Malaysian Journal of Computer Science, Vol. 16 No. 2, December 2003, pp. 11-20

PRO-ACTIVE QoS RESOURCE MANAGEMENT SCHEMES FOR FUTURE INTEGRATED PACKET-SWITCHED NETWORKS

Shamala Subramaniam
Dept. of Communication Technology and Networks
Faculty of Computer Science & Information Technology
Universiti Putra Malaysia
43400 Serdang, Selangor
Tel: 603-89466512
Fax: 603-89466572
email: shamala@fsktm.upm.edu.my

Mohamed Othman
Dept. of Communication Technology and Networks
Faculty of Computer Science & Information Technology
Universiti Putra Malaysia
43400 Serdang, Selangor
Tel: 603-89466512
Fax: 603-89466572
email: mothman@fsktm.upm.edu.my

Rozita Johari
Dept. of Communication Technology and Networks
Faculty of Computer Science & Information Technology
Universiti Putra Malaysia
43400 Serdang, Selangor
Tel: 603-89466512
Fax: 603-89466572
email: rozita@fsktm.upm.edu.my

Mohd Yazid Mohd Saman
Faculty of Computer Science and Information Technology
College Universiti Sains Terengganu
Kuala Terengganu

ABSTRACT

In this research two pro-active dynamic QoS resource management schemes are designed, namely the dynamic QoS control scheme with delay estimation, and the hybrid dynamic QoS control scheme. In both schemes, every new packet arrival is compared against the computed estimated delay it will experience, prior to being admitted into the buffer. If the computed estimated delay expires the requested delay bound, then the packet is dropped. In the hybrid scheme, every packet is first assessed for the estimated delay prior to being admitted into the buffer, subsequently the packets which have been successfully admitted into the buffer are evaluated on the actual delay experienced before being transmitted to the receiver. The paper studies the performance of the two proposed schemes with a dynamic resource management scheme, known as the OCcuPancy_Adjusting (OCP_A). The results obtained through the simulation models show that the proposed schemes have significantly improved the average delay for different traffic patterns. In addition to improving the average delay in delay sensitive traffic, improvement is seen in the average packet loss ratio, and subsequently increasing the throughput of delay sensitive traffic.

Keywords: ***Performance analysis, Resource management and QoS algorithms***

1.0 INTRODUCTION

The nature of communication systems, the constantly evolving and growing demands of the diligent 'on-line' population coupled with the stringent requirements of future applications are causing a pressing need for existing algorithms, protocols and architectures to be revamped and designed [1]. Packet-switching networks have been long dominated by the advent features of data applications. These applications advocate low delay, low packet loss and high throughput. Algorithms and protocols that were developed to realise these pre-requisites employed the 'best-effort' service. This service provided no service guarantees to the clients, allowed drastic service degradation when networks are overloaded, required no resource reservations and employed the First Come First Serve packet scheduling (FCFS) algorithms [1, 2, 3, 4]. The requirements of low speed data applications such as telnet and file transfer protocol (ftp) were catered for efficiently via these 'best-effort' services.

Emerging multimedia technology used in computing and communication system, offers a wide spectrum of opportunity which are equally challenging. These challenges are largely related to the characteristics and nature of multimedia traffic [4, 5]. Among the prominent features are as follows: it is composed of multiple traffic patterns, demands stringent quality requirements, created bursty network traffic, requires differentiated communication modes and needs integrated services in a common network. Efforts to integrate multimedia applications into traditional data architectures proved to be unsuccessful [6]. This was partially caused by the fact that traditional architectures provided single level best-effort service [7]. Multimedia applications which constitutes of various data imposes different QoS. Among the implications of this, is the need for resource reservation and traffic dependent services.

