Traffic Models and Admission Control for Variable Bit Rate Continuous Media Transmission with Deterministic Service

Sambit Sahu, Victor Firoiu, Don Towsley, Jim Kurose
{sahu, vfiroiu, towsley, kurose }@cs.umass.edu
Department of Computer Science
University of Massachusetts
Amherst, MA 01003

Extended Abstract

1 Introduction

Future high-speed computer networks are projected to carry a dizzying array of stored continuous media (CM) flows emanating from sources such as digital libraries and video servers. These flows are often variable bit rate in nature and are characterized by tight delay requirements. In order to guarantee such delay requirements, the network must reserve resources at the links of the path for the given CM flow. The development of an efficient algorithm for deciding whether a new flow can be supported while guaranteeing delay requirements for it and all existing flows is a challenging problem which has been the subject of considerable research [3, 6, 5, 11, 12, 13].

Much of the research in recent years has focussed on resource reservation and call admission for the case when hard guarantees are required. In this case much of the attention has focussed on the development of algorithms that take *time-invariant* descriptions of CM flows as inputs [5, 12, 13]. A CM flow description is said to be *time-invariant* if the description over an interval of any arbitrary length does not depend on the starting point of the interval. The focus of much of these works has been on the development of parsimonious time invariant flow descriptions that give rise to efficient call admission algorithms that provide high resource utilization. Unfortunately these algorithms are unable to generate high resource utilization when offered variable bit rate flows with stringent delay requirements (100- 500ms).

In this paper we take a different approach, namely, we address the problem of developing call admission algorithms that use more general descriptions of stored CM flows which are not necessarily time-invariant. We begin by demonstrating that the use of a complete description of a stored CM flow (e.g., frame sizes in the case of video) can result in a substantial increase in the number of supported flows over what is possible using time-invariant flow descriptions. Since the computational requirements of the algorithm are proportional to the length of the CM flow (size of the video), we present an algorithm for producing a parsimonious flow description which still results in efficient call admission. We show that using this simple algorithm, the number of supported flows is substantially larger than the number of flows that can be supported using time-invariant flow descriptions.

In addition to the contributions described above, we also present the admissibility conditions for flows with more general descriptions that need not be time-invariant, where packets are scheduled according to the earliest-deadline-first (EDF) scheduling policy. This generalizes earlier results in [6] to accommodate a larger set of descriptions. Furthermore, we present an algorithm for admissibility checks whose computational complexity is linear in the number of flows.

The rest of the abstract is organized as follows: In Section 2, we briefly describe the admission control framework and show that using time-invariant workload descriptions grossly underutilizes the network capacity. In Section 3, we develop efficient admission control for CM flow descriptions that are piecewise linear, but are not necessarily time-invariant. In Section 4, we propose an *average-rate interpolation* workload model that provides a more accurate representation of the workload than that provided by any time-invariant descriptions. In Section 5, we evaluate the performance of our flow description model and compare its performance to the time-invariant description. Section 6 briefly discusses the potential benefits of our work in the context of new service models being proposed for the Internet, namely, differentiated services. Section 7 presents conclusions and discusses future extensions of our work.

2 Admission Control Framework

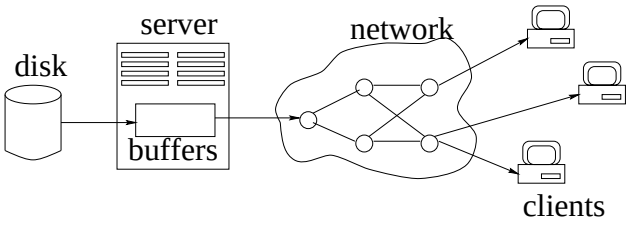

Figure 1: Illustration of a Continuous Media Application

Figure 1 illustrates a typical stored continuous media application. The continuous media is stored at the server in a disk subsystem. A client sends a request for the transmission of continuous media via the attached network. The server then retrieves the continuous media from the disk subsystem for timely transmission over the network. In order to meet the end-to-end service guarantees, it is necessary that sufficient resources are available at all the links on the path between the server and the client. This can be achieved by admission control at each link which admits only those requests for transmission for which enough resources are available. Although the focus of this study is on network admission control, the proposed solutions are sufficiently general to be applicable to the reservation of server resources as well. For example, the solutions can be adapted for admission control in a disk subsystem where the disk bandwidth is shared by multiple requests for timely retrieval of CM flows onto the server buffer.

2.1 Network Model and Admission Control Framework

We model a network link as a server with a rate c where packets are scheduled for transmission according to a Rate-Controlled EDF policy [14]. We are interested in the maximum end-to-end packet delay as the quality of service guarantee. We assume that the maximum allowable end-to-end packet delay has been decomposed apriori into maximum per-link delays for the link on the end-to-end path. Thus end-to-end admission control reduces to EDF schedulability verification at each link, i.e., it is verified that the flow can be supported with the allocated per-link maximum packet delay. The ability to do link-by-link admission control is due to the rate-controlled EDF scheduler at each link that has an independent contribution to the end-to-end delay guarantee of a flow. We do not address how the maximum end-to-end delay is divided into per-link maximum packet delays. For a detailed discussion on this reader may refer to [2]. With the above assumptions, it suffices to study admission control issues at a single link.

