Impact of Synchronization on the Allocation of Bandwidth for Multiplexed MPEG Streams *

Marwan Krunz† and Satish Tripathi‡
† University of Maryland Institute for Advanced Computer Studies
‡ Department of Computer Science
University of Maryland
College Park, MD 20742
Email: [krunz,tripathi]@cs.umd.edu

Abstract

In an MPEG encoder, three types of frames (I, P, and B) are periodically generated according to a pre-specified *compression pattern*. As a result, an MPEG sequence is periodic in its compression pattern, and this periodicity can be used to reduce the bandwidth requirements of multiplexed MPEG streams. By exploiting the deterministic and periodic nature of the compression pattern, we show that it is possible to provide stringent deterministic guarantees (no cell losses and no queueing delay) to MPEG connections without the need to allocate the peak rates of individual sources. Instead, a stream is allocated its *effective bandwidth*, which is the *aggregate* peak rate of the multiplexed streams divided by the number of streams. The aggregate peak rate depends on the *arrangement* of the multiplexed streams which is a measure of the degree of synchronization among the compression patterns of different streams. It is found that in most cases, the effective bandwidth is smaller than the source peak rate. For a given arrangement, we provide a procedure to compute the effective bandwidth. We also give an expression for the 'best' arrangement that results in the 'optimal' effective bandwidth. Examples of real MPEG sequences are used to show the bandwidth gains that can be achieved through proper scheduling of the starting times of MPEG connections.

Keywords: bandwidth allocation, MPEG, statistical multiplexing, deterministic guarantees

* This research was partially supported by the NSF grant # CCR 9318933.

1 Introduction

One of the major challenges in designing a BISDN/ATM network is to guarantee the quality-of-service (QoS) requirements for all transported streams without underutilizing the available bandwidth capacity. The QoS requirements, which are often measured by the cell loss rate and cell delay, can be easily satisfied by allocating bandwidth based on the peak rates of the individual sources. However, due to the burstiness of many sources (i.e., large peak rate to mean rate ratio), source-peak-rate allocation results in low utilization. To increase the utilization, statistical multiplexing can be used, which allows the available bandwidth to be shared among various streams on a need basis. By means of statistical multiplexing, the network can allocate an aggregate amount of bandwidth that is less than the sum of peak rates of the individual streams. This conventional use of statistical multiplexing results in possible cell queueing and buffer overflow. The amount of cell delay and cell loss depend on the traffic model used to characterize the multiplexed streams. Because of the statistical nature of commonly used traffic models, the use of statistical multiplexing is usually limited to sources with *statistical* QoS requirements. Typically, a stream with deterministic QoS requirements (e.g., no cell losses) is not statistically multiplexed with other streams. Depending on its delay requirement, such a stream is either allocated its peak rate, or (if the some buffering delay can be tolerated) its peak rate over a finite interval [5].

In this paper, we focus on compressed video streams that are generated by MPEG encoders. We show that, contrary to the general belief, statistical multiplexing can be used to an advantage with this type of traffic while providing stringent and deterministic QoS guarantees. By exploiting the deterministic and periodic manner in which frame types in an MPEG stream are generated, we show that MPEG streams can be statistically multiplexed (with an effective bandwidth per source that is less than the source peak rate) without experiencing any cell losses or delays. The effective bandwidth depends on the relative degree of synchronization among the multiplexed streams. We provide a simple algorithm for computing the effective bandwidth for an arbitrary synchronization structure. This algorithm can be used as part of call admission control at a switching/multiplexing network node. In situations where it is possible to have some control on the starting times of MPEG streams (e.g., in a VOD system), we give the form of the best synchronization structure for the multiplexed MPEG streams that has the optimal (minimum) effective bandwidth.

The rest of the paper is structured as follows. The deterministic traffic model that is used to characterize MPEG sources is described in Section 2. The effective bandwidth for multiplexed MPEG streams is discussed in Section 3. In the same section, we derive the formulae for the 'best' synchronization structure and the associated optimal effective bandwidth. The paper is concluded in Section 4.

2 Characterization of an MPEG Stream

2.1 Compression Pattern

An MPEG encoder employs several modes of compression to generate three types of compressed frames: Intra-coded (I), Predictive (P), and Bidirectional (B) frames. I frames are compressed using intraframe coding (e.g., DCT) only, while P and B frames are compressed using intraframe coding as well as motion compensation techniques (prediction techniques for P frames and both prediction and interpolation techniques for B frames). As a result, I frames are, in general, the largest in size, followed by P frames, and finally B frames (the frame size refers to the number of bits used to encode the frame). To maintain a constant-quality motion picture, frames are compressed at a constant frame rate (e.g., 30 frames/sec).

An important feature of MPEG encoders is the manner in which frame types are generated. When compressing a video sequence, the encoder uses a pre-defined *compression pattern* to determine the types of the compressed frames. The compression pattern defines the number and temporal order of P and B frames to be generated between two successive I frames. The same compression pattern is used repeatedly to compress the whole video sequence. An example of the compression pattern is shown in Figure 1-a. Because the sizes of compressed frames are largely affected by their types (as well as the scene dynamics), one should expect a significant impact of the periodicity of the compression pattern on the characteristics of the traffic and, consequently, the bandwidth allocation strategies.

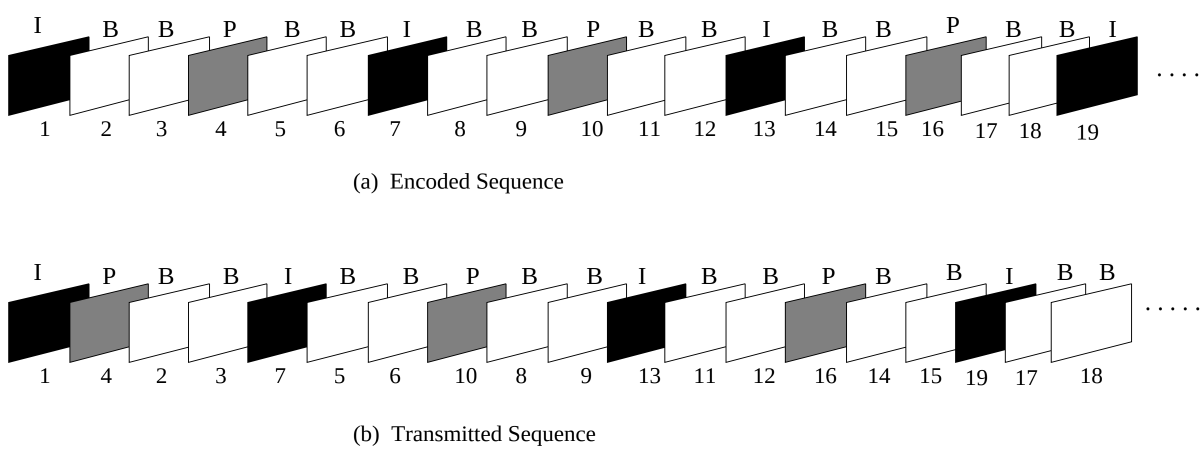


Figure 1: An example of a compression pattern that defines the frame types in an MPEG stream.

