

## Performance Comparison of Packet-Level Multiplexing Algorithms with Bursty Traffic

M. M. N. Aldeer\*

MEC Telematik Telecommunications Consultancy, P O Box 773, Ashar Post Office, Basrah, Iraq.

Received 29 December 2009; Revised 25 February 2010; Accepted 3 March 2010

### Abstract

Packet-based networks have opened the possibility of carrying a great variety of applications with different traffic profiles. Traffic profiles such as bursty traffic is essential for the analysis and evaluation of network performance. This paper is a simulation study devoted to comparing the performance of a range of packet-level multiplexing algorithms with bursty traffic. Four multiplexing algorithms are studied: round robin, random selection, weighted random selection and longest queue. Bursty traffic in this study is specified using two parameters: average packet rate and mean burst length. The algorithms performance has been measured in terms of three performance measures: average delay, delay variation and packet loss. The performance of multiplexing algorithms deteriorated when traffic intensity and burstiness of traffic increase. The algorithms competed between each other. Each algorithm showed better performance for a certain measure, however, with the expense of deterioration in other performance measures.

**Keywords:** Packet-based networks, multiplexing algorithm, bursty traffic, performance.

### 1. Introduction

Nowadays, most telecommunication networks are being modified to be packet-based networks due to the advances over the conventional circuit-switched networks. One of these advances is the possibility of carrying different traffic types those originated from different applications. However, that comes on the price of different QoS requirements that must be met, which are the performance requirements from the network's viewpoint [1] [2]. For instance, traffic originated from applications such as file transfer is known by being bursty and requires a bounded packet loss, while have no bounds on the delay or delay variation. In contrast, real-time applications such as real-time voice require the delay and delay variation to be at their minimum values while small amount of packet loss is tolerable. Real-time video are known by its sensitivity to delay variation and packet loss since it requires regular packet delivery and losing a piece of video information may result in degrading the QoS of the application.

The performance of a network is a function of set of factors. The two main factors are the characteristics of the carried traffic and the scheduling algorithm employed. The diverse of applications supported by the packet-based networks have different traffic characteristics. Understating the characteristics of the traffic is of major concern since it helps in enhancing the network performance [3]. One of the main traffic categories that existed in packet-based networks is the variable bit rate (VBR) traffic.

VBR traffic generate by sources that alternate between the ON and OFF states. During the ON period, the source generates a stream of packets at its peak rate while no packets are generated during the OFF period [2]. An example for VBR sources is the video applications, where their packet rate is changing significantly during scene changes [4]. The behaviour of such traffic source results in an instantaneous rate of packet generation which varies widely from the average packet rate for these traffic sources. The variability in traffic generation is often referred to as burstiness. Practically, the burstiness factor of VBR sources is greater than one, thus VBR sources are known by being bursty [1] [5]. The bursty traffic denotes a traffic flow made up of periods of heavy traffic separated by long periods of light or no traffic [6] [7]. In packet-based networks, the information is sent from the source node to the destination node as small chunks which are the packets. Multiple packets that originate from different traffic sources on a network node, to a designated node can be concentrated. The process of traffic concentration is known as multiplexing [4]. Within a packet multiplexer, the time is divided into short time intervals named time slots. In each time slot, only one packet can be picked from one of the input queues. The multiplexing process is controlled by some scheduling or multiplexing algorithm. The function of the multiplexing algorithm is to select, from the set of input ports, the packet to be transmitted over the transmission link. The multiplexing algorithm for a multiplexer can have a critical influence on the multiplexer's overall performance. Therefore it has formed the focus of many developers and researchers.

\* E-mail address: m.aldirawy@yahoo.com

ISSN: 1791-2377 © 2010 Kavala Institute of Technology. All rights reserved.

Comparing the performance of a range of packet-level multiplexing algorithms with bursty traffic forms the focus of this research. The reminding of this paper is organised as follows. Section 2 briefly introduces the investigations of the effects of bursty traffic on the network performance found in the literature. Section 3 presents a series of packet-level multiplexing algorithms that were studied in this research. The structure of the simulation program is presented in Section 4. In Section 5, the bursty traffic source model that is applied in the simulation is presented. Simulation results and analysis are discussed in Section 6. Section 7 concludes this paper.

