Exploiting the Temporal Structure of MPEG Video for the Reduction of Bandwidth Requirements*

Marwan Krunz†and Satish K. Tripathi
Institute for Advanced Computer Studies
Department of Computer Science
University of Maryland
College Park, MD 20742
{krunz,tripathi}@cs.umd.edu

Abstract

We present a novel bandwidth allocation scheme for transporting variable-bit-rate MPEG traffic from a video server. Using time-varying envelopes to characterize the traffic, this scheme achieves significant bandwidth gain, via statistical multiplexing, while supporting stringent, deterministic QoS guarantees. The gain can be maximized by allowing the server to appropriately schedule the starting times of video sources, at the expense of some negligible startup delay. For homogeneous streams, we give the optimal schedule that results in the minimum allocated bandwidth. A sub-optimal schedule is given in the heterogeneous case, which is shown to be asymptotically optimal. Efficient online procedures for bandwidth computation are provided. Numerical examples based on traces of MPEG-coded movies are used to demonstrate the benefits of our allocation strategy.

1 Introduction

A large volume of traffic in future BISDN/ATM networks will be generated by video applications. Providing network support for video traffic without underutilizing bandwidth resources is a major challenge, particularly for variable-bit-rate (VBR) video. Although ATM networks are expected to provide a VBR network service, it is unlikely that such a service will support deterministic quality-of-service (QoS) guarantees. Such guarantees can be supported using a constant-bit-rate (CBR) network service.

Efficient bandwidth allocation for VBR-coded video transported using a CBR network service requires reducing the variability of the bit rate. In principle, the variability can be reduced by means of temporal averaging (or smoothing) on a stream-by-stream basis or spatial averaging (or aggregation) by means of statistical multiplexing (**SM**). Temporal smoothing has been used for both real-time [6, 3, 8] and stored video [1, 7, 10]. Our primary interest here is in the transport of stored video in video-on-demand (VOD) systems. Temporal smoothing of stored video takes the form of a "work-ahead" approach, where frames are sent ahead of their playback time. While smoothing has several attractive features, it also has some drawbacks, including excessive buffers at the set-top box and the need for an exact knowledge of the end-to-end network delay to avoid buffer overflow and underflow at the client side.

As an alternative to temporal smoothing, we investigate the use of SM to reduce the variability in video, *while providing stringent deterministic guarantees.* In principle, supporting deterministic guarantees necessitates the use of deterministic traffic models. One such model was suggested by Knightly et al. [4], where a video stream is characterized by a time-invariant traffic envelope. Using a time-varying version of this envelope, we provide an efficient allocation scheme for MPEG video that achieves significant multiplexing gain while guaranteeing zero loss rate and small bounded delay. Although the scheme is primarily tailored for archived video, it can also be used for real-

* This research was supported by NSF grant # CCR 9318933, and by a Shared University Research IBM grant.

†Currently with the Dept. of ECE, University of Arizona.

time MPEG-coded video if the envelope can be conservatively estimated, policed based on some declared values, or enforced by the encoder. We investigate the performance of the allocation scheme when used at a video server that distributes precoded video movies on demand. Two possible scenarios are investigated. In the first scenario, requests are served immediately (the *non-aligned case*), or delayed by no more than the duration of a frame (the *aligned case*). In the second scenario, a request can be delayed by no more than a GOP period. This gives the server the flexibility to efficiently schedule video streams at the multiplexer to maximize the multiplexing gain. We provide algorithms for online computation of allocated bandwidth under both scenarios. Even without scheduling (first scenario), some gain can still be achieved depending on the "arrangement" of the multiplexed streams (a measure of the lags between their GOPs). When stream scheduling is performed, we provide the optimal schedule for homogeneous streams, which achieves the minimum allocated bandwidth. For heterogeneous streams, we provide a suboptimal schedule, which is shown to be asymptotically optimal.

The paper is structured as follows. Section 2 describes the allocation of bandwidth based on time-varying envelopes. Optimal and suboptimal scheduling of video streams at the server are are provided in Section 3. Section 4 gives some numerical examples. We summarize our findings in Section 5.

2 SM with Time-Varying Envelopes

2.1 Traffic-Envelope Model

A standard MPEG encoder generates three types of compressed frames: Intra-coded (I), Predictive (P), and Bidirectional (B). In general, I frames are larger than P frames which, in turn, are larger than B frames. To simplify the hardware design, many MPEG encoders use a fixed Group-of-Pictures (GOP) pattern when compressing a video sequence. Moreover, these fixed GOP patterns are often "regular" in the sense that the number of successive B frames between two reference frames is constant. In this paper, we assume fixed and regular GOP patterns, and use that to allocate resources efficiently. A regular GOP pattern is specified by two parameters: N (the *I-to-I* frame distance) and M (the *P-to-P* frame distance), where N is a multiple of M.

We model a video stream by the following traffic envelope: For the ith video stream, s_i, the envelope is the time-varying periodic function $\overline{b}_i(t)$ that is parameterized by the 5-tuple $\boldsymbol{E}_i = \left(I^{(i)}_{max},\ P^{(i)}_{max},\ B^{(i)}_{max},\ N^{(i)},\ M^{(i)}\right)$. Here, $I^{(i)}_{max}$ is the largest frame in the sequence (typically an I frame), $P^{(i)}_{max}$ is the largest frame among P and B frames (typically a P frame), and $B^{(i)}_{max}$ is the largest B frame (we assume that frames are generated at a constant frame rate, their sizes are given in cells, and cells are evenly distributed over a frame period). The last two parameters describe the GOP pattern of s_i. By definition $I^{(i)}_{max} \geq P^{(i)}_{max} \geq B^{(i)}_{max}$. Our model is similar to the D-BIND model [4]. However, the D-BIND model provides a time-invariant bound on the cumulative arrivals, rather than a time-varying bound on the arrival rate. An example of $\overline{b}_i(t)$ is shown in Figure 1.