Since B frames are coded using future I/P frames, the order in which frames are sent over the network (i.e., the transmission order) is different from their encoding order (see Figure 1-b). However, starting from the second I frame, the transmission and encoding orders look similar with respect to frame types. Therefore, we will ignore the first few frames in a stream, and assume, for simplicity, that frame types in an MPEG stream are represented by exact replications of the compression pattern.

2.2 Deterministic Traffic Model

Several traffic models were proposed for the characterization of compressed video traffic (see [2] and the references therein). Most of these models are probabilistic in nature, and thus, cannot be used to provide deterministic guarantees. To support deterministic QoS requirements, which is the focus of this paper, a deterministic model for an MPEG source is needed. Intuitively, a deterministic characterization can be constructed using bounds on the actual bit rate. In [5], Knightly et al. proposed a simple deterministic approach to characterize a VBR stream using a traffic constraint function. The same methodology is adopted in this paper, and is explained below.

Consider an MPEG stream that consists of a sequence of frames that are generated at a constant rate f. We assume that video bits in a frame are packetized into ATM cells (each 48 bytes of video corresponds to one cell). Cells within a frame are distributed evenly over the frame period. Hence, the bit rate over a frame period is given by f times the frame size. Let I_{max}, P_{max}, and B_{max} be, respectively, the maximum sizes of I, P, and B frames in the stream. Using these three values and the compression pattern of the stream, a traffic constraint function, $\overline{b}(t)$, can be defined which bounds the actual bit rate (t here is measured in units of frame periods). An example of $\overline{b}(t)$ is shown in Figure 2 based on the compression pattern of Figure 1-a. Note that $\overline{b}(t)$ is a piecewise constant function.

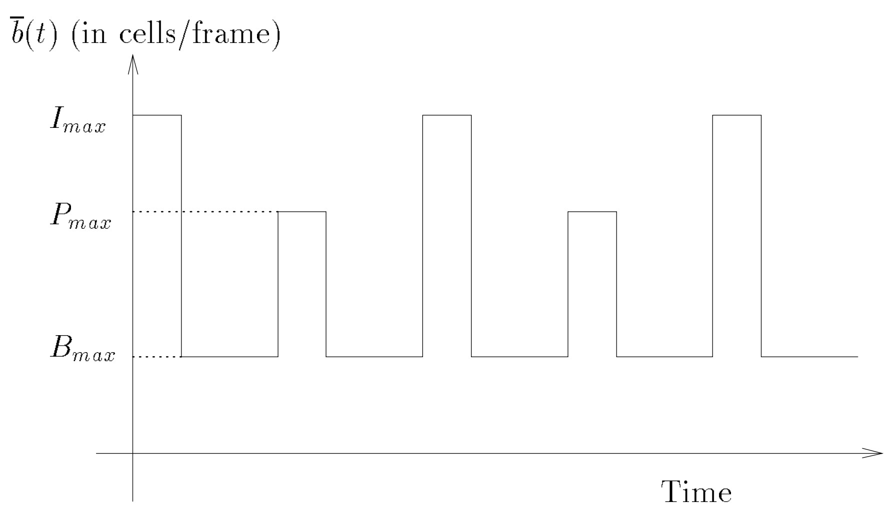


Figure 2: Traffic constraint function based on the compression pattern of Figure 1(a).

Because of its regularity, the compression pattern can be characterized by two parameters:

L : number of frames between two consecutive I frames in an MPEG stream.

Q : number of frames between an I frame and the subsequent I/P frames (whichever comes first) in an MPEG stream.

Examples of various compression patterns and their associated L and Q values are shown in Table 1. The regularity of the compression pattern means that L is an integer multiple of Q. Notice that it is possible to have $L = Q = 1$, in which case only I frames are generated (this is similar to a stream generated by a JPEG encoder).

Compression Pattern	L	Q
$IBBPBB$–$IBBPBB \cdots$	6	3
$IBBPBBPBBPBBPBB$–$IBBPBBPBB \cdots$	15	3
$IBBBPBBB$–$IBBBPBBB \cdots$	8	4
$IPPP$–$IPPP \cdots$	4	1
I–I–I–I–$I \cdots$	1	1

Table 1: Compression patterns and their associated L and Q values.

Consequently, $\overline{b}(t)$ is fully specified by five parameters: I_{max}, P_{max}, B_{max}, L, and Q. In this paper, the traffic constraint function is used to characterize an MPEG stream.

3 Statistical Multiplexing of MPEG Streams

3.1 Preliminaries

We consider MPEG streams with very stringent requirements that consist of no losses and no queueing delays. Typically, these requirements are met by allocating bandwidth based on the peak bit rate of each source, resulting in very low utilization. Network utilization can be improved by temporal averaging in which video frames are buffered before entering the network. However, the queueing delay incurred in buffering precludes its use for delay-sensitive traffic. Moreover, an excessive amount of buffer is often needed to maintain a reasonable level of utilization. Network utilization can also be improved using statistical multiplexing which spatially averages the bit rate of several streams. Typically, statistical multiplexing have not been used in conjunction with deterministic QoS guarantees, mainly due to the statistical behavior of the sources. The situation is different in the case of MPEG streams. Because of the deterministic structure of the compression pattern, and using the traffic constraint function to characterize an MPEG source, we show that statistical multiplexing can be used advantageously with MPEG sources while supporting stringent, deterministic QoS requirements.

The benefits of statistical multiplexing for MPEG streams with deterministic QoS guarantees can be demonstrated by the following example. Consider two streams that are characterized by the same traffic constraint function, $\overline{b}(t)$, with $L = 6$, $Q = 3$, and $I_{max} > P_{max} > B_{max}$. Suppose that the second stream started *exactly* one frame period after the start of the first stream (see Figure 3). Then, an upper bound on the bit rate of the *superposition* of the two streams is given by:

$$\overline{b}_{tot}(t) = \overline{b}(t) + \overline{b}(t-1) \tag{1}$$

which is clearly less than $2\overline{b}(t)$. In fact, it is easy to see that:

$$C \triangleq \frac{1}{N} \max_{t \geq 0} \overline{b}_{tot}(t) = \frac{I_{max} + B_{max}}{2} < I_{max} \tag{2}$$

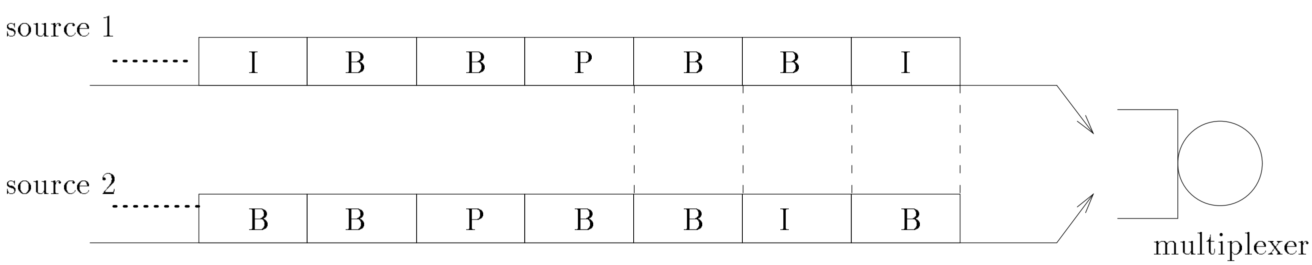


Figure 3: An example that shows the reduction in bandwidth requirements when two MPEG streams are multiplexed.