## 2. Background

Burstiness has impacts on the network performance. Many investigations have been contributed towards studying the effects of burstiness on the network's performance. Zhang has investigated the influence of bursty traffic in a finite size buffer environment in an ATM network [8]. The results showed that the probability of cell loss depends on the ratio of buffer capacity to burst length. The author deduced that the cell loss probability increases as the burst length increases. Similar study has been done by Takano *et al.* [9]; they investigated the relation between burstiness and packet loss for internet traffic. The obtained results showed that the higher the burstiness is the higher the packet losses. In [10], Wang studied the impacts of burstiness on delay variation for real-time communications and deduced that the variation in delay increases as the traffic becomes burstier. Mazraani and Parulkar [11] measured the Ethernet performance under conditions of bursty traffic. The results implied that packet delay, queue length and packet loss increase with the burstiness of traffic sources. The authors concluded that the deterioration of the Ethernet performance is faster with the burstiness of traffic sources. The study carried out in this paper is an investigation of the influences of bursty traffic on the performance of packet-based networks from a specific point which is the multiplexing algorithm. The performance has been measured via simulation in a single node environment and only one multiplexer is employed. The performance was measured against three metrics: average delay, delay variation and packet loss.

## 3. The Algorithms Being Studied

A typical multiplexing system is a multi queue system in which a single server serves  $N$  queues, where  $N$  is at least two. The mechanism by which the server serves the queues significantly affects the performance of such systems [12] [13]. Multiplexing algorithms or schemes define these mechanisms. These schemes behave in different ways when there are multiple queues that are not empty. This difference results from the fact that each scheme has its own policy to deal with such situation. The algorithms to be studied are described in this section.

### 3.1 Round Robin

Round robin scheme treats each queue equally by serving each

queue in turn. For  $N$  sources there are  $N$  buffer queues to store the arriving packets [14]. The queues' lengths may differ from each other since the traffic is generated from independent sources that may behave differently. Round robin scheme scans the queues in a cyclic manner looking for non-empty queue to remove a packet from. Thus each queue ensures to be served again after the completion of one cycle, where the period to complete one cycle may change depending on the number of non-empty queues in each cycle. Hence, the larger the number of non-empty queues, the longer the time a packet must wait in the queue in order to be removed.

### 3.2 Random Selection

Random selection scheme polls the queue to be served randomly, where the queue to be polled must not be empty. In each time slot, random selection scheme chooses a queue randomly to remove a packet from, and this process is independent of which queue was chosen in the previous time slots, thus it became possible for a queue to be chosen in  $t$  consecutive time slots, which is not possible with round robin (unless it is the only non-empty queue among  $N$  queues). Generally speaking, with this scheme no assumption can be made on which queue to serve first, since the selection process is to be done randomly. As a result, for a source that builds up its queue size quickly, it may suffer from long delays and excessive packet loss while other shorter queues are being served. By contrast, a large size queue may dominate the attention of the server while other queues are delayed for long periods.

### 3.3 Weighted Random Selection

The random selection scheme just discussed does not ensure a better service for applications that may generate bursty traffic. For such applications, weighted random selection scheme seems to be more appropriate. Weighted random selection scheme attempts to provide service to the queues with longer lengths than other shorter queues. Consider a random selection system in which every queue  $i$ , has a number  $\lambda_i$  associated with it.  $\lambda_i$  represents the possibility of that queue to be selected and it is proportional to the length of queue  $i$ .  $\lambda_i$  can be found using the formula given below:

$$\lambda_i = \frac{L_i}{\sum_i \lambda_i} \quad (1)$$

Where  $L_i$  is the queue length of queue  $i$ . Hence,  $\sum_i \lambda_i = 1$ .

The concept of the selection possibilities is central to defining the weighted random selection technique: Suppose there are three queues  $A$ ,  $B$  and  $C$ . Their lengths at a given time slot are  $L_A=1$ ,  $L_B=3$  and  $L_C=5$ . Then the resultant possibilities are  $1/9$ ,  $1/3$  and  $5/9$  respectively. This is illustrated in Fig. 1.

