

DESIGN AND IMPLEMENTATION OF AN IMPROVED TIME SLIDING WINDOW TRAFFIC METER FOR QOS ASSURANCE.

^aOYETUNJI M.O., ^bEMUOYIBOFARHE O.J, ^cOLADEJI F. O ^dOLAJIDE O.J

^{ad}Department of Computer Science and Information Technology

Bowen University, Iwo, Osun State, Nigeria.

^bDepartment of Computer Science and Engineering

Ladoke Akintola University of Technology Ogbomoso, Oyo State, Nigeria.

^cDepartment of Computer Science

University of Lagos, Akoka, Lagos State, Nigeria.

kunleoyetunji@gmail.com

ABSTRACT

The deployment newly evolving multimedia application over Transport Control Protocol/Internet Protocol (TCP/IP) poses serious challenge to the quality of service (QoS) of the conventional Internet. Despite so many congestion control and QoS assurance schemes proposed by so many researchers, congestion is still an open research issues. This paper is set to develop and implement an improved Time Sliding Window 3 Color Meter (ITSW3CM) for quality of service assurance in the internet.

This research was carried out by developing a new logic for the existing meter-time sliding window three color meter (ITSW3CM) using object tool command language (OTCL) together with C++ and evaluate its performance against the existing model with the use of the following performance metrics: delay, fairness, throughput, and loss rate. Network simulator 2 (ns-2) was used for the simulation

After simulation, the results of the improved meter (ITSW3CM) and the existing meter (TSW3CM) were evaluated based on aforementioned performance metrics. The higher the throughput and fairness the better the meter and the lower the delay and loss rate the better the meter. In terms of throughput and fairness, ITSW3CM had the higher value (99.97 and 37.5%) while TSW3CM had a lower value (99.73 and 36.76) using TCP as the traffic agent. In terms of delay TSW3CM had the lower delay value of 0.066582 while the ITSW3CM had 0.097707 using TCP as the traffic agent. In terms of loss rate, ITSW3CM had the lower value of 0.03261%, with TSW3CM with 0.04891%. In short, better QoS assurance can be achieved in differentiated service with the use of the proposed improved traffic metering algorithm that guarantees better fairness.

Keywords: Quality of service (QoS), Differentiated service, Traffic meter, Time Sliding Window 3 color meter, improved Time Sliding Window Three Color Meter, TCP and UDP Protocols.

INTRODUCTION

The internet is the most used network and since its inception in 1969 as ARPANET. It has done nothing but grow exponentially in terms of users and applications. In the last 18 years the usage of data services has increased tremendously. In 1995, 16 million people used the Internet, in December 2013 this number had increased to 2,749 million (Internet world users, 2013). From a simple project created by the Advanced Research Projects Agency (ARPA) in the United States in 1967 just to send information between computers to the largest collection of public information the success of the Internet has increased its vulnerability to misuse and performance problems. Internet service providers are now faced with the challenging task of continuous monitoring of their network to ensure that security is maintained, and customers are obtaining their agreed-upon service [1].

With the Internet's 'best-effort' policy, there is little it can do when it has to deal with applications that requires guarantee of quality of service (QoS). Best-effort service, a situation where network traffic is processed as fast as possible with no assurance that the packets sent from source gets to the destination and even if it does, whether some of it was not lost along the way. Quality of service (QoS) refers to the ability of a network to provide selected traffic with better service. It has been proven that the conventional internet has

no architecture to provide or guarantee quality of service (QoS) for real-time interactive applications such as video calls, online gaming, voice over IP (VoIP), e-business etc. which have high demands for quality of service. QoS provides a hierarchical control on how its bandwidth is shared among different users of the internet [2].