The quantity C is the *effective bandwidth* (measured in cells per frame period) that must be allocated to each stream to guarantee lossless transmission with no delay. The notion of *effective bandwidth* (also known as *equivalent capacity*) was investigated in several previous studies in a stochastic framework (for example, see [4] and [1]). In this paper, the effective bandwidth is used in a deterministic framework to guarantee zero cell loss rate and no queueing delays. By superposing the two streams and allocating bandwidth for the aggregate traffic, the required amount of bandwidth per source decreased from I_{max} (source-peak-rate allocation) to $(I_{max}+B_{max})/2$. The superposition can be achieved via statistical multiplexing. A very small buffer is needed at the input to the multiplexer in case two cells from both streams arrive simultaneously at the multiplexer (the size of this buffer is one cell). Notice that bandwidth gains from statistical multiplexing are obtained via spatial averaging, and *not* temporal averaging (i.e., buffering).

From the above example, it is clear that bandwidth gains from statistical multiplexing depend on the degree of synchronization among the multiplexed MPEG streams. In this context, the degree of synchronization is used to measure the differences in the starting times of the multiplexed streams.

If the two streams were sending I frames simultaneously, then $C = I_{max}$, and statistical multiplexing introduces no advantages over source-peak-rate allocation. Fortunately, the probability that both streams are in the same phase (i.e., sending I frames simultaneously) is small.

Next, we formally quantify the bandwidth gains obtained by multiplexing an arbitrary number of MPEG streams having an arbitrary synchronization structure. Two cases are considered. The first case is when MPEG sources have an arbitrary synchronization structure with regard to their compression patterns, but the boundaries of the frames are exactly aligned. In this case the system is slotted with a time unit of one frame period. The second case is more general than the first case, where frames boundaries need not be necessarily aligned. Intuitively, the first case results in better bandwidth gains (or higher probability to achieve these gains) than the second case. In a video server, exact alignment of frame boundaries can be imposed by delaying the starting time of an MPEG stream by no more than a frame period. This delay amounts to less than 1/30 of a second (at $f = 30$ frames/sec), and will not be noticeable by a user of a video-on-demand system. If multiplexing is to take place at an intermediate network node, then a small amount of buffering (less than a frame) is needed to exactly align frame boundaries. Even when frame boundaries are not aligned, some bandwidth gains may still be obtained.

3.2 Effective Bandwidth for Multiplexed MPEG Streams

3.2.1 Case of Aligned Frame Boundaries

Let N be the number of multiplexed MPEG streams. We assume that all streams are characterized by the same constraint function, $\bar{b}(t)$, which is specified by $(I_{max},\ P_{max},\ B_{max},\ L,\ Q)$, with $I_{max} > P_{max} > B_{max}$. When the streams have different maximum values for the sizes of I, P, and B frames (but the same L and Q), a common constraint function can be obtained by taking I_{max} as the largest I frame in all the streams (similarly, for P_{max} and B_{max}). Let u_j be the difference (in frame periods) between the arrival time of an I frame from the jth MPEG stream and the arrival time of the most recent I frame from the first stream. Because of the periodicity of the compression pattern and the fact that the same compression pattern is used in all the streams, u_j is constant throughout the connection hold time. For the present case of aligned frame boundaries, u_j can take any integer value in $\{0, \ldots, L-1\}$, $u_1 \stackrel{\triangle}{=} 0$. In the example of Figure 3, $u_2 = 1$. The synchronization structure of the N sources can be completely specified by the $(N-1)$-tuple $(u_2, u_3, \ldots, u_N)$. Such a tuple will be referred to as an *arrangement*. Let $\mathcal{U}$ be the set of all possible distinct arrangements of N streams. For a given arrangement $u = (u_2, u_3, \ldots, u_N)$ and N, we define the effective bandwidth per source as:

$$C(u,N) \triangleq \frac{1}{N} \max_{t \geq 0} \left(\sum_{i=1}^{N} \overline{b}(t-u_i) \right) \tag{3}$$

which can also be written as:

$$C(u,N) = \frac{n_I I_{max} + n_P P_{max} + n_B B_{max}}{N} \tag{4}$$

for some nonnegative integers n_I, n_P, and n_B where $n_I + n_P + n_B = N$. In the worst situation, all the streams send I frames simultaneously, in which case, $n_I = N$, $n_P = n_B = 0$, and the effective bandwidth is the same as the source peak rate. Notice that $C(u,N)$ is given in units of cells/frame period which translates to $f * C(u,N) * 53 * 8$ bits/sec.

Computing $C(u,N)$ from (3) requires taking the maximum of L terms, each of which is the sum of N values. Note that the sum in the RHS of (3) is a piece-wise constant function that is periodic in L. A numerical procedure for computing $C(u,N)$ is given in Figure 4 and is illustrated below.

Input: u, L, Q, N, I_{max}, P_{max}, B_{max}

Ouput: $C(u,N)$

Initialization:
```
for k = 0,1,...,L-1 do
    r_k := number of streams that are in phase k
end for
C(u,N) := B_max              /* a lower bound on C(u,N) */
```

Loop:
```
for k = 0,1,...,L-1 do
    z := 0 /* number of streams that generate P frames during phase k */
    for j = 1,2,...,L/Q-1 do        /* compute z for phase k */
        x := mod_L(k + jQ)
        z := z + r_x                /* update z */
    end for
    Y := (r_k I_max + z P_max + (N - r_k - z) B_max)/N
    C(u,N) := max(C(u,N), Y)        /* update C(u,N) */
end for
```

Figure 4: Algorithm for computing $C(u,N)$.

For $i = 2,\ldots,N$, u_i can be obtained by implementing a counter at the multiplexing node, which is initialized with the first frame of the first admitted stream, and is incremented every frame period. The counter goes from 0 up to $L-1$ and then starts again. When the ith stream arrives, its u_i is set to the current value of the counter. A table of (i, u_i) pairs is maintained, and is used to execute the algorithm in Figure 4. The ith stream is said to be in phase k if $u_i = k$. The algorithm requires the computation of r_k which is the number of streams that are in the same phase.

The inner 'for' loop in the algorithm computes the number of streams that generate P frames during phase k. Such streams must be at frame distances of Q or multiples of Q from streams of phase k. These distances are given by x in the 'for' loop. The number of computations required to obtain $C(u, N)$ is $\mathcal{O}(L^2/Q)$ which is relatively small (assuming that the values for r_k are pre-computed, and are updated whenever a new stream is added). To further reduce the required number of computations, we introduce the following results.

Proposition 1 *Consider any two streams i and j with $u_i = k_1$ and $u_j = k_2$, $k_1 \neq k_2$. If during phase k_1 stream j sends a B frame, then during phase k_2 stream i sends a B frame. Similarly, if during phase k_1 stream j sends a P frame, then during phase k_2 stream i sends a P frame.* $\square$

The proof Proposition 1 follows immediately from the fact that all the streams have the same *periodic* compression pattern.