Clearly the possibility for queue  $C$  to be selected is the largest since its length is the longest ( $=5$ ) as compared with the other two queues. Meanwhile, the possibility of selecting queue  $A$  is the smallest since it is the shortest ( $=1$ ). However, the queue with the longest length may not always be selected, because the selection process is still done by the server randomly, but the longer a queue is the more likely to be selected. Thus the weighted random se-

lection scheme provides a random selection service but with bias towards the longer queue.

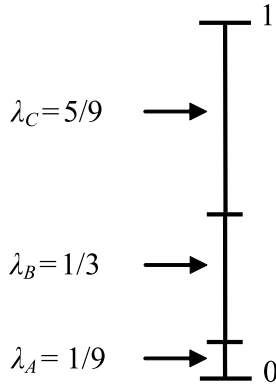


Figure 1. Selection mechanism in weighted random selection scheme.

### 3.4 Longest Queue

The longest queue scheme was originally proposed by Fan as a service policy for multi queue systems [12]. Longest queue scheme aims to provide better service for the longest queue. For  $N$  independent sources, newly arriving packets queue up in  $N$  independent queues. In each time slot the server selects the longest queue from the  $N$  queues and serves a packet from it. Thus, if that queue remains the longest in the next  $t$  time slots then it will be served again (if there are several queues of equal lengths at the same time slot, then one of these queues will be selected randomly) [12].

The major advantage of the longest queue technique is that it tries to prevent packet loss that may be incurred from a source generating traffic at high rate by giving higher service priority for its corresponding queue. However, there is a trade off between packet loss and packet delay, whereas if the difference between the length of the longest queue and other queues is big this could lead to an increase in the delay of packets from other shorter queues.

### 4. Simulation Model

The multiplexing system model considered in the simulation is assumed to be consisting of  $N$  homogenous but independent ON-OFF bursty traffic sources and an input buffer consisting of  $N$  separated queues, Fig.2. All sources are identical by their average rate and mean burst length and have the ability to produce bursty traffic. The information generated by each source is divided into small chunks, those are the packets. All sources states' are assumed to be OFF, at the beginning of the simulation.

The time is slotted into fixed-size units those are the time slots, where the width of one time slot equals the time that a source needs to generate one packet. Then the generated packets are assumed to be of fixed size rather than variable size. Also one time slot equal to the time that the server requires to pick a packet from one of the non-empty queues, where only one packet can be removed from one non-empty queue in each time slot. This is the aggregation process, and it is assumed to occur at the beginning of the time slot. The order of serving multi non-empty queues is controlled by the multiplexing algorithm employed. The multiplexed traffic is

assumed to form a stream of packets rather than being divided into frames, each of which contains multiple packets. At a given time slot, if all buffer queues are empty, a *NULL* packet is generated by the server to preserve the constant packet rate of the output line. The event when the source generates a packet is assumed to occur at the middle of the time slot. However, in every time slot there may be more than one source generating packets. The arrival of a generated packet at the buffer queue is independent from other buffer queues and can only occur at the end of the time slot. Each packet queues up in the queue allocated for its source. Within each buffer queue, packets are being served in FCFS manner.

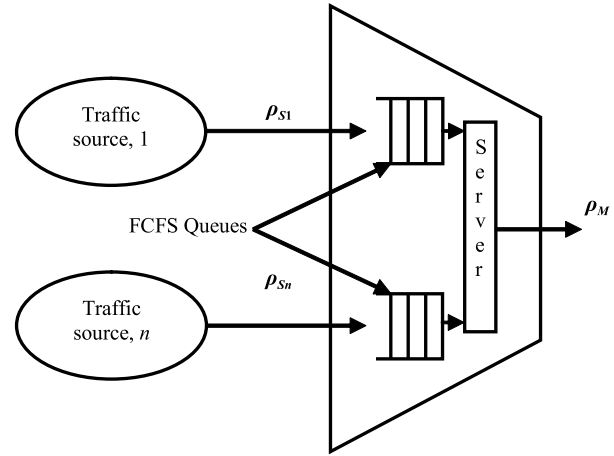


Figure 2. Simulation model for a packet multiplexer.

