Admission Control for Multimedia Delivery Over Deadline-Based Networks

Yanni Ellen Liu
Department of Computer Science
University of Manitoba
Winnipeg, MB R3T 2N2, Canada
Email: yliu@cs.umanitoba.ca

Jie Wu
Department of Computer Science
University of Manitoba
Winnipeg, MB R3T 2N2, Canada
Email: jiewu@cs.umanitoba.ca

***Abstract*—Increasing demand to transmit real-time data over packet-switched networks calls for quality-of-service support from the underlying network. Deadline-based networks were developed for this purpose. In a deadline-based network, each application data unit (ADU) is associated with a delivery deadline, it is specified by the sending application and represents the time at which the ADU should be delivered at the receiver. The ADU deadline is mapped to packet deadlines, which are carried by packets and used by routers for channel scheduling; deadline-based scheduling is employed in routers. It was shown that a deadline-based network provides better support to real-time data delivery than a first-come-first-served network. We study how to effectively and efficiently deliver multimedia data in deadline-based networks, especially when at heavy load. When network load is high, congestion may occur. Multimedia data may miss their delivery deadlines due to excessive queueing delays and high packet loss ratios. This would directly affect the playback quality at the application layer. To control the level of load and improve performance, two end-system based admission control algorithms are developed. Their performance is evaluated using simulation. Both schemes are shown to improve the performance of multimedia delivery over deadline-based networks.**

I. Introduction

Multimedia delivery over packet-switched networks is becoming more and more popular. Example multimedia applications include video-on-demand, voice over IP, Internet radio, video conferencing, multi-player interactive online games, remote medical surgery, and high definition TV. Some of these applications, for example, download-and-then-display movie trailer viewing, do not have stringent requirement on network delivery. Other multimedia applications, on the other hand, may have stringent quality of service (QoS) requirements on the underlying network, and are referred to as real-time multimedia applications. Real-time multimedia applications can be classified into three types: streaming stored multimedia, streaming live multimedia, and real-time interactive multimedia [10]. Users of streaming stored multimedia applications such as video-on-demand usually can handle up to 10 seconds initial delay [10]. Streaming live multimedia applications such as live soccer game broadcast can lag tens of seconds after the first data frame arrives at the receiver. In real-time interactive multimedia applications, e.g., Internet telephony, in order to maintain interactivity, an end-to-end delay of less than 150 ms is often required [5].

In multimedia network applications, multimedia documents are encoded into frames before transmission. Different types of frames are defined to take advantage of spatial and temporal redundancy within multimedia data, so that compression is achieved yet still maintaining good multimedia quality at decoding. After encoding, multimedia frames are sent to the network for transmission. Depending on the size of the maximum transfer unit of the underlying network, each frame is subject to fragmentation at the sender and reassembly into frames at the receiver. Received frames are played back at a constant frame rate [9]. To ensure a good playback quality, the fraction of frames that miss their playback times should be kept low. The goal for the network is to deliver as many frames as possible to the receiver by their playback times.

Our work addresses effective and efficient support to multimedia delivery over deadline-based networks. Deadline-based network resource management is a novel framework that was designed to support real-time document delivery applications over packet networks [13], [14], [19]. In this framework, each application data unit (ADU), e.g., a file, or an audio or video frame, is associated with a delivery deadline; it is specified by the sending application and represents the time at which the ADU should be delivered at the receiver. The ADU deadline is mapped into packet deadlines, which are carried by packets and used by routers for channel scheduling. Deadline-based channel scheduling is employed in routers; packets with more urgent deadlines are serviced first. Such deadline-based scheduling has been shown to provide superior service to first-come-first-served in real-time data delivery [19].

Previous studies on deadline-based networks have mainly focused on real-time discrete ADUs. We focus on real-time multimedia data which is characterized by a sequence of ADUs sent at a fixed rate. We develop two application-layer admission control algorithms to address the performance degradation issue at heavy load. The admission control is at the multimedia document level. Algorithm I utilizes a network condition indicator "positive ACK rate" to make admission control decisions. Algorithm II makes use of additional information, namely bandwidth requirements of both the arriving multimedia document and current outstanding documents, when making admission control decisions. The performance of both schemes are evaluated with simulation.