Proposition 2 *In (4), $n_I \geq 1$ for any arrangement $u = (u_1, \ldots, u_N)$.*

Proof: Suppose that $n_I = 0$.
First, consider the case when $n_P = 0$. Then $C(u, N) = NB_{max}/N$. Since $u_1 = 0$ (by definition), $r_0 \geq 1$. Thus, during phase 0 the aggregate peak rate $\overline{b}_{tot}(t) \geq I_{max} + (N-1)B_{max} > NC(u, N)$, which is contrary to the definition of $C(u, N)$.
Next, consider the case when $n_P \geq 1$. Let phase k be the phase during which $\overline{b}_{tot}(t)/N = C(u, N)$. By assumption, $r_k = 0$ (since $n_I = 0$). Since $n_P \geq 1$, there exists at least one stream j with $|u_j - u_k| =$ a multiple of Q. During phase u_j, source j sends I frames. Also, any other stream which sends P frames during phase u_k will be sending either I frames or P frames during phase u_j. Thus, $\overline{b}_{tot}(t)$ during phase u_j is larger than $\overline{b}_{tot}(t)$ during phase u_k, which is contrary to the definition of $C(u, N)$. Hence, $n_I \geq 1$. $\square$

Since $n_I \geq 1$, the computational requirements of $C(u, N)$ can be further reduced by excluding any phase k that has $r_k = 0$ from the body of the outer 'for' loop in Figure 4.

The following example demonstrates the dependency of $C(u, N)$ on the arrangement u. Using $N = 3$, $L = 15$, and $Q = 3$, the effective capacity $C(u, N)$ was computed for different arrangements

$u = (u_1, u_2, u_3)$ (with $u_1 \triangleq 0$). Figure 5 shows $C(u, N)$ normalized to I_{max} (the source peak rate) for all possible values of u_3 (from 0 to $L-1$) and $u_2 = 0, 1, 2$. Maximum values for frame sizes were taken from the frame-size trace of the *Wizard of Oz* movie which was compressed using an MPEG encoder [6]. Accordingly, $I_{max} = 894$, $P_{max} = 742$, and $B_{max} = 157$ (in cells). It is clear that except for one possible arrangement, $u = (0, 0, 0)$, statistical multiplexing can reduce the bandwidth requirements without sacrificing any performance guarantees. In fact, even when the number of sources is as small as 3, the bandwidth requirement for a stream can be reduced in some cases to less than 50% of the source peak rate. The exact amount of reduction depends on the synchronization structure. If all possible arrangements can occur with equal probability, then the chance of not achieving any gains from multiplexing is $1/L^{N-1}$ (which is the probability that $u_i = 0$ for all $i \in \{2, \ldots, N\}$).

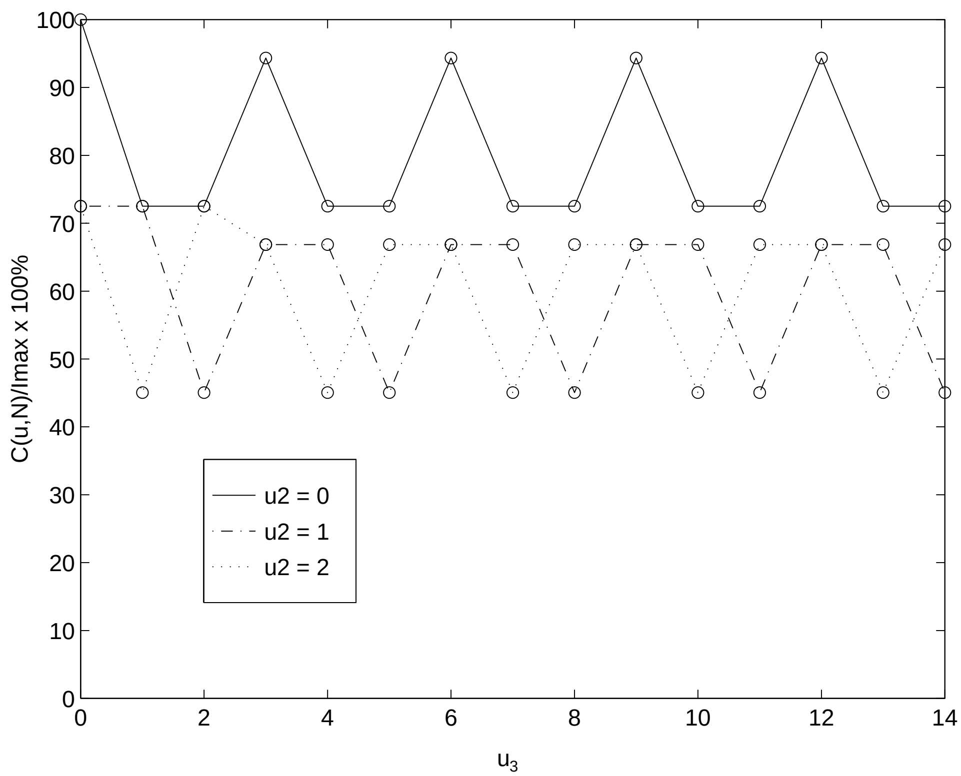


Figure 5: Effective bandwidth for different arrangements in the case of aligned frame boundaries. Different arrangements, $u = (u_1, u_2, u_3)$, are obtained by varying u_3 from 0 to $L-1$ (using only integer values), with $u_2 = 0, 1, 2$ and $u_1 = 0$ ($N = 3$, $L = 15$, $Q = 3$).

3.2.2 Case of Non-Aligned Frame Boundaries

In the previous section, it was assumed that frame boundaries from different streams are aligned. This alignment can be enforced using a frame-buffer for each stream. When frame boundaries are aligned, there is more probability to achieve bandwidth gains through statistical multiplexing than in the case of non-aligned frames. Consider, for example, the two streams in Figure 3. If frame boundaries are aligned, then the probability of having $C(u, N) = I_{max}$ is $\Pr\{u_1 = u_2 = 0\} = 1/L$ (assuming all that all possibilities are equally probable). However, if frame boundaries are not aligned (e.g., source 2 starts after a fraction of a frame period from the start of source 1), then the probability of $C(u, N) = I_{max}$ is the probability that the two sources simultaneously generate I frames for *any* time duration. This is the same as the probability that the two sources overlap in their phases, and is given by $2/L$. In general, the probability of not gaining anything from multiplexing N streams when frame boundaries are arbitrarily aligned is $2/L^{N-1}$ (compared to $1/L^{N-1}$ when frames are aligned).

When frame boundaries are not aligned, the algorithm in Figure 4 can still be used to compute $C(u, N)$, with slight modifications. Since, in this case, the lag between the jth stream ($j = 2, \ldots, N$) and the 1st stream can take non-integer values, two phases are associated with each stream (except for the first stream). Thus, if the lag between the arrival of an I frame in the jth stream and the arrival of the most recent I frame in the first stream is 3.2 frame periods, then the jth stream is assigned to both phases 3 and 4. When computing r_k in Figure 4, a stream with a non-integer phase is counted twice. This does not mean that the stream contributes two terms to the computation of the aggregate bit rate during a given phase, but rather its contribution is the largest of two terms. Hence, the most inner loop in the algorithm must be modified to ensure that z does not exceed $N - r_k$. A statement such as $z = \min(z, N - r_k)$ should be inserted after the end of the most inner 'for' loop. Figure 6 shows an example of the effective bandwidth (given in percentage of source peak rate) for different arrangements in the case of non-aligned frames. The same parameters as in Figure 5 are used. However, in this case u_3 is *continuously* varied, while in Figure 5 u_3 assumed only integer values. Clearly, the proportion of arrangements that result in significant bandwidth gains is smaller in the case of non-aligned frames. In fact, the effective bandwidth in this case is less than half of the source peak rate only at a finite number of values for u_3. For example, when $u_2 = 1$, $C(u, N) < 0.5 I_{max}$ only when $u_3 = 2, 5, 8$, and 11. Assuming that u_3 is random with uniform distribution in $[0, L)$, the probability that $C(u, N) < 50\% I_{max}$ is zero. Nevertheless, there is a high probability that $C(u, N) < 70\%\ I_{max}$.