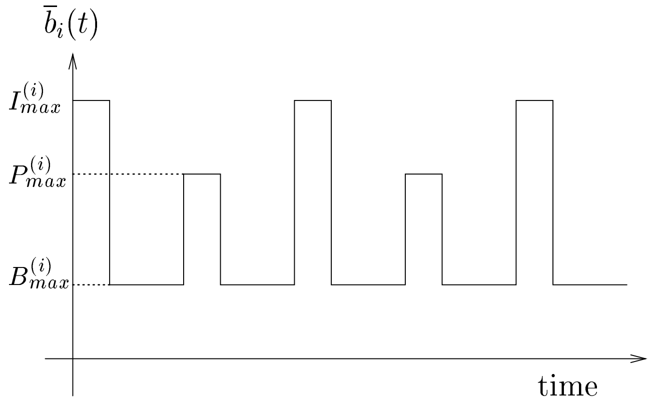

Figure 1: Traffic envelope with $N^{(i)} = 6$ and $M^{(i)} = 3$.

2.2 Effective Bandwidth with Deterministic Guarantees

Consider n video streams, $s_1, \ldots, s_n$ that require stringent QoS of no losses and small, bounded delay. When transported over an ATM network, such requirements are often met by allocating each source its peak rate ($I^{(i)}_{max}$). Let $\overline{b}_i(t)$ be the traffic envelope for s_i, parameterized by $\boldsymbol{E}_i$ (assume, for now, that $N^{(i)} = N$ for all i). Suppose that the first frame of s_i arrives at the multiplexer at time t_i, with $t_1 \stackrel{\triangle}{=} 0$. The lag in frame durations between a GOP of s_1 and the following GOP of s_i is given by $u_i = t_i \bmod N$. The temporal relationships between the GOPs of the n streams are completely specified by $\boldsymbol{u} = (u_2, u_3, \ldots, u_n)$, which we refer to as the *arrangement*. Let $\overline{b}_{tot}(t)$ be the traffic envelope from the superposition of the n streams; $\overline{b}_{tot}(t) = \sum_i \overline{b}_i(t - u_i)$. Note that $\overline{b}_{tot}(t)$ is periodic with period N. We define the *effective bandwidth* for n multiplexed streams with arrangement $\boldsymbol{u}$ by:

$$C(\boldsymbol{u},n) \triangleq \frac{1}{n}\max_{t\geq 0}\overline{b}_{tot}(t) = \frac{1}{n}\max_{t\geq 0}\left(\sum_{i=1}^{n}\overline{b}_i(t-u_i)\right) \quad (1)$$

Since $nC(\boldsymbol{u},n)$ bounds the aggregate bit rate at all times, allocating bandwidth of $C(\boldsymbol{u},n)$ per source ensures that the instantaneous aggregate input rate never exceeds the output rate. A small buffer of n cells is needed in case cells from several sources arrive simultaneously. With this buffer, zero loss rate and a maximum delay of $1/C(\boldsymbol{u},n)$ are guaranteed.

Because of the periodicity of $\overline{b}_{tot}(t)$, it is sufficient to take the maximum in (1) over an interval of N time units. When $N^{(i)}$ varies with i, (1) is still valid with $\widetilde{N}$ replacing N, where $\widetilde{N}$ is the least common multiple of $\{N^{(1)}, N^{(2)}, \ldots, N^{(n)}\}$. Given that $I^{(i)}_{max} \geq P^{(i)}_{max} \geq B^{(i)}_{max}$, it is easy to see that $nC(\boldsymbol{u},n) < \sum_i I^{(i)}_{max}$ for most values of $\boldsymbol{u}$. The extreme case is when $\boldsymbol{u} = (0,0,\ldots,0)$ (e.g., when all streams start simultaneously), which results in $nC(\boldsymbol{u},n) = \sum_i I^{(i)}_{max}$.

2.3 Online Bandwidth Computation

The utilized bandwidth at the server must be updated dynamically upon the addition of a new video stream or the removal of an ongoing one. We now consider the situation when video requests arriving at the server are promptly served (i.e., without scheduling delay), given that resources are available. Allocation can then be based on $C(\boldsymbol{u},n)$. We consider two cases.

2.3.1 Aligned Boundaries Case

Suppose that $\boldsymbol{u}$ takes only integer values in $\{0,1,\ldots,\widetilde{N}-1\}$, which means that frames' boundaries in different sources are aligned in time. Alignment of frame boundaries can be enforced by delaying the servicing of a request by no more than a frame period. Then for a fixed integer k, $\overline{b}_{tot}(t)$ is constant for all $t \in (k, k+1)$ (the time unit is a frame period). Because of its periodicity, $\overline{b}_{tot}(t)$ is specified by $\widetilde{N}$ values. Hence, computation of $C(\boldsymbol{u},n)$ requires maintaining only the values of the traffic envelopes for the first $\widetilde{N}$ slots (from 0 to $\widetilde{N}-1$). We refer to such slots as *phases*. Let $\overline{b}_{i,j}$ be the value of $\overline{b}_i(t)$ during phase j. Thus,

$$\overline{b}_{i,j} = \overline{b}_i(\tau - u_i) \ \ \forall \tau \in (j, j+1) \quad (2)$$