Many QoS control algorithms were derived to enable conducive platforms for the deployments of networked multimedia system. Initial schemes revolved from the high weight-age placed on timing requirements of real-time applications. These schemes were oriented in guaranteeing the requested services at no compromise. Subsequently, resource reservations were done based on worst-case analysis and these reservations were maintained throughout the entire lifetime of the transmission. Among the schemes that practiced this mode of resource management were the Weighted Fair Queuing [8], Virtual Clock [9], Self-Clocked Fair Queuing [10] and Burst Scheduling [11].

These schemes provided service guarantees, at the expense of resource utilisation [12]. Eventually, with the rapid increase of packet-switching networks (i.e. Internet) users, schemes required two mutually exclusive performance parameters to be supported. The parameters were providing service guarantees and high resource utilisation. The development of these schemes were further motivated by the development of adaptive applications [13]. Predictive services, whereby the amount of resources allocated to a flow can be dynamically tuned. The reallocation of the resources is based on the actual service experienced in comparison to the requested service [12]. Predictive schemes have proven to satisfy the requested service whilst simultaneously achieving high resource utilisation [14]. In this paper, we proposed two new QoS control algorithms which are based on the predictive service schemes. The organisation of the paper is as follows: Section 2 presents the model development of a dynamic QoS control algorithm and the two newly proposed QoS control algorithms. Section 3, discusses the system description and simulation model under consideration. The results obtained from the performance analysis are presented in Section 4. The paper is concluded in Section 5.

2.0 MODEL DEVELOPMENT

This section presents the development of three models which are as follows: the dynamic QoS control model (OCP_A), the dynamic QoS control scheme with delay estimation model and the hybrid dynamic QoS control scheme model. The first model algorithm is taken from [15] as the reference model for evaluation. The sub-sections below provide a detail description of the algorithms.

2.1 Dynamic QoS Control Model

The dynamic QoS control scheme presented in [15] has two objectives, which are: high resource utilisation and satisfying the requested QoS. These objectives are achieved by dynamically allocating the amount of resources needed to satisfy the requested QoS. The user generated flows are associated with the desired QoS tuple $\{D_i , L_i\}$, where D_i is the delay bound/deadline and L_i is the tolerated packet loss ratio. Based upon the specified QoS tuple, a referenced amount of resources are allocated. Two resources are considered: the bandwidth resource and the buffer resource. These resources are specified in a form known as the Current Service Environment, CSE $\{B_i, \gamma_i\}$, where B_i is the allocated buffer size and γ_i is the allocated service rate in packets per second.

The resource reallocation in [12] is done based on the observed packet loss ratio. There are two situations that can cause a packet loss: a new packet arrival is discarded due to a full buffer; and a buffered packet that has exceeded the requested delay bound. The latter is included into the packet loss consideration because transmitting a packet that has expired the requested delay will not contribute to the receiver's desired quality, it will unnecessarily consume resources which will subsequently affect other applications QoS. The dynamic resource allocation is performed to ensure that the observed loss ratio performance is within the range by using only the minimum amount of resources. In the event of a new packet arrival and a full buffer is observed, then a check on the observed loss ratio performance will be conducted. This step is performed to ensure that if the new packet is dropped, the total observed loss ratio will not exceed the requested loss ratio performance, L_i. Thus, if the observed loss ratio performance is below the requested loss ratio performance and discarding the newly arrived packet will not violate the specified loss ratio (L_i) then the new packet is dropped. However, if by dropping the newly arrived packet the requested loss ratio performance will be exceeded, then the allocated buffer size (B_i) is increased such that, the new buffer size is increased by an amount equivalent to the upper threshold value. The dynamic QoS control algorithm is shown in Fig. 1.

