

Meta-QoS Performance of Earliest-Deadline-First and Rate-Monotonic Scheduling of Smoothed Video Data in a Client-Server Environment

KELVIN YIU-LUN TSOI AND YU-KWONG KWOK

Department of Electrical and Electronic Engineering
The University of Hong Kong, Pokfulam Road, Hong Kong

Email: {yltsoi, ykwok}@eee.hku.hk

Abstract[†]

In this paper we present an extensive performance study of two modified EDF and RM scheduling algorithms which are enhanced to provide *quality of service* (QoS) guarantees for smoothed video data. With a probabilistic definition of QoS, we incorporate admission control conditions into the two algorithms. Furthermore, we also include a counter-based scheduling module as the core scheduling mechanism which adaptively adjusts the actual QoS levels assigned to requests. Our theoretical analysis of the two enhanced algorithms, called QEDF and QRM, shows that the QRM algorithm is more robust than the QEDF algorithm for different workload and utilization conditions. We also propose to use a new metric called *meta-QoS* to quantify the overall performance of a packet scheduler given a set of simultaneous requests. In our experiments, we find that the QRM algorithm can sustain a rather stable level of meta-QoS even when the workload and utilization levels are increased. On the other hand, the QEDF algorithm is found to be less desirable for a high level of utilization and a large number of requests.

1 Introduction

Simultaneous transmission of multiple high quality video streams from the symmetric multiprocessor (SMP) server [3], [4] to the clients is becoming an increasingly important class of traffic in a typical local area network (LAN) environment. However, the implementation of such a video data transmission system entails tackling a number of difficult issues. One of the most crucial issues is the support for *quality-of-service* (QoS) guarantees for real-time transport of stored or live video over the underlying high-speed network [16]. For continuous playback of video at the client, strict QoS must be provided. In a high-speed LAN environment, we usually have several clients requesting different video streams simultaneously. Thus, the server has to exercise some form of admission control policy in order to judge whether a certain set of requests can be admitted without degrading the QoS of existing transmissions [8], [15]. On the other hand, to maximize the utilization of the server, it is desirable to support as many different streams as possible. Obviously, strict QoS guarantees using tight admission control and high utilization are two conflicting goals. As video data are usually compressed using

prediction-based techniques such as MPEG-2, the rate of a video stream is usually varying [16]. Admission control, therefore, is not a trivial task because it is difficult to predict whether a newly accepted request will affect the service levels of existing requests due to rate variability, which has been shown to be quite substantial [7]. Thus, much work has been done in using buffering techniques to *smooth* the video streams—to make each video stream less bursty or even have a somewhat constant bit rate (CBR). However, even after smoothing, at the network level, to ensure the QoS of a request can be achieved, some form of packet scheduling technique has to be used [7].

Although many smoothing techniques have been proposed [7], [16], the issue of how to judiciously multiplex the smoothed video streams to the network is largely unexplored. Many researchers have mentioned that traditional real-time scheduling approaches such as earliest-deadline-first (EDF) and rate-monotonic (RM) [9], [10] can be used. Indeed, these two hard real-time scheduling techniques have been increasingly used in a soft real-time environment [6], [13]. But it is not clear which approach is more suitable in the environment considered in our study.

In order to choose a more suitable approach for implementation in our video server, we have conducted a performance study of our enhanced EDF and RM schedulers for providing QoS guarantees in the delivery of smoothed video streams. In particular, we are interested in the overall performance of the packet scheduling technique given a set of simultaneous requests. Thus, to quantify such performance, we define a metric called *meta-QoS* as an indicator of how much surplus of service a scheduler can allow. This metric is useful for practitioners to choose a suitable scheduling technique because as the number of requests increases (i.e., increasing utilization), meta-QoS indicates to what extent the scheduler can provide fair overall service guarantees to the requests of a group of users. Thus, the capacity of the video server, in terms of the number users, can be determined. Our extensive simulation study reveals that the enhanced RM scheduling technique is much more robust than the EDF approach in that the former can sustain a high level of meta-QoS and requests admission percentage even with a large number of requests and a high level of request utilization.

