A New Approach to Support Delay-Sensitive VBR Video in Packet-Switched Networks

Hui Zhang
School of Computer Science
Carnegie Mellon University
hzhang@cs.cmu.edu

Edward W. Knightly
EECS Department, UC Berkeley
and Sandia National Laboratories
knightly@tenet.berkeley.edu

Abstract. Previous approaches to supporting video on packet-switched networks include deterministic service, statistical service, predicted service, and feedback-based schemes. These schemes represent different tradeoffs in quality of service (QOS), achievable network utilization, and method of dealing with overload. In this paper, we propose a new service that attempts to strike an efficient balance with the above tradeoffs. The approach is based on deterministic guarantees with client controlled renegotiation of QOS parameters and graceful adaptation during overload periods. We evaluate the scheme using two traces of MPEG-compressed video and show that, even with simple renegotiation polices and relatively low renegotiation frequencies, high network utilization in the range of 50% to 80% can be achieved. For traffic that is bursty over long intervals, this represents a 100% to 150% improvement in network utilization compared to deterministic service. Compared to statistical and predicted service, our approach allows more graceful and client-controlled QOS degradation during overload period.

1 Introduction and Motivation

Future integrated services networks will have to support applications with diverse traffic characteristics and performance requirements. There are three important types of traffic for future integrated services networks: constant bit rate CBR traffic, delay-sensitive variable bit rate or VBR traffic, and best-effort ABR or available bit rate traffic. Among these, delay-sensitive VBR traffic poses a unique challenge. While resource reservation schemes work best for CBR traffic, and there are many congestion control algorithms based on feedback and re-transmission for best-effort traffic, there is no consensus on which strategy should be used for VBR traffic, in particular, compressed video. This is due mainly to two conflicting design goals: good quality of service and high network utilization.

Achieving both goals with bursty traffic is fundamentally difficult. Since a bursty source may generate various amounts of data during different time periods, the aggregate amount of traffic generated by many sources sharing the same network resources also varies over time. When the amount of aggregate incoming traffic is greater than the outgoing link speed, packets have to be buffered. If the situation persists, packets will be dropped due to buffer overflows, which will in turn cause the application's quality of service (QOS) to suffer. This problem is compounded by the nature of VBR video traffic: depending on the underlying information content of the video stream, bursts of high rate can persist for time scales on the order of many seconds over the duration of an entire complex, high-motion scene. Bursts of this time scale cannot be absorbed by network buffers or smoothed at the source because of the excessive delay that this would introduce and the excessive buffer sizes that it would require.

Thus, the fundamental problem is that when bursts from many sources collide inside the network, if the rate of the aggregated traffic is greater than the link speed and the situation persists for a certain period, the QOS of some or all connections will suffer. Various solutions have been proposed to address the problem, and they represent different ways of dealing with the tradeoff between QOS and network utilization. Previous solutions can be classified according to the following four categories: deterministic service with peak-rate allocation [3], statistical service with probabilistic allocation [5, 9, 13], predicted service with observation-based admission control [2], and feedback based scheme with no resource reservation [4, 6].