User specifies QoS tuple {D_i, L_i};
Determine the initial amount of resources to be allocated {B_{iref}, γ_{ref}};
Determine the upper and lower buffer occupancy threshold value;
Determine the minimum amount of resources to be allocated at any one time {B_{min}, $\gamma_{\min}$ };
Determine the appropriate interval for performing resource reallocation;

Start with an initial reference amount of resources;
Serve packets from this flow under the current allocation;

For each packet arrival:

Total packet arrived = packet arrived + 1;

/* dynamic resource allocations */

If (total packet arrived = interval variable)
If (the allocated buffer is full and losing this arrival will cause the loss ratio performance than L_i)

Increase B_i to an amount so that the new occupancy is OCP_u;

Perform i $\leftarrow B_i/D_i$;

Before transmission of each buffered packet:

Calculate the delay experienced by the packet;

/* dropping of transit packets that have expired their delay bound */

If (the experienced delay > requested delay bound (L_i))

The packet is dropped;

After the transmission of each packet:

Total packet transmitted = packet transmitted +1;
If the (total packet transmitted = confidence time)
Decrease B_i by an amount but make it no lower than B_{imin}

Perform i $\leftarrow B_i / D_i$, when there are no packets in the buffer;

Fig. 1: The dynamic QoS control algorithm

2.2 Dynamic QoS Control Scheme with Delay Estimation

The objective of the second model (model 2) is to improve the dual objective of higher resource utilisation and satisfying the requested QoS, as in the previous model. The focus of achieving this objective is on the packet loss ratio calculation and the delay management schemes implemented in the dynamic QoS control model. In the dynamic QoS control scheme, two reasons cause a packet loss to occur: the arrival of a new packet is discarded in the event of a full buffer and if the delay experienced by a buffered packet at the transmission moment exceeds the specified delay bound, then the packet is dropped. With this approach, a packet may be unnecessarily buffered, even if it is likely to be rejected because it may exceed the specified time delay. Therefore, in the second model, an additional feature is integrated in the algorithm. Now, the proposed packet loss ratio calculation comprises of the following: in the event of an arrival of a new packet, the estimated delay to be experienced by the packet is calculated. The calculation of the estimated delay is performed before the new packet may be admitted into the buffer.

If the calculated estimated delay exceeds the requested delay bound, then the packet is discarded. It is proposed that by implementing this control mechanism, all the packets that are admitted into the buffer will be transmitted within the specified delay bound. In the second model, a new control parameter is introduced, which is the estimated delay control parameter (δ). The estimated delay parameter is derived based on the delay experienced by the packets that have been transmitted and the buffered packets.

The algorithm for the dynamic QoS control algorithm with delay estimation is shown in Fig. 2. Fig. 3 displays the delay estimation algorithm.

User specifies QoS tuple $\{D_i, L_i\}$*;*
Determine the initial amount of resources to be allocated $\{B_{iref}, \gamma_{ref}\}$*;*
Determine the upper and lower buffer occupancy threshold value;
Determine the minimum amount of resources to be allocated at any one time $\{B_{min}, \gamma_{\min}\}$*;*
Determine the appropriate interval for performing resource reallocation;

Start with an initial reference amount of resources;
Serve packets from this flow under the current allocation;

/* The integration of the estimation algorithm */

For each packet arrival:
Calculate the average delay experienced by the existing packets;
Calculate the estimated delay (δ) for the new packet arrived
(If the estimated delay (δ) > specified delay bound, L_i*)*
Drop the packet;

Total packet arrived = packet arrived + 1;

/* dynamic resource allocations */

If (total packet arrived = interval variable)
If (the allocated buffer is full and losing this arrival will cause the loss ratio performance than L_i*)*

Increase Bi to an amount so that the new occupancy is OCP_u*;*

Perform $i \leftarrow B_i/D_i$*;*

After the transmission of each packet:

Total packet transmitted = packet transmitted +1;
If the (total packet transmitted = confidence time)
Decrease B_i *by an amount but make it no lower than* B_{imin}