The remainder of this paper is organized as follows. We

[†] Kelvin Tsoi was supported by HKU EEE postgraduate studentship.

provide some related work in Section 2. In Section 3, we first discuss how we enhance the two algorithms using a probabilistic definition of QoS. The enhancements include incorporating admission control conditions and a QoS-based scheduling module. We then present a theoretical analysis of the performance of the two enhanced algorithms. Section 4 contains the results of our extensive simulation study of the two algorithms. The last section concludes the paper.

2 Related Work

Much work has been done in providing QoS guarantees using packet scheduling techniques based on the smoothing concept. Lam *et al.* [7] suggested an algorithm for transmitting MPEG encoded video streams by using buffering techniques. The algorithm is designed based on the premise that MPEG encoded video streams are composed of repeating patterns of picture types. Thus, using buffering, the pictures in the same pattern can be transmitted at approximately the same rate. However, they do not consider the issue of transmitting several video streams simultaneously. Salehi *et al.* [11] have recently addressed the problem of smoothing a given video stream given a fixed buffer size. Their technique is based on a work-ahead scheduling method in which the rate profile of a video stream is analyzed to determine a feasible rate schedule with the greatest possible reduction in rate variability.

For packet scheduling and multiplexing, Hyman *et al.* [5] have addressed the issue of determining the schedulable region in the space of workload with QoS requirements. They evaluated the MAGNET II scheduling, static priority, and minimum laxity threshold schemes. Their results indicate that the size of schedulable region is a useful metric for admission control for transmissions segregated into multiple classes. However, such concept cannot be applied in the environments considered in our study because we do not assume a classification of video requests. Georgiadis *et al.* [2] also investigated the problem of multiplexing several packet streams onto a single link. They showed that a standard non-preemptive EDF policy and a tracking policy based on preemptive EDF are delay-optimal. Their results are useful in a system with limited buffers.

Many researchers have extended the EDF and RM algorithms in various other situations [1], [10]. A more recent extension of the RM algorithm is suggested by Shih *et al.* [12] for scheduling requests with deferred deadlines. Modified EDF schedulers are also widely adopted in a rate-controlled manner in ATM packets networks.

3 Soft Real-Time Scheduling with QoS Guarantees

In this section, we first provide a definition of QoS for applying the two algorithms in the soft real-time environment considered in our research. This is followed by

a derivation of the admission control criteria used in the two algorithms.

3.1 QoS and Admission Control

In our study, we aim at studying the efficacy of the classical preemptive EDF and RM approaches [9], [10] for packet scheduling in a scalable video data server which has large enough packet buffers for zero-loss scheduling (this assumption is valid in a SMP server). As in most video data transmission systems, the loss of certain amount of data packets only lead to a degradation of the display quality at the client rather than a catastrophe. Thus, such a system is commonly referred to as a soft real-time environment [6], [13]. Specifically, we assume that during the connection setup phase of a particular video request from a client, the client machine informs the server the level of tolerance of packets loss. In response to the client's request, the server executes an admission control policy (i.e., a schedulability test) in order to determine whether or not the client's request can be accommodated without degrading the previous commitments to the existing services. If schedulability test is positive, then the server will start to schedule the requested video; otherwise, the result is sent back to the client and re-negotiation may be done. In our study, we use the following definition of QoS to quantify the client's requirement of the scheduling of a particular packet stream.

DEFINITION 1: QoS is defined as the probability that an arbitrary packet meets its transmission deadline.

Thus, QoS can be interpreted as the expected percentage of packets that meet the transmission deadlines. With the notion of QoS, we can derive analytical expressions for guiding the admission control process. In the following, we assume that there are n video requests $\{V_1, V_2, \dots, V_n\}$ and v_{ij} denotes the j th packet of video request V_i . As discussed earlier, we assume that the video streams are already smoothed so that the packet transmission time and period are constants which are denoted by C_i and P_i , respectively. Without loss of generality, we assume that $P_1 \leq P_2 \leq \dots \leq P_n$. As in most real-time scheduling environments, we assume that a packet is ready for transmission as soon as the previous packet has been transmitted. In addition, the transmission deadline of a packet is assumed to be the beginning of the next period.