3.3 "Optimal" Effective Bandwidth

From the previous section, it is clear that $C(u, N)$ varies with u. It is, thus, natural to seek the "best" arrangement that produces the "optimal" effective bandwidth for N multiplexed MPEG streams ($C_{opt}(N)$). In this section, we obtain the expression for $C_{opt}(N)$ and the arrangement

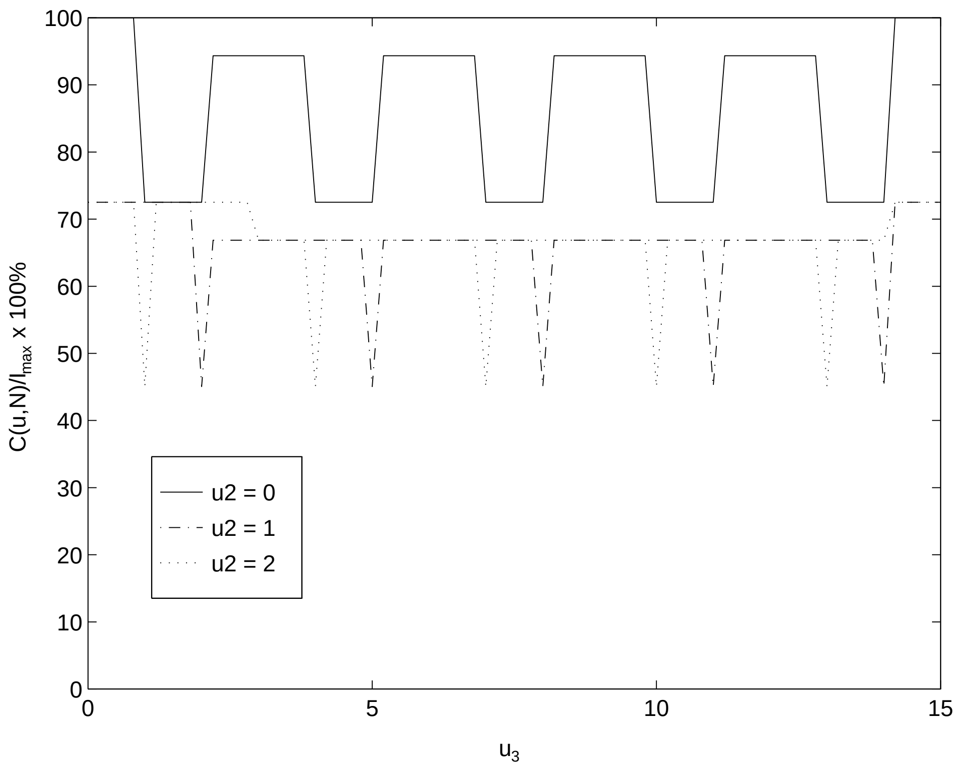


Figure 6: Effective bandwidth in the case of non-aligned frame boundaries. u_3 is continuously varied in the interval $[0, L)$ with $u_2 = 0, 1, 2$ and $u_1 = 0$ ($L = 15$, $Q = 3$).

(in fact, one of the arrangements) that results in $C_{opt}(N)$. We also address the issues related to computing $C_{opt}(N)$. First, lets define the optimal effective bandwidth:

$$C_{opt}(N) \stackrel{\triangle}{=} \min_{u \in \mathcal{U}} C(u, N) \tag{5}$$

where $\mathcal{U}$ is the set of all possible *distinct* arrangements of N streams (distinct in the sense that all permutations of the elements of a given u are counted as one distinct arrangement). We will assume throughout this section that frame boundaries are aligned. Extension to the case of non-aligned frame boundaries is straightforward, since, as we show later, the optimal bandwidth when frame boundaries are generally non-aligned occurs for an arrangement with aligned frame boundaries (note that aligned frame boundaries are special cases of generally non-aligned boundaries).

Computing $C_{opt}(N)$ directly from (5) requires an exhaustive search for the minimum $C(u, N)$ among all possible distinct u. The number of computations is proportional to the size of $\mathcal{U}$ which is given by:

$$|\mathcal{U}| = \sum_{i=1}^{m} \left(\begin{array}{c} L \\ i \end{array} \right) \left(\begin{array}{c} N-2 \\ i-1 \end{array} \right) \tag{6}$$

where $m = \min\{N-1, L\}$. The number of distinct arrangements grows very fast with N (e.g., for $L = 12$ and $N = 6$, $|\mathcal{U}| = 4368$). This is computationally prohibitive if $C_{opt}(N)$ is to be obtained during connection setup time. Of course, if the traffic constraint function is known before connection setup time, then $C_{opt}(N)$ can be computed off-line. It should be note that although the number of distinct arrangements is that large, the number of possible values for $C_{opt}(N)$ no more than $N(N+1)/2$. This number is obtained by considering all possible values for $C(u, N)$. From (4) and Proposition 2, the number of possible values for $C(u, N)$ is $\sum_{i=1}^{N}(N-j+1) = N(N+1)/2$.

Suppose that $C_{opt}(N)$ is known for a given N. Let $u^* \in \mathcal{U}$ be a 'best' arrangement such that $C(u^*, N) = C_{opt}(N)$. It remains questionable if $C_{opt}(N-1)$ and $C_{opt}(N+1)$ can be obtained by, respectively, dropping or adding a stream to the N streams that are arranged according to u^*. This is an important issue because if it was possible to start N streams with the 'best' arrangement, it is desirable that the $N-1$ streams that remains after a video stream terminates still form a 'best' arrangement of $N-1$ streams. Similarly, it is desirable that a 'best' arrangement of $N+1$ streams can be obtained by adding a stream to a 'best' arrangement of N streams. There is a better chance to obtain a best arrangement when adding a stream than when dropping a stream. For example, if $u^* = (u_1^*, u_2^*, \ldots, u_N^*)$ is a 'best' arrangement of N streams, then the $(N+1)$th stream can be added with $u_{N+1} = k$ such that

$$C((u^*, u_{N+1} = k), N+1) = \min_{j \in L} C((u^*, u_{N+1} = j), N+1) \tag{7}$$

which requires only L computations of the effective bandwidth. However, the resulting arrangement $(u_1^*, u_2^*, \ldots, u_N^*, u_{N+1} = k)$ may not produce $C_{opt}(N+1)$, but rather a suboptimal value.