Perform $i \leftarrow B_i / D_i$*, when there are no packets in the buffer;*

Fig. 2: The dynamic QoS control algorithm with delay estimation

2.3 Hybrid Dynamic QoS Control

The third model proposes to achieve high resource utilisation and satisfaction of the requested QoS by integrating the mechanics of the dynamic QoS control (model 1) and the approach adopted in the second model. In this hybrid scheme (model 3), it is assumed that although packets admitted into the buffer are estimated to be served within the specified delay bound, there will still be the occurrence of buffered packets that have expired the specified delay bound. Thus, there are two mechanisms for computing the delay associated with a packet. First, before a new packet is admitted into the buffer, the estimated delay is computed. The second computation of the delay experienced by a packet is performed before the transmission of a packet to the receiver. In both cases, if the estimated delay and the actual delay experienced by a packet exceed the specified delay bound, the packet is dropped. Thus, in this model there are three reasons that cause a packet loss: (i) the arrival of a new packet is discarded due to a full buffer; (ii) the arrival of a new packet is discarded as the estimated delay that will be experienced by the packet exceeds the specified delay bound and (iii) buffered packets that have exceeded the specified delay bound are dropped. The Hybrid dynamic QoS control is elaborated in Fig. 4.

Estimate ( ) {

If (first packet)

Estimated delay (δ) *= total delay;*

Else If (no packets have departed)

$$\text{Estimated delay } (\delta) = \frac{\text{Total delay}}{\text{number of packets transmitted}}$$

else

$$\text{Estimated delay } (\delta) = \frac{\Sigma\,(\text{simulation time - packet } i \text{ arrival time})}{\text{number of packets in the buffer}}$$

}

Fig. 3: The delay estimation algorithm

3.0 SIMULATION MODEL

The performance evaluation of the three algorithms discussed in Section 2 was carried out using a discrete-event simulation [16, 17]. The simulation model developed comprises of three sub-modules: the source sub-module, the resource sub-module and the dynamic resource allocation sub-module. The source sub-module function is to generate packets for a given flow. Four traffic patterns are used for this purpose, which are: traffic pattern 1 $\{D_i=0.01s, L_i=0.1\}$, traffic pattern 2 $\{D_i=0.03s, L_i=0.01\}$, traffic pattern 3 $\{D_i=0.06s, L_i=0.001\}$ and traffic pattern 4 $\{D_i=0.1s, L_i=0.0001\}$. Based upon the generated flow specification, the resource model is used to allocate the initial amount of resources. The dynamic resource allocation model comprises of the three algorithms that were discussed above.

The performance metrics evaluated from the model are: average packet loss ratio, average packet loss ratio due to expired delay, average packet loss ratio due to buffer overflow, average allocated buffer, average delay and throughput which are used to evaluate the designs. The average delay that is calculated in the three algorithms is defined as the waiting time of a packet in the buffer until it is transmitted. The average packet loss ratio computed for each of the three algorithms is different. In the first model, the packet loss is caused by two factors: (i) a buffer that is full which subsequently discards a new packet arrival and (ii) a buffered packet that has expired the requested delay bound. The second model computes the packet loss based on the following reasons: (i) packet loss due to buffer overflow and (ii) packet loss due to the value of the estimated delay of a packet, which exceeds the specified delay bound. The third model computes the packet loss based on the following criteria: (i) packet loss due to buffer overflow; (ii) packet loss due to the value of the estimated delay of a packet, which exceeds the specified delay bound and (iii) a buffered packet that has expired the requested delay bound.

The control parameters used in the three models are: the upper and lower buffer occupancy thresholds, the resource monitoring intervals and the selection of the appropriate confidence time used for decreasing the buffer occupancy.