In the best-effort service used by the internet, packet loss is a constant phenomenon when there is congestion in the router. This brought about the need of QoS assurance and the development of DiffServ where a user's need could be granted according to the traffic profile at the routers. Specifically, problems that can be faced in this project work are;

- 1) The problem of congestion in the differentiated service due to the influx of more users and applications that demand quality of service.
- 2) It has been demonstrated that the standardized IETF Diffserv conditioners were not good TCP traffic descriptors. Thus, the IETF, through its request for comment (RFC) for attaining QoS is searching for better algorithm to achieve QoS, maximize the use of scarce resources, minimize packet delay, loss rate and improve fairness in the network. (Oladeji et al. 2010).
- 3) Many algorithms have been proposed to curb network congestion but they have all not completely accomplished that yet, therefore the need to develop and prove the validity of a new algorithm that provides quality of service assurance.

I. TRAFFIC CLASSIFICATION AND CONDITIONING

Establishment of a Service Level Specification (SLS) between the upstream and the downstream domains is used to express Differentiated services across a domain boundary. Traffic conditioning normally takes place at the DS Domain boundary. SLS in some other references are known as Service Level Agreement (SLA). SLAs are used to define packet classifications, marking and re-marking rules, traffic profiles and actions to be performed to in-profile and out-of-profile traffic. Traffic conditioning specifications (TCS), also known as Traffic conditioning Agreement (TCA) make up an integral part of the SLS.

From the TCA we can further define the role of traffic classification, as this will identify a subset of traffic that will receive a differentiated service such that the identified traffic will then be conditioned and/or mapped to one or more behaviour aggregates. The term traffic conditioning actually relates to the metering, shaping, policing and marking of traffic in order to ensure that traffic entering a domain conforms to the rules derived in the TCA [4]. In the simplest model, each packet is either in-profile or out-of-profile based on the metering result at the arrival time of the packet. In-profile packets obtain better traffic conditioning and forwarding treatment than out-of-profile packets.

Classifiers: Classifiers are devices that forward traffic that matches specified rules for further processing. In operation these functions are located within devices at DS ingress or egress nodes. Packets are selected in a traffic stream based upon some content within the packet header [5]. Two types of classifiers are defined based upon the types of information they choose to make decisions on:

1. Behavior Aggregate (BA) Classifier

This classifier classifies packets based upon the contents of the Differentiated Services Codepoint (DSCP) field only.

2. Multi-Field (MF) Classifier

The multi-field classifier classifies packets based upon the value (s) of the combination of one or more header fields.

In conclusion, classifiers are devices that select specific packets from a stream of packets based on the content of a portion of the packet header.

Traffic Profiles: This provides the rules and conditions to decide whether a packet is in or out-of-profile. It also specifies the characteristics or properties of a stream of packets chosen by the classifier.

Traffic Conditioners: This is a collection of several tools used to check whether a packet or stream of packets sent by the classifier is in-or out-of-profile and carry out several decisions on the stream depending on whether it is in-profile or out-of-profile. These components are:

- 1) A Meter
- 2) A Shaper
- 3) A Dropper
- 4) A Marker

Traffic conditioners can be represented by the diagram shown in Figure 2.5 below. Meter measure the temporal properties of the stream of packets selected by a classifier against a traffic

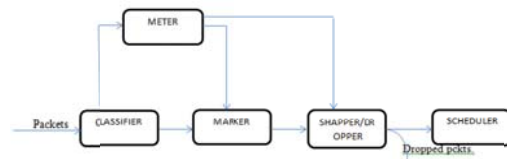


Figure 2.6: DiffServ Traffic conditioner components
Traffic conditioner design for assured forwarding in differentiated services networks
(Habib, Fahmy and Bhargava, 2000)

Figure 1: Traffic conditioning in DiffServ

profile specified in a TCA. A meter passes state information to other conditioning functions to trigger a particular action for each packet which is either in- or out-of-profile (to some extent).

The Marker is used to set the DS field of a packet to a particular codepoint, such that the marked packet can be associated with a particular DS behaviour aggregate. The marker may be configured to mark all packets which are steered to it to a single codepoint, or may be configured to mark a packet to one of a set of codepoints used to select a PHB in a PHB group, according to the state of a meter. When the marker changes the codepoint in a packet it is said to have "re-marked" the packet.