For EDF scheduling, consider an arbitrary packet v_{ij} , whose ready time is $(j-1)P_i$ and deadline is jP_i . The number of packets whose deadlines are before jP_i is given by:

$$\sum_{\substack{k \\ k \neq i}} \left\lfloor \frac{jP_i}{P_k} \right\rfloor$$

and the number of packets transmitted before $(j-1)P_i$ is

given by:

$$\sum_{\substack{\forall k \\ k \neq i}} \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor$$

Thus, during the time period from $(j-1)P_i$ to jP_i , the number of packets need to be transmitted:

$$\sum_{\substack{\forall k \\ k \neq i}} \left(\left\lfloor \frac{jP_i}{P_k} \right\rfloor - \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor \right)$$

The expected number of packets that have been transmitted is therefore given by:

$$\sum_{\substack{\forall k \\ k \neq i}} \left(\left\lfloor \frac{jP_i}{P_k} \right\rfloor - \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor \right) \text{QoS}_k$$

Using the above observation, the EDF scheduler can check whether packet v_{ij} can be successfully transmitted (i.e., meeting its transmission deadline) by using the following inequality:

$$P_i - \sum_{\substack{\forall k \\ k \neq i}} \left(\left\lfloor \frac{jP_i}{P_k} \right\rfloor - \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor \right) \text{QoS}_k C_k \geq C_i$$

For RM scheduling, we can derive an admission control policy using a similar method. Again, consider an arbitrary packet v_{ij} , whose ready time is $(j-1)P_i$ and deadline is jP_i . The number of packets whose transmission ready times are before jP_i and have higher priority is given by:

$$\sum_{k < i} \left\lceil \frac{jP_i}{P_k} \right\rceil$$

and the number of packets transmitted before $(j-1)P_i$ is given by:

$$\sum_{k < i} \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor$$

Thus, the RM scheduler can check whether packet v_{ij} can be successfully transmitted (i.e., meeting its transmission deadline) by using the following inequality:

$$P_i - \sum_{k < i} \left(\left\lceil \frac{jP_i}{P_k} \right\rceil - \left\lfloor \frac{(j-1)P_i}{P_k} \right\rfloor \right) \text{QoS}_k C_k \geq C_i$$

It should be noted that the packet streams included in the summation are different for EDF and RM scheduling algorithms.

3.2 Scheduling

The admission controller of the scheduling system uses the conditions in the previous section to determine whether a set of requests can be accepted or not. Once the requests are accepted, it is the responsibility of the scheduler to actually dispatch the requests onto the communication link. During this scheduling process, each request V_i is associated with a

counter variable ζ_i which is defined as:

$$\zeta_i = \frac{1}{(1 - \text{QoS}_i)}$$

Thus, ζ_i indicates that, for every ζ_i packets of V_i , at most one packet can be discarded in order to meet the requirement. However, it should be noted that the scheduler keeps on increasing the value of ζ_i even after the requested QoS level is achieved. In essence, the higher the value of ζ_i , the greater the actual QoS level achieved for request V_i . During this initial transient scheduling phase for actual QoS assignment, if a certain request's QoS assigned decreases because some later requests are using more resources, the scheduler will choose the most recently incremented ζ_i and decrement it. This transient scheduling phase lasts only for a short time (in our simulation experiments, the transient phase is about four to five times the least common multiple of all the periods), and the counter of each request will then be fixed for subsequent dispatching of packets.

With the above mechanism for providing QoS guarantees and dynamically adjusted actual QoS levels, hereafter, we call the enhanced EDF and RM scheduling algorithms as QEDF and QRM, respectively, to indicate that they take into account the QoS of video requests in scheduling.

3.3 Analysis

Observe that for any non-negative number x , we have $x \geq \lfloor x \rfloor > x - 1$ and $x + 1 > \lceil x \rceil \geq x$. Thus, as an approximation, we can simplify the admission control inequalities of QEDF and QRM by removing the ceiling and floor operators so that we have, for QEDF:

$$P_i - \sum_{\substack{\forall k \\ k \neq i}} \left(1 + \frac{P_i}{P_k} \right) \text{QoS}_k C_k \geq C_i$$

and for QRM:

$$P_i - \sum_{k < i} \left(2 + \frac{P_i}{P_k} \right) \text{QoS}_k C_k \geq C_i$$