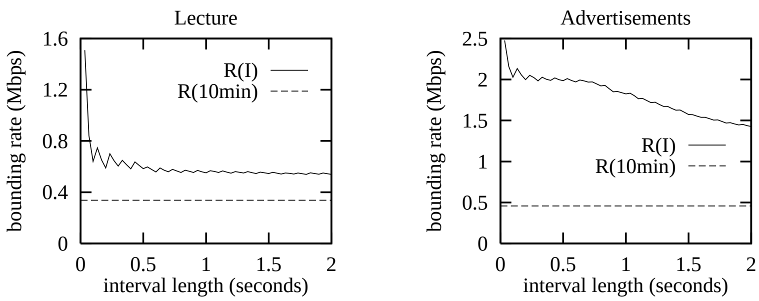
Previously proposed solutions for deterministic service *eliminate* the occurrence of overload situations by reserving resources at the sources' peak rates. While this approach provides the best QOS, it does so at the expense of having low network utilization when peak-to-average rate ratios are high. In various ways, the other three approaches trade a higher network utilization for a potential degradation of QOS. However, they all suffer from some limitations. Statistical and predicted services try to *control* the frequency of the overload situation by exploiting statistical multiplexing (respectively using knowledge of source statistics and queue measurements). However, the overload situation may still happen, and at unexpected times. Additionally, during the overload period, QOS is likely to suffer significantly for all connections in an uncontrolled and difficult-to-predict way. As well, the QOS may drop significantly due to *consecutive* packet losses. This last problem is exacerbated for VBR *video* since VBR video may have very long burst lengths, on the order of scene lengths, possibly causing a persistent degradation in service when the bursts do collide. Feedback schemes with no reservations try to *adapt* and *react* to overload situations by using network congestion signals to reduce the rates of sources. Such schemes have the advantage that they can *gracefully* degrade QOS during an overload situation by exploring an important property of the compressed video: most of video compression algorithms have a quality control parameter that, when tuned, will output compressed video at different rates and qualities. The drawback of a feedback-based scheme is that, without some round robin type of scheduler at the switch, it won't work unless *all* sources cooperate. Even with switch support, it still has the fundamental problem that it is impossible to provide different types of QOS to different applications.

In this paper, we study a new approach to support VBR video that utilizes the following two important observations. The first one is that although compressed video is bursty, it is much more structured than data traffic. While compressed video is bursty because the size of a compressed frame varies from one frame to the next, there is an underlying structure in that a new frame is generated every 33 msec. More importantly, for an MPEG source, the largest local variation between frame sizes is due to the alternation of inter-frame coded frames with intra-frame coded frames. That is, a larger (intra-frame coded) I-frame is immediately followed by a smaller (inter-frame coded) B-frame so that the micro-level burst does not persist for very long. In addition to such burstiness on a shorter time scale, there is also burstiness on a longer time scale due to scene changes [10]. The second observation is that most of the video compression algorithms have some type of quality control factor (Q-factor). By tuning this factor, a video source can tradeoff its bit rate for perceptual quality.

In order to characterize the property that VBR video has different bounding rates over different interval lengths, we use the Deterministic Bounding Interval-Dependent (D-BIND) traffic model [8] to characterize sources. With the D-BIND model and the new admission control algorithms derived recently [12], we show that, contrary to common belief, no-loss deterministic service can be provided without reservation based on peak-rate. We study the average network utilization that can be achieved for two 10-minute MPEG compressed video sequences. We observe that, for video sequences that have smaller burstiness over longer time intervals, high average network utilization can be achieved even for deterministic services.

We also observe that if there are large traffic-rate variations over longer interval lengths, deterministic service will result in *low* average network utilization. To increase network utilization in this case, we propose that the application renegotiate its traffic parameters and QOS with the network when there is a *significant* change of long term traffic rate. Such a scheme is possible for two reasons: (1) since renegotiations need to happen only when the traffic rate changes over long term, such renegotiation is not very frequent; (2) even if the renegotiation request for more resources cannot be satisfied, the application can *adjust* the Q-factor of its compression algorithm, and gracefully degrade its QOS based on the currently available resources.

While traditional reservation-less approaches do statistical multiplexing at the packet level (packets may be dropped), and traditional reservation-based service can be viewed as doing statistical multiplexing at the connection level (connection requests may be denied), our approach can be seen as doing statistical multiplexing at the segment level — resources are reserved on a per segment basis and reservation requests for a *segment* may be denied when the network is overloaded. An important feature of such an approach is that each individual application *determines for itself* the tradeoff between QOS and price-of-service by defining its own segmenting algorithm. The approach is statistical in that a renegotiation request for more network resources can be denied. However, compared to statistical service and predicted service, we avoid the uncontrollable and unpredictable packet drop behavior and the extended drop periods by introducing graceful degradation during overload situations.