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

Both are shown to prevent throughput degradation and improve multimedia delivery performance in deadline-based networks.

The rest of paper is organized as follows. Section II reviews related literature. In Section III, the performance of multimedia delivery in deadline-based networks when without admission control is studied. In Section IV, the two admission control algorithms developed are described. Performance evaluation is presented in Section V. Finally, Section VI contains a summary of this work and a discussion of future research.

II. Related Work

Existing QoS strategies for multimedia delivery over the Internet can be classified into *end-system based* and *network-layer schemes*. End-system based schemes do not have explicit support from the underlying network, thus no changes need to be made in routers. Examples are rate adaptation and Content Delivery Network (CDN). The former adapts multimedia encoding rate based either on end-to-end feedbacks received during normal data transmission (e.g., [16]) or active probing (e.g., [11]). The latter overcomes network congestion and server overload, and improves multimedia delivery performance by doing content replication and load balancing at the application layer (e.g., [18]). Rate control can also be accomplished by dropping frames selectively (e.g., [4]). Network-layer schemes include IETF IntServ/RSVP and DiffServ frameworks [20]. Both are designed to provide QoS support such as guaranteed delay or bandwidth to traffic flows or flow aggregates, and can be utilized to deliver multimedia (e.g., [21]). In contrast, we are motivated to study an alternative QoS framework, namely, deadline-based networks.

Admission control determines whether a new traffic stream can be admitted to the network for delivery without jeopardizing the QoS assurances granted to earlier accepted streams. Admission control schemes can be classified into two categories: *measurement-based* and *parameter-based*. Measurement-based admission control is based on on-line measurement either at network routers (e.g., [12], [1], [8]) or at end systems (e.g., [6]). The former may measure traffic rate or residue bandwidth at a channel; the latter relies on sending probe packets and parsing corresponding receiver reports. Parameter-based admission control for multimedia traffic uses traffic descriptors such as peak rate, average rate to achieve either deterministic or statistical QoS guarantees. Newer approaches endeavor to devise more sophisticated rate estimation schemes. E.g., in [3], a neural network approach is adopted to predict the required bandwidth dynamically. Admission control in deadline-based networks was previously studied also [15]. Two application-layer admission control algorithms are developed to alleviate load and improve performance. However, that work targeted at *discrete* real-time data only and the admission control is at the ADU level.

III. Multimedia Delivery in Deadline Networks

We first study the performance of multimedia delivery in deadline-based networks when no special resource management strategy is in place. The results from this section will serve as the benchmark for evaluating the performance of the strategies that are developed later. Discrete event simulation is used. The simulation model, experiments, and results are presented in sequence.

A. Performance Model

We simulate video transmission over a deadline-based network. At a sender, an arriving video clip is characterized by four attributes: source, destination, arrival time, and deadline. The first frame is assumed to be sent at the video clip arrival time. All subsequent frames are sent at regular time intervals. The video clip deadline specifies the time at which the first frame needs to be received by the receiver. This deadline is an application-layer one and is normally application dependent. Deadlines for subsequent frames are the video clip deadline plus their time offsets from the first frame. Denote a video clip's deadline with D, assume the time interval between two consecutive frames is t, then the ith frame's deadline is $D + (i - 1)t$. Each frame is passed from the application layer to the layers below. A frame may be segmented into multiple transport segments, each of which in turn may again incur segmentation. We assume a one-to-one mapping between transport segment and packet. The frame deadline is mapped onto packet deadlines. For simplicity, we assume that all packets of a frame carry the frame deadline. Packets are routed through the network until they reach their destination node. At the receiver, all the packets that belong to the same frame are re-assembled and sent to the application layer. For simplicity, the processing times at the sender and the receiver are not modeled. Once the frame is received by the receiving application, an ACK is generated and returned to the sender. A frame is on-time if all its packets are received on-time. If the frame is on-time, a positive ACK is returned; otherwise, a negative ACK is returned. It is possible that some packets of a frame are dropped inside the network due to buffer overflow, in this case, the frame can not be re-assembled; no ACK will be returned to the sender. Such a frame is assumed to be lost.

