To appear: ACM/Springer Multimedia Systems Journal.

Traffic Characterization Algorithms for VBR Video in Multimedia Networks*

*Jörg Liebeherr** *Dallas E. Wrege*†

* Department of Computer Science
University of Virginia
Charlottesville, VA 22903
Email: jorg@cs.virginia.edu

† IBM Corporation
3039 Cornwallis Road
Research Triangle Park, NC 27709
Email: Wrege@raleigh.ibm.com

Abstract

A network that offers deterministic, i.e., worst-case, quality-of-service (QoS) guarantees to variable-bit-rate (VBR) video must provide a resource reservation mechanism that allocates bandwidth, buffer space, and other resources for each video stream. Such a resource reservation scheme must be carefully designed, otherwise network resources are wasted. A key component for the design of a resource reservation scheme is the *traffic characterization* method that specifies the traffic arrivals on a video stream. The traffic characterization should accurately describe the actual arrivals so that a large number of streams can be supported; but it must also map directly into efficient traffic policing mechanisms that monitor arrivals on each stream. In this study, we present a fast and accurate traffic characterization method for stored VBR video in networks with a deterministic service. We use this approximation to obtain a traffic characterization that can be efficiently policed by a small number of leaky buckets. We present a case study where we apply our characterization method to networks that employ a dynamic resource reservation scheme with renegotiation. We use traces from a set of 25-30 minute MPEG sequences to evaluate our method against other characterization schemes from the literature.

Key Words: Multimedia Networks, Traffic Characterization, Bounded Delay Service, Quality-of-Service, Deterministic Service, Renegotiation, Resource Reservation, VBR Video.

*This work is supported in part by the National Science Foundation under Grants No. NCR-9309224 and NCR-9624106. Some results in this paper were presented at the IEEE Infocom '96 Conference.

1 Introduction

With the advent of integrated-services packet-switching networks, it has become feasible to design multimedia networks that can transmit compressed variable-bit-rate (VBR) video in real time. Since video has strict *quality-of-service* (QoS) requirements, a multimedia network must maintain a resource reservation scheme [10] that can allocate network resources for each video stream. In this study we consider resource reservation schemes for QoS networks with a *deterministic service*, that is, a service that can provide worst-case delay guarantees to all packets on a video stream. Since overallocating resources to a stream results in a low network utilization, a resource reservation scheme must be carefully designed.

The amount of resources to be reserved for a video stream is largely dependent on the *traffic characterization* method used to describe the traffic on a video stream. For a deterministic service we require a *deterministic traffic characterization* method that gives an upper bound on the traffic [7, 9, 17, 26, 34]. The selection of particular traffic characterization method must trade-off between the following design goals:

1. *Accuracy:* The traffic characterization method should describe the actual traffic on a stream as accurately as possible. A pessimistic characterization will overestimate the resource requirements of a stream, resulting in poor network utilization. Recently, several traffic characterization methods for VBR video have been proposed that can express the complex timely correlations of VBR video sequences [11, 12, 16, 28, 29].

2. *Simplicity:* Traffic policing mechanisms which ensure that all traffic submitted to the network conforms to its traffic characterization must be able to operate at the speed of the transmission links. Therefore, the traffic characterization method must be simple enough so that fast traffic policing algorithms can be designed.

3. *Speed:* The effort to compute a traffic characterization for a video stream must be reasonably small. While traffic characterization for stored video can be computed off-line, networks that offer dynamic renegotiation of traffic parameters (see Section 6) require availability of fast traffic characterization algorithms.

Possibly, the single most important approach to traffic characterization of VBR video is derived from the so-called *leaky bucket*[1] which describes traffic in terms of rate and burstiness parameters. With leaky bucket traffic characterizations, the worst-case traffic on a stream is given in terms of piecewise-linear functions [7]. Since each leaky bucket can be implemented with a single counter and a single timer [27], the traffic policing algorithm for a single leaky bucket satisfies the demand for simplicity.

In a previous study [34] we showed that concave piecewise-linear functions ("leaky buckets") can also satisfy the need for accuracy. However, the number of leaky buckets needed for an accurate

[1]In this paper, a *leaky bucket* [33] is equivalent to the *Generic Cell Rate Algorithm (GCRA)* as specified in [1].

characterization was too large to be of practical value. For example, up to a dozen leaky buckets were needed to accurately characterize an MPEG-I video stream [34]. Since practical considerations limit the number of available leaky buckets to a small value, e.g., the limit is two for ATM streams [1], methods are needed that yield an accurate VBR traffic characterization, yet, with only a small number of leaky buckets.

In this study, we will show that concave piecewise-linear functions can be used for traffic characterization of VBR video traffic while satisfying all the goals of accuracy, simplicity, and speed. We present an algorithm that takes as input a VBR video sequence and generates the parameters for a specified number of leaky buckets. Using empirical examples, we will demonstrate that three leaky bucket are sufficient to produce traffic characterizations leading to utilizations within 91% of the best possible characterization. The computation time necessary to generate a traffic characterization for a feature length movie is less than a second, thus, satisfying the demand for a fast characterization method.

The presented work is based on an observation of a characteristic of MPEG video traces. Studying 'best-possible' time-invariant bounds, so-called *empirical envelopes* [34], of several 30-minute MPEG traces, we found that the first few hundred data points of the empirical envelope can provide a traffic characterization that has almost the same accuracy as the entire envelope with 40,000 points. In other words, we are able to show that, for the purposes of deterministic traffic characterization, complete information on a video trace is retained in very few parameters.

The traffic characterization method in this paper will be evaluated against existing methods that have applied leaky buckets for VBR video characterizations. Note, however, that most methods proposed in the literature do not strive for high network utilization as a design goal, and, instead, select the traffic parameters according to network resource availability. Also, most studies use only a single leaky bucket [23, 25, 28, 29, 31]. Pancha and El Zarki [25] choose parameters by fixing the burstiness parameter of a leaky bucket according to available buffer space. In [3], the choice of the traffic parameters depends on the relative availability of unallocated bandwidth and buffer space. An approach discussed by Guillemin et. al. in [15] assigns relative importance parameters α and β to buffer space and bandwidth, respectively; the parameters are selected such that the product of burstiness and rate is minimized. Guillemin et. al. [15] presented two heuristic algorithms that select leaky bucket parameters to approximate an "ideal" probabilistic traffic characterization, the so-called time-ϵ quantile function. The heuristic algorithms take an approach to traffic characterization that is similar to ours, in that they first determine a function that describes the traffic on a stream and then choose parameters based on this function.

Finally, we want to point to studies that have explored the benefits of reducing the burstiness of VBR video traffic by either (1) *shaping* the traffic by spacing packets before submitting them to the network [13, 18, 19] or (2) sending packets early with respect to their playback time at the receiving application via *workahead smoothing* [24, 29, 32]. These techniques involve modification of the traffic that is submitted to the network by buffering at either the sender, receiver, or a combination of both. While shaping and smoothing techniques have been shown to increase the achievable network utilization, these methods are orthogonal to the traffic characterization problem studied in this paper. Note that even after traffic is shaped or smoothed, a characterization method such as the one developed in this paper must be available to determine an accurate and policable characterization for the traffic submitted to the network.

The remainder of this paper is structured as follows. In Section 2 we review the issues involved in deterministic traffic characterizations for VBR video traffic. We present our traffic characterization method in Sections 3 and 4. In Section 3 we describe a method for approximating the empirical envelope using only a small number of envelope parameters. In Section 4 we present an algorithm for reducing the number of leaky buckets of a given traffic characterization. In Section 5 we evaluate our characterization method by comparing it with previously proposed schemes. In Section 6 we discuss a case study where we apply our method to construct a so-called deterministic renegotiation scheme. In Section 7 we summarize the findings of this paper.

2 Deterministic Traffic Characterizations

In this section, we review the key concepts of traffic characterizations for a deterministic service, referred to as *deterministic traffic characterizations*. We first provide necessary background on deterministic traffic characterizations. Then we discuss our previous work with Zhang and Knightly [34] on best possible deterministic traffic characterizations for traffic traces with so-called *empirical envelopes*.