(a) (b)

Fig. 1. D-BIND Characterization for Lecture and Advertisements

2 Deterministic QOS with the D-BIND Model

While the conventional wisdom has been that a deterministic service requires peak rate allocation, we will show in this section that this is not necessarily the case. A deterministic service will ensure that no packets are dropped or excessively delayed, even in the *worst case*. As alluded to in Section 1, compressed MPEG video has the important property that micro-level bursts do not persist very long. By developing a more accurate traffic model to capture such a property and utilizing this information in the connection admission control algorithm, considerably higher network utilization can be achieved compared to a peak-rate-allocation scheme, even while providing a deterministic QOS guarantee.

In [8], we propose a Deterministic Bounding Interval Dependent (D-BIND) traffic model to characterize sources. This model captures the intuitive property that over longer interval-lengths, a source's bounding rate decreases. With the D-BIND model, source j is described by the curve $R_j(I)$ where $R_j(I)$ is the bounding rate over an interval of length I. Dropping the source j subscript, if $A[t_1, t_2]$ represents the total number of bits transmitted by a source in the interval $[t_1, t_2]$, then $A[t, t+I]/I \leq R(I), \quad \forall\ t, I > 0$. Thus, the source is constrained to transmit no more than $I \cdot R(I)$ bits during any interval of length I. In practice, a traffic source must be able to specify its traffic with a small number of parameters. For this reason, the D-BIND *model* consists of N rate-interval pairs, i.e., $\{(R_n, I_n) | n = 1, 2, \cdots N\}$, with an appropriate interpolation between pairs.

Figure 1 shows the D-BIND $R(I)$ curve for two 10-minute traces of MPEG compressed video: a lecture and a series of advertisements. The horizontal axis is interval length and the vertical axis is the bounding rate over the interval length as defined above. As shown in the figure, the general trend of the curves is that the bounding rates approach the source's peak rate for small interval lengths and the long-term average rate for longer interval lengths. Of note is the difference between the two curves. For the lecture sequence of Figure 1(a), there is not a great deal of action, mostly the camera panning and zooming between the speaker and his transparencies. As a result, the $R(I)$ curve drops sharply from the peak rate to close to the long-term average rate indicating that its

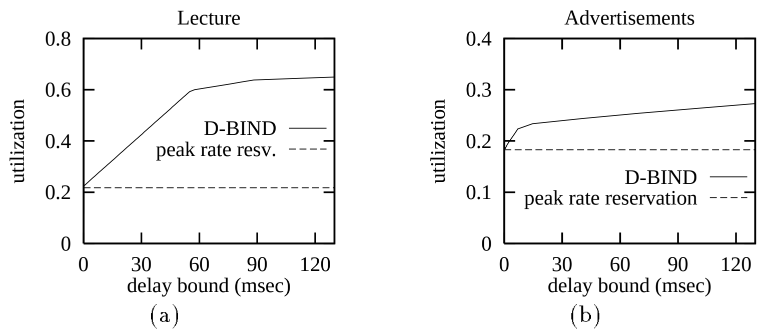


Fig. 2. Achievable Utilization for Lecture and Advertisements

high-rate bursts are of limited duration. Because of the shape of the D-BIND curve, one expects that this source will be a good candidate for achieving a reasonable utilization for deterministic service. Alternatively, because of the fast motion and many scene changes of the advertisement sequence, its $R(I)$ curve, shown in Figure 1(b), shows that the bounding rate very slowly decreases from the peak rate to the long-term average rate. This indicates that it will be difficult to achieve high utilization when multiplexing such a source since bursts of high rate and duration cannot be effectively absorbed by network buffers.