The network that we simulated consists of 12 nodes and 21 links (see Fig. 1). Each link consists of two channels, one per direction. At each channel, we assume that the capacity is 10 Mbit/sec; the propagation delay is 10 ms; the buffer size is 50 packets. A deadline-based scheduling algorithm, namely T/H [14], is implemented. T is time left, H is number of hops

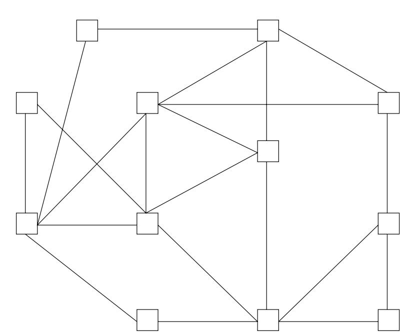

Fig. 1. Network model

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

remaining. T/H is calculated for each arriving packet. The packet with the lowest T/H value is given the highest priority. The maximum packet size at the network layer is assumed to be 1000 bytes. Fixed routing is assumed. 34 MPEG-4 video traces from [17] are used. They include drama, action, cartoon, news, and sports. Each trace contains 20 seconds' video data, encoded at 25 frames per second. Frame sizes in bytes are given. Three types of frame in each trace are I-frame, P-frame, and B-frame. The average bit rates of these traces ranges from 84.5 Kbps to 1.34 Mbps, with an average of 0.55 Mbps. Over all traces, frame size ranges from a few hundred bytes to over 15000 bytes, the average frame size is 2884 bytes. It is assumed that video clip transmission requests arrive at each sender at a rate of λ clips/sec. The video clip inter-arrival time is assumed to be exponentially distributed. At each sender, for each arriving video clip, the destination node is selected at random. Data for each video clip is selected from the 34 trace at random. Sender sends frames at a constant rate - one frame every 40 milliseconds. We define "class" to be all the traffic between a given source and destination pair. Given average arrival rate per class and fixed routing, the bottleneck channels that carry most traffic can be identified. There is one bottleneck channel in the network. 13 classes are carried on this channel.

For each arriving video clip, the delivery deadline is modeled as follows. Let $x(y)$ be the end-to-end latency to transmit a frame of size y when there is no queueing and no segmentation. Let x_p be the end-to-end propagation delay, c_j the capacity of the j-th channel along the path based on shortest-path routing. Then $x(y)$ can be estimated by $x(y) = x_p + \sum_j y/c_j$. For each video clip i, we assume that the maximum frame size y_i is known or can be estimated at the video clip arrival time. Hence, the delivery deadline for the first frame, i.e., the video clip deadline, is given by $d = arrival\ time + k \times x(y_i)$, where k is referred to as a "deadline parameter". It specifies the urgency level of a deadline. k is modeled as 1.0 plus an exponentially distributed random variable with mean ε, i.e., $k = 1 + expo(\varepsilon)$. Deadlines are at least as large as the end-to-end latency. By varying ε, a wide variety of deadline urgency can be modeled. ACKs are treated the same as data packets except that they carry slightly more urgent deadlines. The assumption is that ACKs are normally small in size, and are useful in network resource management, thus they are given slightly higher priority in channel scheduling than data packets.

The performance measure of interest is *bottleneck on-time throughput*. This is defined to be the number of frames that are delivered on-time per unit time for those video clips that pass through the bottleneck channel. The NS2 simulator [7] is used. UDP is used at the transport layer. The simple drop-tail buffer management scheme is employed. For obtaining steady-state results, the simulation run length was chosen to be 300 seconds. Each experiment was repeated six times. The sample mean and confidence intervals were calculated. Because the width of the confidence interval is very small compared to the sample mean, only sample mean results are reported.