2.1 Subadditive Traffic Characterizations

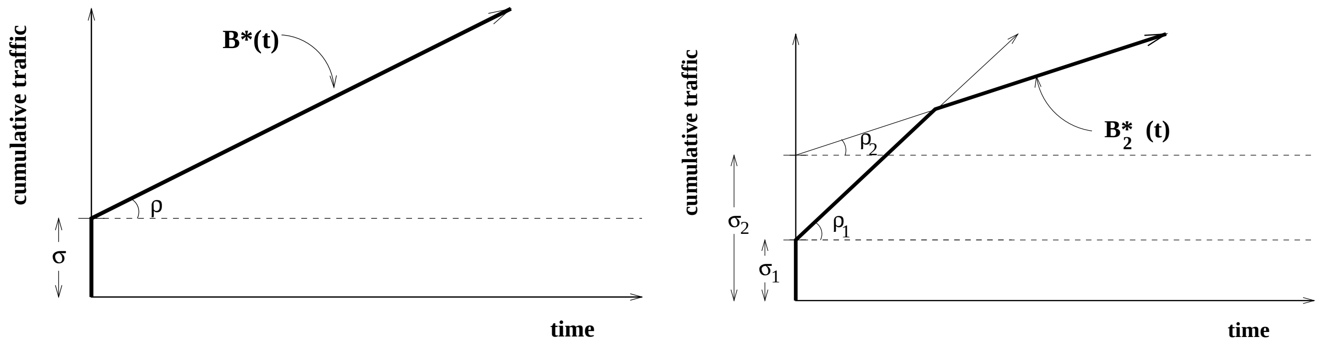
Let A denote the actual traffic on a stream, where $A[\tau, \tau+t]$ denotes the traffic arrivals in time interval $[\tau, \tau+t]$. Then, a worst-case characterization of the traffic A is given by a *characterization function* A^* which provides an upper bound on A. A characterization function A^* should satisfy two important properties, namely *time-invariance* and *subadditivity* [7, 22].
A function A^* provides a time-invariant bound for A if for all times $\tau \geq 0$ and $t \geq 0$ the following holds [7]:

$$A[\tau, \tau+t] \leq A^*(t) \tag{1}$$

Since a time-invariant function A^* bounds the maximum traffic over any time interval of length t, the traffic description is independent of the starting time of a stream. A characterization function A^* is subadditive if it satisfies the following inequality:

$$A^*(t_1) + A^*(t_2) \geq A^*(t_1+t_2) \qquad \forall t_1, t_2 \geq 0 \tag{2}$$

A subadditive characterization function allows the arrivals on a stream to attain the bound given by A^*. In other words, it is feasible that $A[\tau, \tau+t] = A^*(t)$ for any $t \geq 0$. Even though characterization functions that are time-invariant but not subadditive have been proposed in the literature,



(a) Single Leaky Bucket. (b) Multi-Leaky Bucket.

Figure 1: Characterization functions for leaky buckets.

e.g., [16], we point out that any such traffic characterization function A_1^* can be replaced by a subadditive function A_2^* such that $A_2^*(t) \leq A_1^*(t)$ for all $t \geq 0$. In the following, we call a characterization function A^* for A *viable* if it satisfies both equations (1) and (2).

Practical traffic characterizations are derived from parameterized *traffic models* which can describe a characterization function with a few parameters. In this study, we are interested in traffic models that correspond to leaky bucket policing mechanisms. The (σ, ρ) traffic model [7] describes the worst-case traffic admitted by a leaky bucket with a burstiness parameter σ and a rate parameter ρ. We denote the characterization function that provides a bound on the maximum traffic conforming to the (σ, ρ) model by B^*, where B^* is given by the following linear constraint [7]:

$$B^*(t) = \sigma + \rho t \qquad \text{for all } t \geq 0 \tag{3}$$

The characterization function is illustrated in Figure 1(a).

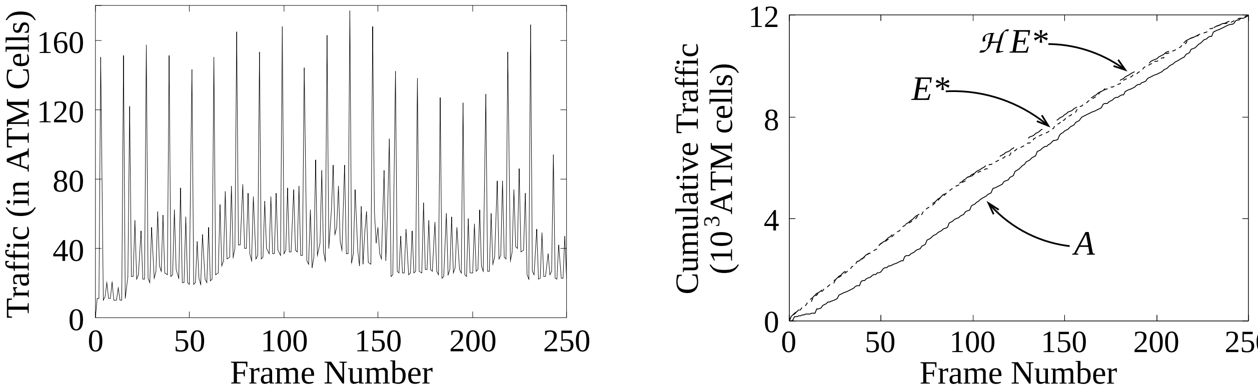
The leaky bucket traffic model can be generalized to the $(\vec{\sigma}, \vec{\rho})$ traffic model [8, 34] which corresponds to traffic policing by multiple leaky buckets connected in series. For a stream that conforms to the $(\vec{\sigma}, \vec{\rho})$ traffic model with a set of m pairs $\{(\sigma_i, \rho_i)\}_{1 \leq i \leq m}$, the amount of traffic admitted to the network is limited by each of the (σ_i, ρ_i)-pairs. The resulting characterization function, illustrated in Figure 1(b) and denoted as B_m^*, is a concave function consisting of m piecewise-linear segments [8, 34]:

$$B_m^*(t) = \min_{1 \leq i \leq m} \{\sigma_i + \rho_i t\} \tag{4}$$

All traffic characterization methods considered in this paper determine characterization functions that conform to the $(\vec{\sigma}, \vec{\rho})$ traffic model.

2.2 The Empirical Envelope E^*

In previous work with Knightly and Zhang [34], we presented a traffic characterization, referred to as the "empirical envelope", denoted by E^*, which is the tightest characterization function for a given traffic source [2, 34]. The empirical envelope E^* of a video sequence is optimal in the sense



(a) MPEG traffic trace.

(b) Cumulative traffic and characterization functions.

Figure 2: Functions A, E^* and $\mathcal{H}E^*$ for an actual MPEG trace.

that, for all subadditive characterization functions A^* that satisfy Equation (1), E^* is minimal, that is, $E^*(t) \leq A^*(t)$ for all t. The empirical envelope E^* is obtained as follows [2, 34]:

$$E^*(t) = \sup_{\tau \geq 0} A[\tau, \tau + t] \qquad \forall t \geq 0 \tag{5}$$

To obtain the empirical envelope of a given video sequence consisting of N frames with fixed inter-frame time r, we assume that frames are fragmented into 53-byte ATM cells with a payload of 48 bytes each, and these cells are transmitted at equally-spaced intervals over the frame time r. If the sequence of frame sizes of a video sequence is given by $\{f_1, f_2, \ldots, f_N\}$, then the empirical envelope E^* can be constructed by calculating [34]:

$$E^*(ir) = \max_{0<k<N-i+1} \sum_{j=k}^{k+i-1} f_j \qquad \text{for } i = 1, 2, \ldots N \tag{6}$$

Note that Equation (6) defines N parameters $\{E^*(ir) \mid 1 \leq i \leq N\}$ for the empirical envelope, where $E^*(r)$ is equal to the largest frame in the video sequence, $E^*(2r)$ is equal to the largest two-frame sequence, etc. The values of the empirical envelope at times that are not multiples of the frame time are obtained by spacing the cells in $E^*(ir) - E^*((i-1)r)$ evenly over the frame time $[(i-1)r, ir]$.

Policing of the empirical envelope E^* is impractical since it has one parameter for each frame of a video sequence, resulting in about 1,800 parameters per minute of video. In [34], we solved this problem by taking the *concave hull of* E^*, which we denote by $\mathcal{H}E^*$ [34][2] we obtain a $(\vec{\sigma}, \vec{\rho})$-model traffic characterization [34]. Since $\mathcal{H}E^*$ is the smallest piece-wise linear function larger than E^* [6], $\mathcal{H}E^*$ is the most accurate traffic characterization that can be policed by leaky buckets. In Figure 2 we illustrate the traffic characterization method from [34] for a trace of 250 frames from an MPEG movie. In Figure 2(a) we plot the number of cells (with 48-byte payloads) as a function of the frame sequence numbers. In Figure 2(b) we illustrate the cumulative arrivals A for the trace in Figure 2(a), the empirical envelope E^* of the trace, and its concave hull $\mathcal{H}E^*$.

[2]In this document, we use $\mathcal{H}$ to denote the concave hull operator, that is, $\mathcal{H}f$ is the concave hull of the function f.

A drawback of the concave hull $\mathcal{H}E^*$ is that it generates too many piecewise-linear segments. In most of our examples, the concave hull of the empirical envelope had more than ten piecewise-linear segments, resulting in a requirement for as many leaky buckets to police a single video stream. Clearly, this violates the requirement for simplicity of the traffic characterization methods.

As a second drawback, the computation of the empirical envelope, needed for deriving the concave hull, is computationally expensive. The number of operations required to compute all N parameters of the empirical envelope E^* for a video sequence with N frames is $O(N^2)$. Since N is generally large, e.g., it exceeds 200,000 for most feature-length motion pictures, it is not possible to calculate the empirical envelope in real-time, violating the requirement for a speedy calculation of the characterization functions.

In the next sections, we will provide solutions that overcome these drawbacks. The resulting characterization method will satisfy all of the requirements for accuracy, simplicity, and speed, that we specified in Section 1.