Let $U_i = C_i/P_i$ be the utilization of individual video stream. We can rewrite the admission condition of QEDF as:

$$U_i \leq \frac{1 - \sum_{\substack{\forall k \\ k \neq i}} U_k \text{QoS}_k}{1 + \sum_{\substack{\forall k \\ k \neq i}} \left(\frac{C_k}{C_i} \right) \text{QoS}_k}$$

and that of QRM as:

$$U_i \leq \frac{1 - \sum_{k < i} U_k \text{QoS}_k}{1 + \sum_{k < i} 2 \left(\frac{C_k}{C_i} \right) \text{QoS}_k}$$

Given the above two expressions, suppose the scheduler has to decide whether it can accept a new request V_x with utilization $U_x = C_x/P_x$. We can deduce a number of observations:

- if QoS_x , U_x , and C_x are small (implying that P_x is large), then both scheduling algorithms will tend to accept the new request;
- if U_x is high but C_x is small (implying that P_x is small also), then QEDF will be less likely to accept the new request because the sum in the denominator is large; however, QRM can still accept such a request because its priority is high (due to the small P_x) and the sum in the denominator will not be too large (note that the sum just includes those requests with higher priority);
- if both U_x and C_x are high (implying that P_x is also large), then both algorithms may or may not accept the new request, depending upon the utilization levels of the existing requests (the numerator);

Based on the above observations, it seems that QRM is more robust than QEDF.

4 Results

In this section, we present the experimental results (more detailed results can be found in [14]) of applying the EDF and RM scheduling algorithms in a simulated client-server environment, in which the server dispatches multiple smoothed video packet streams to the network for several clients in a LAN using ATM. The simulation parameters for the system model are based on the characteristics of the PearlCluster environment [3], [14]. The objective was to investigate the performance of the two methods for scheduling the outgoing link of the server under different QoS requirements. Specifically, we simulated the transmission of multiple video packet streams from a server to multiple clients using the following sets of parameters:

- number of video packet streams varied from 2 to 8 with increments of 2;
- the QoS of each stream was randomly selected from a uniform distribution with range 0.2, and the means were 0.4, 0.6, and 0.8;
- the ratio of period to transmission time (i.e., $1/\text{utilization}$) varied from 5 to 10 with increments of 1; thus, utilization varied from 0.1 to 0.2;
- the period of each stream was randomly chosen from a uniform distribution with range 20, and the means were 20, 40, 60, and 80.

Thus, there were a total of 288 ($4 \times 3 \times 6 \times 4$) test cases for each algorithm. To quantify the overall performance of the scheduling algorithms in each case, we computed the *meta-QoS* of n video packet streams as follows:

$$\text{meta-QoS} = \frac{1}{n} \sum_{i=1}^n (QoS'_i - QoS_i)$$

where QoS'_i is the actual quality of service provided by the scheduler for stream i . Here, if a request is not admitted at all, its QoS'_i is defined as zero which is also taken into account in the sum. The significance of the meta-QoS value is that it reveals the overall quality of service provided to a set of different requests and also indicates the amount of extra requests the scheduler can accommodate.

Figure 1 shows the average meta-QoS values of the two scheduling algorithms for different number of video streams with various mean requested QoS. Notice that each point on the curves is an average of 24 meta-QoS values. It can be seen that the meta-QoS performance of QRM is about the same as that of QEDF when the number of requests is small (i.e., two requests). However, for larger number of requests, QRM outperforms QEDF considerably. Indeed, we can see that QRM is quite robust in that it can sustain a certain level of meta-QoS even with increased number of requests. The effect of mean requested QoS is also apparent: the higher the requested QoS, the lower the meta-QoS. It is worth mentioning that even when the mean requested QoS is 0.8 and the number of requests is 8, the QRM algorithm still provides some “room” for accommodating a low-QoS request because the meta-QoS level is about 0.16 which is not a very low value.