Let $A_i(t,s)$ denote the workload offered by a CM flow i during an interval $(t,s), \forall t, s \geq 0, t < s$ and d_i the required maximum packet delay. Using appropriate EDF schedulability conditions, the admission control process verifies if there are sufficient resources available to service this flow while meeting the maximum packet delay d_i. Assuming a fluid model where packet size is of negligible length, we derive the necessary and sufficient conditions for EDF schedulability which are stated as:

Theorem 1 *A set of N flows that are characterized by any general workload function $A_i(0,t)$ and maximum packet delay d_i, for flow $i = 1, \ldots, N$, are EDF-schedulable if and only if:*

$$\sum_{i=1}^{N} A_i(u, v - d_i) \leq c(v-u), \quad \forall u, v \in R^+, \quad u < v \tag{1}$$

If there are sufficient resources on each link along the end-to-end path, the flow is admitted for transmission. The admission control process then updates the information regarding the availability of resources to reflect the reservation of resources for the new flow.

Often, however, it is not practical to use $A_i(t,s)$ as the workload function for admission control because for compressed video, $A_i(t,s)$ would refer to a frame-level description of the workload function which results in high computational overhead for admissibility check [1]. Instead the admission control process often uses an approximation to the actual workload $A_i(t,s)$ that is easy to police and permits computationally efficient admission control. Let $A_i^r(t,s)$ denote the approximation to $A_i(t,s)$ that admission control process uses as the estimate of the required workload by a flow i. Henceforth we refer to $A_i^r(t,s)$ as the workload requirement function. Note that how well the link is utilized depends on how close $A_i^r(t,s)$ is to $A_i(t,s), \forall t, s \geq 0, t < s$.

[1] This follows from the result shown later in Section 3 that the computational complexity of admissibility check depends on the number of segments that are required to describe a workload function.

Observation 1 *For deterministic guarantees,* $A_i^r(t,s) \geq A_i(t,s), \forall t, s \geq 0, t < s$.

This is true because for deterministic guarantee, the workload requirement function $A_i^r(t,s)$ has to be at least the actual workload $A_i(t,s)$. If $A_i^r(t,s) = A_i(t,s)$, then the link utilization is maximized because if a flow i is not schedulable with $A_i(t,s)$, it is not schedulable with any $A_i^r(t,s)$ that can assure deterministic guarantees. Hence the problem is to choose $A_i^r(t,s)$ to accurately model the exact workload $A_i(t,s)$, while being easy to police and permitting computationally efficient admission control.

We examine the workload requirement functions that have been proposed in recent works [11, 5, 12] for compressed video admission control. We need the following definition:

Definition 1 (Time-Invariant) *A workload function* A_i^r *is said to be time-invariant if* $A_i^r(t, t+\tau) = A_i^r(s, s+\tau), \forall t, s, \tau > 0$.

All of the previous results [11, 5, 12] have used time-invariant workload functions as a measure of the required workload for performing admission control. Assuming a fluid model where packet size is of negligible length, the EDF schedulability conditions [6] for time-invariant workload functions are stated as:

Theorem 2 *A set of* N *flows with* $A_i^r(0,t)$ *as the time-invariant workload function and* d_i *maximum packet delay, for flow* $i = 1, \ldots, N$*, are EDF-schedulable if and only if:*

$$ct \geq \sum_{i=1}^{N} A_i^r(0, t - d_i), \quad \forall t \geq 0 \tag{2}$$

Note that our result in Theorem 1 generalizes Theorem 2 as Theorem 1 provides schedulability conditions for any general workload function that need not be time-invariant. Let us examine how well a time-invariant function estimates the exact workload for applications like stored-video when detailed frame-level information is available apriori. Note that the lowest upper-bound on the exact workload that a time-invariant function provides is limited by the *empirical envelope* $\mathcal{E}(\tau)$ which is given by $\mathcal{E}(\tau) = \max_{t \geq 0}\{A(t, t+\tau)\}$. Hence the following result:

Lemma 1 *Let* $\mathcal{S}$ *denote the set of all time-invariant functions* f_i *such that* $A_i(t,s) \leq f_i(t,s)$*, for any* $t, s, s > t$*. Then a flow* i *is admissible using* $\mathcal{E}_i(\tau)$ *whenever it is admissible using any function* $f_i \in \mathcal{S}$ *with any scheduling policy that is used to schedule the flow.*