B. Congestion In Deadline-Based Networks

We consider the case when no resource management strategy is used to transmit multimedia data in deadline-based networks. The objective is to study the effect of congestion on the on-time performance in delivering multimedia. To achieve so, we select average k of 1.2, we vary the video clip arrival rate λ from 0.25 to 1.43 clips/sec. In our network, the bottleneck channel is saturated when $\lambda \approx 0.77$ clips/sec. Thus the arrival rates selected represent a wide range of load levels to the network. The bottleneck on-time throughput results when λ is varied are shown in Fig. 2. It can be observed that as

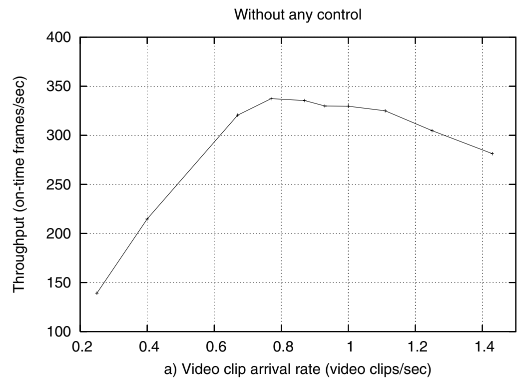

Fig. 2. Performance when without any QoS schemes

the load increases, the bottleneck on-time throughput continues to increase until λ reaches 0.77, then it starts to degrade. This is because when the system is at heavy load, the queueing delay at the routers becomes excessive, resulting in more late frames. At the same time, buffers at network channels start to become full, resulting in dropped packets. This would cause more frames not to be delivered on-time. The objective of admission control is to prevent throughput degradation in the presence of increased video clip arrival rates.

IV. Admission Control for Multimedia Delivery

Two admission control algorithms are developed in this section. Both perform admission control at the application layer of a sending host based on an end-to-end feedback called "positive ACK rate". Admission control decisions are made on a per source/destination pair basis. This is because for a network with fixed routing, packets to the same destination will traverse the same path inside the network. The ACK information returned from previous transmissions therefore reflects the congestion condition along the source/destination path, and may be used as indication for path conditions during upcoming transmissions. The "Positive ACK Rate" R is defined as the ratio between the number of positive ACKs received and the total number of frames sent. A sample of R is collected every Q seconds, referred to as an update interval. At the end of update interval i, R_his is computed as a weighted average between its old value R_his_old and R which is collected in the current update interval (see (1) below).

$$R\_his = (1 - \beta) * R\_his\_old + \beta * R, R\_his\_old = R\_his \quad (1)$$

R_his is used by both admission control algorithms. β is fixed at 0.25. The update interval Q is a tunable parameter.

A. Algorithm I - Positive ACK Rate

The first algorithm only makes use of R_his. It consists of two parts: admission test and adjustment of R_thresh. Upon a video clip arrival, the current R_his is compared with a threshold R_thresh. If $R\_his \geq R\_thresh$, the video clip is accepted; otherwise, it is rejected. R_thresh acts as a doorsill of the network; if it is higher, then less video clips can be admitted, and vice versa. R_his is refreshed periodically and then is used to update R_thresh. If $R\_his > R\_his\_old$, that means more positive ACKs are coming back, and network condition is becoming better. Thus, the "doorsill" R_thresh can be adjusted to a lower value and more video clips can be accepted. If $R\_his < R\_his\_old$, that indicates that network condition is becoming worse. In this case, R_thresh should be raised in order to stop more video clips from entering the network. The adjustment in each case is ΔD. If R_his is equal to R_his_old, R_thresh remains the same. An upper bound 0.9 and a lower bound 0.2 are set for R_thresh to keep it in a reasonable range. Through adjusting R_thresh, network load is controlled. The details of Algorithm I are shown below.

```
/* INITIALIZATION */
    R_thresh = R_thresh_init;
/* ADMISSION TEST */
    if (R_his ≥ R_thresh) accept;
    else reject;
/* ADJUSTMENT OF R_THRESH PERIODICALLY */
    if (R_his > R_his_old )
        R_thresh -= ΔD;
        if (R_thresh < 0.2) R_thresh = 0.2;
    else if (R_his < R_his_old )
        R_thresh += ΔD;
        if (R_thresh > 0.9) R_thresh = 0.9;
```