To compute $C(\boldsymbol{u},n)$, the server maintains a matrix $M = [m_{ij}]$ of size $n \times \widetilde{N}$. Each video stream is associated with one row in the table. For $i = 1,\ldots,n$, and $j = 1,\ldots,\widetilde{N}$, $m_{ij} = \overline{b}_{i,j-1}$. In addition, the node maintains a row vector $V = \left[v_1,\ldots,v_{\widetilde{N}}\right]$, where

$$v_j = \sum_{i=1}^{n} m_{ij} \ \ \forall \ j \quad (3)$$

v_j gives the value of $\overline{b}_{tot}(t)$ during phase $j-1$. Now, $C(\boldsymbol{u},n)$ is simply given by:

$$C(\boldsymbol{u},n) = \frac{1}{n}\max_{1\leq j\leq \widetilde{N}} v_j \quad (4)$$

Upon the arrival of the $(n+1)$th stream, a row is added to M based on $\overline{b}_{n+1}(t)$ and u_{n+1}. For heterogeneous streams with different $N^{(i)}$ values, the updating of M can be simplified by choosing $\widetilde{N}$ based on all anticipated values of $N^{(i)}$ (which are few in practice). Thus, the number of columns of M is kept constant and only the rows are added or deleted during the updating process. The effective bandwidth is recomputed by updating V (using $v_j := v_j + m_{n+1,j}$) and applying (4) with $n+1$ replacing n. A similar procedure is used to update $C(\boldsymbol{u},n)$ when an ongoing connection is terminated. Clearly, very few operations are needed to recompute the effective bandwidth upon adding/dropping a video stream.

2.3.2 Non-Aligned Boundaries Case

Consider the general case in which $u_2, u_3, \ldots$, can take any real values in $[0, \widetilde{N})$. Maintaining a table of $\widetilde{N}$ columns as in the previous case is not sufficient to compute $C(\boldsymbol{u},n)$ since $\overline{b}_{tot}(t)$ can take up to $n\widetilde{N}$ different values in a period of $\widetilde{N}$. With n continuously varying, the size of the table and the cost of updating it become impractical for online computations. Instead, we provide an upper bound on the effective bandwidth that can be efficiently updated. A matrix $\widehat{M} = [\widehat{m}_{ij}]$ of dimensions $2n \times \widetilde{N}$ is maintained at the server. An ongoing stream s_i is associated with two adjacent rows of $\widehat{M}$; the $(2i-1)$th and the $(2i)$th rows which contain the the $\widetilde{N}$ values of $\overline{b}_i(t)$ assuming s_i is in phase $\lfloor u_i \rfloor$ and $\lceil u_i \rceil$ mod $\widetilde{N}$, respectively. Hence,

$$\widehat{m}_{ij} = \begin{cases} \overline{b}_{(i+1)/2,j-1} & \text{if } i \text{ is odd} \\ \overline{b}_{i/2,j-2} & \text{if } i \text{ is even} \end{cases} \quad (5)$$

where $\overline{b}_{i,j}$ is now defined by $\overline{b}_{i,j} \triangleq \overline{b}_i\left(\tau - \lfloor u_i \rfloor\right)$ for all $\tau \in (j, j+1)$. In addition to $\widehat{M}$, the node maintains a row vector $\widetilde{V} = \left[\widetilde{v}_1,\ldots,\widetilde{v}_{\widetilde{N}}\right]$, where

$$\widetilde{v}_j = \sum_{i=1}^{n} \max\left\{\widehat{m}_{2i-1,j},\ \widehat{m}_{2i,j}\right\} \ \ \forall \ j \quad (6)$$

$\widetilde{v}_j$ gives a bound on the aggregate bit rate during phase $j-1$. It can be shown that

$$\overline{C}(\boldsymbol{u},n) = \frac{1}{n} \max_{1\le j\le \widetilde{N}} \widetilde{v}_j \ge C(\boldsymbol{u},n) \qquad (7)$$

Upon the arrival of a new video stream to a node with n ongoing streams, two rows are added to $\widehat{M}$ based on $\overline{b}_{n+1}(t)$ and u_{n+1}, and $\widetilde{v}_j$ is updated using

$$\widetilde{v}_j := \widetilde{v}_j + \max\left\{\widehat{m}_{2(n+1)-1,j},\ \widehat{m}_{2(n+1),j}\right\} \quad \forall\, j \qquad (8)$$

The bound on the effective bandwidth is updated using (7) (with $n+1$ replacing n). When an ongoing connection s_i is terminated, $\widetilde{v}_j$ is updated using

$$\widetilde{v}_j := \widetilde{v}_j - \max\left\{\widehat{m}_{2i-1,j},\ \widehat{m}_{2i,j}\right\} \quad \forall\, j \qquad (9)$$

3 Scheduling of Video Streams

Since the effective bandwidth depends on the arrangement of the multiplexed video streams, it is natural to look for the *best* arrangement that results in the minimal effective bandwidth. A best arrangement can be used in a VOD system to provide optimal scheduling of video streams for transmission over the network. The server in a VOD system has some flexibility in controlling the starting times of new connections. This flexibility allows the server to efficiently schedule the transmission of requested movies at the expense of delaying the start of a new stream by no more than a GOP period (1/2 second). Efficient scheduling schemes for video are given in this section. In the homogeneous case (identical envelopes), our scheduling scheme is proven to be optimal. A suboptimal scheme is provided for heterogeneous envelopes.