### 5. Bursty Traffic Source Model

Evaluating the performance of telecommunications networks requires an efficient traffic model that can accurately represent the actual traffic characteristics. The model used for modelling bursty traffic sources, is a stochastic model, that is the two-state Markov Modulated Bernoulli Process (MMBP-2) [2] [15]. Fig.3 illustrates the MMBP-2 traffic model.

As illustrated in Fig.3, there are two states in this model. In state 1, the traffic model produces traffic (packets) with probability  $\alpha$ , and in state 2, the traffic model produces traffic with probability  $\beta$  [15]. The traffic generated in a time slot behaves like a Bernoulli process and depends on the current state of the model (state 1 or state 2) [15] [16].

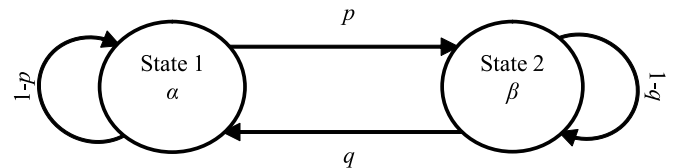


Figure 3. The MMBP-2 traffic model.

The transitions between the two states are governed by a stochastic process, which is Markov process. The notion of Markov process depends on the rule that the future of a process is only decided based on the current state of the system and not on the past states. In other words, when applying Markov property in modelling traffic sources, it can be said that at time slot  $t_n$ , the source is in one of the states (state 1 or state 2, i.e. it is a 2-state Markov

process), the state of the source at time slot  $t_{n+1}$  depends on the current state and not on the state in time slot  $t_{n-1}$  or earlier [7].

The transition probability matrix  $T$ , which governs the 2-state Markov process, is defined as following:

$$T = \begin{bmatrix} 1-q & \rho \\ q & 1-\rho \end{bmatrix}$$

If the source is in state 1, it will remain in the same state with probability  $1-p$  and transit to state 2 with probability  $p$ . In similar, if the source is in state 2, it will remain in the same state with probability  $1-q$ , and changes to state 1 with probability  $q$ .

In the simulation, a more specific version of MMBP-2 traffic source is used, which is the ON-OFF model [15]. This model has been used extensively in previous works since it can produce the bursty nature of bursty applications. In the ON-OFF model, the probabilities  $\alpha$  and  $\beta$  are set to 0 and 1 respectively. It means that the probability of generating traffic in the OFF state is 0, meanwhile the probability of generating traffic in the ON state is 1 [15]. While the source is in the ON state it considered active and it emits fixed-size packets. In contrast, in the OFF state the source is considered silent or idle and does not emit packets at all [17] [18]. In the simulation, mean sojourn time that is the mean of the time required for transiting between states is included in the length of the time slot. The average number of packets being generated in a time slot by one ON-OFF source (the offered load of the traffic source) denoted by  $\rho_s$ , is given by:

$$\rho_s = \frac{\rho}{\rho + q} \quad (2)$$

The source may remain in the ON state for one time slot or  $n$  successive time slots and then changes to the OFF state. The packets generated in the ON state are referred to as a burst. Then for the ON-OFF source model, it can be said that  $p$  denotes the probability of starting a new burst and the termination of that burst depends on the probability of going to the OFF state,  $q$ . Hence, the larger the number of time slots that a source spends in the ON state, the longer the generated burst. The mean burst length  $L$  of an ON-OFF traffic source is given below:

$$L = \frac{1}{q} \quad (3)$$

Therefore, the applied ON-OFF source model can be characterised by specifying value for the two parameters, traffic load of the source,  $\rho_s$  and mean burst length,  $L$ . However, for the simulation  $\rho_s$  does not need to be specified, since it can be deduced by using the following formula:

$$\rho_s = \frac{\rho_M}{N} \quad (4)$$

Where  $\rho_M$  ( $0 < \rho_M < 1$ ) is the traffic load of the multiplex and  $N$  ( $N \geq 2$ ) is the number of sources. Refer to [19] for the proof of formulas (2), (3) and (4).