The above result suggests that, if a time-invariant workload function is to be used, then the best possible server utilization occurs when the empirical envelope is used as the workload requirement function. But it is not easy to police such a workload requirement function since the empirical envelope requires many points to represent it. In order to alleviate this problem, there have been numerous efforts at computing approximations to the empirical envelope that are easy to police. In [5], Knightly et al. determine a piecewise linear bound to the envelope. Their result shows that eleven to twelve linear segments are sufficient to closely approximate an empirical envelope which otherwise would include 32768 segments for a 25 minute long

video. In [11], Liebeherr et al. propose a heuristic that even further reduces this to three or four linear segments. In Figure 2 we have plotted the actual workload function $A(t,s)$, i.e., the sum of the frame sizes in the interval (t,s) , the empirical envelope and the approximated time-invariant function that is derived by Liebeherr's scheme [11] for MPEG encoded Jurassic Park movie for the first two seconds of the video. We observe that any time-invariant workload function grossly overestimates the actual workload for compressed video. Figure 3 shows the same result for a different time scale.

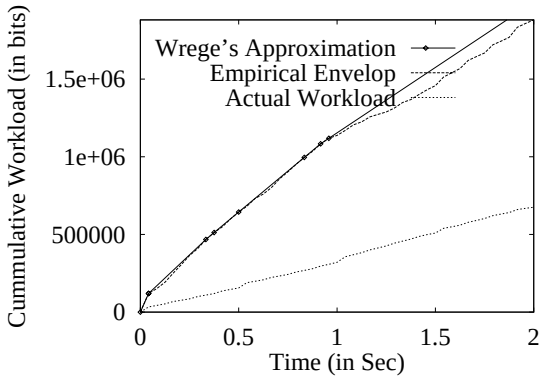

Figure 2: Comparison of Different Time-Invariant Workload Functions with Actual Workload for Jurassic Park Video Stream

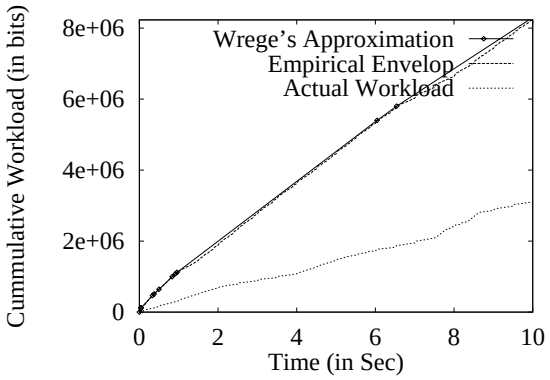

Figure 3: Comparison of Different Time-Invariant Workload Functions with Actual Workload for Jurassic Park Video Stream

2.2 Our Approach: More General Workload Functions

Instead of limiting our investigation to the set of time-invariant functions, we examine a larger set of workload functions which do not necessarily meet the time-invariant property. Without describing the details of the schedulability conditions until later, we illustrate the potential benefits of our approach through a simple example. We find the number of flows that can be supported as a function of maximum packet delay using the actual workload function $A_i(t,s)$ and the empirical envelope $\mathcal{E}(t)$. We use a 25 minute long MPEG encoded video traces of Jurassic Park, MTV, and Star Wars movie [8] for deriving the workload in our simulation. We consider both homogeneous and heterogeneous workloads. In the homogeneous case, the same video is used for each flow, but playback is started at any frame with equal probability. The heterogeneous workload consists of a set of flows, each of which is equally likely to be one of the 3 video types listed above. We determine the maximum number of flows that can be admitted for both workloads. Figure 4 shows the number of flows that can be admitted over an OC-3 link as a function of maximum packet delay for homogeneous workload generated using the Jurassic Park movie. We observe that the channel utilization can be increased by as much as 300%, by using the exact workload function, over what is possible using the best possible time-invariant workload function. We also show the maximum number of flows that can be admitted if the flows were constant bit rate (CBR) and could be admitted based on their long term average rates. The motivation for using this workload function is that, if the videos were of infinite duration, the average rate based admission control would give the absolute upper bound on resource utilization. We ob-

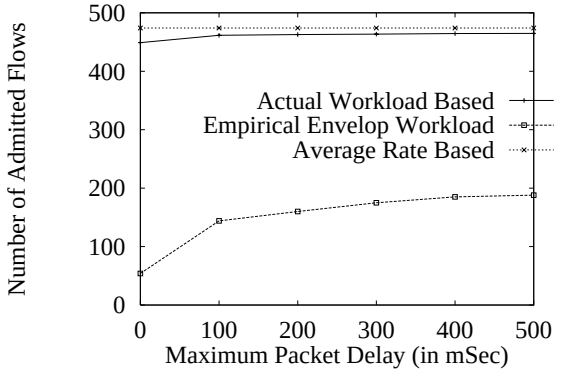

Figure 4: Benefits of Time Dependent Traffic Characterization: Homogeneous Workload

