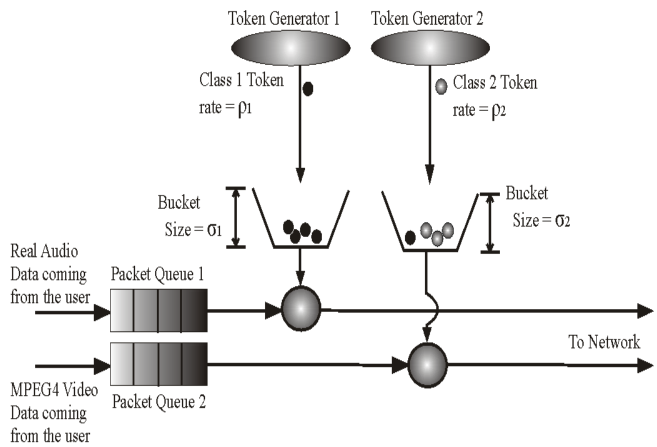

Fig. 1. Multiple Independent Token Bucket Shapers

IV. SIMULATION RESULTS AND OBSERVATIONS

The simulation environment consisted of two source nodes i.e. node '0' and node '2'. Node '3' is the sink which received the data through node '1'. Node '0' is the MPEG4 video source while node '2' is the real audio source. The link between the node '0' and node '1', node '2' and node '1' is 2.20 Mbps and the delay is 100ms. The link between node '1' and node '3' is 0.02Mbps and the delay is 100ms.

Fig. 2 shows the simulation scenario without the token bucket shaper. It can be seen that the data is sent from both nodes '0' and node '2' which is then sent to the sink node '3' through node '1'. At the bottom of the Network Animator (NAM) in Fig. 2, the unshaped and random data can be seen.

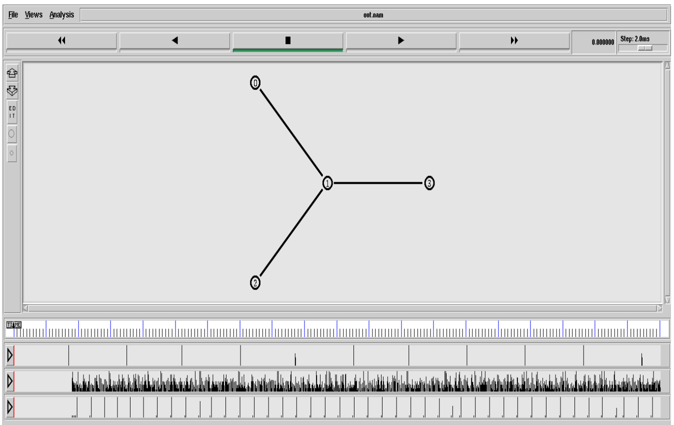

Fig. 2. The simulation scenario without token bucket shaper

Fig. 3 shows the throughput of the unshaped data which is received by the sink 3. It can be seen that the data is bursty and random which does not conform to the network parameters and will jeopardize the QoS guarantee.

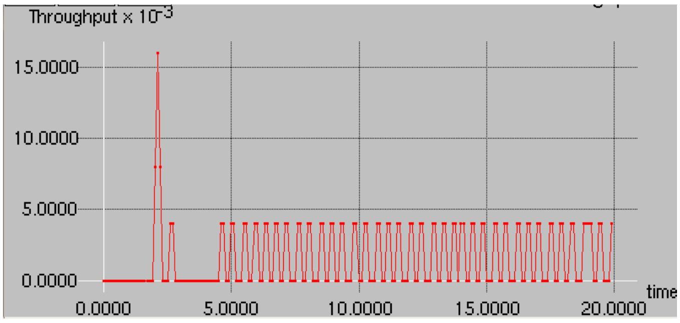

Fig. 3. Throughput without the token bucket shaper

Fig. 4 shows the packet loss which will take place in the absence of the token bucket shaper. These are the nonconforming packets which violate the bilateral agreement between the user and the network, hence they are dropped.

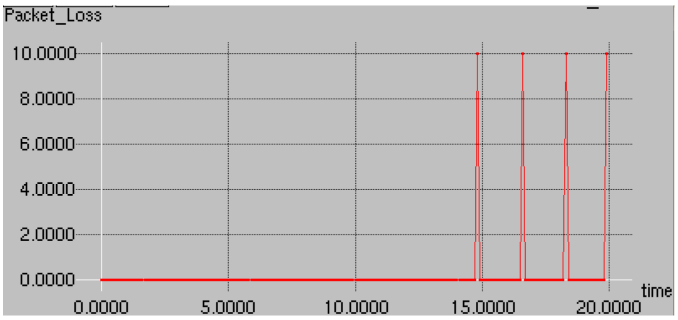

Fig. 4. Packet loss without the use of token bucket shaper

One of the significance of the token bucket shaper is that it reduces the delay which might arise due to network congestion. Network congestion might take place when the sent data is nonconforming to the network parameters.