## 6. Simulation Results and Discussion

The simulation results are presented in this section. In each run, either the traffic load or the mean burst length has been varied. The algorithms' performance were analysed by examining the effect of these two parameters on the performance measures. All results presented in this section were obtained when running the simulation for 10,000 time slots in each case. The size of buffer queues have been dimensioned when running the simulation with flexible buffer (infinite size buffer) on the assumption that the packet loss probability is not more than  $10^{-3}$ . Each queue can hold an equal number of packets. For full simulation results refer to [19].

### 6.1 Effect of Varying Traffic Load on Performance Measures

For this set of simulations, the mean burst length was fixed at 8 packets while the traffic load was varying from 0.1 to 0.9. The size of each buffer queue was dimensioned to 30 packets. A study for the curves in Fig.4 shows that the average delay for all algorithms tends to increase as the traffic load increases. It is clear that the curves are convoluted for light loads.

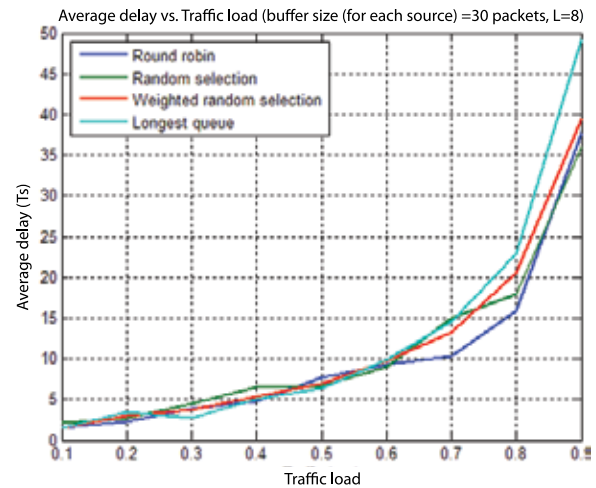


Figure 4. Average delay vs. Traffic load.

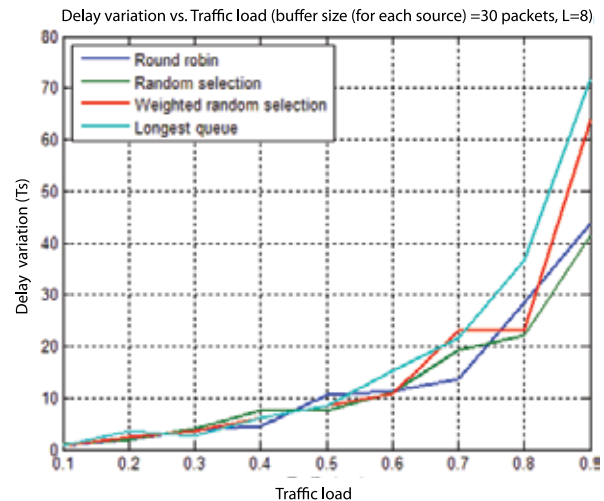


Figure 5. Delay variation vs. Traffic load.

However, as the traffic load becomes higher these curves diverge. As expected, round robin and weighted random selection algorithms performed better than longest queue algorithm which incurred the highest average delays. An interesting result to note here is that for random selection algorithm, the average delay performance was acceptable since it was equivalent to round robin or weighted random selection algorithms performance.

In Fig.5, the effect of increased traffic load on delay variation is illustrated. It is notable that for light traffic loads the algorithms performed similarly. Clearly for all algorithms, the delay variation performance drops as the traffic load increases. However, the worst performance from this point of view was by the longest queue algorithm.

Fig.6 depicts the packet loss ratio result for all algorithms as a function of traffic load.