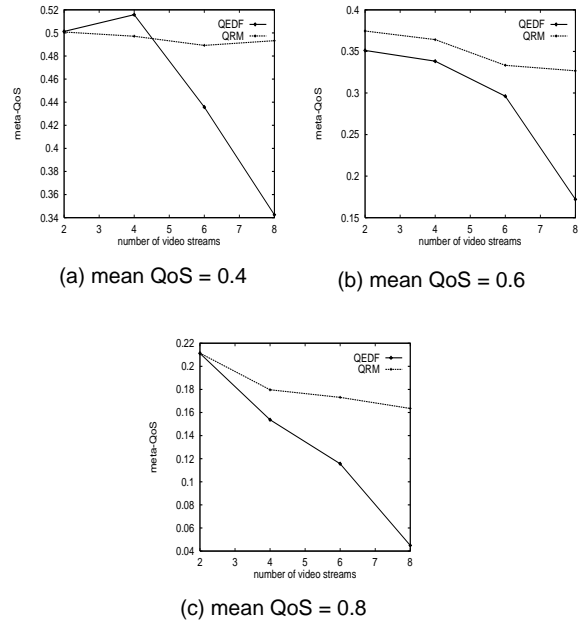


Figure 1: Average meta-QoS of the QEDF and QRM scheduling algorithms for different number of video streams with mean QoS = 0.4, 0.6, and 0.8.

Next we investigate the effect of utilization on the meta-

QoS level achieved. The results are summarized in Figure 2. Each point on the curves is an average of 16 meta-QoS values. Here, it is apparent that as utilization increases, meta-QoS level generally decreases. There are a few cases where the meta-QoS increases, though. This is due to the fact that in some cases, the randomly generated periods are somewhat skewed so that the scheduling algorithms can “pack” the requests more closely and a higher meta-QoS is achieved despite that the utilization is increased. From Figure 2, we can see that QRM is also much more robust than QEDF in that the former can provide a more consistent level of meta-QoS with increasing utilization.

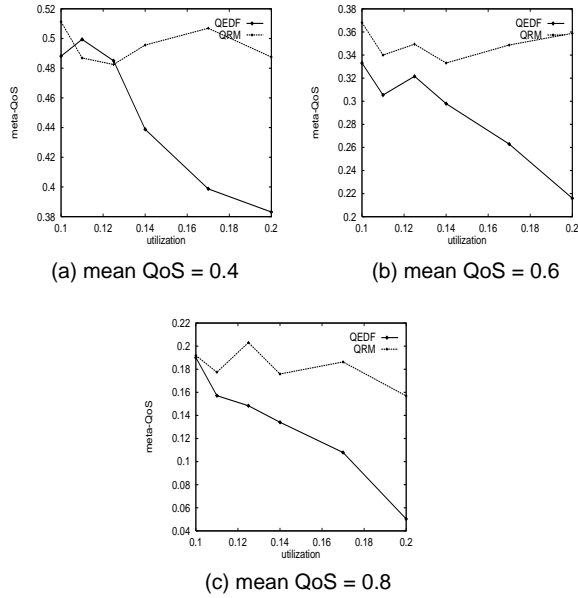


Figure 2: Average meta-QoS of the QEDF and QRM scheduling algorithms for different levels of utilization of individual video streams with mean QoS = 0.4, 0.6, and 0.8.

We are also interested in the number of requests the two scheduling algorithms admitted in each test case. Thus, we determined the percentage of request admission for each case. First, we examine the effect of number of requests on the percentage. These results are shown in Figure 3. Here, again, each point on the curves is an average of 24 admission percentages. From Figure 3(a), we can see that when the mean requested QoS level is low (i.e., 0.4), the requests are all admitted by both scheduling algorithms. When the mean requested QoS is moderate (i.e., 0.6), both algorithms cannot admit all requests when the number of request is high (i.e., 8, see Figure 3(b)). However, the QRM algorithm can admit more than 96% of the requests while QEDF can only admit about 82%. When the mean requested QoS is high, admission percentage of QRM is greater than that of QEDF by a considerable margin.