3 Approximating the Empirical Envelope with Extrapolations

In this section we show that accurate traffic characterizations can be calculated without computing the entire empirical envelope. We present two methods for obtaining viable characterization functions that are derived from only the first k parameters of the empirical envelope, i.e., $E^*(r)$, $E^*(2r)$, $\ldots$, $E^*(kr)$. Both methods construct a characterization function through extrapolation of the first k values of the empirical envelope.

We first discuss the best-possible extrapolation based on the first k parameters of E^*. Then we present a simple characterization that can be obtained with a fast extrapolation technique. The evaluation of the extrapolation methods at the end of this section will show that with $k \geq 200$ the extrapolation methods will result in very accurate traffic characterizations.

3.1 Largest Subadditive Extrapolation

Any viable characterization function obtained from the first k parameters of the envelope must be at least as large as the empirical envelope E^* for all times t. Since we know that E^* is a subadditive function, the best extrapolation is given by the *largest subadditive extrapolation of* $\{E^*(ir)\}_{1\leq i\leq k}$. We denote this largest subadditive extrapolation by E^*_k, where E^*_k is obtained by calculating:

$$E^*_k(ir) = \begin{cases} E^*(ir) & \text{for } i \leq k \\ \min\limits_{1\leq j<i} \{E^*_k(jr) + E^*_k((i-j)r)\} & \text{for } i > k \end{cases} \qquad (7)$$

E^*_k is equal to the empirical envelope for the first k frame times, and E^*_k is defined for subsequent times by exploiting the requirement for subadditivity of E^*_k.[3]

[3]Note that equation (7) only defines E^*_k for times that are multiples of the frame time r. Similar to the production of the empirical envelope in Equation (6), the values for intermediate values of E^*_k are determined by spacing cells evenly over each frame.

Although the function E_k^* is the tightest characterization function that can be obtained directly from the first k parameters of the envelope, the production of E_k^* is expensive. Specifically, we see from Equation (7) that the number of computations required to construct E_k^* is $O(N^2)$, the same number required for computing the empirical envelope itself. Since we seek an approximation that can be computed efficiently, we search for other approximation schemes, and we will use E_k^* as a benchmark.

3.2 Repetition Extrapolation

Next we present a more efficient approach to obtain a deterministic traffic characterization based on knowledge of only the first k frames of the empirical envelope. Consider the function that is obtained by simply repeating the first k parameters $\{E^*(ir)\}_{1\leq i\leq k}$ for all times t. We call such a function the *repetition extrapolation*, which we denote by R_k^*. R_k^* is given as follows:

$$R_k^*(t) = \lfloor \frac{t}{kr} \rfloor E^*(kr) + E^*(t - \lfloor \frac{t}{kr} \rfloor (kr)) \quad \text{for } t \geq 0 \tag{8}$$

Observe that R_k^* can be immediately obtained from the first k parameters of the envelope, and so the computational complexity of computing R_k^* is $O(kN)$.

However, although R_k^* provides a time-invariant bound on the traffic arrivals A in terms of Equation (1), it is not necessarily subadditive, and, hence, does not satisfy our requirement for a viable traffic characterization. Therefore, an additional step is required to make the repetition extrapolation a viable characterization. The steps are summarized as follows.

- Step 1: Calculate the repetition extrapolation R_k^* given by Equation (8).

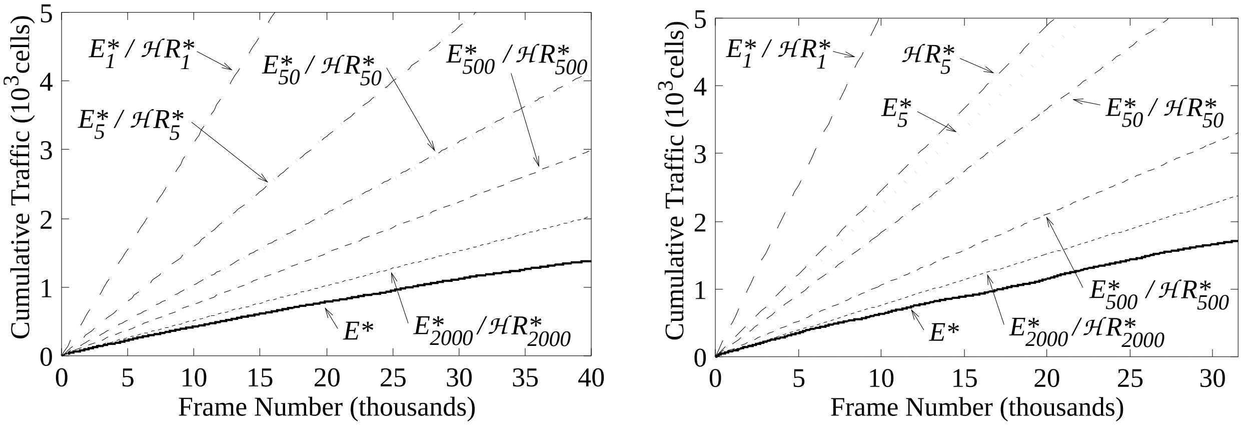
- Step 2: Determine a subadditive function from the repetition extrapolation R_k^* by taking the *concave hull of* R_k^*. Since the concave hull $\mathcal{H}R_k^*$ is a concave function, it is clearly subadditive. Recall that the concave hull is a piecewise linear function that can be expressed in terms of the $(\vec{\sigma}, \vec{\rho})$ model.

A problem that remains to be solved is the potentially large number of leaky buckets needed to police the concave hull $\mathcal{H}R_k^*$. We defer our solution to this problem to Section 4. In the next subsection we present an empirical evaluation of the extrapolation methods of Subsections 3.1 and 3.2

3.3 Evaluation

Here we evaluate the accuracy of traffic characterizations E_k^* and $\mathcal{H}R_k^*$ as approximations of the empirical envelope using actual traces of MPEG-compressed video. We want to find answers to the following questions:

- How closely do the largest subadditive extrapolation from Subsection 3.1 and the repetitive extrapolation from Subsection 3.2 approximate the empirical envelope for different values of k?



(a) *Park*

(b) *News*

Figure 3: Characterization functions.

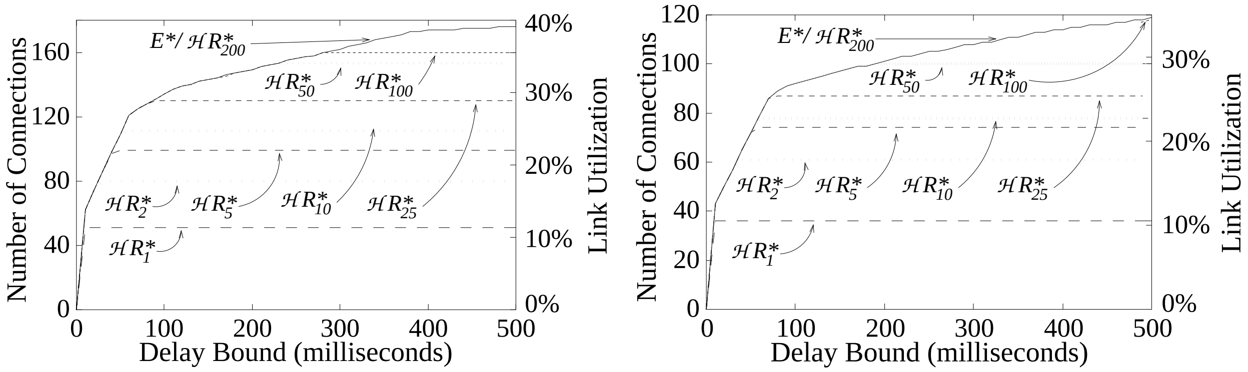
- How well does the repetitive extrapolation serve as an approximation of the best possible (largest subadditive) extrapolation?
- How much information of the empirical envelope is needed by the extrapolations to generate an accurate characterization?

We use two MPEG traces in the evaluation: one from the entertainment film *Jurassic Park* ("*Park*"), and the second from a news broadcast ("*News*"). These sequences were encoded in software with the Berkeley MPEG-encoder [30]. Both *Park* and *News* are thirty-minute video sequences with a frame size of 384x288 and frame pattern IBBPBBPBBPBB. We note that *News* generates burstier traffic than *Park*; the ratio of the peak rate to the average rate for *News* and *Park* are 6 and 4, respectively.

To address the first and second question above we show in Figures 3(a) and 3(b) the characterization functions for the *News* and *Park* traces. The figures depict the empirical envelope E^* as well as E^*_k and $\mathcal{H}R^*_k$ for $k \in \{1, 5, 50, 500, 2000\}$. For each characterization function, we plot the cumulative number of cells as a function of the frame sequence number. In both graphs, the empirical envelope E^* is shown as a bold solid curve, while the functions E^*_k and $\mathcal{H}R^*_k$ are depicted by dotted and dashed curves, respectively. As expected, the approximation functions estimate the empirical envelope E^* accurately as k is increased.