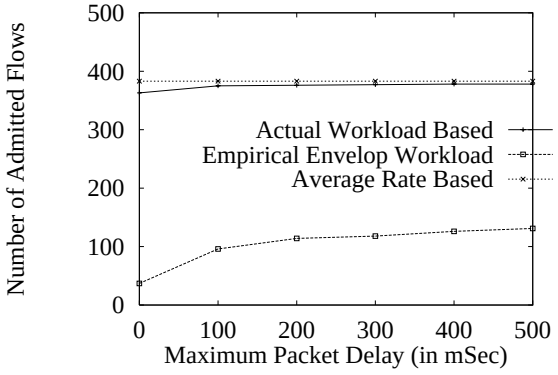

Figure 5: Benefits of Time Dependent Traffic Characterization: Heterogeneous Workload

serve in Figure 4 that, by using the exact workload function, it is possible to achieve almost full utilization of the channel. Figure 5 shows similar results for heterogeneous workloads. We have performed detailed simulations of other kind of compressed video such as MJPEG and have observed similar trends.

The results from the above experiments indicate that: 1) time-invariant functions grossly overestimate the exact workload function, 2) by using the exact workload information, it is possible to increase the channel utilization. The new challenges that we need to address are: 1) the need for computationally efficient admissibility tests and 2) the need for parsimonious workload functions that use very few segments and yet are able to achieve the benefits similar to exact workload.

3 Efficient Admissibility Conditions

In this section, based on the EDF schedulability conditions in Theorem 1 we develop efficient admissibility conditions that can be used to perform admission control for the set of all piecewise linear workload functions. We need the following two definitions.

Definition 2 (Relative Work Availability Function) *The relative work availability function $R(u,v), \forall u < v$, is a measure of resource availability in the interval (u,v) and is defined as:*

$$R(u,v) = c(v-u) - \sum_{i \in N} A_i^r(u, v-d_i) \tag{3}$$

Definition 3 (Absolute Work Availability Function) *The absolute work availability function $F(t)$ is defined as the difference between the total resource capacity and the total minimum workload that is required to be served for any $t > 0$,*

$$F(t) = ct - \sum_{i \in N} A_i^r(0, t-d_i) \tag{4}$$

Using the above definitions, the schedulability condition in Theorem 1 can be expressed as $F(v) - F(u) + H(u) \geq 0, \quad \forall u, v \in R^+, u < v$ where $H(u) = \sum_{i \in N} A_i^r(t - d_i, t)$. In order for a set of N flows to be admissible, the above condition has to be satisfied for every $u, v \in R^+, u < v$.

Before proceeding with EDF-schedulability conditions for piecewise linear workload functions, we introduce the following definition:

Definition 4 (Piecewise Linear Function) *A function $f : R^+ \rightarrow R^+$ is said to be piecewise linear if there exists a set of points $\mathcal{L} = \{t_1 < t_2 < \ldots < t_n\}, t_i \in R^+$ and $0 < n < \infty$ such that $f(t)$ is linear over $(-\infty, t_1], [t_1, t_2], \ldots, [t_n, \infty)$. If $\frac{f(t_{i+1}) - f(t_i)}{t_{i+1} - t_i} \neq \frac{f(t_i) - f(t_{i-1})}{t_i - t_{i-1}}$, then the set of points in $\mathcal{L}$ are denoted as flexion points.*

Note that the slope of a piecewise linear function f is a constant between any two consecutive flexion points $l_i, l_{i+1} \in \mathcal{L}$. The slope of f changes exactly at every flexion point. A $k-$piecewise linear function f thus represented by these constant slopes is said to consist of k linear segments where the i^{th} linear segment is represented by $\{(l_{i-1}, f(l_{i-1})), (l_i, f(l_i))\}$.

Now we develop computationally efficient admissibility conditions for piecewise linear workload functions. Suppose the workload function $A^r(0, t)$ is a $k-$piecewise linear function. Let $\mathcal{L}_F$ and $\mathcal{L}_H$ denote the set of flexion points in $F(t)$ and $H(t)$ respectively. It can be shown that for a set of N flows, the number of flexion points in $\mathcal{L}_F$ and $\mathcal{L}_H$ is upper-bounded by kN and $2kN$ respectively. Using the above property, the schedulability condition for a piecewise linear workload function can be stated as:

Proposition 1 *The schedulability condition for piecewise linear workload functions is given by:*

$$D(u) - F(u) + H(u) \geq 0, \forall u \in \mathcal{L}_H \tag{5}$$

where $D(u) = \min_{v \in \mathcal{L}_F, v > u} F(v)$

The above schedulability condition has to be verified at each point in $\mathcal{L}_H$. As $|\mathcal{L}_H| \leq 2kN$, the number of checks needed is bounded by $2kN$. It can be shown that with appropriate data structures, the total computation needed is of $O(kN)$. Note that the computational complexity of admissibility check is similar to the one for "time-invariant" functions that is reported in [3]. The details of the algorithms and the complexity analysis for performing the admissibility check are found in [9].