User specifies QoS tuple {D_i, L_i};
Determine the initial amount of resources to be allocated {B_{iref}, γ_{ref}};
Determine the upper and lower buffer occupancy threshold value;
Determine the minimum amount of resources to be allocated at any one time {B_{min}, $\gamma_{\min}$};
Determine the appropriate interval for performing resource reallocation;
Start with an initial reference amount of resources;
Serve packets from this flow under the current allocation;
/* The integration of the estimation algorithm */
For each packet arrival:
Calculate the average delay experienced by the existing packets;
Calculate the estimated delay for the new packet arrived
(If the estimated delay > specified delay bound, L_i)
Drop the packet;
Total packet arrived = packet arrived + 1;
/* dynamic resource allocations */

If (total packet arrived = interval variable)
If (the allocated buffer is full and losing this arrival will cause the loss ration performance than L_i)

Increase B_i to an amount so that the new occupancy is OCP_u;
Perform i $\leftarrow B_i/D_i$;
Before transmission of each buffered packet:

Calculate the delay experienced by the packet;

/* dropping of transit packets that have expired their delay bound */

If (the experienced delay > requested delay bound (L_i))

The packet is dropped;

After the transmission of each packet:

Total packet transmitted = packet transmitted +1;
If the (total packet transmitted = confidence time)
Decrease B_i by an amount but make it no lower than B_{imin}
Perform i $\leftarrow B_i / D_i$, when there are no packets in the buffer;

Fig. 4: The hybrid dynamic QoS control algorithm

4.0 EXPERIMENTAL RESULTS

The objective of conducting the performance analysis was to identify and distinguish the appropriate scheme in its ability to meet the desired QoS, for various types of traffic associated with multimedia. There are two dimensions in which the performances of three models are evaluated. First, is the ability of a model to sustain the requested level of QoS and second, the implication that the model has on resource utilisation in its effort to realise the requested QoS. The input parameters used in the simulation models are the requested QoS, specifying the average delay and the average packet loss ratio. The performance parameters to quantify the results are: the average delay, average packet loss ratio, the average buffer allocation and the throughput.

From the results it can be concluded that the dynamic QoS with delay estimation (model 2) and the hybrid dynamic QoS model (model 3) have significantly improved performance for the delay parameter in all the four traffic classes, as shown in Fig. 5 - Fig. 8. The implementation of the delay estimation algorithm reduced successfully the propagated delay, caused by buffered packets that are eventually dropped due to expired delay. The actual delay experienced by successfully transmitted packets to the receiver, permit the calculation of estimated delay that is likely to be experienced by new packets. Therefore, by discarding packets prior to admission into the buffer, the average delay has been reduced significantly. In addition, the buffer needed to store the packets now has a fairly small and constant size. The features of model 2 and model 3 have proven to be well suited for delay sensitive traffic such as video and audio.

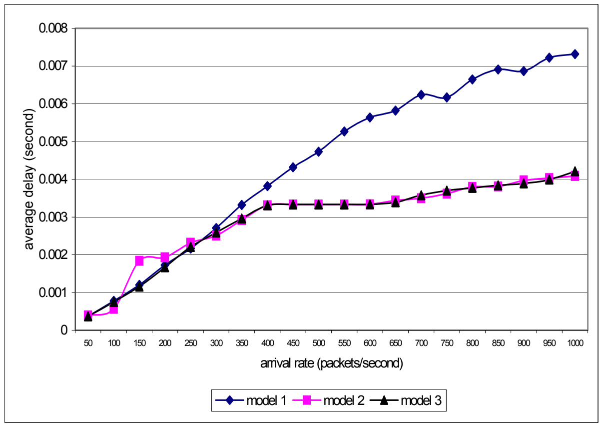

Fig. 5: Average delay of traffic pattern 1 versus arrival rate for models 1, 2 and 3