In the following, we provide a closed-form expression for $C_{opt}(N)$ in terms of I_{max}, P_{max}, B_{max}, L, Q, and N. More importantly, we show that it is possible to obtain $C_{opt}(N+1)$ ($C_{opt}(N-1)$) by adding (removing) a stream to a 'best' arrangement of N streams. Since $I_{max} > P_{max} > B_{max}$, a lower bound on $C_{opt}(N)$ can be deduced from Proposition 2 and Equations (4) and (5):

$$C_{opt}(N) \geq \frac{I_{max} + (N-1)B_{max}}{N} \tag{8}$$

Hence, an arrangement u with $C(u, N)$ that equals the RHS of (8) must be a 'best' arrangement.

Let $u^* = (u_1^*, \ldots, u_N^*)$ be an arrangement of N streams that is given by:

$$u^* = (\underbrace{0, 1, 2, \ldots, L-1, 0, 1, 2, \ldots, L-1, \ldots,}_{w \text{ times}} 0, 1, 2, \ldots, N - wL - 1) \tag{9}$$

where

$$w \stackrel{\triangle}{=} \text{largest nonnegative integer } k \text{ that satisfies } N > kL \tag{10}$$

We will show that u^* is a 'best' arrangement of N streams. Notice that when $N \leq L$, then $w = 0$ and u^* reduces to

$$u^* = (0, 1, 2, \ldots, N-1) \tag{11}$$

Consider the following cases:

Case 1: $1 \leq N \leq Q$

Let the N streams be arranged according to u^* in (11). Since $N - 1 < Q$, for any streams i and j, it must be true that $\left|u_i^* - u_j^*\right| \neq$ a multiple of Q. From Proposition 2, $C(u^*, N)$ must come from a phase k with $r_k \geq 1$. But for u^*, $r_k = 0$ or $r_k = 1$ (each stream is in a distinct phase). Thus, during any phase k with $r_k = 1$, there is exactly one stream that generates an I frame, and $N - 1$ streams that generate B frames, implying that:

$$C(u^*, N) = \frac{I_{max} + (N-1)B_{max}}{N} \tag{12}$$

From (8), u^* must be a 'best' arrangement. Notice that there can be more than one arrangement that produces $C_{opt}(N)$. The rationale behind the particular choice of u^* will be discussed later.

Case 2: $N = Q + 1$

In this case the lower bound of (8) cannot be achieved. To see that, consider an arrangement of N streams, $u = (u_1, \ldots, u_N)$. If two streams or more are in the same phase, then $C(u, N) \geq (2I_{max} + (N-2)B_{max})/N$. On the other hand, if each stream is in a distinct phase, then there must be at least two streams, i and j, such that $|u_i - u_j|$ = a multiple of Q [1], in which case, $C(u, N) \geq (I_{max} + P_{max} + (N-2)B_{max})/N$. Either way,

$$C(u, N) \geq C_{opt}(N) = \frac{I_{max} + P_{max} + (N-2)B_{max}}{N} = C(u^*, N) \tag{13}$$

The last equality can be easily verified by examining the structure of u^* in (11). Since $N = Q + 1$, for any phase k with $r_k = 1$, there is exactly one stream delivering an I frame, one stream delivering a P frame, and $N - 2$ streams delivering B frames. Hence, u^* is a best arrangement.

[1] In general, a set of distinct $kX + 1$ integers, where k is a nonnegative integer, must have at least $k + 1$ elements which differ, pairwise, by a multiple of X.

Case 3: $Q+1 < N \le L$

This case is a generalization of Case 2. Let m be the largest integer such that $N > mQ$. Similar to the argument used in Case 2, it is easy to see that if every stream of an arbitrary arrangement u is in a distinct phase, then there must be at least $m+1$ streams whose phases differ, pairwise, by a multiple of Q. In this case,

$$C(u,N) \ge \frac{I_{max} + mP_{max} + (N-1-m)B_{max}}{N} \tag{14}$$

On the other hand, suppose that at least two streams are in the the same phase. It can be shown (see the appendix for details) that $C(u,N)$ satisfies:

$$C(u,N) \ge \frac{sI_{max} + lP_{max} + (N-s-l)B_{max}}{N} \tag{15}$$

where $s \ge 2$ and $l+s \ge m+1$. Since $s > 1$, the RHS of (15) is greater than the RHS of (14), and thus either way $C(u,N)$ cannot be less than the RHS of (14). Thus, $C_{opt}(N)$ is given by the RHS of (14). Now consider u^* as given in (11). Each stream in u^* is in a different phase. Moreover, there is exactly $m+1$ streams whose phases differ, pairwise, by a multiple of Q. Hence,

$$C(u^*,N) = C_{opt}(N) = \frac{I_{max} + mP_{max} + (N-1-m)B_{max}}{N} \tag{16}$$

Case 4: $N > L$

In this case it is not possible to assign each stream to a distinct phase. For u^* in (9), it can be shown that

$$C_{opt}(N) = \frac{(w+1)I_{max} + (m-w)P_{max} + (N-1-m)B_{max}}{N} = C(u^*,N) \tag{17}$$

The proof to the first equality is similar to the proof of Case 3 (given in the appendix), and is skipped for brevity. The second equality can be readily deduced from the structure of u^*. Figure 7 summarizes the results from the above four cases.

As mentioned earlier, there can be several 'best' arrangements for a given N. u^* was chosen because it has the following important properties. If N streams are arranged as in (18), the $(N+1)$th stream can be added resulting in a best arrangement of $(N+1)$ streams without disrupting the original structure of the N streams. In other words, u^* of $(N+1)$ streams can be obtained by simply concatenating a single number to u^* of N streams. When N streams are arranged according to u^* and $N \le L$, the removal of *any* stream will still result in a best arrangement. When $N > L$, A best arrangement of N streams for $N = 1, 2, \ldots$, is given by:

$$u^* = (\underbrace{0, 1, 2, \ldots, L-1, 0, 1, 2, \ldots, L-1, \ldots,}_{w \text{ times}} 0, 1, 2, \ldots, N - wL - 1) \quad (18)$$

Optimal effective bandwidth is:

$$C_{opt}(N) = \frac{(w+1)I_{max} + (m-w)P_{max} + (N-1-m)B_{max}}{N} \quad (19)$$

where

$$w \triangleq \text{ largest nonnegative integer } k \text{ that satisfies } N > kL$$
$$m \triangleq \text{ largest nonnegative integer } k \text{ that satisfies } N > kQ$$

Figure 7: Best arrangement of N streams and the associated optimal effective bandwidth.

only the removal of certain streams preserves the optimality of the arrangement.

So far, we assumed that frame boundaries are aligned. However, even if frame boundaries are allowed to have any arbitrary alignment, the best arrangement and the optimal bandwidth in this case are still the same as in (18) and (19), respectively. The justification is quite simple; when frames boundaries are not aligned, the effective bandwidth is greater than or equal the effective bandwidth of some arrangement with aligned boundaries (for example, compare Figure 5 and 6). Since the case of aligned boundaries is included in the generally non-aligned boundaries case, the results in Figure 7 apply as well to the case of non-aligned frame boundaries.

In Figure 8, the variation of $C_{opt}(N)$ (given as a percentage of the source peak rate, I_{max}) is shown as a function of N, using different L and Q values. Maximum frame sizes (I_{max}, P_{max}, and B_{max}) are taken from the *Wizard of Oz* trace (see Section 3.2), which was compressed using $L = 15$ and $Q = 3$. For simplicity, the same maximum sizes are used in to obtain $C_{opt}(N)$ under other L and Q values. Although one might expect that for a given movie, the maximum sizes of compressed frames vary with L and Q, our experiments (discussed below) suggest that compressing a video segment using different (L, Q) pairs has little impact on I_{max}, P_{max}, and B_{max}.