4 Time-Dependent Workload Characterization

Note that the computational complexity of the admissibility condition in Proposition 1 depends on the number of linear segments that are needed to represent the workload function. The success of our admission control algorithm depends on reducing the number of linear segments needed while accurately representing the exact workload function. The results in Figure 4 indicate that as much as 300% improvement in server utilization is possible by using exact workload function instead of the "time-invariant" empirical envelope. But the exact workload function contains as many as 32768 segments for a 25 minute long movie. The focus

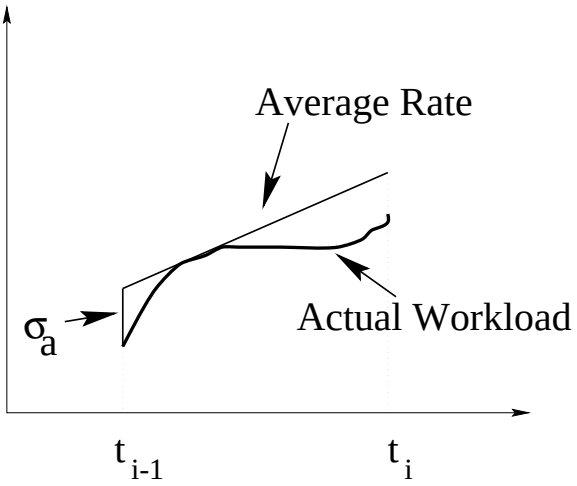

Figure 6: Illustration of Average-Rate Interpolation in (t_{i-1}, t_i)

of this section is on constructing accurate workload functions but with a smaller number of appropriately chosen linear segments.

Given a flow characterized by workload function $A(t,s), t \leq s, t, s = 1, 2, \ldots, n$, we want to construct a workload function $A^r(t_i, t_j), t_i \leq t_j, i, j = 1, 2, \ldots, m$ where $m < n$. There are two issues that need to be addressed for constructing $A^r()$:

- Choosing a set of m points $\mathcal{L}^{'} = \{0 < t_1 \leq t_2 \leq \ldots \leq t_m = n\}$.
- Choosing an interpolating function that represents the workload during the interval (t_{i-1}, t_i), for $0 < i \leq m$. The interpolating function which can be computed apriori should provide a tight upper bound to the exact workload. It should also be easily policeable.

We experimented with several interpolating functions [9]. We report here on the most promising one, the *average-rate* interpolation. The basic idea of the average-rate interpolation is the following: we determine the minimum offset $\sigma_a(l)$ that is needed for the linear segment $(k, A(0,k) + \sigma_a(l)), (k+l, A(0,k+l) + \sigma_a(l))$ over an interval of length l to be an upper bound of the exact workload for all $0 \leq k \leq n-l$. The minimum offset $\sigma_a(l)$ for an interval of length l is given as follows:

$$\sigma_a(l) = \max_{0 \leq t \leq n-l} \left\{ \max_{0 \leq j \leq l} \left\{ A(0, t+j) - \left\{ A(0,t) + j \frac{A(0,t+l) - A(0,t)}{l} \right\} \right\} \right\} \tag{6}$$

Note that $\sigma_a(l)$ is a time-invariant function of interval length l and hence can be calculated off-line. We determine the average rate ρ_i from the exact workload function, where $\rho_i = \frac{A(0,t_i) - A(0,t_{i-1})}{t_i - t_{i-1}}$, for any $i = 1, 2, \ldots, m$. Let l_i denote the i^{th} interval length, i.e., $l_i = t_i - t_{i-1}$. The workload function $A^r()$ is now defined as:

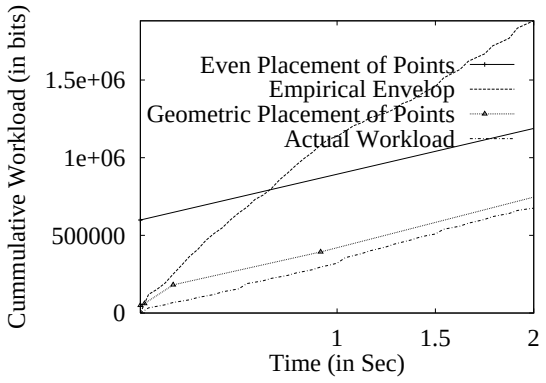

Figure 7: Comparison of the Cover Generated with 8 Segments for Geometrically Placed Points with Evenly Placed Points