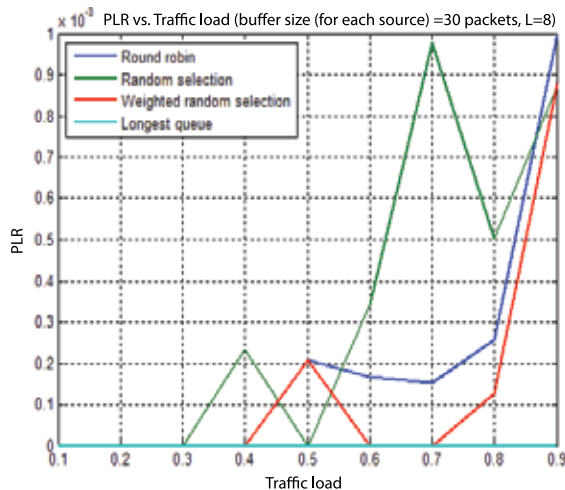


Figure 6. Packet loss vs. Traffic load.

All algorithms tend to loss packets with the increase in traffic load. Clearly, the longest queue algorithm performance is better than other multiplexing algorithms since it did not incur any packet losses. These results are comparable to those reported by Fan in [12]. And this is anticipated because the longest queue algorithm always tends to serve the longest queue first while other shorter queues are being delayed, which in this case acts as a trade off between delay and packet loss. The only expectation for losing a packet with the longest queue algorithm is that due to the appearance of two or more queues those are the longest at a given time slot, where in such case the longest queue algorithm selects one of these queues randomly to be served while the others will loss packets if the size of the longest queues were equal to the buffer size. The weighted random selection algorithm performance was better than the round robin and random selection algorithms. However, that came on the expense of higher maximum delay and delay variation.

## 6.2 Effect of Varying Mean Burst Length on Performance Measures

To examine the effect of varying the mean burst length on the performance of multiplexing algorithms, a number of experiments were conducted when the traffic load is fixed and the mean burst

length varied. The size of each buffer queue was dimensioned to 20 packets. Fig.7 illustrates the average delay comparison results as a function of mean burst length for a traffic load of 0.5.

From this Figure, it can be observed that the average delay for all multiplexing algorithms scale up accordingly with the increase of traffic burstiness. Clearly, round robin algorithm performed well for all values of mean burst length, however that came at the expense of bad packet loss performance, as it will be seen hereafter.

The delay variation is depicted in Fig.8. It is observed that for all multiplexing algorithms, the delay variation increases as the mean burst length increases. The algorithms competed between each other, whereas the better performance was by the round robin and weighted random selection algorithms and the worst was by the longest queue algorithm.

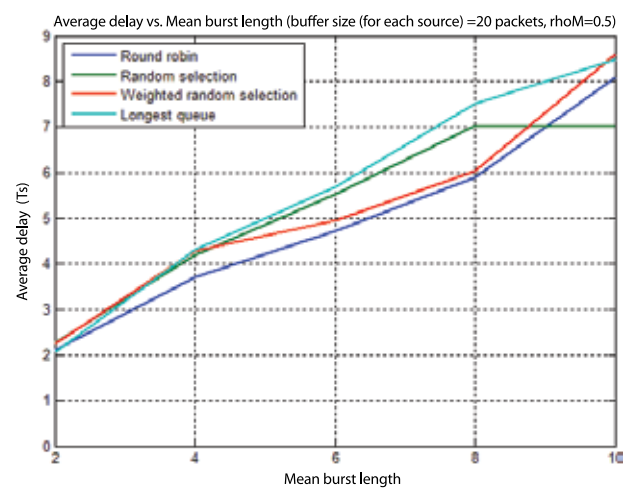


Figure 7. Average delay vs. Mean burst length.

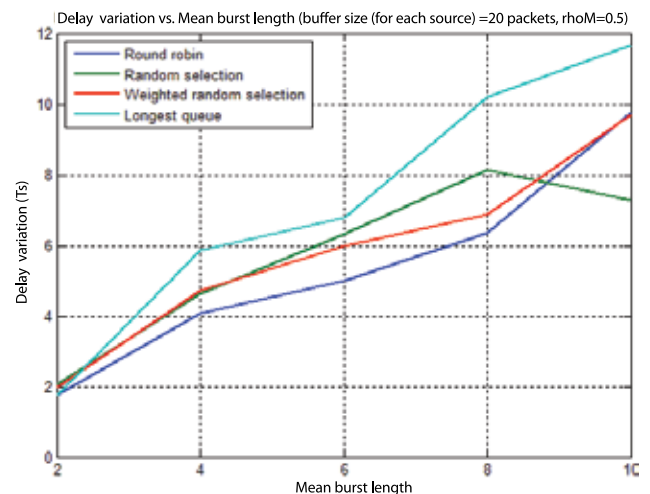


Figure 8. Delay variation vs. Mean burst length.