Fig. 5 shows the delay without the token bucket shaper. There is a significant delay at the start of the simulation which reduces significantly towards the end.

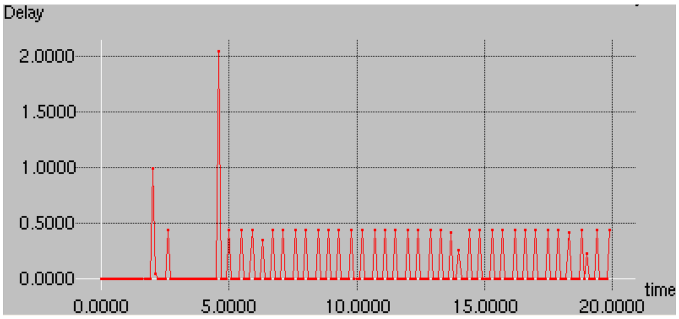

Fig. 5. The delay without the token bucket shaper

Once the token bucket shaper is added, the bursty and random data is shaped into a data which conforms to the network parameters; hence a decrease in packet loss and a shaped data which conforms to the network parameters can be expected. Fig. 6 shows the simulation scenario with the token bucket shaper attached the source nodes. The bottom part of the figure clearly shows the shaped data. In contrast with Fig. 2, the random and bursty data has been shaped according to the network parameters.

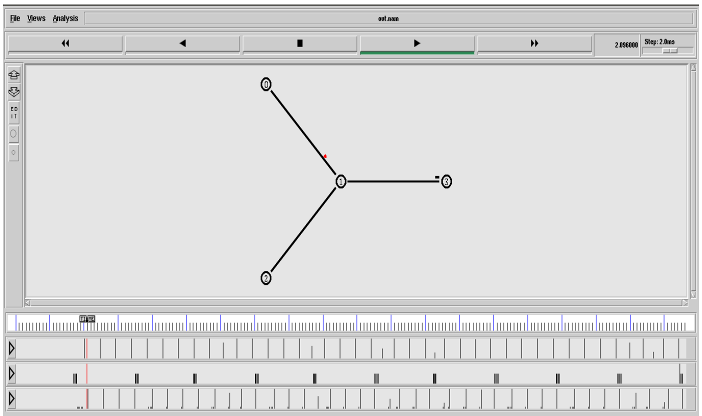

Fig. 6. Simulation scenario with token bucket shaper

Fig. 7 shows the throughput at node '3' after the token bucket shapers are installed at the source nodes. It can be seen that the data is no more random and bursty as it used to be. It has been leveled according to the network parameters.

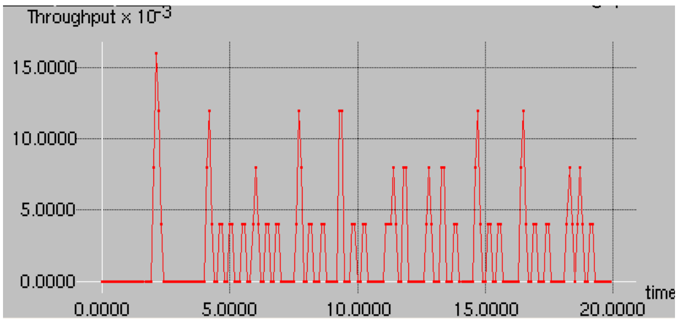

Fig. 7. The throughput with the token bucket shaper

The shaping of data will make the packets conforming to the network parameters and hence the packet loss will be reduced. Fig. 8 shows the packet loss which is zero now due to the use of the token bucket shaper. This indicates that the all the data packets will be received by node ‘3’ and no packet will be lost.

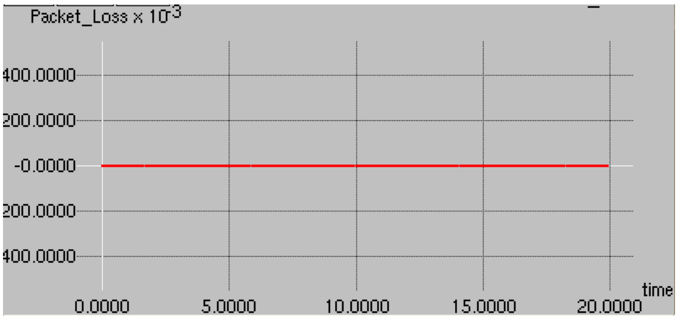

Fig. 8. Packet loss with token bucket shaper

The delay with the use of token bucket shaper occurs because the packets have to wait in packet queue till sufficient tokens are available in the token bucket. However adjusting the token bucket shaper parameters will result an optimum performing token bucket shaper which will give the optimum delay.