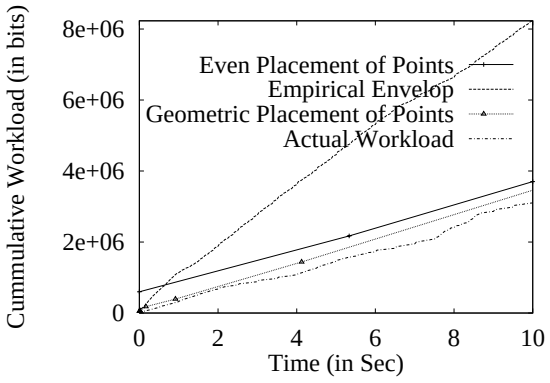

Figure 8: Comparison of the Cover Generated with 8 Segments for Geometrically Placed Points with Evenly Placed Points

$$A^r(0,t) = \begin{cases} \sigma_a(l_1) & : \quad t = 0 \\ \max\{A(0,t_i) + \sigma_a(l_i), A(0,t_i) + \sigma_a(l_{i-1})\} & : \quad t = t_i \\ A^r(0,t_i) + \rho_{i+1}(t - t_i) & : \quad t_i < t < t_{i+1} \end{cases}$$

Figure 6 illustrates the interpolation method. The above workload function can be constructed in $O(m)$ time using the precomputed σ_a. The set $\mathcal{L}^{'} = \{0 \leq t_1 \leq t_2 \leq \ldots \leq t_m = n\}$ is chosen so that the interval lengths $l_1, l_2, \ldots, l_m$ form a geometric series with $\sum_{i=1}^{m} l_i = n$, i.e., $\frac{l_{i+1}}{l_i} = \frac{l_{i+2}}{l_{i+1}}$. The details of the reasoning behind this is provided in [9]. Figure 7 shows the comparison of the workload function generated with eight segments using the average-rate interpolation with points chosen such that they are evenly spaced and geometrically spaced for Jurassic Park movie. Figure 8 shows the same result for a different time scale.

5 Performance Evaluations

We evaluate the benefits of the average-rate interpolation workload model via simulation using the workloads introduced in Section 2. Two types of arrival patterns are considered. In the bunched arrival pattern, all requests arrive before any request receives service. In the staggered arrival pattern, requests arrive according to some arrival process. We use the "average-rate" interpolation workload function with 14 segments and compare the performance improvements with the case when the exact workload and the exact empirical envelope are used. We determine the number of flows that can be admitted as a function of maximum packet delay under each workload function.

For homogeneous workloads, Figure 9 shows the number of flows that can be admitted for the three types of workload functions (that are mentioned above) when the requests are bunched. We observe a gain of $100 - 125\%$ over the best possible performance that can be achieved using any time-invariant workload function. For the staggered case, we assume that the requests arrive at 10 second intervals. Figure 10

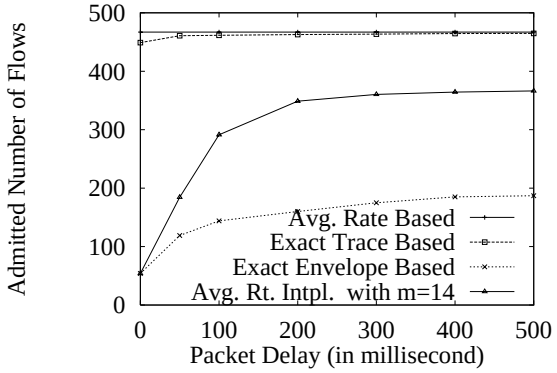

Figure 9: Comparison of Flows Admitted for Bunched Arrivals

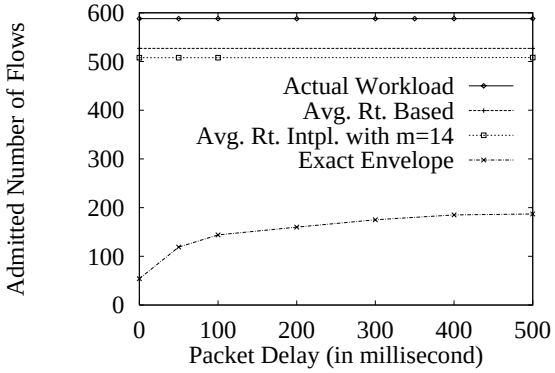

Figure 10: Comparison of Flows Admitted for Staggered Arrivals

illustrates similar results with staggered arrival of requests. We observe a gain in the range of $250 - 300\%$ over the best possible performance that can be obtained by using any time-invariant workload function. The reason for such a large performance improvement for staggered arrivals is that the admissibility conditions do not require check for worst-case workload requirement on arrival of each new request. They need to check against the aggregation of workload functions that are staggered. For time-invariant functions, it is necessary to check for worst-case workload requirement every time a new flow request arrives even though the requests are not bunched. This difference becomes more important if the workload requirement is relatively large during the initial few segments of a flow, especially when the maximum packet delay is very low. We observe similar gains in performance for heterogeneous workload [9].