In Fig.9, the packet loss ratio as a function of mean burst length is illustrated. From this figure it can be said that all algorithms except the longest queue algorithm have the tendency to loss packets as the burst length increases. It is remarkable that increasing the burstiness is less pronounced on the longest queue performance from this viewpoint since it did not incur packet losses.



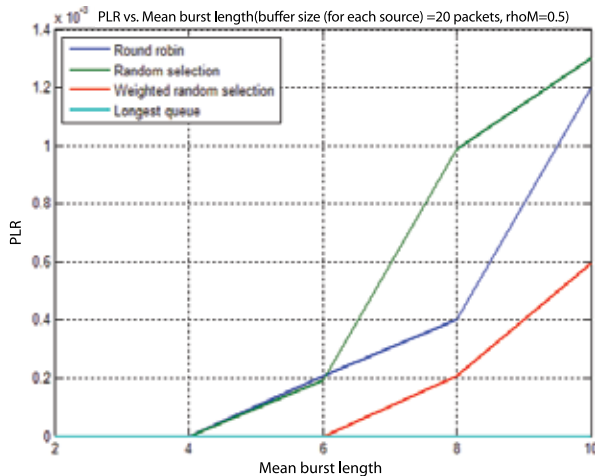


Figure 9. Packet loss ratio vs. Mean burst length.

However, this came with the cost of high average delay and delay variation as it was shown in Figures 7~8. Also for the weighted random selection algorithm it is notable that it gave less packet loss than the round robin and random selection algorithms. This anticipated because as it was claimed in Section 3, this algorithm has the tendency to serve the longer queue.

From the results observed from the experiments carried out in this section, it is clear that there was underestimating for the buffer size since the packet loss exceeded  $10^{-3}$ . Thus, the buffer size must be dimensioned properly in order to obtain better results for packet loss. Overall, the simulation results show that for a given average burst length, the performance of multiplexing algorithms deteriorates as the traffic intensity increases. Similarly, for a given traffic load, the performance of the multiplexing algorithms deteriorates as the traffic burstiness increases.

## 7. Conclusion

Packet-based networks allow combining data originating from dif-

ferent applications such as video, voice and data applications with different traffic patterns. It is claimed that most applications generate bursty traffic, and it is essential to characterise the multiplexing algorithms performance under such traffic profiles. The simulation study reported in this paper has examined the performance of four packet-level multiplexing algorithms with bursty traffic. Results show that the performance of multiplexing algorithms changes with bursty traffic, which is an expected result. For low traffic conditions the drop in performance measures are minimal. For normal and heavy traffic, the drop in performance measure becomes more pronounced. The important conclusion of this simulation study is that the multiplexing algorithms performance deteriorates faster with the burstiness of traffic sources. For example, average delay, delay variation increase accordingly with the burstiness of traffic sources. It has been shown that round robin algorithm performs better than other multiplexing algorithms from the delay performance viewpoint at cost of high packet loss. From the packet loss point of view, longest queue algorithm performs well, where the packet loss is minimal but at cost of high average delay and delay variation. Weighted random selection algorithm shows a balanced performance since it gives good average delay and delay variation performance that is similar to round robin and good packet loss performance that is similar to longest queue algorithm. The performance of random selection algorithm gives an acceptable delay performance and good delay performance in finite buffer environment.

From these observations it can be concluded that with bursty traffic, all multiplexing algorithms have drawbacks in specific performance measure. However, these drawbacks may not be of major impacts on the applications supported by the network. This depends on the application itself.

## Acknowledgment

The author would like to express his great thanks to Dr. Alan Hewitt from the ECE department in the University of Portsmouth-UK, for his support in all parts of the topic in [19].