3.1 Optimal Scheduling of Homogeneous Streams

Suppose that all streams are characterized by the same envelope $\overline{b}(t)$ with parameters $\boldsymbol{E} = (I_{max},\ P_{max},\ B_{max},\ N,\ M)$. This homogeneous case occurs when several copies of the same movie are requested at different instants of time. In addition, it can be enforced by using a slightly conservative common traffic envelope to characterize heterogeneous streams with relatively close but different maximum frame sizes and similar N and M values. Such an envelope is constructed by taking I_{max} to be the largest $I^{(i)}_{max}$ over all i, and similarly for P_{max} and B_{max}. We define the minimal effective bandwidth by:

$$C_{min}(n) = C(\boldsymbol{u}^*,n) \stackrel{\triangle}{=} \min_{u\in\mathcal{U}} C(\boldsymbol{u},n) \qquad (10)$$

where $\mathcal{U}$ is the set of all possible *distinct* arrangements of n streams, and $\boldsymbol{u}^*$ is a *best* arrangement that results in the minimal effective bandwidth. Using combinatorial techniques, it can be shown that the size of the set $\mathcal{U}$ is given by

$$\sum_{i=1}^{m} \binom{N}{i}\binom{n-2}{i-1} \quad \text{where } m = \min\{n-1, N\}$$

which increases rapidly with n. Therefore, obtaining $C_{min}(n)$ from (10) by exhaustive search is computationally prohibitive for moderate and large n. Instead, we give a closed-form expression for $\boldsymbol{u}^*$. We assume, without loss of generality, that frame boundaries are aligned. This assumption is justified by the fact that the effective bandwidth for an arrangement with non-aligned boundaries can be shown to be greater than or equal the effective bandwidth for some arrangement with aligned boundaries. Thus, $\boldsymbol{u}^*$ is necessarily an arrangement with aligned boundaries.

Proposition 1 *A best arrangement of n streams, $n \ge 1$, with identical traffic envelopes is given by:*

$$\boldsymbol{u}^* = (\underbrace{0,1,\ldots,N-1,0,1,\ldots,N-1,\ldots,}_{w\text{times}}0,1,\ldots,n-wN-1)$$

and the minimal effective bandwidth $C_{min}(n) = C(\boldsymbol{u}^*,n)$ is given by:

$$\frac{(w+1)I_{max} + (m-w)P_{max} + (n-1-m)B_{max}}{n} \qquad (11)$$

where

$$w \stackrel{\triangle}{=} \text{largest nonnegative integer } k \text{ that satisfies } n > kN$$

$$m \stackrel{\triangle}{=} \text{largest nonnegative integer } k \text{ that satisfies } n > kM$$

Although the form of $\boldsymbol{u}^*$ is quite intuitive, proving its optimality is not trivial. The proof is given in the next section. Note that $\boldsymbol{u}^*$ is not necessarily unique.

Given that n ongoing streams are scheduled according to $\boldsymbol{u}^*$, a new stream can be added to the existing ones resulting in a *best* arrangement of $(n+1)$ streams without disrupting the original structure of the n streams. When n streams are arranged according to $\boldsymbol{u}^*$ and $n \le N$, the removal of *any* stream will still result in a best arrangement of $n-1$ streams. When $n > N$, only the removal of certain streams preserves the optimality of the arrangement.

As n increases, $C_{min}(n)$ decreases slowly in a non-monotonic manner. The asymptotic value of $C_{min}(n)$

can be obtained by taking the limit in (11) with respect to n. For large n, $w \approx n/N$ and $m \approx n/M$. Thus,

$$C^*_{min} \triangleq \lim_{n\to\infty} C_{min}(n) = (1/N)I_{max} + (1/M - 1/N)P_{max} + (1 - 1/M)B_{max}$$

In fact, this limiting value is reachable when $n = kN$, for $k = 1, 2, 3, \ldots$, implying that the highest gain from multiplexing in the homogeneous case is achieved whenever the number of multiplexed streams is a multiple of N.

3.2 Proof of Optimality

We now prove the optimality of $\boldsymbol{u}^*$. In the homogeneous case with aligned boundaries, $C(\boldsymbol{u}, n)$ can be written as

$$C(\boldsymbol{u}, n) = \frac{n_I I_{max} + n_P P_{max} + (n - n_I - n_P)B_{max}}{n} \tag{12}$$

for some nonnegative integers n_I and n_P. We say that a stream s_i is in phase k where $k = 0, \ldots, N-1$, if $u_i = k$, i.e., s_i sends its I frames during phase k. We use $\bar{b}_{tot,i}$ to refer to $\bar{b}_{tot}(\tau)$ for $\tau \in (i, i+1)$. Let

$r_k \triangleq$ number of streams in phase k

$z_k \triangleq$ number of streams in phases that differ from phase k by a nonzero multiple of M

Proposition 2 *In (12), $n_I \geq 1$ for any arrangement $\boldsymbol{u} = (u_1, \ldots, u_n)$.*