Droppers perform a very simple and straightforward task. They discard packets in a stream to bring them in compliance with a profile. This is generally known as policing the stream.

Shapers are used to introduce delays to some or all of the packets within a particular stream in order to ensure compliance with a traffic profile. Shapers usually have a finite-sized buffer, so there is a possibility of packets been discarded if there is insufficient space to hold them while they are being delayed.

II. TRAFFIC METER CONSIDERED

The already existing Time sliding window three color (TWS3CM), marker were simulated, analyzed and evaluated against the improved Time sliding window three color marker (ITWS3CM)

Time Sliding Window Three Color Marker is intended to mark packets that will be treated by the Assured Forwarding (AF) Per Hop Behavior (PHB) in downstream routers. The TSWTCM meters a traffic stream and marks packets to be green, yellow or red based on the measured throughput relative to two specified rates: Committed Target Rate (CTR) and Peak Target Rate (PTR). The TSWTCM is designed to mark packets contributing to sending rate below or equal to the CTR with green color. Packets contributing to the portion of the rate between the CTR and PTR are marked yellow. Packets causing the rate to exceed PTR are marked with red color. The TSWTCM consists of two independent components: a rate estimator, and a marker to associate a color (drop precedence) with each packet. If the marker is used with the AF PHB, each color would correspond to a level of drop precedence. The rate estimator provides an estimate of the running average bandwidth. It takes into account burstiness and smoothes out its estimate to approximate the longer-term measured sending rate of the traffic stream (Fang et al, 2000). The marking algorithm used in this meter is:

Avg_rate = Estimated average traffic sending rate

if (avg_rate ≤ CIR)

the packet is marked as green;

else if (avg_rate ≤ a * PIR) AND (avg_rate > CIR)

Calculate $P_0 = (\text{avg_rate} - \text{CIR} / \text{avg_rate})$

With probability P_0 the packet is marked as yellow;

With probability $(1 - P_0)$ the packet is marked as green;

Else

Calculate $P1 = (avg_rate - PIR / avg_rate)$

Calculate $P2 = (PIR - CIR / avg_rate)$

With probability $P1$ the packet is marked as red;

With probability $P2$ the packet is marked as yellow;

With probability $(1 - (P1+P2))$ the packet is marked as green;

III. IMPROVED TRAFFIC METER

To overcome the problem of unfair sharing of the excess bandwidth among aggregates the TSWTCM algorithm is improved. The basic idea is to allow an aggregate to inject yellow packets in proportion to its CIR. This means that a large service subscriber have more of its packet marked yellow than a small service subscriber. The improved TSWTCM algorithm can therefore, improve the inherent unfairness of TCP congestion algorithm, and result in better fair share of excess bandwidth. In this new algorithm, for $avg_rate < CIR$, all packets are marked as green packets (just like the original algorithm), which guarantees the service rate.

The main difference between the proposed algorithm and the previous algorithm is in the region $avg_rate > CIR$:

In this algorithm, the region $avg_rate > CIR$ is divided into two sub regions:

For $CIR < avg_rate < a * CIR$, all packets from an aggregate are marked as yellow. The reason to introduce this region is to alleviate the unfairness of TCP congestion control algorithms, and allow large flows to have a better chance to compete with small ones.

For $avg_rate > a * CIR$, packets in an aggregate are marked as yellow with probability $a * CIR / avg_rate$ and the rest of the traffic in that aggregate are marked as red.