A key observation from Figure 3 is that $\mathcal{H}R^*_k \approx E^*_k$ for most values of k; only $\mathcal{H}R^*_5$ and E^*_5 for the *News* sequence in Figure 3(b) differ considerably. Since E^*_k is the tightest traffic characterization that can be produced from k frames of the empirical envelope, we conclude that the concave hull of the repetition extrapolation $\mathcal{H}R^*_k$ is an accurate approximation of E^*_k.

To address our third question above, we next consider the maximum resource utilization that can be achieved at a video multiplexer that uses characterization functions $\mathcal{H}R^*_k$. We assume that the multiplexer operates at rate $r = 155$ Mbps, and that the packet scheduling method of the multiplexer is FCFS. Let d denote the maximum delay bound of a video stream. Then a set $\mathcal{C}$ of



(a) *Park* (b) *News*

Figure 4: Utilization comparison.

streams can be supported by a FCFS multiplexer without delay bound violations if and only if [7]:

$$d \geq \sum_{j \in \mathcal{C}} A_j^*(t) - t \cdot r \qquad , \forall t \geq 0 \tag{9}$$

Figure 4 illustrates the network utilization obtained at a multiplexer using E^* as well as $\mathcal{HR}_k^*$ for various values of k. All streams at a multiplexer are assumed to be of the same type (either *Park* or *News*) and have identical delay bounds (in the range $0 \leq d \leq 500$ msec). For each characterization, we plot both the maximum number of admissible streams as well as the link utilization as a function of the delay bounds of those streams. For example, Figure 4(b) shows that the characterization function $\mathcal{HR}_2^*$ can be used to support 61 *News* streams for delay bounds larger than 35 ms, thereby utilizing 18% of the 155 Mbps transmission link. Note that the maximum link utilization that is available for a deterministic service is below 50%. However, the unused bandwidth can be utilized by other traffic classes, such as best-effort traffic [10] or predictive traffic [5].

The general trend in both graphs is that the number of streams accepted using $\mathcal{HR}_k^*$ as the characterization function increases with k. An important observation is that the function $\mathcal{HR}_{200}^*$ admits the same number of streams as the empirical envelope E^* for delay bounds up to 500 milliseconds. Thus, we can use our approximation function $\mathcal{HR}_k^*$ based on the first 200 parameters of the envelope (i.e., $\mathcal{HR}_{200}^*$) to characterize both video sequences for the delay bound range considered; we achieve the same utilization using $\mathcal{HR}_{200}^*$ as we would using the empirical envelope with 40,000 parameters.

However, while the function $\mathcal{HR}_k^*$ provides an accurate traffic characterization for VBR video, the number of leaky buckets required to enforce $\mathcal{HR}_k^*$ may be excessive. For example, 12 leaky buckets are needed to police $\mathcal{HR}_{200}^*$ for the *News* sequence (i.e., $\mathcal{HR}_{200}^* \equiv B_{12}^*$). To solve this problem, we next present an algorithm that reduces the number of leaky buckets needed to police a video stream.

4 An Algorithm for Reducing the Number of Leaky Buckets

The discussion in the last section has shown that the repetitive extrapolation approach yields an accurate traffic characterization. However, the concave hull of the extrapolation generated a large number of piecewise linear segments, up to 12 for the examples in Section 3. Since actual networks typically limit the number of leaky buckets that are available for monitoring a stream to only two or three [1], we present a method to reduce the number of leaky buckets needed for policing. The problem of reducing the number of leaky buckets is expressed as follows:

> *Given an n-piecewise linear function B_n^*, find an m-piecewise linear function B_m^* with $m < n$ such that $B_m^*(t) \geq B_n^*(t)$ for all $t \geq 0$.*

Since we want to select a function B_m^* that is 'close' to B_n^*, we introduce a cost function $\Delta(B_n^*, B_m^*)$ to express the "difference" between B_n^* and B_m^*. With Δ we can formulate our problem as the solution to the following optimization problem.

> **Optimization Problem.** *Given an n-piecewise linear function B_n^* find the m-piecewise linear function B_m^* that minimizes $\Delta(B_n^*, B_m^*)$ with $m < n$ subject to the constraint: $B_m^*(t) \geq B_n^*(t)$ for all $t \geq 0$.*

The goal for selecting a cost function Δ is that the characterization function B_m^* allows the admittance of a large number of streams. However, the selection of a 'good' cost function Δ is not obvious. For example, since the burstiness of VBR video limits the number of admitted streams at small delay bounds, it is important that the function B_m^* approximates B_n^* closely for small values of t. We have evaluated a large set of cost functions[4] and found that the following function yields excellent results.

$$\Delta(B_n^*, B_m^*) = \int_0^T \frac{B_m^*(t) - B_n^*(t)}{B_n^*(t)}\, dt, \tag{10}$$

where T bounds the time interval over which the cost Δ is selected. In our examples, T was selected to the time interval that corresponds to 200 frame times.

Since the above cost function does not allow us to solve the optimization problem directly, we present a heuristic algorithm that finds an approximate solution.

The heuristic algorithm is presented in Figure 5. The initialization of the pairs (σ_i, ρ_i) is shown in Steps 2 through 5 of Figure 5. Observe that the initial values are m pairs selected from $\{(\overline{\sigma}_j, \overline{\rho}_j) \mid j = 1, \ldots, n\}$ of B_n^*. The heuristic improves the initial selection by altering the (σ_i, ρ_i)-pairs using the iteration shown in Steps 6 through 11 of the figure. In each iteration step, the (σ_i, ρ_i)-pairs are modified to reduce the cost function Δ. The iteration terminates when the cost cannot be significantly reduced. The crucial step of the algorithm is Step 9, where a single pair (σ_l, ρ_l) is modified to minimize the cost function subject to the constraint that $B_m^*(t) \geq B_n^*(t)\ \forall t$. During this step, the values of all pairs $\{(\sigma_i, \rho_i) \mid i \neq l\}$ are kept constant, and the pair (σ_l, ρ_l) is selected

[4]The results of the evaluation are presented in Appendix A.

Input: An n-piecewise linear function B_n^*, given by parameters $\{(\overline{\sigma}_j, \overline{\rho}_j) \mid j = 1, \ldots, n\}$ with $\overline{\sigma}_i \leq \overline{\sigma}_j$ and $\overline{\rho}_i \geq \overline{\rho}_j$ whenever $i < j$, an integer m $(m < n)$, a cost function Δ, and a sensitivity parameter ϵ.

Output: An m-piecewise linear function B_m^*, given by m parameters $\{(\sigma_i, \rho_i) \mid i = 1, \ldots, m\}$.

1. **Procedure** Parameterize (B_n^*, m, Δ, ϵ)
2.   **For** $i = 1$ **To** m /* Initialize (σ_i, ρ_i) */
3. $\sigma_i \leftarrow \overline{\sigma}_{\lfloor \frac{in}{m} \rfloor}$
4. $\rho_i \leftarrow \overline{\rho}_{\lfloor \frac{in}{m} \rfloor}$
5.   **End For**
6.   **Do**   /* Greedy modifications */
7.   Cost $\leftarrow \Delta$
8.   **For** $i = m$ **Down To** 1
9.   Select (σ_i, ρ_i) to minimize Δ, where $\sigma_{i-1} \leq \sigma_i \leq \sigma_{i+1}$
10.   **End For**
11.   **While** (Cost $- \Delta > \epsilon$)
12.   **Output** $B_m^* \leftarrow \min_{1 \leq i \leq m}\{\sigma_i + \rho_i t\}$
13. **End Procedure**

Figure 5: Parameterization algorithm.

such that $\sigma_{l-1} < \sigma_l < \sigma_{l+1}$ (with boundary conditions $\sigma_1 \geq 0$ and $\sigma_m \leq \overline{\sigma}_n$). Note that the choice of ρ_l is dependent on σ_l.

REMARKS: In the empirical evaluation presented in Section 5, we select the (σ_i, ρ_i) pair of minimum cost in Step 9 through an exhaustive search through all possible values of σ_i. However, with ρ_i expressed in terms of σ_i, it is possible to write Δ with σ_i as the only independent variable, and the selection can be determined analytically by setting $\frac{\partial \Delta}{\partial \sigma_i} = 0$. Also, while we do not make guarantees on the running time of the algorithm, the examples that we ran converged rapidly. In all examples, no more than six iterations were required even when we set $\epsilon = 0$.

5 Empirical Evaluation

We are now ready to evaluate our fast traffic characterization method for VBR video sources by comparing it with other traffic characterization schemes from the literature. The steps of our approximation method are summarized in Figure 6.

With the results from Sections 3 and 4, we compute a piecewise-linear function B_m^* based on

$$\{E^*(jr) \mid j = 1, \ldots, k\} \xrightarrow{\text{Repetition Extension}} R_k^* \xrightarrow{\text{Concave Hull}} \mathcal{H}R_k^* = B_n^* \xrightarrow{\text{Heuristic Algorithm}} B_m^*$$

Figure 6: Overview of traffic characterization method.

the function $\mathcal{H}R_{200}^* \equiv B_n^*$. Recall from the results of Figure 4 that for $k \geq 200$, the concave hull

$\mathcal{H}R_k^*$ provided results that were identical to those attained with the (best possible) characterization using the empirical envelope E^*.