B. Algorithm II - Bandwidth Left

The second algorithm also consists of two components. The first one is again an admission test. At each sender, a real-valued variable Sum is maintained for every destination d. Sum denotes the total bandwidth of all outstanding video clips that are destined to d. Outstanding video clips are those that have been accepted and are being delivered. Upon the arrival of a video clip, its destination d is retrieved. The bandwidth requirement (denoted by X) of the arriving video clip is estimated. Define two algorithm parameters C_{min} and C_{max}. If $X \leq C_{min}$, the video clip is accepted without further consideration, and its estimated bandwidth is entered into a committed bandwidth table. The idea is that video clips with a very small bandwidth requirement are always admitted. If $X > C_{min}$, the sum of X and Sum is calculated. If it is less than or equal to C_{max}, the arriving video clip is admitted, otherwise it is rejected. C_{max} serves as an upper bound on committed bandwidth for this destination.

The second component is concerned with the update of C_{max} and Sum. Initially, C_{max} is set to a pre-specified upper bound C_{max0}. If R_his is relatively high, network condition is considered to be good. Thus, more video clips can be accepted. Otherwise, fewer video clips should be accepted. When the value of R_his changes, the parameter C_{max} is adjusted. Specifically, C_{max} is increased by ΔD if $R\_his \geq 0.9$, and decreased by ΔD if $R\_his < 0.9$. At no time can C_{max} be larger than the pre-specified upper bound C_{max0}. Also, it can never be smaller than C_{min}. The update of Sum is as follows. When a video clip is fully transmitted, the bandwidth that was used by the video clip should be made available to the upcoming video clips; thus once the last frame of a video clip is transmitted, its bandwidth is deducted from Sum. The details of Algorithm II are described below.

```
/* INITIALIZATION */
    Cmax = Cmax0;
/* ADMISSION TEST */
    if (X ≤ Cmin)
        accept; Sum += X;
    else if (Sum + X ≤ Cmax )
        accept; Sum += X;
    else reject;
/* ADJUSTMENT OF SUM */
    if (last frame of a video clip is sent)
        Sum -= X;
/* ADJUSTMENT OF CMAX PERIODICALLY */
    if (R_his ≥ 0.9 )
        Cmax += ΔD;
        if (Cmax > Cmax0) Cmax = Cmax0;
    else Cmax -= ΔD;
        if (Cmax < Cmin ) Cmax = Cmin;
```

V. Performance Evaluation

In this section, the performance of the two proposed admission control algorithms is evaluated by simulation.

A. Algorithm I

For admission control Algorithm I, the following algorithm parameters may affect its performance: R_thresh_init and ΔD. We evaluate the impact of each in sequence. In our experiments, k and λ are the same as those in Section III-B. Let ΔD = 0.01 initially. We vary R_thresh_init from 0.3 to 0.9. This covers a large range of initial threshold values. The bottleneck on-time throughput is plotted in Fig. 3. The case of no admission control is included. It can be observed that

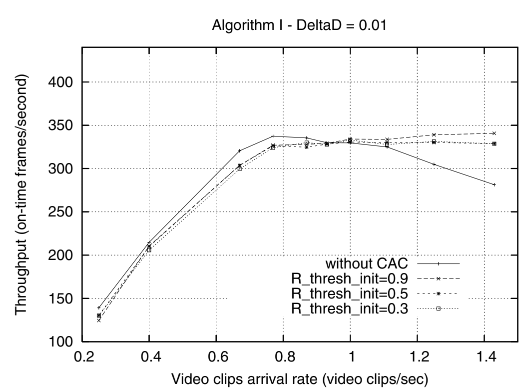

Fig. 3. Algorithm I: effect of R_thresh_init

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

(i) compared with the case of no admission control, all three values of R_thresh_init result in throughput improvement when load is heavy ($\lambda > 0.9$); at low to medium load ($\lambda \leq 0.9$), there is a small degree of performance degradation. This is a side effect for a network that employs admission control [2]. (ii) The value of R_thresh_init does not incur significant changes in performance. We found that these hold for other values of ΔD as well.