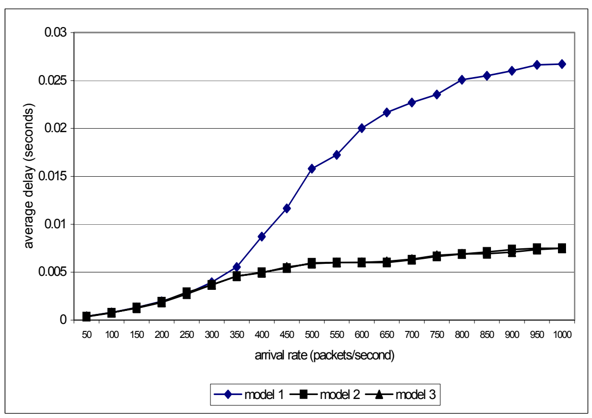

Fig. 6: Average delay of traffic pattern 2 against arrival rate for models 1, 2 and 3

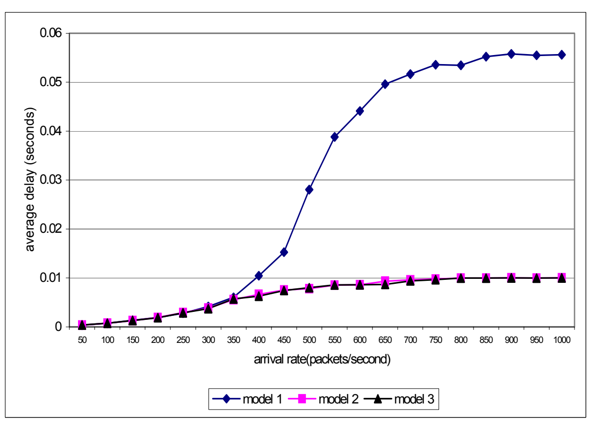

Fig. 7: Average delay for traffic pattern 3 against arrival rate for models 1, 2 and 3

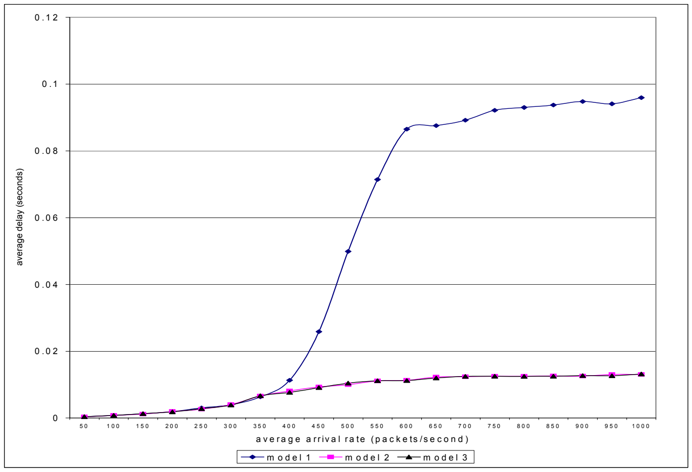

Fig. 8: Average delay for traffic pattern 4 against arrival rate for model 1, 2 and 3

The ability to have a pro-active approach for the discarding of packets due to expired delay has contributed to the success of model 2 and model 3 in improving the average delay as compared to model 1. However, the improvement on delay by discarding packets will have an impact on the average packet loss ratio. An important feature observed from the results is that, in addition to distinctly improving the average delay for delay sensitive traffic (traffic pattern 1) the average packet loss ratio has also been improved as compared to other traffic patterns, as shown in Fig. 9 – Fig. 10. This is directly caused by the effect of discarding packets that are estimated to violate the requested delay bound on the buffer occupancy. In model 1 the policy of admitting all packets into the buffer, result in packets being dropped due to buffer overflow and packets discarded due to expired delay. In models 2 and 3, for traffic pattern 1, which is highly delay sensitive, packet loss is caused only by packets discarded due to estimated delay and therefore, resulting in lower packet loss for models 2 and 3 as compared to model 1, for delay sensitive traffic. This is an attractive conclusion, suggesting that with the proposed scheme now the real-time packets will have benefit in both delay limits as well as improved packet loss, consequentially improving throughput.