We evaluate the characterization method using the MPEG video traces *Park* and *News* described in Section 3.3 and a single FCFS multiplexer at a switch that operates at 155 Mbps. As before, we use the condition in Equation (9) as the admission control test.

We compare our traffic characterization with previously proposed schemes for generating leaky bucket parameters. In the following, we discuss the schemes and their parameters:

(a) **Peak-rate**: A peak-rate characterization is determined by a single rate parameter ρ_1 which is assumed to be the ratio of the size of the largest video frame f_j and the inter-frame time r, i.e., $\rho_1 = \frac{1}{r} \max_{0<j\leq N} f_j$.

(b) **Dual bucket**: This scheme results from a straightforward calculation of two leaky buckets pairs (σ_1, ρ_1) and (σ_2, ρ_2). The first pair describes the peak rate with ρ_1 as described in (a) and $\sigma_1 = 0$. The second pair describes the average rate with $\rho_2 = \frac{1}{N \cdot r} \sum_{j=1}^{N} f_j$ and $\sigma_2 = \inf\{x \mid x + \rho_2 t \geq E^*[\tau, \tau + t], \ \forall t, \ \tau \geq 0\}$.

(c) **Fixed burst**: This scheme was outlined in [25] and uses a pair (σ_2, ρ_2) with the burst parameter σ_2 set equal to a "reasonable" buffer size suggested to be 1000 cells. We have improved the scheme by adding a cell-spacer that enforces the peak rate. We obtain as leaky bucket parameters: ρ_1 as in (a), $\sigma_1 = 0$, $\sigma_2 = 1000$ cells and $\rho_2 = \inf\{y \mid \sigma_2 + yt \geq E^*[\tau, \tau + t], \ \forall t, \ \tau \geq 0\}$.

(d) **Concave hull**: The concave hull approach in [34] selects the first m (σ, ρ) pairs from the concave hull of the empirical envelope. The scheme was outlined in Section 2.

(e) **Product**: The *Product* scheme from [15] uses a peak rate ρ_1 and a pair (σ_2, ρ_2), where σ_2 and ρ_2 are chosen from the candidate set of leaky buckets $\{(\sigma, \rho) \mid \sigma + \rho t \geq E^*[\tau, \tau + t], \ \forall t, \ \tau \geq 0\}$ such that the product $\sigma_2 \cdot \rho_2$ is minimized.

(f) **Maximum distance**: This scheme from [15] uses the peak rate ρ_1 and a pair (σ_2, ρ_2) where ρ_2 is selected such that $\delta = \sup_t \{\frac{1}{t}(\sigma_2 + \rho_2 - M_\epsilon^*(t))\}$ is minimized. The quantity $M_\epsilon(t)/t$ specifies the rate of the video sequence over multiple time scales t. More precisely, if N_t denotes a random variable specifying the number of packets generated over any interval of length t, then $M_\epsilon(t)$ specifies with probability $1 - \epsilon$ the maximum traffic arrivals n in any interval of length t: $M_\epsilon(t) = \inf\{n, \ Pr\{N_t \geq n\} \leq \epsilon\}$ [15, 31]. This heuristic attempts to minimize the maximum difference between $M_\epsilon(t)/t$ and the 'normalized' leaky bucket $(\sigma_2 + \rho_2 t)/t$. Since we are interested in worst-case bounds, we need to set $\epsilon = 0$.

We evaluate the accuracy of an arbitrary characterization function A^* as follows. We assume that all traffic has the same traffic characterization A^* and identical delay bounds, and we compute the maximum number of admissible streams for all delay bounds as before. Since we wish to evaluate the ability of a particular characterization function to approximate the empirical envelope, we plot the ratio of the number of admissible streams using A^* to the number obtained using the empirical envelope E^*, all as a function of the delay bound. In particular, for a given function A^* we plot:

$$\text{Utilization Ratio}(A^*, d) = \frac{\#\text{ admissible streams with } A^* \text{ at delay bound } d}{\#\text{ admissible streams with } E^* \text{ at delay bound } d} \tag{11}$$

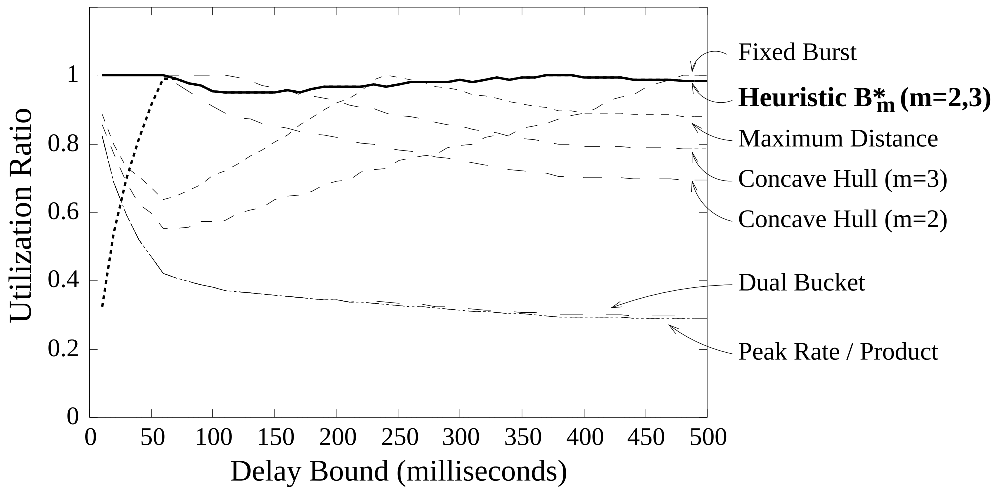
Since no characterizations A^* will admit more streams than the empirical envelope, the metric in (11) allows us to determine how closely a particular characterization approximates the empirical envelope. For example, a traffic characterization A^* that admits the same number of streams as the empirical envelope would result in a constant curve Utilization Ratio$(A^*, d) = 1$.

Figures 7(a) and 7(b) show the utilization ratios of *Park* and *News* streams, respectively, for the entire suite of traffic characterizations described above.

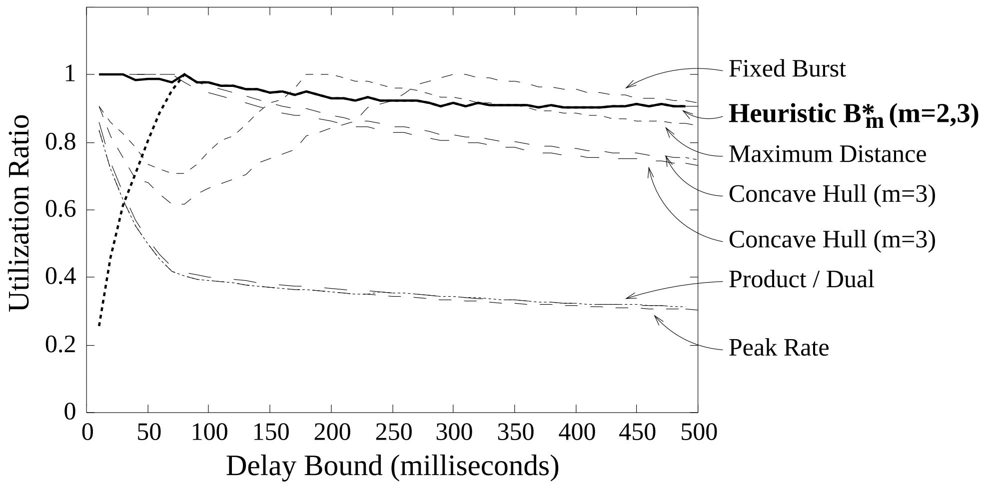
The results for our heuristic characterization are shown in Figure 7 as thick dashed and solid lines for $m = 2$ and and $m = 3$ (σ, ρ) pairs, respectively. For two (σ, ρ) pairs, note that our heuristic yields a poor utilization for small delay bounds, while it is superior to other characterizations for most delay bounds greater than 50 ms. This poor utilization at smaller delay bounds is due to the fact that our heuristic does not select a (σ, ρ) pair with $\rho = \rho_{peak}$. With three (σ, ρ) pairs, our heuristic achieves a utilization ratio of over 95% and 91% for all delay bounds in the *Park* and *News* sequences, respectively.

Notice the poor performance of the three characterizations *Peak*, *Dual Bucket*, *Product* in both graphs. Recall that the Dual Bucket scheme, which selects one bucket for the peak rate and one bucket for the average rate, is the scheme proposed for policing ATM traffic [1]. While a peak-rate characterization yields relatively high utilizations for small delay bounds, it results in a utilization ratio of less than 40% for delay bounds greater than 60 ms for these video sequences. The additional leaky bucket employed in *Dual Bucket* and *Product* does not yield significant utilization gains.

The traffic characterization obtained with our heuristic method for $m = 3$ leaky buckets is clearly the best characterization under consideration. For $m = 2$ leaky buckets our method provides the best results if the delay bounds are large ($d \geq 100$ msec). Since our characterization method can compute the parameters of the characterization in a fraction of the time needed by most other schemes, we conclude that the newly proposed method has the best overall performance.