Fig. 9 shows the delay after the use of the token bucket shaper. The peak delay value is less compared to the delay without token bucket shaper as shown in Fig. 5, however on average the delay has increased; which can be taken care of by adjusting the token bucket shaper parameters.

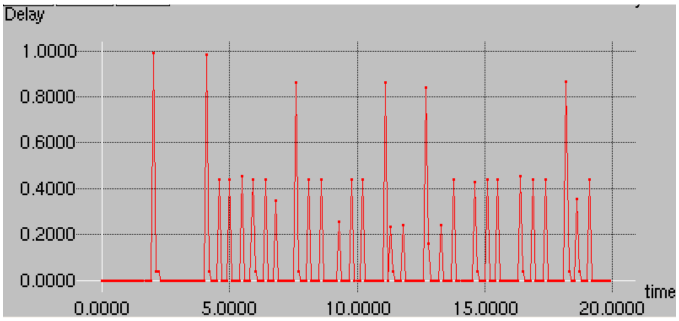

Fig. 9. Delay with token bucket shaper

V. Effect of Token Bucket Shaper Parameters Variations

The simulation results and observations shown in section IV provided us with satisfactory shaping of data and reducing the packet loss. The delay increased somewhat which can be taken care of by varying the token bucket shaper parameter. Using the proper token bucket shaper parameters will result in optimum performing token bucket shaper which will perform to its maximum potential. The effect of varying the bucket size, token generation rate and queue length on throughput, packet loss and delay will be discussed.

A. Varying the bucket size

Varying the bucket size has a significant effect on the performance of the token bucket shaper in terms of delay, throughput and the packet loss. The decrease of bucket size below a certain value, for the specific, data causes packet loss. This is clearly understood as the decrease of the bucket size means the accommodation of fewer tokens in bucket which indicates that data packets with large size will be non-conformant and they will be discarded.

Increasing the bucket size also slightly improves the delay performance where as it adds slight burstiness in the data.

B. Varying the queue length

The decrease in the queue length will cause an increase in the number of packets dropped. The queue is used to keep the non-conformant packets. A smaller queue length will fill up quickly and the incoming non-conformant packets will be discarded. Increasing the queue length will improve the packet loss performance.

The delay performance also improves with the increasing queue length whereas the burstiness increases and randomness slightly decreases with the increase in queue length.

C. Varying the token generation rate

Increasing the token generation rate improves the delay performance; however it increases the burstiness and randomness of the data as tokens are readily available for the data to flow into the network.

Similarly the packet loss increases with increase in the token generation rate as the data gets random and bursty, so the network parameters will be violated and the network will discard those packets.

VI. Conclusion

This paper presented the performance analysis of the token bucket shaper for real world data i.e. MPEG4 video and real audio data. The analysis, based on real world data showed that token bucket shaper shaped the random and bursty data and made the data conform to the network parameters. This resulted in an improved and leveled throughput. The packet loss also decreased while delay on average increased. Varying the token bucket shaper parameters assisted in taking care of the increased delay as well as provided us with optimum performing token bucket shaper. In future, there is need to come up with a mechanism which can adjust the token bucket shaper parameters automatically as required, to make it perform at optimum level.

References

[1] E.P. Rathgeb, “Modeling and Performance Comparison of Policing Mechanisms for ATM Networks”, *IEEE J. Select. Areas Commun.*, vol. 9, no. 3, pp. 325-334, Apr. 1991.

[2] T.H. Lee, “Correlated token bucket shapers for multiple traffic classes”, *IEEE VTC*, Los Angeles, Sept. 2004.

[3] W.N. Gansterer, H. Hlavacs, M. Ilger, P. Lechner and J. Strauß, “Token buckets for outgoing spam prevention”, *CNIS '05: Proceedings of the IASTED International Conference on Communication, Network, and Information Security*. ACTA Press, 2005.

[4] I. Shames, N. Najmaei, M. Zamani and A.A. Safavi: “Application of Reinforcement Learning in Development of a New Adaptive Intelligent Traffic Shaper”, *ICMLA 2006*, pp. 117-122, 2006.

[5] P.P. Tang and T.C. Tai, “Network traffic characterization using token bucket model”, *INFOCOM 1999*, pp. 51-62, 1999.

[6] NS-2 (The Network Simulator 2), http://www.isi.edu/nsnam/ns/