Figure 2 shows the achievable multiplexer utilization when a number of such sources are multiplexed. The figure shows deterministic delay bound on the horizontal axis and achievable utilization on the vertical axis. Points on the curve represent the maximum average utilization that can by achieved multiplexing homogeneous connections so that no packets are dropped or violate their delay bounds (details of the admission control algorithm may be found in [7, 8, 12]. As expected from the D-BIND $R(I)$ curves of Figure 1, there is a considerable difference in achievable utilization for the two streams that is caused by the inherent information content of the different videos. For example, for a delay bound of 60 msec, the multiplexed lecture sequence can achieve an average utilization of 60%. However, for the same delay bound, the advertisement sequence achieves an average utilization below 25%. Even with a more accurate traffic model, a more elaborate admission control algorithm, and a more sophisticated scheduling algorithm, the increase of network utilization for a deterministic service will be very small for the advertisement sequence. In fact, there is a fundamental limit to the utilization that can be achieved by a deterministic service, and the limit for the advertisement sequence is very close to the curve shown in Figure 2 [7].

Thus, if a source has long-duration bursts of high rate (i.e., a $R(I)$ curve that decreases slowly), it will be difficult to achieve high network utilization. Intuitively, since resource allocation for deterministic service is based on an upper bound of the source, a source's traffic specification is dominated by the worst-case segment, i.e., the segment with the highest rates over a longer interval. If the bounding rates in the worst-case segment are significantly above the long-term average rate (as for the advertisement sequence), low utilization may occur. In order to achieve

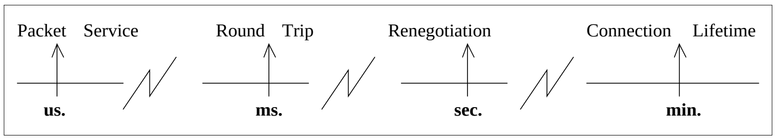


Fig. 3. Important Control Time Scales

higher utilization, some statistical multiplexing has to be introduced. In the next section, we present an approach to support VBR video that is based on deterministic guarantees with client-controlled renegotiation of QOS parameters and graceful degradation during overload situations.

3 Deterministic Service with Renegotiation and Graceful Adaptation

As discussed in Section 2, there are at least two levels of burstiness of VBR video that are important for a resource allocation algorithm: burstiness on a shorter time scale due to the coding algorithm and small-time-scale variations in picture information content, and burstiness on a longer time scale due to scene changes. Burstiness on shorter time scales is effectively taken into account with the D-BIND model and the tighter admission control algorithms. It is burstiness on longer time scales that will result in a low network utilization for deterministic service.

To increase the network utilization in this case, we propose that the application renegotiate its traffic specification and QOS with the network on a per segment basis, where the policy of choosing segments is decided by each individual application. Between each pair of negotiating points for each application, a deterministic service is provided. If a request for more resources is denied, the application will adjust the Q-factor of its compression algorithm and lower the transmission rate, which will gracefully lower the perceptual quality of the compressed video. Renegotiations are accomplished via the signaling mechanism such as the Dynamic Connection Management scheme in the Tenet Protocol Suite [11] or via an ATM signaling protocol in an ATM network.

The scheme can be better understood by considering the timescales that are important for network control as shown in Figure 3. Packet service disciplines at the switches operate at the timescale of a packet transmission time by determining which packet to service next when there is more than one packet in the queue. Connection admission control algorithms operate at the timescale of the connection life-time by deciding whether there are enough network resources to accept a new connection. While traditional resource reservation algorithms effect control at these two timescales, and feedback algorithms do control at the timescale of multiple round-trip times, our approach introduces a new control timescale that is between the round-trip time and connection life-time. It corresponds to the time scale over which the rate of compressed video changes significantly, where “significantly” is defined by the individual application.