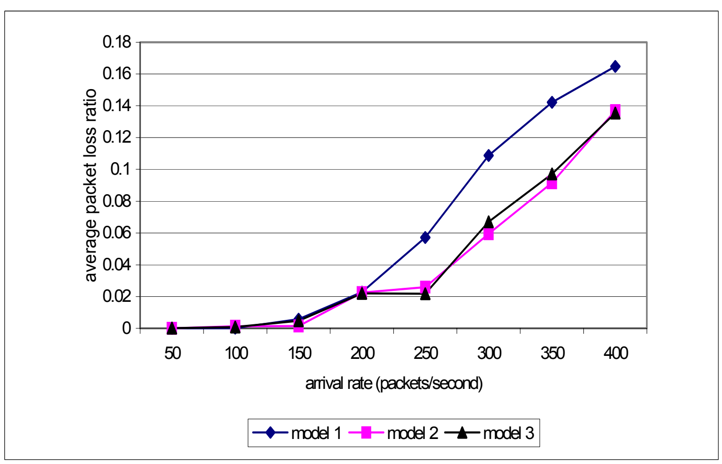

Fig. 9: Average packet loss ratio against arrival rate for traffic pattern 1

In the case of the traffic patterns 2, 3 and 4, model 1 from [15] has shown to produce better average packet loss ratio as compared to model 2 and model 3. However, the delay performance of model 2 and model 3 is still better than model 1, for all the traffic patterns, as mentioned earlier.

The resource requirements (buffer allocated) in model 2 and model 3 are significantly lower than model 1. The delay estimation algorithm in models 2 and 3 limits and justifies the admission control of packets into the buffer.

Therefore, by discarding packets prior to being admitted into the buffer, the average allocated buffer needed is smaller and remains nearly same size. In this case, the users can control the management of the buffer resource to enable high resource utilisation in achieving the requested QoS.

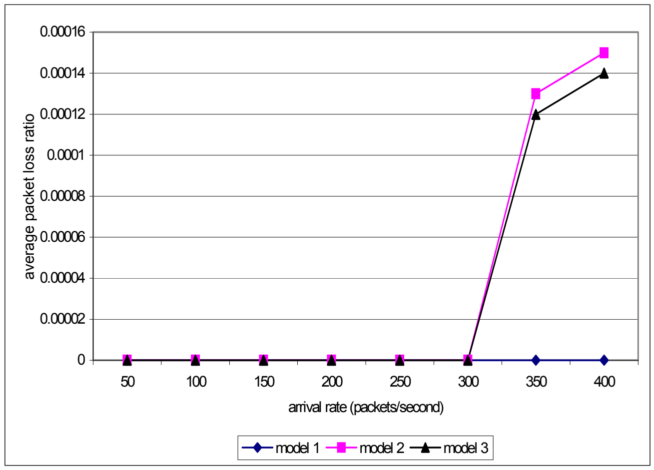

Fig. 10: Average packet loss ratio against arrival rate for traffic pattern 4

5.0 CONCLUSIONS

From the various results discussed in the previous section, it can be concluded that the proposed algorithms are able to enhance successfully the QoS needs of DMS, as well as benefit into low resource requirements. The improvement in delay performance characteristics was seen for all four traffic classes. For traffic pattern 1, which represents the delay sensitive traffic, even the improvement was observed for packet loss characteristics and throughput. This feature of the proposed algorithms will result in additional benefits of dealing with real-time traffic. The old scheme remains beneficial in handling the packet loss sensitive traffic only. Therefore, from the present studies it can be concluded that a QoS sensitive scheme can be dynamically chosen from among the newly proposed scheme and the old scheme, to take advantage under all traffic patterns.

REFERENCES

[1] H. Zhang and S. Keshav, "Comparison of Rate Based Service Disciplines". *ACM SigComm 1991,* September 1991.

[2] Y. Bao, "Efficient Resource Management for Satisfying Diversified QoS Guarantees", *in Proceeding Eleventh IEEE Workshop on Computer Communications,* Reston. VA, September 1996.