(a) *Park*



(b) *News*

Figure 7: Evaluation of characterization schemes.

6 Case Study: VBR Service with Deterministic Renegotiation

Dynamic resource allocation schemes are motivated by studies showing that correlations of VBR video traffic occur over long time scales due to the extended duration of scenes [12, 20, 21]. By renegotiating the traffic characterization, for example, after each scene change, one can more accurately specify the traffic on a stream, resulting in a tighter characterization and hence higher network utilization. Since a renegotiation scheme requires multiple traffic characterizations for a single stream, a fast traffic characterization scheme such as the one described in this paper is needed to renegotiate traffic parameters.

In this section we present and evaluate a new approach to dynamic resource allocation schemes, called *deterministic renegotiation*, that applies the traffic characterization method described in the previous sections.

6.1 Renegotiation of Traffic Characterizations

Most renegotiation schemes that have been proposed attempt to renegotiate the traffic characterization of a stream whenever its long-term rate changes significantly [4, 14, 35]. Chong et. al. address the problem of predicting the rate changes of a live video source [4]. They consider both a recursive least-square method and an neural network approach for the prediction. In [14], Grossglauer et. al. propose a *Renegotiated Constant Bit Rate* (RCBR) scheme for both stored and live video which adds renegotiation and buffer monitoring to a static CBR service. They present algorithms for partitioning a video sequence into segments based on cost functions for both bandwidth allocation and number of renegotiations. Zhang and Knightly study a renegotiated VBR service for both stored and live video in [35]. Their algorithm for stored video proceeds by identifying the worst-case segment of the video sequence, characterizing this worst-case segment, and then repeatedly applying the procedure on the remaining video sequence after this worst-case segment is removed.

Although the above renegotiation schemes were shown to increase network utilization significantly, they all share a common drawback. Since these schemes calculate a traffic characterization independently for each video segment, it is possible that a stream needs to increase its resource allocation even if sufficient resources are not available. In such a scenario, the renegotiation requests cannot be accommodated, and either the video quality or the QoS must be compromised, resulting in a violation of the worst-case guarantees in a deterministic service.

In contrast, the deterministic renegotiation scheme proposed here ensures that the traffic characterization for a stream does not increase. That is, renegotiation always involves a release of network resources. In this case, all renegotiation requests can be satisfied and deterministic QoS guarantees are always maintained.

6.2 Deterministic Renegotiation

We assume that the characterization functions for a VBR video stream are newly negotiated at the renegotiation instants $\tau_0, \tau_1, \ldots, \tau_u$, where $\tau_i < \tau_j$ if $i < j$. Let $A^*_{\tau_i}$ denote the characterization function that is negotiated at time τ_i. Characterization functions $A^*_{\tau_i}$ must satisfy two criteria in order to qualify for a deterministic renegotiation scheme. First, the traffic characterization $A^*_{\tau_i}$ must provide a bound on the worst-case traffic for the remainder of the video sequence, that is, for all i:

$$A^*_{\tau_i}(t) \geq A[\tau_i + \tau, \tau_i + \tau + t] \qquad \forall \tau, t \geq 0 \tag{12}$$

Second, to ensure that all renegotiation requests can be satisfied, a newly-computed traffic characterization may not request additional resources, that is, we enforce that for all $\tau_i < \tau_j$:

$$A^*_{\tau_i}(t) \geq A^*_{\tau_j}(t) \qquad \text{for all } t \geq 0 \tag{13}$$

The condition in Equation (12) ensures that any function $A^*_{\tau_i}$ is a viable characterization function, while equation (13) guarantees that the renegotiation requests can always be satisfied. To show that a set of traffic characterizations $\{A^*_{\tau_i}\}$ can be used in a deterministic renegotiation scheme, it is sufficient to show that Equations (12) and (13) are satisfied.

Consider the class of characterization functions $\{E^*_{\tau_i}\}$, where $E^*_{\tau_i}$ defines the empirical envelope of the sequence A for all times $t \geq \tau_i$:

$$E^*_{\tau_i}(t) = \sup_{\tau \geq 0} A[\tau_i + \tau, \tau_i + \tau + t] \qquad \forall t \geq \tau_i \tag{14}$$

If a renegotiation occurs τ_i time units into a video sequence, the resource allocation can be calculated according to $E^*_{\tau_i}$. However, the functions $E^*_{\tau_i}$ are similar to the empirical envelope E^* in that they employ a large number of parameters that are expensive to compute. In the next section we show how to apply our fast traffic characterization method to approximate these functions $\{E^*_{\tau_i}\}$.

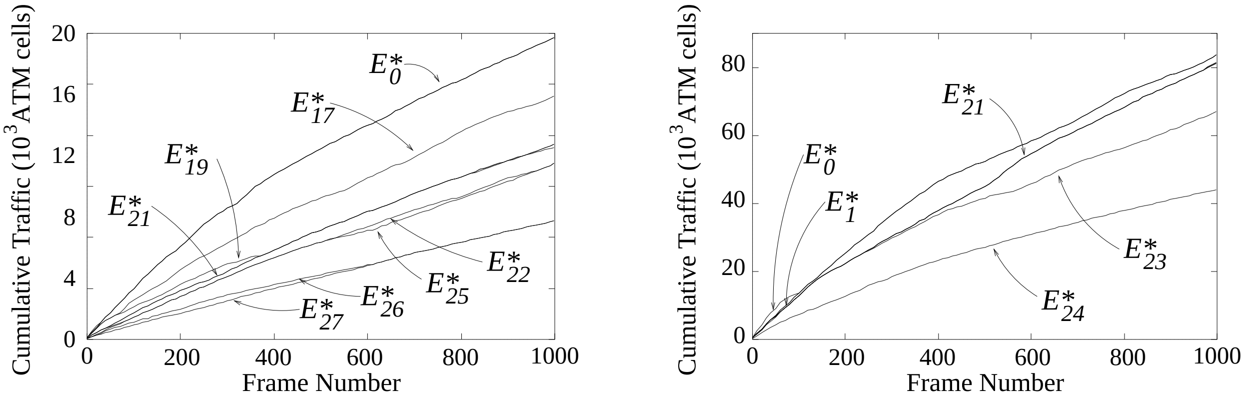
To illustrate the potential gain of deterministic renegotiation we show for the *Park* and the *News* video sequence, how the traffic characterization changes as it is renegotiated. We consider the characterizations $\{E^*_{\tau_i}\}$, where $\tau_i = i$ minutes for $i = 0, \ldots T - 1$ and T is the length of the movie in minutes. Since the *Park* and *News* sequences are 28 and 25 minutes long, respectively, we consider 28 different traffic characterizations for *Park* and 25 for *News*. The resulting functions are depicted in Figure 8. In the figures, we only plot traffic characterizations E^*_i that are visibly smaller than all traffic characterizations E^*_j with $j < i$.

Figures 8(a) and 8(b) show that the decrease of $\{E^*_{\tau_i}\}$ for increasing values of i can be substantial. The deterministic renegotiation scheme exploits this decrease of the functions in $\{E^*_{\tau_i}\}$. Thus, network capacity becomes available that can be utilized by other connections.

6.3 Application of the Fast Video Characterization Method

At each renegotiation instant τ_i we apply our characterization method by performing the following steps:

1. Calculate R^*_{k,τ_i}, the repetition extrapolation of the first k parameters of $E^*_{\tau_i}$.



(a) *Park* (b) *News*

Figure 8: Characterization functions $E^*_{\tau_i}$.

2. Determine the concave hull $\mathcal{H}R^*_{k,\tau_i} = B^*_{\tau_i,n}$.

3. Apply Algorithm 1 in Figure 5 to obtain a function $B^*_{\tau_i,m}$ with $m < n$.

It is easy to verify that each $\mathcal{H}R^*_{k,\tau_i}$ satisfies Equation (12) and (13). However, since the heuristic algorithm as presented determines a function $B^*_{\tau_i,m}$ based only on $\mathcal{H}R^*_{n,\tau_i}$, and independent of the previous approximation $B^*_{\tau_{i-1},m}$, it is possible that the algorithm in Figure 5 selects an approximation with $B^*_{\tau_i,m} > B^*_{\tau_{i-1},m}$ for some values of t. Thus, the condition in Equation (13) does not necessarily hold. Therefore, we suppress renegotiation attempts at times τ_i whenever $B^*_{\tau_{i-1},m}(t) < B^*_{\tau_i,m}(t)$ for some t.