An important feature of this approach is that each individual application

	traffic known in advance	traffic unknown in advance
not delay-sensitive	video playback	live video recording
delay-sensitive	interactive video browsing	video conferencing

Fig. 4. Classification of Video Transmission

determines for itself the tradeoff between QOS and price-of-service by defining its own segmenting algorithm. In one extreme, a video source which does not want to compromise its QOS at any time may have only one segment for the entire sequence. This is equivalent to the traditional deterministic service with no renegotiations. In the other extreme, a video source that wants to minimize reserved resources may want to renegotiate very frequently. Assuming that there is a pricing policy based on the amount of resources reserved, the first source will have the highest quality but more expensive service while the second source will have a cheaper service with the risk that it may have to degrade its QOS during periods of network overload if a renegotiation fails. If most applications are willing to pay for a more expensive service for better quality, the network may operate at a relatively low utilization. Alternatively, if most applications prefer a cheaper service but are willing to risk that they may have to gracefully degrade their QOS, the network will be able to operate at a relatively high utilization. In contrast, a deterministic service allows only the most expensive service with the best QOS.

Thus, the approach provides a *statistical* service on the level of user-defined *segments* in that it is possible that at a transition from a low-bit-rate to a high-bit-rate video segment, the renegotiation request for more network resources will fail. However, unlike traditional statistical service, this approach gives a higher level of control to individual users and avoids uncontrollable packet drop behavior and extended drop periods by using a *deterministic* service as its foundation.

Regarding the speed of renegotiation, our experience with the Tenet Real-Time Protocol Suite indicates that the latency of signaling is dominated by propagation delay [1]. Even for a wide-area network, this latency is on the order of tens of milliseconds which is equal to one or two frame times. We expect that this will be acceptable for most applications.

4 Empirical Evaluation of Scheme

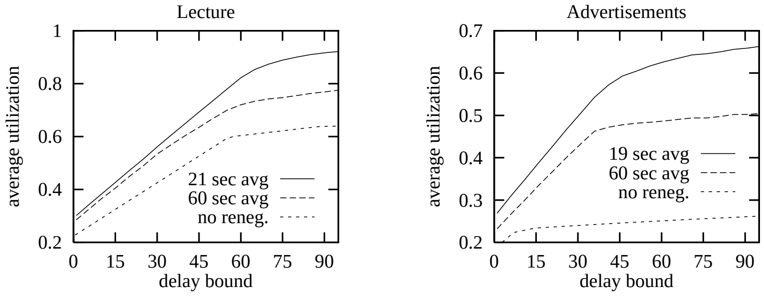
In this section, we evaluate the proposed solution by examining the following questions: (a) how much network utilization can be achieved with the new approach? (b) how often should a source renegotiate, or what is its segmenting algorithm? (c) how can a source derive its D-BIND parameters for each segment *before* a renegotiation?

To address these questions, Figure 4 shows how compressed video transmission can be classified according to whether the traffic is known in advance and whether the transmission is delay-sensitive. The degree of difficulty in solving the above problems varies according to which categories an application belongs to. For applications with traffic known in advance, problems (b) and (c) can be solved by off-line algorithms.

For this case, Section 4.2 describes a a heuristic *off-line* algorithm that illustrates the effectiveness of renegotiations in increasing network utilization. These problems become harder when the traffic is delay-sensitive and unknown in advance. Specifying traffic parameters for live video is a difficult problem that is shared by most of the existing resource allocation schemes. In Section 4.3, we present a heuristic *off-line* algorithm that adaptively chooses traffic parameters for an unknown sequence. Both of the schemes are evaluated according to the performance metric described in the next section.