Proof (by contradiction): Suppose that $n_I = 0$. First, consider the case when $n_P = 0$. Then $C(\boldsymbol{u}, n) = nB_{max}/n$. Since $u_1 \triangleq 0$, $r_0 \geq 1$. Thus, during phase 0 the aggregate peak rate $\bar{b}_{tot,0} \geq I_{max} + (n-1)B_{max} > nC(\boldsymbol{u}, n)$, which contradicts the definition of $C(\boldsymbol{u}, n)$. Next, consider the case when $n_P \geq 1$. Let phase k be the phase for which $\bar{b}_{tot,k}/n = C(\boldsymbol{u}, n)$. By our assumption, $r_k = 0$. Since $n_P \geq 1$, there exists at least one stream, say s_j, with phase j such that $|j - k| =$ a multiple of M. During phase j, s_j sends I frames. Also, any other stream that sends P frames during phase k will be sending either I frames or P frames during phase j. Thus, $\bar{b}_{tot,j} > \bar{b}_{tot,k}$, which contradicts the definition of $C(\boldsymbol{u}, n)$. Hence, $n_I \geq 1$. □

To prove the optimality of $\boldsymbol{u}^*$, we first show that $C(\boldsymbol{u}^*, n)$ is given by (11). Then, we show that $C_{min}(n)$ is also given by (11). When n streams are arranged according to $\boldsymbol{u}^*$, there are exactly $m+1$ streams whose phases differ, pairwise, by a nonnegative multiple of M. Among those, $w+1$ streams belong to the same phase (m and w were defined in Proposition 1). It is obvious that $C(\boldsymbol{u}^*, n)$ is obtained from a phase i in which $r_i = w + 1$ and $z_i = m + 1 - (w + 1) = m - w$. Thus, $C(\boldsymbol{u}^*, n)$ is given by (11).

Now consider an arbitrary arrangement $\boldsymbol{u} = (u_1, \ldots, u_n)$. If we can show that $C(\boldsymbol{u}, n)$ satisfies

$$C(\boldsymbol{u}, n) \geq \frac{sI_{max} + lP_{max} + (n - s - l)B_{max}}{n} \tag{13}$$

with $s \geq w + 1$ and $s + l \geq m + 1$, then $C(\boldsymbol{u}, n)$ must be greater than or equal (11), which proves the optimality of $\boldsymbol{u}^*$. To prove (13) for an arbitrary $\boldsymbol{u}$, we consider two cases.

3.2.1 Arrangement with Distinct Elements

Suppose that the elements of $\boldsymbol{u}$ are distinct (i.e., $u_i \neq u_j$ for all $i \neq j$), which is only possible when $n \leq N$ (thus, $w = 0$). At least $m + 1$ of these streams belong to phases that differ pairwise by a multiple of M. (In general, for a set of distinct $kX + 1$ integers where k and X are nonnegative integers and $X \neq 0$, there are at least $k+1$ integers that differ pairwise by a multiple of X). Thus, $\bar{b}_{tot,j} \geq I_{max} + mP_{max} + (n-1-m)B_{max}$ for some phase j. By definition, $C(\boldsymbol{u}, n) \geq \bar{b}_{tot,i}/n$ for all i, and thus, $C(\boldsymbol{u}, n) \geq \bar{b}_{tot,j}/n \geq (I_{max} + mP_{max} + (n-1-m)B_{max})/n$. Therefore, $C(\boldsymbol{u}, n)$ satisfies (13) with $s = w + 1$ and $l = m - w$ ($w = 0$ in this case).

3.2.2 Arrangement with Repeated Elements

Suppose that the elements of $\boldsymbol{u}$ are not distinct. Let

$$\alpha \triangleq \max_{0 \leq j \leq N-1} r_j \tag{14}$$

Clearly, $\alpha \geq \max\{2, w+1\}$. We use the term *chain* to refer to a subset of the n streams whose phases differ pairwise by a multiple of M (including the ones that belong to the same phase). For example, if $n = 9$, $N = 15$, $M = 3$, and $\boldsymbol{u} = (0, 0, 0, 1, 2, 3, 4, 5, 6)$, then the first chain consists of the sources $\{s_1, s_2, s_3, s_6, s_9\}$, the second chain consists of $\{s_4, s_7\}$, and the last chain consists of $\{s_5, s_8\}$ as shown in Figure 2. Here, $\alpha = 3$. Observe that no more than M chains can exist in any arrangement. Let q be the number of chains in $\boldsymbol{u}$ ($q \leq M$). Denote these chains by $W_1, W_2, \ldots, W_q$, with

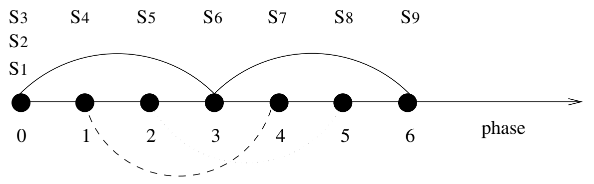

Figure 2: Chains for $\boldsymbol{u} = (0,0,0,1,2,3,4,5,6)$, $N = 15$, and $M = 3$.

corresponding sizes $\eta_1, \eta_2, \ldots, \eta_q$ ($\sum_j \eta_j = n$). If two streams with phases x and y, respectively, are in the same chain, say W_i, then $r_x + z_x = r_y + z_y = \eta_i$. For each chain W_j, let $C_j(\boldsymbol{u}, n)$ be the maximum aggregate peak rate divided by n, with the maximization taken only over the phases of the streams in W_j. For $j = 1, \ldots, q$, $C_j(\boldsymbol{u}, n)$ can be written as

$$C_j(\boldsymbol{u}, n) = \frac{n_I^{(j)} I_{max} + n_P^{(j)} P_{max} + n_B^{(j)} B_{max}}{n} \tag{15}$$