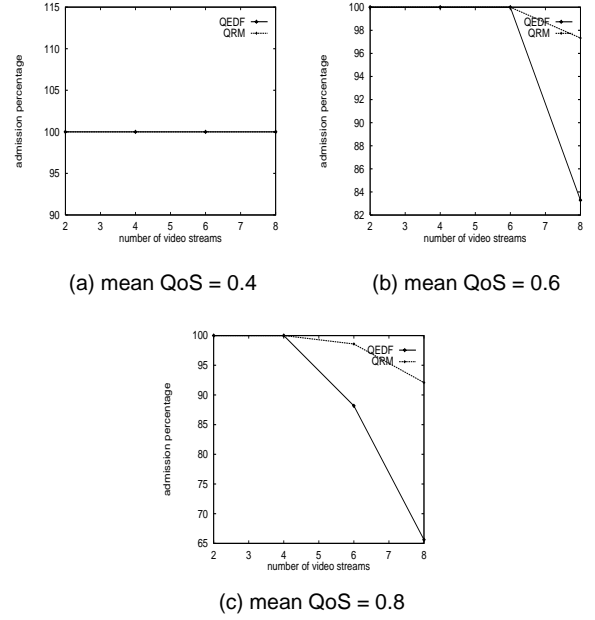


Figure 3: Average request admission percentage of the QEDF and QRM scheduling algorithms for different number of video streams with mean QoS = 0.4, 0.6, and 0.8.

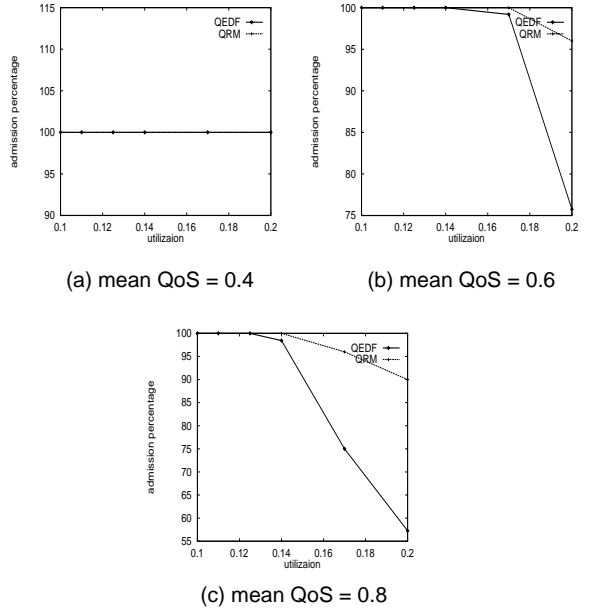


Figure 4: Average request admission percentage of the QEDF and QRM scheduling algorithms for different levels of utilization of individual video streams with mean QoS = 0.4, 0.6, and 0.8.

Figure 4 shows the results of admission percentage against the request utilization. Again, we can see that when the mean requested QoS level is low, all the requests can be accommodated. When the mean requested QoS is moderate and the average utilization of request is high, some requests

cannot be admitted. However, under such conditions, the QRM algorithm is superior to the QEDF algorithm. When the mean requested QoS is high, the QRM algorithm again outperforms the QEDF algorithm by a greater margin.

In view of the above results, it is found that the QRM algorithm is more robust than the QEDF algorithm. This observation concurs with the conclusion derived by the theoretical analysis in Section 3.3. One plausible explanation of the empirical observation is that in the QEDF algorithm, the aggressive commitment of resources to requests with earlier deadlines leads to a more conservative admission control policy. Thus, when the number of requests is high or the utilization of request is high, the QEDF algorithm tends not to admit all the requests. This in turn implies a much lower meta-QoS level as well as admission percentage. On the other hand, the QRM scheduling algorithm is less conservative in admission control. An additional advantage of the QRM scheduling algorithm not demonstrated here is its ease of implementation, which makes scheduling very efficient.

5 Conclusions

In this paper, we have presented an extensive performance study of the EDF and RM scheduling algorithms which are enhanced to provide QoS guarantees. With a probabilistic definition of QoS, admission control conditions are incorporated into the two algorithms. Furthermore, a counter-based scheduling module is included as the core scheduling mechanism which adaptively adjusts the actual QoS levels assigned to requests. Our theoretical analysis of the two enhanced algorithms, called QEDF and QRM, has already shown that the QRM algorithm is more robust than the QEDF algorithm for different workload and utilization conditions. We also propose to use a new metric called meta-QoS to quantify the overall performance of a packet scheduler given a set of simultaneous requests. In our experiments, we find that the QRM algorithm can sustain a rather stable level of meta-QoS even when the workload and utilization levels are increased. On the other hand, the QEDF algorithm, due to its conservative admission control policy, is found to be less desirable for a high level of utilization and a large number of requests. These findings concur with our theoretical analysis. In view of the lower complexity of the QRM algorithm, it seems that the QRM approach is a more suitable candidate for packet scheduling in the client-server environment considered in our study.