4.1 Performance Metric

In order to evaluate the renegotiation schemes, we use a weighted average performance metric that is computed as follows. If a trace, T is segmented into S segments (by either an off-line or on-line algorithm), each segment j that is of duration t_j will have its own D-BIND traffic specification $spec_j$, and delay bound d_j (for the experiments presented here we keep d_j the same for all segments). The QOS guarantee is deterministic for the duration of the segment in that no packets will be delayed beyond d_j and none will be dropped due to buffer overflows. Given the link speed, scheduling algorithm (assumed to be FCFS for connections with guaranteed QOS), and buffer size, we can calculate (using the admission control equations in [7, 8]) N_j, the maximum number of connections with $spec_j$ that can be multiplexed so that all packets of all connections meet their desired QOS. A measure of the efficiency of the segmenting algorithm is a weighted average of the number of admissible connections across all of the segments, i.e., $\eta_T = \frac{\sum_{j=1}^{S} N_j \cdot t_j}{\sum_{j=1}^{S} t_j}$. Note that if a user does not want to do any renegotiations, there will be one segment ($S = 1$) so that $spec_1$ (the only traffic specification) is the worst case parameterization over the entire stream. In this case, the corresponding number of admissible connections is as shown in Figure 2. Alternatively, if an application or user is willing to renegotiate its parameterization, more connections can be accepted during periods where $spec_j$ is less bursty. The equation above can be extended to the case of heterogeneous sources by summing over multiple streams or traces T. The number of connections η_T is then converted to average utilization with knowledge of the source's long term average rate and the link speed. An alternative view of the metric above is that if a number of homogeneous streams with trace T are multiplexed at a queue with each stream having a uniformly random start time, then η_T represents the average number of connections that could be simultaneously established with no denied renegotiations. The reason for this is that at a random time τ, the length of a segment t_j corresponds to how likely it is that the randomly offset source is transmitting segment j at time τ, i.e., $Pr\{$source transmitting segment j at $\tau\} = t_j / \sum_{i=1}^{S} t_i$. Thus, at time τ the different streams will be transmitting a different segment of the trace and will therefore have a different negotiated traffic specification.



(a) (b)

Fig. 5. Achievable Utilization for Off-line Segmenting Algorithm

4.2 Off-line Algorithm

The off-line algorithm segments the video sequence assuming the entire sequence is known in advance. Such an algorithm is interesting not only because it can be used for video playback applications, but also because it can serve as a benchmark for comparing the performance of on-line algorithms. Various algorithms are possible to decide if the difference between the actual future traffic and the current traffic specification is large enough to merit a renegotiation. For the experiments presented here, we use a recursive algorithm that segments a video sequence according to a fixed QOS and a target network utilization.

The off-line algorithm may be described in the following manner. The algorithm takes as its input a sequence of frame sizes and a parameter ψ $(0 \leq \psi \leq 1)$ that indicates how aggressively to segment. A higher ψ will generate more segments and thus will potentially achieve a higher network utilization and less expensive service for the user. First, the algorithm calculates the D-BIND parameters for the entire sequence. It then identifies the worst-case segment as the segment that achieves this worst-case parameterization. This segment is then extended to the left and right until the average rate of that segment has decreased to $\psi \cdot R_N$, where R_N is the bounding rate over the longest parameterized interval length I_N. With this segment isolated, the procedure is iteratively repeated over the remaining two segments until the sequence is completely segmented.

For the two video traces, Figure 5 depicts the achievable network utilization (as defined in Section 4.1) for the off-line segmentation algorithm. For various values of ψ, the figure shows utilization vs. delay bound as in Figure 2. The lower curves depict the case of no renegotiations or $\psi = 0$ as in Figure 2. In Figure 5(a) for the lecture sequence, the upper curves represent higher values of ψ such that higher utilizations are achieved. However, this is at the expense of requiring a smaller average renegotiation interval. For example, for a 50 msec delay bound, without renegotiations, a 56% utilization is achievable. Alternatively, using renegotiations with an average renegotiation interval of 60 seconds a 67% utilization is achieved. With more frequent renegotiations averaging 21 seconds apart, a 75% utilization is achieved.

For the advertisement sequence of Figure 5(b), the improvements for using the renegotiations are even more pronounced. For example, without renegotiations and a delay bound of 50 msec, the average utilization is 24%. Alternatively, using renegotiations, average utilizations of 48% and 60% are achieved for respective average renegotiation intervals of 60 and 19 seconds. This represents respective improvements of 100% and 150% over the utilization achievable with deterministic service.