[3] G. G. Xie and S. Lam, "Delay Guarantee of Virtual Clock Server". *IEEE/ACM Transactions on Networking,* Vol. 3, No. 6, December 1995.

[4] D. D. Clark, S. Shenker and L. Zhang, "Supporting Real-Time Applications in an Integrated Service Packet Network: Architecture and Mechanism", *in Proceeding ACM SIGCOMM'92,* Baltimore, M.D., August 1992, pp. 14–26.

[5] F. A. Tobagi, I. Dalgic, *High-Speed for Multimedia Application.* W. Effelsberg et al. (eds), 1996, pp. 61-101.

[6] M.V. Harrick, "Multimedia Systems Architecture", in *Proceedings of the SPIE International Symposium on Photonics for Industrial Applications, Critical Review on Defining Global Information Infrastructure: Infrastructure, Systems and Services,* Boston, November, 1994.

[7] C. Jin, and S. Jamin, "Design, Implementation, and End-to-End Evaluation of a Measurement-Based Admission Control Algorithm for Controlled-Load Service", *Proc. of the 8th Int'l Workshop on Network and Operating Systems Support for Digital Audio and Video*, 1998.

[8] A. Demers, S. Keshav, and S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm", *in Journal of Internetworking Research and Experience*, October 1990, pp. 3-12.

[9] L. Zhang, "Virtual Clock: A New Traffic Control Algorithm for Packet-Switching Networks", *in Proceedings of ACM SIGCOMM'90*, Philadelphia, PA, September 1990, pp 19-29.

[10] S. Golestani, "A Self-Clocked Fair Queuing Scheme for Broadband Applications", *in Proceedings of IEEE Infocom'94.* Toronto, CA, June 1994, pp. 8-18.

[11] S. S. Lam and G. G. Xie, "Burst Scheduling: Architecture and Algorithm for Switching Packet Video", *in Proceedings of IEEE Infocom'95,* April 1995.

[12] Y. Bao and A. Sethi, "A Real-Time QoS Control Scheme for Satisfying Diversified Delay Bound and Loss Ratio Requirements". *Technical Report No. 96-08,* University of Delaware, October 1996.

[13] S. Jamin, S. J. Shenker and P. B. Danzig, "Comparison of Measurement-Based Admission Control Algorithms for Controlled-Load Service", *in Proceedings of IEEE INFOCOM'97,* April 1997.

[14] Y. Bao and A. Sethi, "OCP_A : An Efficient QoS Control for Real-Time Multimedia Communications", *in Proceedings of IEEE Global Communications Conference 1997,* November 1997.

[15] Y. Bao, "Quality of Service Control for Real-Time Multimedia Applications in Packet-Switched Networks". *Ph.D. Dissertation,* University of Newark, NY, May 1998.

[16] R. Jain, *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, New York, NJ, 1981.

[17] K. Watkins, *Discrete Event Simulation in C*. McGraw-Hill, England 1993.

BIOGRAPHY

Shamala Subramaniam completed her PhD from Universiti Putra Malaysia in 2002. Currently, she is a lecturer at the Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University Putra Malaysia. Her research interest includes scheduling algorithms, congestion control, real-time systems, modeling and simulation.

Mohamed Othman completed his PhD from Universiti Kebangsaan Malaysia in 1999. Currently, he is the Head of Department of the Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University Putra Malaysia. His research interest includes parallel computing, high speed network computing (cluster computing), artificial intelligence, expert system design and programming in logic (parallel).

Rozita Johari obtained her M.Sc in Computer Science from Illinois Institute of Technology, Chicago in 1987, and currently pursuing her PhD at Universiti Putra Malaysia. Her main research areas include parallel computing and operating system.

Mohd Yazid Mohd Saman obtained his PhD in Parallel Processing from Loughborough University UK in 1993. Research interests include parallel processing, simulation, computer networks and computer graphics.