Fig. 4 plots the bottleneck on-time throughput when ΔD is

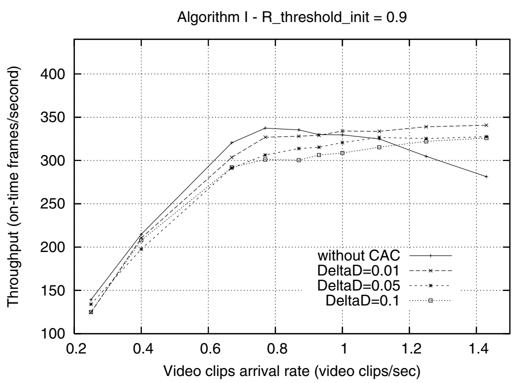

Fig. 4. Algorithm I: effect of ΔD

varied. R_thresh_init equals to 0.9. It can be observed that (i) Algorithm I achieves better performance than the case of no admission control when load is heavy ($\lambda > 0.9$), below that load level, throughput is lower than the case of no admission control. (ii) a smaller ΔD results in better performance; this is especially true when load is light or medium. Thus a low ΔD should be used in Algorithm I.

We conclude that Algorithm I can be used to prevent throughput degradation at heavy load. At medium and light load, it results in inferior performance to the case of no admission control. Between the two algorithm parameters, ΔD should be kept low while R_thresh_init has little effect on performance.

B. Algorithm II

The second admission control algorithm has three tunable parameters: C_{min}, C_{max0}, and ΔD. All are in terms of bandwidth in Mbps. From a factorial experimental design, we found that C_{min} and C_{max0} account for significant portion of variation in results and will be further studied. In contrast, ΔD had little impact thus will not be further studied.

We first vary the value of C_{min} from 0.30 to 0.45. They are lower than the average video clip bit rate which is 0.55. The values of other parameters are: C_{max0} = 2.5 and 6.0; these correspond to 1/2 and 3/5 of the channel bandwidth respectively, ΔD = 0.025. Fig. 5 plots the bottleneck on-time throughput against λ for C_{max0} = 2.5. Results for C_{max0} = 6.0 are similar. It can be observed that all values of C_{min} result in significant improvement in performance over no admission control when $\lambda > 0.8$. The degradation in performance at medium and light load ($\lambda < 0.8$) is minor. Among the four

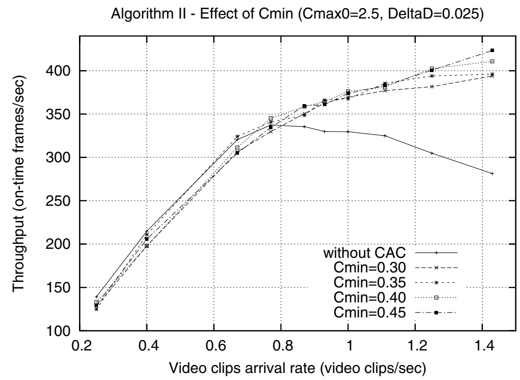

Fig. 5. Algorithm II: effect of C_{min} (C_{max0} = 2.5)

values of C_{min}, 0.45 results in the best performance. In our traces, about 1/3 have bit rate lower than or equal to 0.45.

The effect of C_{max0} is shown in Fig. 6 for C_{min} = 0.45.