4.3 On-line Algorithm

In the off-line algorithm, we assume that the entire video sequence is known advance before transmission starts. In this section, we consider the more difficult case such as live video transmission where traffic parameters are not known in advance. We call the algorithm that dynamically computes traffic specification and issues renegotiation requests an on-line algorithm. As in the case of the off-line algorithm, many algorithms are possible for detecting scene changes or significant changes in traffic parameters. For this experiment, we again present a heuristic algorithm to obtain insight into achievable utilizations for different renegotiation intervals.

The on-line algorithm maintains the currently reserved D-BIND parameters and dynamically computes the D-BIND parameters of the previous N frames. The algorithm needs to make the following policy decisions based on the two sets of D-BIND parameters: (a) when to ask for more resources, and how much more? (b) when to ask for less resources, and how much less? In our heuristic algorithm, three parameters α, β (≥ 1) and K are used to control the policies. If any rate in the measured D-BIND curve exceeds the corresponding rate in the reserved D-BIND curve (i.e., not enough resources are reserved), a renegotiation immediately takes place. The new traffic specification is chosen so that each bounding rate R_n is α times its currently measured value. Thus, in the case of increasing reserved resources, we reserve beyond the current requirements by a factor α so that numerous consecutive increases are not required. If the measured D-BIND parameters have fallen below the currently reserved D-BIND parameters by a factor of β for K consecutive frames, the algorithm will renegotiate to a lower reserved D-BIND parameterization. In the current heuristic algorithm, the lower D-BIND parameters are computed as the average of the currently reserved and currently measured D-BIND parameters.

Figure 6 shows the performance of the on-line algorithm for the lecture and advertisement sequences. As shown, the on-line algorithm can achieve utilizations similar to that of the off-line algorithm. However, the on-line algorithm must renegotiate more frequently to achieve a utilization similar to that achieved by the off-line algorithm. This is as expected since the on-line algorithm does not have knowledge of "future" frame sizes. The figure shows that for the on-line algorithm to achieve utilizations similar to those achieved by the off-line algorithm with an average renegotiation interval of 60 seconds, the lecture sequence must renegotiate with an average interval of 23 seconds and the advertisement sequence must renegotiate with an average interval of 11 seconds.

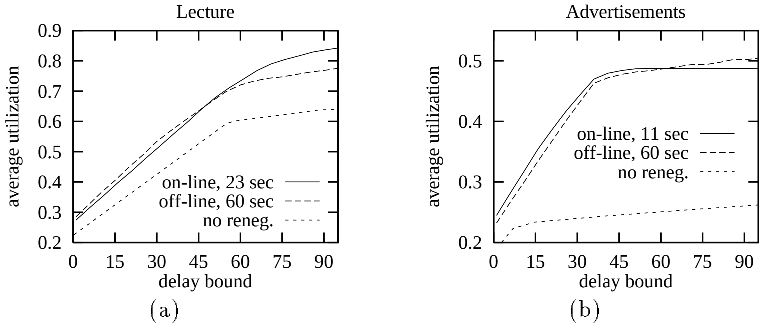


Fig. 6. Achievable Utilization for On-line Segmenting Algorithm

In the above experiments, there were no denials of requests for resources so that no sources were required to scale back their Q-factor. By allowing sources to gracefully adapt, even higher network utilization can be achieved.