In this algorithm, the yellow packet marking algorithm is proportional to its CIR instead of the PIR in order to achieve a fair way of sharing excess bandwidth. The algorithm is:

Avg_rate = Estimated average traffic sending rate

a = a constant ($a > 0$)

if ($avg_rate \leq CIR$)

the packet is marked as green;

else if ($avg_rate \leq a * CIR$)

the packet is marked as yellow; else

Calculate $P = (a * CIR / avg_rate)$

With probability P the packet is marked as yellow;

With probability $1 - P$ the packet is marked as red;



Figure 2: Flow chart for improved traffic meter

IV. EXPERIMENTAL SETUP

The network topology in figure 3.2 below consists of six sources which are combined into two sets of aggregates. There are two groups of sources, called aggregate 1 and aggregate 2. Aggregate 1 sends traffic through the edge router E1 to the destination D1, while aggregate 2 sends traffic through the edge router E2 to the destination D2. The node-to-node network links from sources to destinations were configured with bandwidth of 20Mbps and link delays of 5ms except from the core router C1 to edge router E2 which was configured as 10Mbps of bandwidth and 5ms of link delay. Each aggregate contains a number of flows from different source nodes (S). E1 and E2 are the edge routers responsible for supervising and marking traffic aggregates 1 and 2, respectively. C1 is the core router that implements active queue management, and provides service differentiation among aggregates according to the drop precedencies carried in the packet headers. E3 is the receiving edge router that that is an aggregate of two destinations.

The sources generated traffic streams with Exponential ON/OFF using TCP traffic agent and send them to ingress edge router. At the ingress routers (E1 and E2), the incoming traffic streams are classified and conditioned with traffic conditioning functions such as metering, marking policing and shaping using associated DSCP to carry out traffic profiles enforcement before sending them to the core router for forwarding. The core router buffers the packets into respective queues using priority scheduling discipline to forward them to respective destinations through the egress router (E3).

In this simulation, the multi-RED queue management scheme is used. Duplex transmission mode was used for the links between sources and edge routers while simplex modes were used for link between core router and edge routers. The traffic for this simulation were generated using FTP (file transfer protocol). FTP is the traffic generator used and supported by TCP.

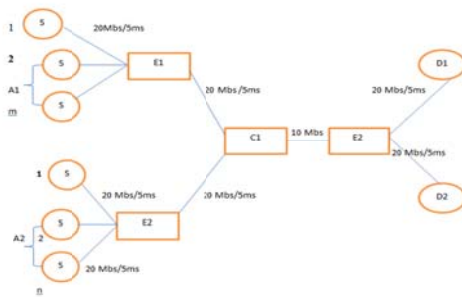


Figure 3: Simulation topology

V. RESULT ANALYSIS

PERFORMANCE BASED ON FAIRNESS

Fairness can be calculated using the most recent fairness index. Given x_i as the amount of Scheduled Traffics from queue i in the system, it is expressed as:

$$\text{Fairness} = \frac{(\sum_i x_i)^2}{N * \sum_i x_i^2}$$

VI. Where, $0 < FI < 1$,

VII. x_i = excess bandwidth obtained by aggregate i / CIR of aggregate

PERFORMANCE BASED ON PACKET DELAY

Packet delay is calculated by subtracting the time a packet is enqueued $a(n)$ from the time the packet is dequeued $d(n)$. Thus the packet delay δ is

$$\delta = d(n) - a(n)$$

PERFORMANCE BASED ON THROUGHPUT

The estimation model for throughput as described in (Kurose and Ross, 2001) was used. This is expressed as follows:

$$\text{Throughput (t)} = \lim_{t \rightarrow \infty} \frac{D(t)}{t}$$

Where $D(t)$ is the number of items departed over time interval t .

By this definition, the number of items transferred divided by the time interval involved in the transmission gives the throughput.

PERFORMANCE BASED ON LOSS

Loss can be calculated as the difference between the input function $x(t)$ and the output function $y(t)$. Similarly, the percentage of lost packets is computed as the number of packets that were eventually dropped divided by total number of packets that arrived i.e.

$$\text{Loss \%} = \frac{N_L * 100}{N_A}$$

VII. RESULT DISCUSSION

From the results obtained from the simulation, it was observed that both traffic meters used in the simulation including the improved one had areas where they perform better. The higher the throughput and fairness, the better the meter. Also, the lower the delay and loss rate, the better the meter. From the compiled results, it was revealed that in terms of throughput and fairness using TCP as traffic agent, ITSW3CM had the higher values of 99.97% and 37.5% than TSW3CM with 99.73% and 36.76% respectively.