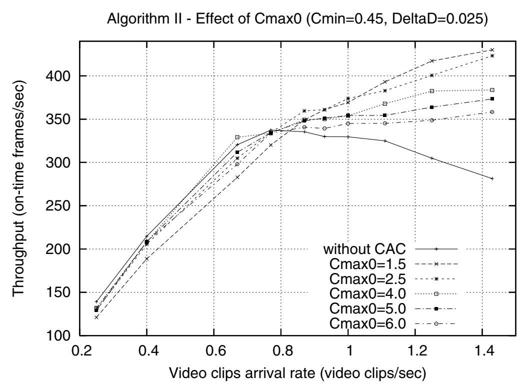

Fig. 6. Algorithm II: effect of C_{max0} (C_{min} = 0.45)

We also experimented with C_{min} = 0.3, results show similar trend. It can be observed that all experiments achieve superior performance to the case of no admission control. Throughput degradation is essentially prevented. C_{max0} has large impact on performance, a lower value of C_{max0} results in better performance when network load is heavy. This suggests that a conservative upper bound C_{max0} should be used. Considering all load levels, C_{max0} of 2.5 achieves the best balance.

We conclude that admission control using Algorithm II can effectively prevent throughput degradation and significantly improve the video clip on-time performance. Among the algorithm parameters, both C_{min} and C_{max0} have large impact on performance. The values of C_{min} and ΔD should be small compared to bottleneck channel capacity. C_{min} should also be lower than the average clip bit rate. C_{max0} can be at the same order as the bottleneck capacity, the value that is one fourth of the channel capacity resulted in the best performance.

C. Discussion and the Effect of Q

Comparing the results of both admission control algorithms, we can see that in almost all cases, Algorithm II achieves better performance than Algorithm I. This is because Algorithm II

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

utilizes more information when making admission decisions - it makes use of the bandwidth requirement of an arriving video clip as well as the bandwidth requirements of all the outstanding video clips. The advantage of Algorithm I is that it does not store the bandwidth requirement of all accepted video clips, thus is simpler to implement. In both algorithms, the key algorithm parameters such as R_thresh and C_{max} are adjusted periodically to accommodate dynamic traffic and network conditions. We discuss three other issues. First, our algorithms are end-system based, which may raise concerns on unresponsive admission control and fairness. This may be dealt with if proper pricing and charging schemes are in place. Second, all our experiments assume 100% real-time traffic. When sharing with best-effort data, a separate FIFO queue with lower priority than the single T/H priority queue can be added at each channel. This would increase channel utilization while maintaining good video performance. Thirdly, our schemes assume fixed routing. When used in an application-layer overlay network, admission control parameters should be further refined by an overlay engine before usage, taking into account load conditions along multiple paths.

Besides the tuning parameters of the two algorithms discussed, we also studied the effect of the update interval Q on performance. Q is utilized by both algorithms, it determines how often R is calculated. A smaller Q will lead to a more accurate indication of current network condition, while a larger Q incurs less overhead. To evaluate the effect of update interval Q, we select the best case of Algorithm II in our experiments. The parameters are: C_{min} = 0.45, C_{max0} = 2.5, and ΔD = 0.025. Q is varied among 1 second, 3 seconds, and 5 seconds. It was found that Q does not have great impact on performance. A smaller value of Q leads to somewhat better performance when load is heavy. When load is light, a larger Q is slightly better. Consider both, a smaller Q is recommended if it is affordable in terms of implementation.

VI. Conclusion and Future Work

In this research, how to effectively and efficiently deliver multimedia documents over deadline-based networks is studied. Of interest are heavy load conditions. Two application-layer admission control algorithms are developed and evaluated. Both algorithms are found to result in better performance than the case of no admission control when network is heavily loaded. The performance degradation at medium and light load is small. Between the two algorithms, the one that utilizes the bandwidth requirements of both arriving transmission request and previously accepted outstanding requests achieves better performance. The one that relies solely on network congestion condition feedback does not perform as well. This indicates that with bandwidth requirement information, the admission control entity can make more informed admission decisions and achieve better performance.

In the experiments to evaluate the performance of the two developed admission control algorithms, Algorithm II takes into account bandwidth requirement in making admission control decisions. In current experiments, only average bit rates are used, a possible future research is to come up with better estimates than the first moment on video clip bit rate so that the on-time performance can be maximized.

References

[1] Y. Bao and A. S. Sethi. Performance-driven adaptive admission control for multimedia applications. In *ICC '99: Proceedings of IEEE International Conference on Communications*, volume 1, pages 199–203. IEEE Press, October 1999.