5 Conclusions

We have proposed a service to VBR video based on flexible renegotiation of traffic parameters with graceful degradation of QOS in the case where renegotiations fail. Each client determines its own renegotiation policies (when to renegotiate and what parameters are used). Between two adjacent renegotiation points, a deterministic network service is provided. We have shown that such an approach can achieve significant multiplexing gains without requiring an excessive signaling overhead. For example, with the deterministic service, the network can only achieve 24% utilization when the video traffic is highly bursty over long intervals. However, using our approach with a simple off-line segmenting algorithm and relatively low renegotiation frequencies (20-60 sec/renegotiation), high network utilization in the range of 48% to 60% (50% to 80% for less bursty traffic) can be achieved for connections with delay bounds between 40 and 80 ms. This represents a 100% to 150% improvement of network utilization compared to deterministic service. As well, the on-line algorithm achieves similar improvements but requires a smaller average renegotiation interval. For example, the on-line algorithm required an 11 second average renegotiation interval to achieve utilizations close to those achieved with the off-line algorithm and a 60 second average renegotiation interval. The on-line algorithm also provides a practical solution to address the issue of specifying traffic parameters for live video. Compared to statistical and predicted service, our approach allows more graceful and client-controlled QOS degradation during overload period.

While this paper demonstrates the effectiveness of the approach, many areas remain to be explored in future works: 1) derive more elaborate on-line and off-line algorithms; 2) design algorithms to deal with rejected renegotiation requests; 3) design a mechanism to retry an increase-rate renegotiation for the case when the Q factor has been decreased by a failed renegotiation; 4) implement and experiment.

References

1. A. Banerjea, E. Knightly, F. Templin, and H. Zhang. Experiments with the Tenet real-time protocol suite on the Sequoia 2000 wide area network. In *Proceedings of the 2nd ACM International Conference on Multimedia*, San Francisco, CA, October 1994.
2. D. Clark, S. Shenker, and L. Zhang. Supporting real-time applications in an integrated services packet network: Architecture and mechanism. In *Proceedings of ACM SIGCOMM'92*, pages 14–26, Baltimore, Maryland, August 1992.
3. D. Ferrari and D. Verma. A scheme for real-time channel establishment in wide-area networks. *IEEE Journal on Selected Areas in Communications*, 8(3):368–379, April 1990.
4. M. Gilge and R. Gusella. Motion video coding for packet switching networks – an integrated approach. In *Proceedings of SPIE Visual Communications and Image Processing '91*, pages 592–603, Boston, MA, November 1991.
5. R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE Journal on Selected Areas in Communications*, 9(7):968–981, September 1991.
6. H. Kanakia, P. Mishra, and A. Reibman. An adaptive congestion control scheme for real-time packet video transport. In *Proceedings of ACM SIGCOMM'94*, pages 20–31, San Francisco, CA, September 1993.
7. E. Knightly, D. Wrege, J. Liebeherr, and H. Zhang. Fundamental limits and tradeoffs for providing deterministic guarantees to VBR video traffic. In *Proceedings of ACM SIGMETRICS'95*, Ottowa, Ontario, May 1995.
8. E. Knightly and H. Zhang. Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models. In *Proceedings of IEEE INFOCOM'95*, Boston, MA, April 1995.
9. J. Kurose. On computing per-session performance bounds in high-speed multi-hop computer networks. In *Proceedings of ACM SIGMETRICS'92*, pages 128–139, Newport, Rhode Island, June 1992.
10. A. Lazar, G. Pacifici, and D. Pendarakis. Modeling video sources for real-time scheduling. In *Proceedings of IEEE GLOBECOM'93*, pages 835–839, Houston, TX, November 1993.
11. C. Parris, H. Zhang, and D. Ferrari. Dynamic management of guaranteed performance multimedia connections. *Multimedia Systems Journal*, 1:267–283, 1994.
12. H. Zhang and D. Ferrari. Improving utilization for deterministic service in multimedia communication. In *Proceedings of 1994 International Conference on Multimedia Computing and Systems*, pages 295–304, Boston, MA, May 1994.
13. H. Zhang and E. Knightly. Providing end-to-end statistical performance guarantees with bounding interval dependent stochastic models. In *Proceedings of ACM SIGMETRICS'94*, pages 211–220, Nashville, TN, May 1994.