References

- [1] A.K. Atlas and A. Bestavros, "Statistical Rate-Monotonic Scheduling," *Proceedings of the 19th IEEE Real-Time Systems Symposium*, Dec. 1998.
- [2] L. Georgiadis, R. Guerin, and A. Parekh, "Optimal Multiplexing on a Single Link: Delay and Buffer Requirements," *IEEE Trans. Information Theory*, vol. 43, no. 5, pp. 1518-1535, Sept. 1997.
- [3] K. Hwang, H. Jin, E. Chow, C.-L. Wang, and Z. Xu, "Designing SSI Clusters with Hierarchical Checkpointing and Single I/O Space," *IEEE Concurrency*, vol. 7, no. 1, pp. 60-69, Jan.-Mar. 1999.
- [4] K. Hwang and Z. Xu, *Scalable Parallel Computing: Technology, Architecture, Programming*, McGraw Hill, 1999.
- [5] J.M. Hyman, A.A. Lazar, and G. Pacifici, "Real-Time Scheduling with Quality of Service Constraints," *IEEE J. Selected Areas in Communications*, vol. 9, no. 7, pp. 1052-1063, Sept. 1991.
- [6] B. Kao and H. Garcia-Molina, "Scheduling Soft Real-Time Jobs Over Dual Non-Real-Time Servers," *IEEE Trans. Parallel and Distributed Systems*, vol. 7, no.1, pp. 56-68, Jan. 1996.
- [7] S.S. Lam, S. Chow, and D.K.Y. Yau, "A Lossless Smoothing Algorithm for Compressed Video," *IEEE/ACM Trans. Networking*, vol. 4, no. 5, pp. 697-708, Oct. 1996.
- [8] J. Liebeherr, D.E. Wrege, and D. Ferrari, "Exact Admission Control for Networks with a Bounded Delay Service," *IEEE/ACM Trans. Networking*, vol. 4, no. 6, pp. 885-901, Dec. 1996.
- [9] C.L. Liu and J.W. Layland, "Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment," *J. ACM*, vol. 20, no. 1, pp. 46-61, Jan. 1973.
- [10] K. Ramamritham, J.A. Stankovic, and P.-F. Shieh, "Scheduling Algorithms and Operating Systems Support for Real-Time Systems," *Proc. IEEE*, vol. 82, no. 1, pp. 55-67, Jan. 1994.
- [11] J.D. Salehi, Z.-L. Zhang, J. Kurose, and D. Towsley, "Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements Through Optimal Smoothing," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 397-410, Aug. 1998.
- [12] W.K. Shih, J.W.S. Liu, and C.L. Liu, "Modified Rate-Monotonic Algorithm for Scheduling Periodic Jobs with Deferred Deadlines," *IEEE Trans. Software Engineering*, vol. 19, no. 12, pp. 1171-1179, Dec. 1993.
- [13] K.G. Shin and P. Ramanathan, "Real-Time Computing: A New Discipline of Computer Science and Engineering," *Proc. IEEE*, vol. 82, no. 1, pp. 6-24, Jan. 1994.
- [14] K.Y.-L. Tsoi and Y.-K. Kwok, "Real-time Scheduling Techniques with QoS Support and Their Applications in Packet Video Transmission," Technical Report, Department of Electrical and Electronic Engineering, The University of Hong Kong, in preparation.
- [15] D.E. Wrege, E.W. Knightly, H. Zhang, and J. Liebeherr, "Deterministic Delay Bounds for VBR Video in Packet-Switching Networks: Fundamental Limits and Practical Trade-Offs," *IEEE/ACM Trans. Networking*, vol. 4, no. 3, pp. 352-362, June 1996.
- [16] Z.-L. Zhang, J. Kurose, J.D. Salehi, and D. Towsley, "Smoothing, Statistical Multiplexing, and Call Admission Control for Stored Video," *IEEE J. Selected Areas in Comm.*, vol. 15, no. 6, pp. 1148-1166, Aug. 1997.