## References

- Chen T. M., Network Traffic Modeling. Ch.15 in H. Bidgoli (Ed.), *The Handbook of Computer Networks*, vol.3, pp. 326-339. John Wiley & Sons, New Jersey, 2008.
- Kwak J. Y., Nam J. S. & Kim D. H., "A Modified Dynamic Weighted Round Robin Cell Scheduling Algorithm". *ETRI journal*, vol.24, no.5, pp. 360-372, 2002.
- Stallings W., *High-Speed Networks and Internets: Performance and Quality of Service*, Prentice-Hall, New Jersey, 2002.
- Chandra K., Statistical Time Division Multiplexing. Ch.38 in H. Bidgoli (Ed.), *The Handbook of Computer Networks*. vol.1, pp. 579-590, John Wiley & Sons, New Jersey, 2008.
- Martins J. S. B., Quality of Service in IP Networks [Electronic version], In A. Aidaraous & T. Plevyak (Eds.), *Managing IP Networks*, pp. 57-142, Wiley Interscience, New Jersey, 2003.
- Gyires T., "Methodology for Modeling the Impact of Traffic Burstiness on High-Speed Networks", In Proceedings of the IEEE SMC'99 Conference, Tokyo, Japan, vol.1, pp. 980-985.
- Tutsch D., *Performance Analysis of Network Architectures* [Electronic version], Springer, Berlin, 2006.
- Zhang Z., "Finite Buffer Discrete-Time Queues with Multiple Markovian Arrivals and Services in ATM Networks", In Proceedings of the IEEE INFOCOM'92 Conference, vol.3, Florence, Italy, pp. 2026-2035, 1992.
- Takano R., Kodama Y., Kudoh T., Matsuda M. & Okazaki F. (2006, February), *Realtime Burstiness Measurement*, Paper presented at the PFLD-net2006 [Nara Japan].
- Wang Z. & Crowcraft J., "Analysis of Burstiness and Jitter in Multimedia Communications", In Proceedings of the IEEE'93 GLOBECOM Conference, Houston, Texas, vol.3, pp. 1496-1500, 1992.

11. Mazraani T. Y. & Parulkar G. M., "Performance Analysis of the Ethernet under Conditions of Bursty Traffic", In Proceedings of the IEEE'92 GLOBECOM Conference, Orlando, Florida, vol.1, pp. 592-596, 1992.
12. Fan Y. H., "New Policy for Multi Queue Systems with Finite Queue Size", *Electronic Letters*, vol.27, no.16, pp. 1434-1435, 1991.
13. Fan Y., Jianli W. & Chuan W., "Performance Analysis of Internal Buffered Crossbar-Based ATM Switches with LOQF Policy", In Proceedings of the IEEE TENCON'93 Conference, Beijing, China, vol. 0, no.3, pp. 255-258, 1993.
14. Chao H. J. & Liu B., *High Performance Switches and Routers*, Wiley Interscience, New Jersey, 2007.
15. Ng C.-H., Yuan L., Fu W. & Zhang L., "Methodology for Traffic Modeling Using Two-State Markov-Modulated Bernoulli Process", *Computer Communication*, vol.22, pp. 1266-1273, 1999.
16. Xiong L., *Scheduling in Packet Switches with Relaxed Constraint*. MSc thesis, University of Wollongong, Wollongong, 2004.
17. Marafih N. M., Pickholtz R.L. & Ya-Qin Zhang, "New Approximation for Analysis the Multiplexing of Bursty Sources in ATM Networks", In Proceedings of the IEEE GLOBECOM'94, San Francisco, California, vol.2, pp. 1085-1089, 1994.
18. Jianxin L., Lemin L. & Hairong S., "The Influence of Burstiness and Correlation of Traffic on an ATM Multiplexer", In Proceedings of the IEEE ICCT'96 Conference, Beijing, China, vol.1, pp. 20-23, 1996.
19. Aldeer M. M. N., *Multiplexing/Concentrating Bursty Packet Data*, MSc dissertation, University of Portsmouth, Portsmouth, 2008.