In terms of delay using TCP as traffic agent, TSW3CM had the lower value of 0.066582 than the ITSW3CM with 0.097707. In terms of loss rate using TCP as traffic agent, ITSW3CM had the lower value of 0.03261%, than TSW3CM with 0.04891.

The ITSW3CM will make real-time applications such as voice over IP, video on demand, online gaming to perform better than TSW3CM. The ITRTCM can meet the IETF search for a better algorithm to achieve QoS, maximize the use of scarce resources, minimize packet delay and loss rate and also improve fairness in the network.

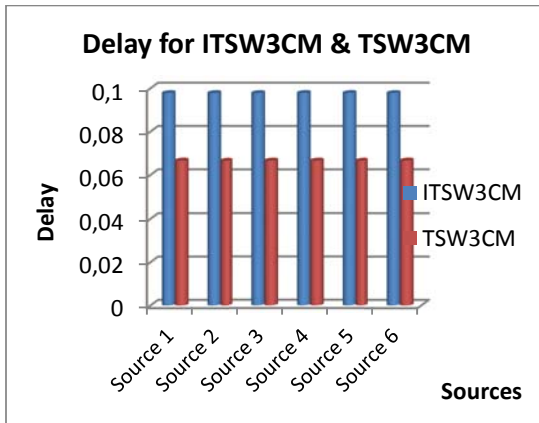


Figure 4: Delay for ITSW3CM and TSW3CM using TCP as traffic agents

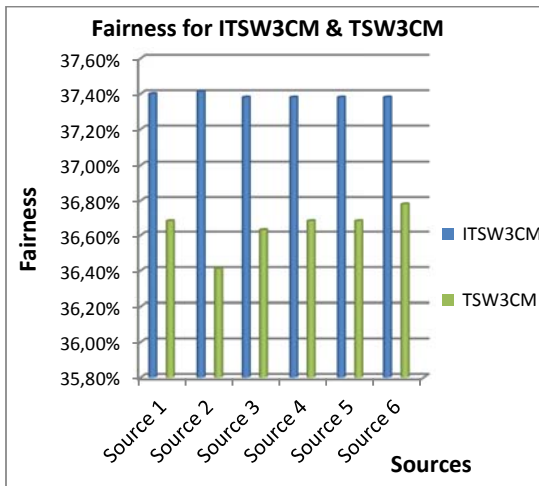


Figure 5: Fairness for ITSW3CM and TSW3CM using TCP as traffic agents

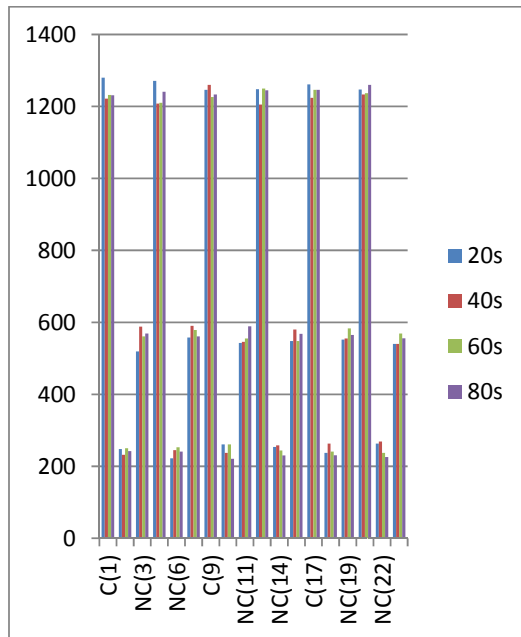


Figure 6: Throughput of ITSWM using TCP agents

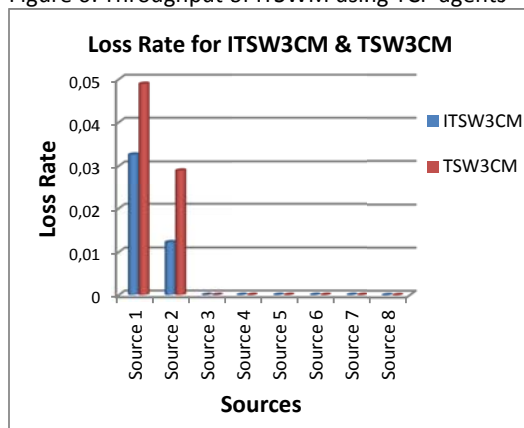


Figure 7: Loss rate for ITSW3CM and TSW3CM using TCP as traffic agents

VIII.CONCLUSION

In conclusion, better QoS assurance can be achieved in differentiated service with the use of the proposed improved traffic metering algorithm that guarantees better fairness. Finally, implementing a differential buffering scheme showed that enforcement of service level agreement can help in improving disciplines on non-compliant sources and thereby reducing congestions in routers.

REFERENCES

- [1] Habib, A., Fahmy, S., Bhargava, B. (2000). Traffic Conditioner Design for Assured Forwarding in Differentiated Services Network.
- [2] Wroclawski, J. (1997). Specification of the controlled-load network.
- [3] Oladeji, F., Uwadia, C., Oyetunji, M., &Adeyemi, M. (2010). The significance of traffic admission control and scheduling mechanisms in a QoS internet router. *International Journal for the Application of Wireless and Mobile Computing*.