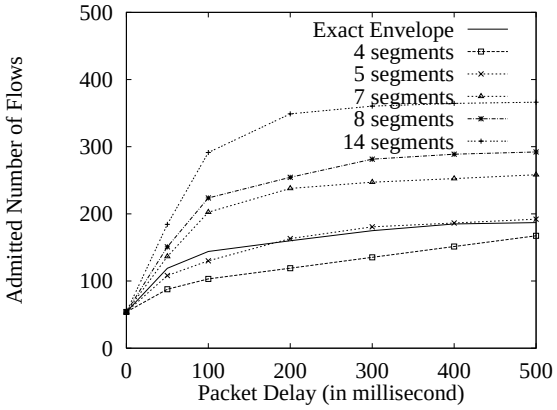

Figure 11: Admitted Number of Flows for Different Number of Segments in the Interpolating Cover

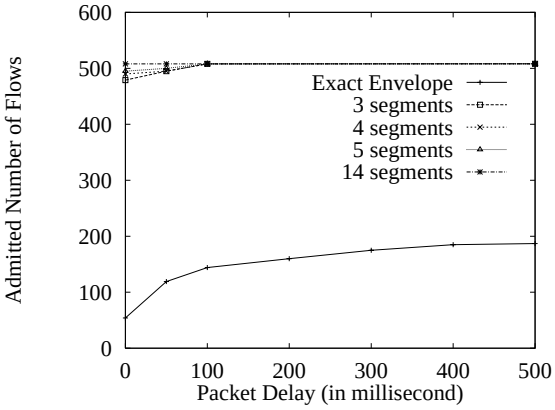

Figure 12: Number of Admitted Flows for Different Number of Segments with Staggered Arrival

Next we examine the sensitivity of our result to the number of linear segments. Figure 11 shows that an average-rate interpolation workload function with 5 or more segments achieves better performance than the best possible time-invariant workload function that has as many as 32768 linear segments. When requests

are staggered, Figure 12 shows that as few as three or four segments are required to achieve as much as 200% gain in performance.

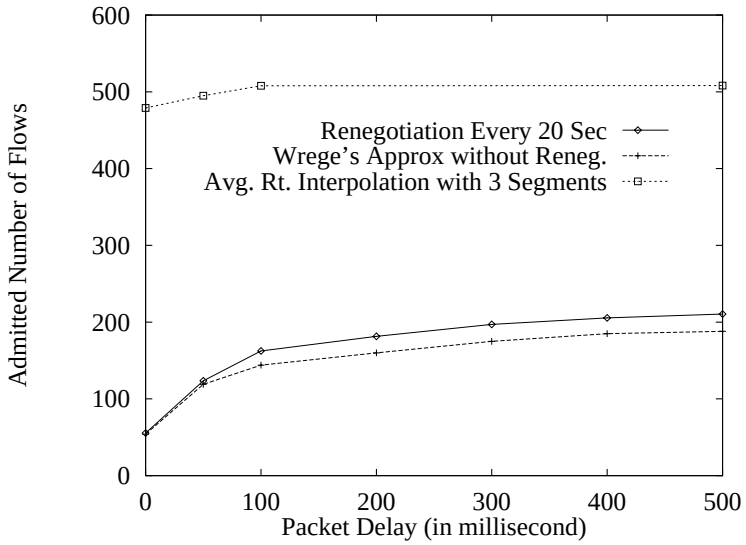

Figure 13: Comparison with Renegotiation Based Approach

Another approach for increasing resource utilization with a time-invariant workload function has been proposed in [13] where a *re-negotiation* based scheme is used to refine the workload function gradually as time of transmission progresses. This re-negotiation scheme provides deterministic service guarantee. The authors in [13] propose to recompute the approximated empirical envelope at periodic intervals using the remainder of the actual workload information. If the period of renegotiation is τ seconds, then after the first τ seconds, the empirical envelope is computed using the workload information during the interval $[\tau, T]$. From this recomputed empirical envelope, a piecewise linear cover is derived which is used as the new resource requirement in the re-negotiation mechanism. It has been shown in [13] that this approach can yield an additional 20% improvement. Figure 13 compares the number of flows that can be admitted by our approach with the re-negotiation in [13]. For comparison, we choose the renegotiation period as 20 second and the approximation parameter $k = 200$. We observe a greater benefit with our time-dependent workload model, both in resource utilization and computational complexity. The renegotiation approach needs $O(kn)$ amount of work every 20 seconds to generate the new workload function, and then perform additional resource reservation during the renegotiation phase. In contrast, our approach needs only $O(m)$ computation to derive a workload function (where m is around $3 - 4$) which is done only on the arrival of a flow request.

We have not compared our results with the re-negotiation scheme proposed in [4] since this does not fit into deterministic service model. Secondly, with our model we are able to achieve performance that is close that achievable with long term average rates and yet provides deterministic guarantee to admitted flows. But the renegotiation approach in [4] can be used in conjunction with our proposed model to further improve the

performance for statistical service model. This is a subject of future work.