6.4 Empirical Examples

We present examples based on MPEG video sequences to demonstrate the impact of renegotiation on the achievable network utilization. We attempt to find answer the following questions:

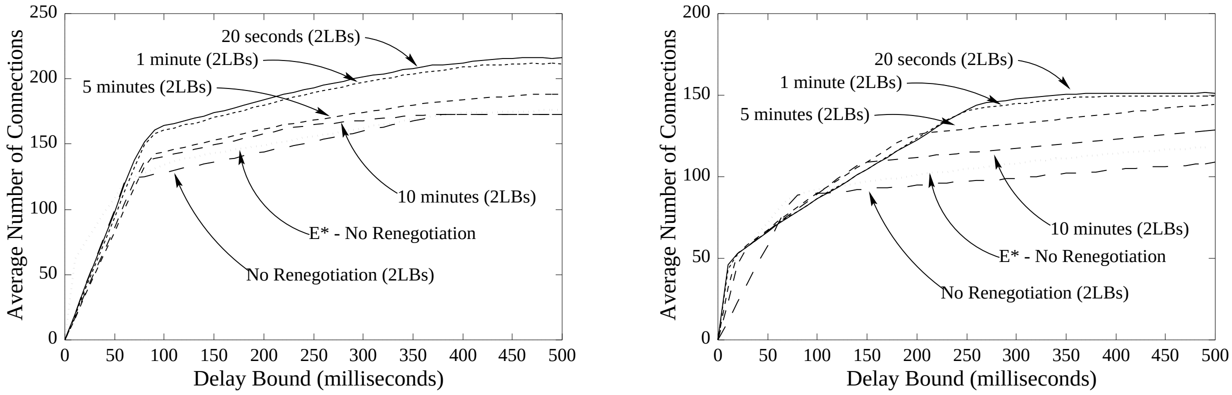
- What is the gain in utilization achievable with a deterministic negotiation scheme?

- What is the frequency of renegotiation needed to obtain a gain in network utilization?

We do not address the question of optimally selecting the renegotiation instants. In our experiments, all renegotiation instants are equally spaced over the length of a video sequence.

As in previous sections, we consider a single 155 Mbps FCFS multiplexer. The multiplexer is loaded with the MPEG streams *Park* and *News*, all with the same delay bound d. To evaluate the performance gain attainable with deterministic renegotiation, we assume that the starting time of video streams are uniformly distributed over the duration of the sequence.

Starting with an empty multiplexer, we add streams to the multiplexer, where each stream has a randomly selected starting time. We continue adding streams as long as the admission control test from Equation (9) is satisfied, that is, as long as all streams are guaranteed a worst-case delay bound of d. We record the maximum number of admissible streams. This process is repeated 1000



(a) *Park.* (b) *News.*

Figure 9: Utilization comparison for different renegotiation periods.

times for each delay bound, and we plot the average number of admissible streams as a function of the delay bound. The above experimental setup allows us to measure the average, as opposed to the worst-case, utilization gain of the renegotiation scheme.

The results of our experiment are shown in Figure 9 where we depict the number of admissible streams for both the *Park* and *News* sequences for several renegotiation periods. The maximum number of admissible streams are plotted as a function of the delay bound. Figures 9(a) and 9(b) show results obtained using our traffic characterization method from Sections 3 and 4. We plot curves corresponding to renegotiation periods of 20 seconds as well as 1, 5, and 10 minutes.

We see in Figure 9 that the renegotiation period significantly impacts the number of admissible streams. In Figure 9(a), note that the number of admissible *Park* streams increases by 20–30% for delay bounds larger than 50 ms if the renegotiation period is less than 1 minute. For the longer renegotiation periods, i.e., 5 minutes and 10 minutes, renegotiation provides gains of about 10%. For the *News* sequence, we see in Figure 9(b) that even infrequent renegotiation results in considerable utilization gains.

7 Concluding Remarks

The traffic characterization used for VBR video streams has a significant impact on the number of admissible streams in a multimedia network with a deterministic service. We presented a method for traffic characterization based on the empirical envelope of a video sequence that uses a two-step process. We first approximated the empirical envelope with a characterization that can be policed by some number of leaky buckets, and we then determined the final characterization that can be policed by a small, fixed number of leaky buckets. Using two MPEG-compressed video sequences, we demonstrated that our characterization method determines accurate characterizations that admit a large number of streams. With the caveat that our experimental evaluation is based on only a small number of video traces, the experiments in this paper gave the following insights:

- For the MPEG video sequences considered in the paper, we saw that as few as 200 parameters of the empirical envelope out of a total of 40,000 are sufficient to yield a characterization that admits the same number of streams as the empirical envelope. This observation suggests that the relevant information of an MPEG sequence is contained in a small segment of the empirical envelope.

- Using our heuristic algorithm, three leaky buckets were shown to be sufficient to admit nearly the same number of streams as the empirical envelope. Based on the good performance of our heuristic algorithm, it may not be worthwhile to investigate more complex algorithms for video characterization.

- Our evaluation in Section 5 showed that a dual leaky bucket scheme with a parameter selection as defined by the ATM Forum [1], i.e., one bucket for the peak rate and one bucket for the average rate, yields poor network utilization. However, the poor performance is not due to the fact that only two leaky buckets are used, but rather to a poor selection of leaky bucket parameters. Using a better heuristic algorithm such as the one developed in this paper, one can achieve markedly higher performance.

- In [34], the numerical examples indicated that a large number of leaky bucket pairs were needed to approximate the empirical envelope. Using our method, the number of leaky buckets needed to achieve performance similar to the envelope is small. This discrepancy can be explained as follows. First, the heuristic algorithm presented here is superior to the concave hull approach from [34]. Second, we note that the examples in this paper only consider delay bounds up to 500 ms, while the examples in [34] consider delay bounds up to 2000 ms. For a larger delay bound range, additional leaky buckets are necessary to closely approximate the empirical envelope.

- The deterministic resource renegotiation scheme that we describe is distinguished from previous approaches in that it is appropriate for services that provide constant video quality and deterministic QoS guarantees. The experimental data suggests that the expected utilization gain from deterministic renegotiation is 20–35%.

References

[1] ATM Forum, ATM Forum Traffic Management Specification Version 4.0, October, 1995.

[2] C.-S. Chang. Stability, Queue Length, and Delay of Deterministic and Stochastic Queueing Networks. *IEEE Transactions on Automatic Control*, 39(5):913–931, May 1994.

[3] S. Chong and S. Li. (σ, ρ)-Characterization Based Connection Control for Guaranteed Services in High Speed Networks. In *Proc. IEEE Infocom '95*, pages 835–844, 1995.

[4] S. Chong, S. Li, and J. Ghosh. Predictive Dynamic Bandwidth Allocation for Efficient Transport of Real-Time VBR Video over ATM. *IEEE Journal on Selected Areas in Communication*, 13(1):12–23, January 1995.

[5] D. D. Clark, S. Shenker, and L. Zhang. Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanisms. In *Proc. Sigcomm '92*, pages 14–26, August 1992.

[6] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms.* MIT Press, 1992.

[7] R. L. Cruz. A Calculus for Network Delay, Part I: Network Elements in Isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, January 1991.

[8] R. L. Cruz. A Calculus for Network Delay, Part II: Network Analysis. *IEEE Transactions on Information Theory*, 37(1):132–141, January 1991.

[9] A. Elwalid, D. Mitra, and R. Wentworth. A New Approach for Allocating Buffers and Bandwidth to Heterogeneous, Regulated Traffic in an ATM Node. *IEEE Journal on Selected Areas in Communications*, 13(6):1115–1127, August 1995.

[10] D. Ferrari and D. C. 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.

[11] V. Frost and B. Melamed. Traffic Modelling for Telecommunications Networks. 32(3):70–81, March 1994.

[12] M. W. Garrett and W. Willinger. Analysis, Modeling and Generation of Self-Similar VBR Video Traffic. In *Proc. ACM Sigcomm '94*, pages 269–280, August 1994.

[13] L. Georgiadis, R. Guerin, V. Peris, and K. N. Sivarajan. Efficient Network QoS Provisioning Based on per Node Traffic Shaping. In *Proc. IEEE Infocom '96*, pages 102–110, March 1996.

[14] M. Grossglauer, S. Keshav, and D. Tse. RCBR: A Simple and Efficient Service for Multiple Time-Scale Traffic. In *Proc. ACM Sigcomm '95*, pages 219–230, August 1995.

[15] F. Guillemin, C. Rosenberg, and J. Mignault. On Characterizing an ATM Source via the Sustainable Cell Rate Traffic Descriptor. In *Proc. IEEE Infocom '95*, pages 1129–1136, April 1995.

[16] E. Knightly and H. Zhang. Traffic characterization and switch utilization using deterministic bounding interval dependent traffic models. In *Proc. IEEE Infocom '95*, pages 1137–1145, April 1995.

[17] E. W. Knightly. H-BIND: A New Approach to Providing Statistical Performance Guarantees to VBR Traffic. In *Proc. IEEE Infocom '96*, pages 1091–1099, March 1996.

[18] E. W. Knightly and P. Rossaro. Effects of Smoothing on End-to-End Performance Guarantees for VBR Video. In *Proc. International Symposium on Multimedia Communications and Video Coding*, October 1995.

[19] S. S. Lam, S. Chow, and D. K. Y. Yau. An Algorithm for Lossless Smoothing of MPEG Video. In *Proc. ACM Sigcomm '94*, pages 281–293, August 1994.

[20] A. Lazar, G. Pacifici, and D. Pendarakis. Modeling Video Sources for Real-Time Scheduling. In *Proc. IEEE Globecom '93*, pages 835–839, December 1993.