- [4] Miller, & Philip, M. (2010). *Tcp/Ip - the Ultimate Protocol Guide* (Vol. 1&2). Universal-Publishers.
- [5] Cisco (2006). Differentiated Services - The Scalable End-to-End Quality of Service model.
- [6] IBM. (2013). *Traffic conditioners*. Retrieved 04 12, 2013, from IBM: <http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/index.jsp?topic=%2Frzak8%2Frzak8conditioners.htm>
- [7] Ferguson, P., & Huston, G. (1998). Quality of Service: Delivering QoS on the Internet and in Corporate Networks. *John Wiley Inc.*
- [8] Shenker, S., Partridge, C., & Guerin, R. (n.d.). *Specification of guaranteed quality of service*. RFC.
- [9] Heinanen, J., T. Finland, and R. Guerin. "A Single Rate Three Color Marker" (Internet draft). May, 1999.
- [10] Fang, W., Seddigh, N., & Nandy, B. (2000, June). A Time Sliding Window Three Colour Marker (TSWTCM). *Request for Comments*.
- [11] Arora, H., Greenwald, L., Rao, U., Novatnack, J. (2002). Performance Comparison and Analysis of Two QoS Schemes: Swan and DiffServ
- [12] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and Weiss, W. (1998). An Architecture for Differentiated Services. *IETF Draft, RFC 2474*.
- [13] Boche, H., Wolisz, A., and Zitterbart, M. (2003, June 10). The Influence of Proportional Jitter Scheduling Algorithms on Differentiated Services Networks.
- [14] Bonald, T., and Roberts, J. (2003). Congestion at Flow Level and the Impact of User Behaviour, Computer Networks. *Science Direct*, 521-536.
- [15] Braden, R., Clark, D., and Shenker, S. (1994). 'Integrated Services in the Internet Architecture: an Overview. *IETF Draft, RFC 1633*.
- [16] Clack, D., and Fang, W. (1998). Explicit allocation of best effort packet delivery service. *IEEE/ACM Transactions on Networking*, 362-374.
- [17] Eitan, A., and Tania, J. (2003). NS Simulator for beginners.
- [18] Evans, J., and Filsfils, C. (2007). *Deploying IP and MPLS QOS for Multiservice Networks*. Elsevier Inc.