6 Discussion and Applications

In this section, we discuss some of the issues and relevance of our work in the context of recent service models proposed for the Internet. The current trend has been to move away from the guaranteed service models such as *Guaranteed Service* (GS) and *Controlled-Load* (CL) that need a detailed end-to-end resource reservation on a per-flow basis. The new diff-serv proposals for supporting differentiated service models aim to provide *assured* throughput [1] and delay guarantees [7]. The *Bandwidth Broker* reservation model in [7] proposes a light weight resource reservation that does per-flow admission control and policing within the administrative domain of an ISP, especially at the first-hop router. Beyond the first-hop router, no per-flow policing or reservation is made. The admission control and policing inside the network is done on the aggregate workload. Our proposed model has the potential of being a good candidate for per-flow admission control at the first-hop router. This is because the first-hop router is within the administrative domain of an ISP and hence can be modified to easily to adapt to our admission control scheme. This could benefit an ISP tremendously by increasing the channel utilization within its administrative domain. The appropriate traffic shaping and regulation that are needed inside the network for such a service model is the subject of future work.

7 Conclusion and Future Work

In this work, we have taken a different approach to address the admission control problem for stored video. Instead of restricting to the set of time-invariant workload functions, we consider more general description of the workload. We have shown that any time-invariant description of the workload for compressed video grossly overestimate the exact workload. By considering more general workload functions that benefit from the detailed available information of the exact workload, we are able to achieve an improvement of $200 - 250\%$ over the best possible utilization that can be obtained using any time-invariant function. Our EDF-admissibility result for a broader range of workload functions generalizes the result in [6]. We have developed computationally efficient algorithms for admissibility check that is linear in the number of flows.

Another approach that has been taken for dealing with the rate variability in stored video is to smooth the original video by using workahead into a client buffer [10]. The evaluation of benefits of our proposed model with smoothed video flows is the subject of future study. We expect to observe the benefits which may not be as dramatic as we observe for unsmoothed video flows. But note that our model provides the flexibility that it can handle a mix of video flows, when some of them are smoothed and some are not.

Acknowledgements

We would like to thank Prof. Zhi-Li Zhang of Univ. of Minnesota for many helpful comments and discussions during the course of this work.

References

[1] D. Clark, and J. Wroclawski. An Approach to Service Allocation in the Internet, *IETF Draft*, July 1997.

[2] V. Firoiu, and D. Towsley. Call Admission and Resource Reservation for Multicast Sessions, *IEEE Infocom 96*, pp. 94-101, April 1996.

[3] V. Firoiu, J. F. Kurose, and D. Towsley Efficient Admission Control of Piece-Wise Linear Traffic Envelopes at EDF Schedulers, *CMPSCI Technical Report TR 97-23*, Dept. of Computer Science, Univ. of Massachusetts, June 1997.

[4] M. Grossglauser, S. Keshav, and D. Tse. RCBR: A Simple and Efficient Service for Multiple Time-Scale Traffic, *IEEE Transaction on Networking*, pp. 741-755, Dec. 1997.

[5] E. Knightly, and H. Zhang. Traffic Characterization switch utilization using deterministic bounding interval dependent traffic models. *IEEE Infocom 95*, pp. 1137-1145.

[6] J. Liebeherr, D. Wrege, and D. Ferrari. Exact Admission Control for Networks with Bounded Delay Services, *ACM/IEEE Transaction on Networking*, pp. 885-901, Dec. 1996.

[7] K. Nichols, and V. Jacobson. A Two-bit Differentiated Services Architecture for the Internet, *IETF Draft*, Nov. 1997.

[8] O. Rose. Statistical Properties of MPEG video traffic and their impact on traffic modeling in ATM systems, *Technical Report 101, Univ. of Wurzburg*, Feb. 1995.

[9] S. Sahu, V. Firoiu, D. Towsley and J. Kurose. Traffic Models and Admission Control for Continuous Media Transmission with Deterministic Services, *CMPSCI Technical Report*, Dept. of Computer Science, Univ. of Massachusetts (In Preparation).

[10] J. D. Salehi, Z.-L. Zhang, J. Kuorse, and D. Towsley. Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements through Optimal Smoothing, *In the Proceedings of ACM Sigmetrics*, pp. 222-231, May 1996.

[11] D. E. .Wrege, and J. Liebeherr. Video Traffic Characterization for Multimedia Networks with a Deterministic Service, *IEEE Infocom 96*, pp. 537-544, April 1996.

[12] 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 Tradeoffs, *ACM/IEEE Transaction on Networking*, pp. 352-362, June 1996.

[13] J. Liebeherr, and D. Wrege. An Efficient Solution to Traffic Characterization of VBR Video in QOS Networks, *Submitted for Publication to Multimedia Systems*.

[14] H. Zhang, and D. Ferrari. Rate-Controlled Service Disciplines, *Journal of High Speed Networks*, Vol. 3(4), 1994.