where for all j, $n_I^{(j)}$, $n_P^{(j)}$, and $n_B^{(j)}$ are nonnegative integers; $n_I^{(j)} \geq 1$ (from Proposition 2); and $n_I^{(j)} + n_P^{(j)} + n_B^{(j)} = n$. The total number of streams sending I or P frames during the phase of any stream in W_j is given by η_j. At least one of the chains, say W_1, contains α streams that belong to the same phase, say phase i. Consequently, $C_1(u, n) = \overline{b}_{tot,i}/n$ and $n_I^{(1)} = \alpha$. Based on the definition of $C(\boldsymbol{u}, n)$,

$$C(\boldsymbol{u}, n) = \max_{1 \leq j \leq q} C_j(u, n) \tag{16}$$

We consider two cases, depending on the value of η_1. First, suppose that $\eta_1 \geq m + 1$. Then,

$$\begin{aligned} C(\boldsymbol{u}, n) &\geq C_1(u, n) \\ &= \frac{\alpha I_{max} + (\eta_1 - \alpha) P_{max} + (n - \eta_1) B_{max}}{n} \end{aligned}$$

Since $\alpha \geq w + 1$ and $\eta_1 \geq m + 1$, $C(\boldsymbol{u}, n)$ satisfies (13), and $\boldsymbol{u}^*$ is optimal.

Next, suppose $\eta_1 < m + 1$. Thus, $\sum_{j=2}^{q} \eta_j \geq n - m$. There must be at least one chain, say W_j, for which

$$\eta_j \geq \frac{n - m}{q - 1}$$

(otherwise, $\sum_{j=2}^{q} \eta_j < n - m$). Accordingly,

$$\eta_j \geq \frac{n - m}{q - 1} > \frac{n - n/M}{q - 1} = \frac{n(M-1)/M}{q - 1} > \frac{n}{M} > m$$

where we use the fact that $m < n/M \leq m + 1$ and $q \leq M$. Since η_j is an integer, $\eta_j > m$ implies that $\eta_j \geq m + 1$. The streams in W_j belong to no more than N/M phases. For at least one of these phases, say phase i, we have $r_i \geq \eta_j/(N/M)$. But $n_I^{(j)} \geq r_k$ for all values of k that represent the phases of streams in W_j. Consequently,

$$n_I^{(j)} \geq \frac{\eta_j}{N/M} \geq \frac{1 + m}{N/M} \geq \frac{n/M}{N/M} = \frac{n}{N} > w$$

The last inequality follows from the definition of w. Accordingly, $n_I^{(j)} \geq w + 1$, and

$$\begin{aligned} C(\boldsymbol{u}, n) &\geq C_j(u, n) \\ &= \frac{n_I^{(j)} I_{max} + (\eta_j - n_I^{(j)}) P_{max} + (n - \eta_j) B_{max}}{n} \end{aligned}$$

Since $n_I^{(j)} \geq w + 1$ and $\eta_j \geq m + 1$, $C(\boldsymbol{u}, n)$ satisfies (13), and $\boldsymbol{u}^*$ is optimal.

3.3 Suboptimal Scheduling of Heterogeneous Streams

Using a single envelope to characterize all streams can be conservative if the videos significantly differ in their maximum frame sizes. In this case, it is more appropriate to use different traffic envelopes. With heterogeneous envelopes, $\boldsymbol{u}^*$ in Proposition 1 is no longer optimal. In fact, it can be shown that the optimal schedule depends on the exact values of the traffic envelopes, and no general expression for the best arrangement is possible. And even if we compute the best arrangement for a fixed n by means of exhaustive search (which is computationally expensive), it is not possible in general to maintain the optimality when a stream is added or terminated without disrupting the original structure of the n streams. Instead of providing an optimal schedule in the heterogeneous case, we provide a suboptimal schedule that gives very close gain to the optimal one. Such a suboptimal schedule is shown to be asymptotically optimal (as the number of sources goes to infinity). These results are stated below without proofs due to space limitation.

The suboptimal schedule does not have a closed-form expression, and is described as follows: The server maintains a matrix M and a vector V similar to the ones in Section 2.3.1. Given n ongoing streams, the server schedules the $(n+1)$ as follows:

$$u_{n+1} = i - 1 \text{ where } v_i = \min_{1 \leq j \leq \widetilde{N}} v_j \tag{17}$$

In other words, the new stream is scheduled in the phase for which the aggregate bit rate (computed based on the envelopes of the n sources) is minimal.

It can be shown that when streams are successively scheduled using this approach, then after each scheduling operation:

$$|v_j - v_k| \leq 2I_{max} \tag{18}$$

where $I_{max} = \max\{I_{max}^{(1)}, \ldots, I_{max}^{(n)}\}$. Let $C_{sub}(n)$ be the allocated bandwidth per source based on the suboptimal schedule. Then (18) can be used to show that:

$$C_{low}(n) \leq C_{sub}(n) \leq C_{low}(n) + 2I_{max} \tag{19}$$

where

$$C_{low}(n) \triangleq \sum_{i=1}^{n} \Big[(1/N^{(i)})I_{max}^{(i)} + (1/M^{(i)} - 1/N^{(i)})P_{max}^{(i)} + (1 - 1/M^{(i)})B_{max}^{(i)}\Big]$$

It is obvious that $C_{low}(n)$ is a lower bound on the minimum effective bandwidth. Thus, $C_{sub}(n)$ is no more than $2I_{max}/n$ from the optimal solution. As $n \to \infty$, $C_{sub}(n) \to C_{low}(n)$, i.e., $C_{sub}(n)$ is asymptotically optimal. The updating procedure is similar to the one in Section 2.3.1.