[21] S. Q. Li, S. Chong, C. Hwang, and X. Zhao. Link Capacity Allocation and Network Control by Filtered Input Rate in High Speed Networks. In *Proc. IEEE Globecom '93*, pages 744–750, December 1993.

[22] J. Liebeherr, D. E. Wrege, and Domenico Ferrari. Exact Admission Control in Networks with Bounded Delay Services. *IEEE/ACM Transactions on Networking*, 4(6):885 – 901, December 1996.

[23] S. H. Low and P. Varaiya. Burst Reducing Servers in ATM Networks. *Queueing Systems*, 20:61–84, 1995.

[24] J. M. McManus and K. W. Ross. Video on Demand over ATM: Constant-Rate Transmission and Transport. To appear: *IEEE Journal on Selected Areas in Communications.*

[25] P. Pancha and M. El Zarki. Leaky Bucket Access Control for VBR MPEG Video. In *Proc. IEEE Infocom*, pages 796–803, April 1995.

[26] A. K. Parekh and R. G. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case. *IEEE/ACM Transactions on Networking*, 1(3):344–357, June 1993.

[27] E. P. Rathgeb. Modeling and Performance Comparison of Policing Mechanisms for ATM Networks. *IEEE Journal on Selected Areas in Communications*, 9(4):325–334, April 1991.

[28] E. P. Rathgeb. Policing of Realistic VBR Video Traffic in an ATM Network. *International Journal of Digital and Analog Communications Systems*, 6:213–226, 1993.

[29] A. R. Reibman and A. W. Berger. Traffic Descriptors for VBR Video Teleconferencing Over ATM Networks. *IEEE/ACM Transactions on Networking*, 3(3):329–339, June 1995.

[30] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. Technical Report 101, Institute of Computer Science, University of Wurzburg, February 1995. The traces used in this paper are available via anonymous ftp from the site ftp-info3.informatik.uni-wuerzburg.de in the directory /pub/MPEG/.

[31] C. Rosenberg and B. Lague. A Heuristic Framework for Source Policing in ATM Networks. *IEEE/ACM Transactions on Networking*, 2(4):387–397, August 1994.

[32] J. Salehi, Z. Zhang, J. Kurose, and D. Towsley. Supporting Stored Video: Reducing Rate Variability and End-to-End Resource Requirements through Optimal Smoothing. To appear: *Proc. ACM Sigmetrics '96*, May 1996.

[33] J. S. Turner. New Directions in Communications (or Which Way to the Information Age?). *IEEE Communications Magazine*, 25(8):8–15, October 1986.

[34] D. E. Wrege, E. W. Knightly, H. Zhang, and J. Liebeherr. Deterministic Delay Bounds for VBR Video in Packet-Switching Networks: Fundamental Limits and Tradeoffs. *IEEE/ACM Transactions on Networking*, 4(3):352–362, June 1996.

[35] H. Zhang and E. Knightly. A New Approach to Support Delay-Sensitive VBR Video in Packet-Switched Networks. In *Proc. 5th Intl. Workshop on Network Operating System Support for Digital Audio and Video (NOSSDAV)*, pages 275–286, April 1995.

A Evaluation of Cost Function

We have evaluated a number of candidate cost functions of the following general form:

$$\Delta(B_n^*, B_m^*) = \int_0^T \frac{(B_m^*(t) - B_n^*(t))^\alpha}{(t+1)^\beta B_n^*(t)^\gamma}\, dt, \tag{15}$$

where T determines the time interval over which the cost Δ is selected, and α, β, and γ are weights that determine the shape of the cost function. For example, a selection of $(2, 0, 0)$ for the (α, β, γ)-tuple minimizes the squared difference of B_m^* and B_n^*.

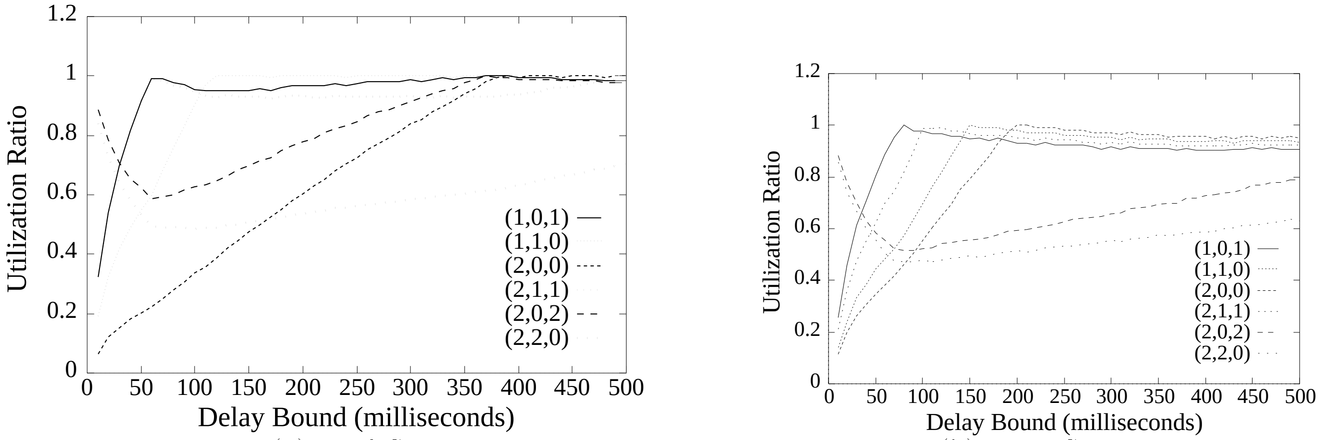
We evaluated the the following choices for parameters (α, β, γ):

$$\begin{array}{ccc} (1,0,1) & (1,1,0) & (2,0,0) \\ (2,1,1) & (2,0,2) & (2,2,0). \end{array}$$

The evaluation was done in a similar fashion as the experiments in Section 3. We consider the MPEG traces *News* and *Park*, that are transmitted over a 155 Mbps FCFS multiplexer. As input function function B_n^* we chose the concave hull $\mathcal{H}R_{200}^*$, thus, T (15) is set to frame times. We always ran the heuristic algorithm to produce m-piecewise linear functions B_m^* with $m = 2$ or $m = 3$. We evaluated the cost functions using the utilization ratio defined in Equation (11) as performance parameter. The results of the evaluation are shown in Figures 10 and 11.

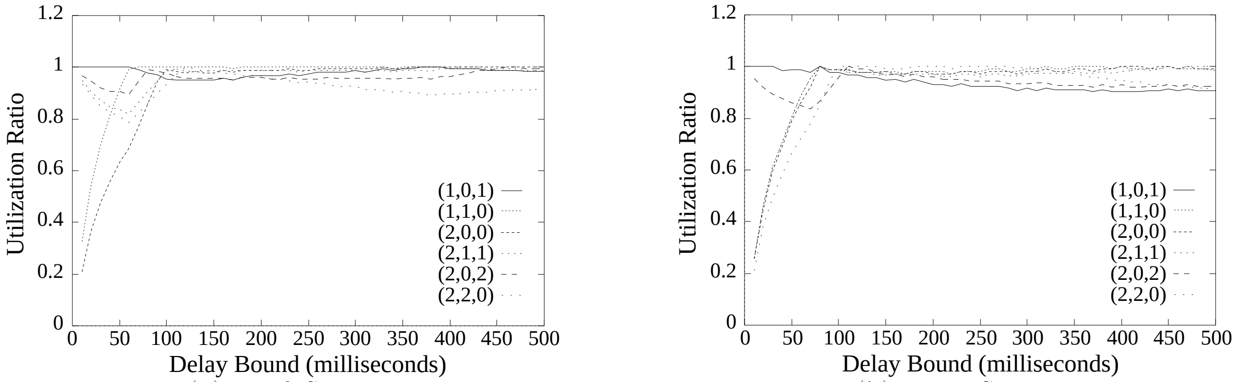
Comparing Figure 10 and Figure 11, it is very noticeable that $m = 3$ leaky buckets yield much better results than $m = 2$ leaky buckets. With two leaky buckets as in Figures 10(a) and (b), the $(1, 0, 1)$ and $(2, 1, 1)$ characterizations have a high utilization ratio for delay bounds greater than 50 ms. With three leaky buckets as in Figures 11(a) and (b), four of the characterizations have a utilization ratio of higher than 80% over the entire delay bound range $[0, 500\text{ms}]$. For $(1, 0, 1)$ we obtain a utilization ratio $\geq 90\%$ over the entire delay bound range.

Based on these experiments, the $(1, 0, 1)$ cost function (shown in all graphs as a solid line) appears to be the best choice. In all four graphs, observe that the utilization ratio for the $(1, 0, 1)$ characterization is above 90% for a large range of delay bounds.



(a) *Park* Sequence. (b) *News* Sequence.

Figure 10: Evaluation of cost functions for $m = 2$ leaky buckets.



(a) *Park* Sequence. (b) *News* Sequence.

Figure 11: Evaluation of cost functions for $m = 3$ leaky buckets.