[2] D. Bertsekas and R. Gallager. *Data networks*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2 edition, 1992.

[3] C. Chang, L. Lin, S. Lin, and R. Cheng. Power-spectrum-based neural-net connection admission control for multimedia networks. In *IEEE Proceedings - Communication*, volume 149 of *2*, pages 70–76. IEEE Press, April 2002.

[4] K. Chebrolu and R. R. Rao. Selective frame discard for interactive video. *Proceeding of IEEE International Conference on Communications*, 27(1):4097–4102, June 2004.

[5] J. Davidson, J. Peters, M. Bhatia, S. Kalidindi, and S. Mukherjee. *Voice over IP Fundamentals*. Cisco Press, second edition, July 2006.

[6] V. Elek, G. Karlsson, and R. Ronngren. Admission control based on end-to-end measurements. In *Proceedings of IEEE Infocom 2000*, pages 623–630, March 2000.

[7] K. Fall and K. Varadhan. *The ns Manual*, April 2002. http://www.isi.edu/nsnam/ns.

[8] S. Jagannathan, A. Tohmaz, A. Chronopoulos, and H. Cheung. Adaptive admission control of multimedia traffic in high-speed networks. In *Proceedings of the 2002 IEEE International Symposium on Intelligent Control*, pages 728–733. IEEE Press, October 2002.

[9] R. Koenen. *Overview of the MPEG-4 Standard*. Internationl Organisation for Standardisation, v.21 edition, March 2002. http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm.

[10] J. F. Kurose and K. W. Ross. *Computer Networking - A Top-Down Approach Featuring the Internet*. Addison Wesley, Second edition, 2003.

[11] T. V. Lakshman, P. P. Mishra, and K. K. Ramakrishnan. Transporting compressed video over ATM networks with explicit-rate feedback control. *IEEE/ACM Transaction of Networking*, 7(5):710–723, 1999.

[12] T. K. Lee, M. Zukerman, and R. Addie. Admission control schemes for bursty multimedia traffic. In *INFOCOM:Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 478–487. IEEE Press, April 2001.

[13] Y. E. Liu. *Deadline-based network resource management*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, 2003.

[14] Y. E. Liu and J. W. Wong. Deadline based channel scheduling. In *Proceedings of the IEEE Global Telecommunications Conference (Globecom'01)*, pages 2358–2362, San Antonio, USA, November 2001.

[15] Y. E. Liu and J. W. Wong. Admission control in deadline-based network resource management. In *Proceedings of the 23rd IEEE International Performance, Computing, and Communications Conference (IPCCC'2004)*, pages 95–102, Phoenix, Arizona, April 2004.

[16] R. Rejaie, M. Handley, and D. Estrin. Rap: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet. In *Proceeding of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies – INFOCOM '99*, volume 3, pages 1337–1345. IEEE Press, March 1999.

[17] P. Seeling, M. Reisslein, and B. Kulapala. Network performance evaluation using frame size and quality traces of single-layer and two-layer video: A tutorial. *IEEE Communications Surveys & Tutorials*, 6(3):58–78, 2004. http://trace.eas.asu.edu/tracemain.html.

[18] S. Tao and R. Guérin. Application-specific path switching: a case study for streaming video. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 136–143. ACM Press, 2004.

[19] J. W. Wong and Y. E. Liu. Deadline based network resource management. In *Proceedings of the Ninth International Conference on Computer Communications and Networks (ICCCN'00)*, pages 264–268, Las Vegas, Nevada, October 2000. IEEE Press.

[20] X. Xiao and L. M. NI. Internet QoS: A big picture. *IEEE Network*, pages 8–18, 1999.

[21] A. Ziviani, J. F. de Rezende, O. C. M. B. Duarte, and S. Fdida. Improving the delivery quality of MPEG video streams by using differentiated services. In *Proceeding of the 2nd European Conference on Multiservice Networks – ECUMN'2002*, pages 107–115, Colmar, France, April 2002.

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.