Several noteworthy observations can be inferred from Figure 8. First, as N increases, $C_{opt}(N)$ decreases, but not monotonically, and converges slowly to some positive value. The limiting value of $C_{opt}(N)$ can be determined from (19). For large N, $w \to N/L$ and $m \to N/Q$. Thus,

$$C^*_{opt} \triangleq \lim_{N\to\infty} C_{opt}(N) = (1/L)I_{max} + (1/Q - 1/L)P_{max} + (1 - 1/Q)B_{max} \quad (20)$$

The limiting value of $C_{opt}(N)$ is, in fact, achievable when $N = kL$ for $k = 1, 2, 3, \ldots$, implying

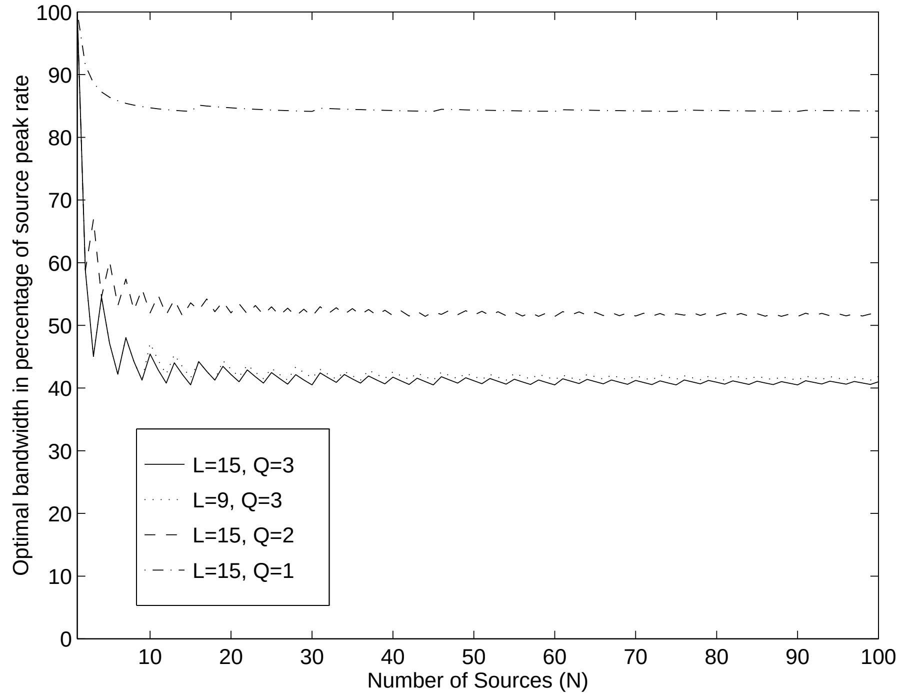


Figure 8: Optimal effective bandwidth, $C_{opt}(N)$ as a function of the number of sources (N), using different L and Q. The values for $C_{opt}(N)$ are given in percentage of I_{max} (source's peak rate). Maximum frame sizes are obtained from Wizard of Oz trace.

that the highest possible multiplexing gains are obtained whenever the number of multiplexed streams is a multiple of L. For moderate and large N, $C_{opt}(N)$ is almost insensitive to N (compare the plots for $(L, Q) = (15, 3)$ and $(L, Q) = (9, 3)$. This is expected since P_{max} is close to (but smaller than) I_{max}. When $P_{max} \approx I_{max}$, C^*_{opt} in (20) reduces to $(1/Q)P_{max} + (1 - 1/Q)B_{max}$ which does not depend on L. On the other hand, the optimal effective bandwidth seems to depend heavily on Q. In the above example, when $L = 15$ and Q is varied from $Q = 1$ (only I and P frames) to $Q = 3$, C^*_{opt} decreased from $C^*_{opt} = 84\%\ I_{max}$ to $C^*_{opt} = 40.5\%\ I_{max}$. Clearly, the relative impact of L and Q depends on the relative values of I_{max}, P_{max}, and B_{max}. In most cases, P_{max} is closer to I_{max} than to B_{max}. We verified that by examining the traces of several MPEG-compressed movies. The movies are listed in Table 2. *Star War* trace was provided by M. Garrett [3]. *Advertisements* and *Lecture* traces were provided by E. Knightly [5]. *Silence of the Lambs* trace was provided by O. Rose [8]. Table 2 gives the maximum frames sizes (in cells) for each trace, the compression parameters, and the limit on the statistical multiplexing gain (given as a percentage of the source peak rate).

To study the impact of L and Q on the maximum sizes of I, P, and B frames, we chose a segment from *Wizard of Oz* movie, and compressed it several times using different L and Q values.

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

Table 2: Empirical MPEG traces for different video movies with various compression patterns (frame sizes in cells). The last column shows C^*_{opt} as a percentage of source peak rate.

The segment corresponds to 12600 frames (from frame No. 29191 to frame No. 41790 in the movie). Table 3 depicts the compression patterns that were used and the measured I_{max}, P_{max}, and B_{max}. In addition, the table gives the limiting value for $C_{opt}(N)$, which is computed from (20). It is clear that the compression pattern has a very insignificant impact on the maximum frames sizes (note, however, that the overall average of frames sizes can considerably vary from one compression pattern to another). This can be intuitively justified by the fact that a movie consists of several 'scenes'. A scene can be loosely defined as a segment of the movie with relatively consistent level of activity. Sizes of I frames (similarly, P and B frames) within a scene are close is value. Since on the average a scene lasts for several seconds [7], changing the compression pattern (whose time scale is smaller than one second) will have little effect on the maximum sizes of I, P, and B frames within a scene.

Compression Pattern	L	Q	I_{max}	P_{max}	B_{max}	$(C^*_{opt}/I_{max}) \times 100\%$
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 3: Encoding of a video segment using different compression patterns.

From the last column of Table 3, it is obvious that L has a very negligible effect on C^*_{opt}, whereas increasing Q results in a significant reduction in C^*_{opt}. However, a large Q means more B frames between successive I/P frames, which is undesirable from the perspective of the decoder. Hence,

Q should be chosen such that it provides a good compromise between the decoder complexity (and the associated decoding delay) and the multiplexing gain.

4 Summary

MPEG encoders often use a pre-specified compression pattern to determine the types of compressed frames. The periodic and deterministic nature of this pattern can be used advantageously in reducing the bandwidth requirements of MPEG traffic streams. By means of statistical multiplexing, we showed that the amount of bandwidth that must be allocated to a source while guaranteeing very stringent QoS requirements (i.e., no cell losses and no queueing delay) can be less than the source peak rate. Bandwidth gains are obtained by exploiting the structure of the compression pattern of the multiplexed streams. The amount of bandwidth gain that can be achieved depends largely of the synchronization structure (i.e., the *arrangement*) of the multiplexed streams. We measure the bandwidth gain using the notion of effective bandwidth. Among all possible *arrangements*, we give the form of the 'best' arrangement that has the optimal (i.e., minimum) effective bandwidth. An expression for the optimal effective bandwidth was also derived. Examples of actual MPEG streams from various compressed movies were presented and used to show the possible bandwidth gains that can be obtained from statistical multiplexing of MPEG streams. The development in this paper assumed that the multiplexed streams are homogeneous with respect to their compression patterns and maximum frames sizes. In a future paper, we extend our development to the case of heterogeneous streams.