4 Numerical Results

We tested the effectiveness of our scheduling schemes using real MPEG traces that were provided by several researchers [2, 4, 5, 9] (see references for compression details). The envelopes for these traces are described in Table 1. The last column of the table depicts the maximum asymptotic gain in the homogeneous case as a percentage of the source peak rate.

Figure 3 depicts the normalized minimum effective bandwidth, $(C_{min}(n)/I_{max}) \times 100\%$, versus n for homogeneous streams. As n increases, $C_{min}(n)$ decreases non-monotonically to C^*_{min}. For large n, $C_{min}(n)$ is very insensitive to the variation in n. Clearly, the gain in bandwidth depends on the values of the traffic envelope parameters. For example, when several *Wizard of Oz* streams are multiplexed, the allocated bandwidth per source for large n is about 41% of the source peak rate, whereas it is about 84% for *Lecture* streams. The multiplexing gain can also be demonstrated by the number of video connections that can be simultaneously transported using a fixed capacity, as shown in Figure 4 (total bandwidth is normalized with respect to I_{max}).

To study the impact of N and M on the minimum effective bandwidth, we examined a segment of 12600 frames from the *Wizard of Oz* movie (frame # 29191 to frame # 41790 in the movie). This segment was

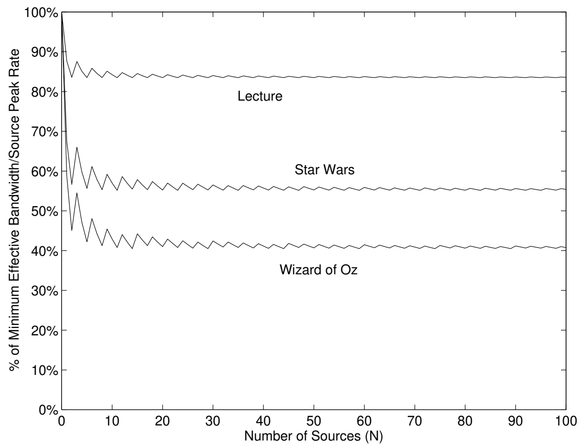

Figure 3: Percentage of $C_{min}(n)/I_{max}$ versus n for different MPEG traces (homogeneous case).

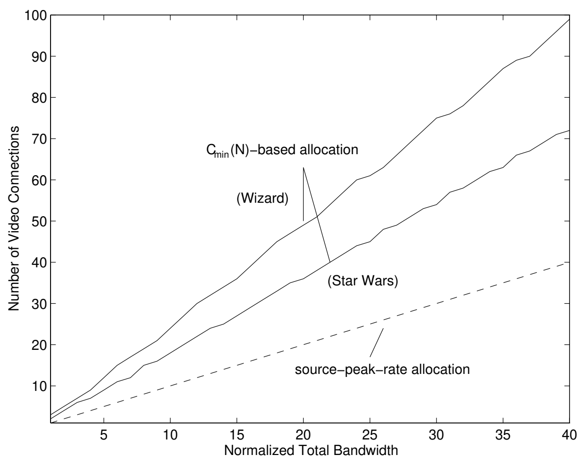

Figure 4: Number of simultaneously connections based on $C_{min}(n)$ versus normalized capacity.

Trace	Length (in frames)	I_{max}	P_{max}	B_{max}	N	M	$(C^*_{min}/I_{max}) \times 100\%$
Star Wars [2]	174136	483	454	169	12	3	55%
Wizard of Oz [5]	41760	894	742	157	15	3	41%
Advertisements [4]	16316	215	214	162	6	3	84%
Lecture [4]	16316	131	92	32	6	3	45%
Silence of the Lambs [9]	40000	350	231	144	12	3	53%

Table 1: Envelopes for empirical MPEG traces and the resulting maximum bandwidth gain (homogeneous case).

compressed several times using different N and M values. Table 2 depicts the GOP patterns that were used and the resulting I_{max}, P_{max}, and B_{max}. Unexpectedly, the GOP pattern seems to have little impact on the maximum frames sizes (note, however, that the GOP pattern has significant impact on the average frame size for each type of frames). This can be justified by the fact that a movie consists of several scenes, where a scene is loosely defined as a segment of the movie that exhibits uniformity in the video dynamics. The sizes of I frames (also P and B frames) within a scene are close in value. On the average, a scene lasts for few seconds. Thus, varying the compression pattern (whose time scale is smaller than one second) has little effect on the maximum sizes of I, P, and B frames within a scene. The last column in Table 2 gives C^*_{min}. It is obvious that N has a very negligible effect on C^*_{min}, whereas increasing M results in a significant reduction in C^*_{min}. This is expected since for the examined traces, P_{max} is closer to I_{max} than to B_{max}. When $P_{max} \approx I_{max}$, C^*_{min} reduces to $(1/M)P_{max} + (1-1/M)B_{max}$ which does not depend on N. In practice, using a large M (i.e., more B frames in a GOP) is not desirable from the decoder's perspective. Hence, M should be chosen so that it provides a good compromise between the decoder complexity (and resulting delay) and the multiplexing gain.

Using the suboptimal scheduling scheme for heterogeneous sources, the normalized effective bandwidth is plotted in Figure 5 as a function of the multiplexed streams. Here, we consider a simple scenario in which the heterogeneous mix consists of two different envelopes (e.g., two movies). Starting with $n = 1$, we increment n by adding streams one at a time to the multiplexer, and recursively computing the effective bandwidth according to the suboptimal scheme. During this process, we alternate between the two movies (for example, we start with an *Advertisement* stream, then add a *Lecture* stream, then add another *Advertisement* stream, and so on). The effective bandwidth is normalized with respect to the average source peak rate $\sum_{i=1}^{n} I^{(i)}_{max}/n$. As in the homogeneous case, it is observed that effective-bandwidth allocation, though not optimal, results in significant bandwidth gain.

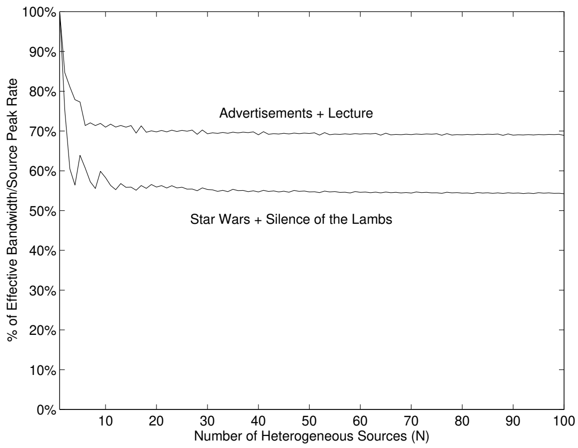

Figure 5: Percentage of $C_{sub}(n)/(\sum_i I^{(i)}_{max}/n)$ versus n for heterogeneous streams.

5 Summary

We presented a bandwidth allocation scheme for VBR MPEG-coded stored video. By exploiting the temporal structure of MPEG compression, this scheme achieves significant bandwidth gain, via statistical multiplexing, while supporting stringent, deterministic QoS guarantees. Our scheme can be implemented at a video server to maximize the number of simultaneously transported video streams between the server and a remote head-end switch. Efficient online procedures for computing and updating the allocated bandwidth were presented. To maximize the achievable gain, the server is given some flexibility in scheduling new requests prior to their multiplexing. For homogeneous sources, an optimal schedule was provided, which produces the minimum effective bandwidth. When tested with real MPEG traces, the optimal schedule sometimes results in more than 50% reduction in the allocated bandwidth compared to the source peak rate. We also presented a suboptimal scheduling scheme for heterogeneous sources, which is

GOP Pattern	N	M	I_{max}	P_{max}	B_{max}	$(C^*_{min}/I_{max}) \times 100\%$
I	1	1	908	—	—	100%
IP	2	1	898	756	—	92.1%
IPP	3	1	898	756	—	89.5%
IPPP	4	1	896	756	—	88.3%
IPPPP	5	1	896	740	—	86.1%
IBPB	4	2	896	733	161	54.4%
IBPBPB	6	2	898	742	161	53.2%
IBPBPBPB	8	2	889	742	161	52.9%
IBPBPBPBPB	10	2	894	742	161	52.2%
IBBPBB	6	3	898	719	157	41.7%
IBBPBBPBB	9	3	896	742	157	41.2%
IBBPBBPBBPBB	12	3	896	742	157	40.7%
IBBPBBPBBPBBPBB	15	3	893	742	157	40.5%

Table 2: Compression of a segment from the *Wizard of Oz* using different GOP patterns.

proved to be asymptotically optimal. A forthcoming paper extends the results to the case when streams are characterized by window-based envelopes, which provide tighter bounds on the bit rate.

Acknowledgment

The authors are grateful to Ron Sass, Mark Garrett, Oliver Rose, and Ed. Knightly for making their video traces available.

References

[1] W. Feng and S. Sechrest. Smoothing and buffering for the delivery of prerecorded compressed video. In *IS&T/SPIE Multimedia Computing and Networking*, pages 234–244, Feb. 1995.

[2] M. W. Garrett and M. Vetterli. Congestion control strategies for packet video. In *Proc. of Fourth Int. Workshop on Packet Video*, Aug. 1991. Kyoto, Japan.

[3] E. W. Knightly, D. Wrege, J. Liebeherr, and H. Zhang. Fundamental limits and tradeoffs of providing deterministic guarantees to VBR video traffic. In *Proc. of the ACM SIGMETRICS/PERFORMANCE '95 Conference*, pages 98–107, May 1995.

[4] E. W. Knightly and H. Zhang. Traffic characterization and switch utilization using a deterministic bounding interval dependent traffic model. In *Proc. of IEEE INFOCOM '95*, pages 1137–1145, 1995.

[5] M. Krunz, R. Sass, and H. Hughes. Statistical characteristics and multiplexing of MPEG streams. In *Proc. of the IEEE INFOCOM '95 Conference*, pages 455–462, Boston, Apr. 1995.

[6] S. S. Lam, S. Chow, and D. K. Y. Yau. An algorithm for lossless smoothing for MPEG video. In *Proceedings of the ACM SIGCOMM '94 Conference*, Aug. 1994.

[7] J. M. McManus and K. W. Ross. Video on demand over ATM: Constant-rate transmission and transport. In *Proc. of IEEE INFOCOM '96*, pages 1357–1362, 1996.

[8] T. Ott, T. Lakshman, and A. Tabatabai. A scheme for smoothing delay-sensitive traffic offered to ATM networks. In *Proc. of IEEE INFOCOM '92*, pages 776–785, May 1992.

[9] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. In *Proceedings of the 20th Annual Conference on Local Computer Networks*, Minneapolis, MN, 1995.

[10] J. D. Salehi, Z.-L. Zhang, J. F. Kurose, and D. Towsley. Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing. In *Proc. of the ACM SIGMETRICS/PERFORMANCE '96 Conference*, May 1996.