Appendix

A Optimal Effective Bandwidth when $Q + 1 < N \leq L$

In this appendix we prove that, when $Q + 1 < N \leq L$, $C(u, N)$ for an arbitrary u satisfies the following inequality:

$$C(u,N) \geq \frac{I_{max} + mP_{max} + (N-1-m)B_{max}}{N} \tag{21}$$

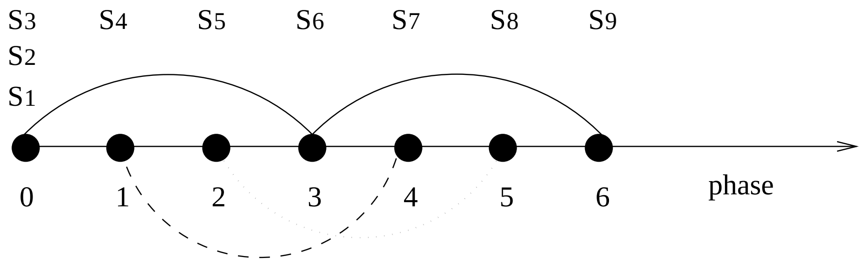
where m is the largest possible integer that satisfies $N > mQ$.

First, suppose that each stream has a distinct phase (i.e., $u_i \neq u_j$ for all $i \neq j$). This possibility can occur because $N \leq L$. Then, there must be at least $m+1$ streams whose phases differ pairwise by a multiple of Q (if $W = \{S_1, S_2, \ldots, S_{m+1}\}$ is the set of such streams, then $|u_i - u_j|$ = a multiple of Q for any S_i and S_j in W, $i \neq j$). Hence, the aggregate bit rate during any phase u_j, where $S_j \in W$, is greater than or equal the RHS of (21). From the definition of the effective bandwidth (see (3)), $C(u,N)$ must be greater than or equal to the aggregate bandwidth during any phase. Therefore, $C(u,N)$ must satisfy (21).

Next, suppose that at least two of the N streams have the same phase, i.e., $\exists$ a phase i such that $r_i \geq 2$. Let

$$\alpha \triangleq \max_{0 \leq j \leq L-1} r_j \tag{22}$$

Note that $\alpha \geq 2$. Denote the jth stream by S_j, $j = 1, 2, \ldots, N$. We will use the term *chain* to refer to a subset of the N streams whose phases differ pairwise by a multiple of Q (including streams that have the same phase). For example, if $N = 9$, $L = 15$, $Q = 3$, and $u = (0,0,0,1,2,3,4,5,6)$, then one chain consists of the sources $\{S_1, S_2, S_3, S_6, S_9\}$, a second chain consists of $\{S_4, S_7\}$, and the last chain consists of $\{S_5, S_8\}$. The three chains are shown below. In this example, $\alpha = 3$.



From the definition of a chain, it is easy to see that there can be no more than Q chains in a given arrangement. Let q be the number of chains ($q \leq Q$). Let the chains be denoted by W_1, $W_2, \ldots, W_q$, with corresponding sizes $\eta_1, \eta_2, \ldots, \eta_q$ ($\sum_j \eta_j = N$). For each chain W_j, let $C_j(u,N)$ be the maximum aggregate peak rate divided by N, where the maximization is taken only over the time intervals that are composed of the phases of the streams in W_j. For $j = 1, \ldots, q$, $C_j(u,N)$ can be given by:

$$C_j(u,N) = \frac{n_I^{(j)} I_{max} + n_P^{(j)} P_{max} + n_B^{(j)} B_{max}}{N} \tag{23}$$

where $n_I^{(j)} + n_P^{(j)} + n_B^{(j)} = N$. For any chain W_j, the total number of streams sending I or P frames during the phase of any stream in W_j is given by η_j. Clearly, η_j is the same for all the phases of streams in W_j (see Proposition 1). At least one of the chains, say W_1, contains α streams that are in the same phase, say phase i. Hence, it must be true that $C_1(u,N)$ results from the aggregate bit rate during phase i. Therefore, $n_I^{(1)} = \alpha$. Based on the definition of $C(u,N)$,

$$C(u,N) = \max_{j \in \{1,\ldots,q\}} C_j(u,N) \tag{24}$$

which implies that

$$C(u,N) \geq \frac{\sum_{j=1}^{q} C_j(u,N)}{q} \tag{25}$$

$$= \frac{1}{q}\frac{I_{max}\sum_{j=1}^{q} n_I^{(j)} + P_{max}\sum_{j=1}^{q} n_P^{(j)} + B_{max}\sum_{j=1}^{q}(N - n_I^{(j)} - n_P^{(j)})}{N} \tag{26}$$

Replacing $n_P^{(j)}$ by $\eta_j - n_I^{(j)}$, and with some rearrangements, (26) becomes:

$$C(u,N) \geq \frac{1}{q}\frac{(I_{max} - P_{max})\sum_{j=1}^{q} n_I^{(j)} + P_{max}\sum_{j=1}^{q} \eta_j + B_{max}\sum_{j=1}^{q}(N - \eta_j)}{N} \tag{27}$$

From Proposition 2, $n_I^{(j)} \geq 1$ for $j = 2, \ldots, q$. Moreover, $n_I^{(1)} = \alpha$. Thus,

$$\sum_{j=1}^{q} n_I^{(j)} \geq \alpha + q - 1 \tag{28}$$

Note that $\sum_{j=1}^{q} \eta_j = N$ and $I_{max} > P_{max}$. Thus, (27) reduces to

$$C(u,N) \geq \frac{1}{q}\frac{(\alpha + q - 1)(I_{max} - P_{max}) + NP_{max} + (qN - N)B_{max}}{N} \tag{29}$$

which can be written as

$$C(u,N) \geq \frac{sI_{max} + lP_{max} + (N - s - l)B_{max}}{N} \tag{30}$$

where

$$s \triangleq \frac{\alpha + q - 1}{q} \tag{31}$$

$$l \triangleq \frac{N - \alpha - q + 1}{q} \tag{32}$$

Note that (30) is the same as (15). We only need to show that $s + l \geq m + 1$. From (31) and (32), we have

$$s + l = \frac{N}{q} \geq \frac{N}{Q} \tag{33}$$

(since $q \leq Q$). But $N > mQ$, or equivalently, $N \geq mQ + 1$. Therefore,

$$s + l \geq m + \frac{1}{Q} \tag{34}$$

However, the expression for the effective bandwidth must consist of integer number of I_{max} and

P_{max} in the numerator of (4). Thus, it must be true that s and l in (30) satisfy $s + l \geq m + 1$. When $s + l \geq m + 1$, the RHS of (21) is smaller than the RHS of (30), which implies that $C(u, N)$ satisfies (21) for any arrangement u. Therefore,

$$C_{opt}(N) = \frac{I_{max} + mP_{max} + (N - 1 - m)B_{max}}{N}. \tag{35}$$

□

References

[1] A. I. Elwalid and D. Mitra. Effective bandwidth for general Markovian traffic sources and admission control of high speed networks. *IEEE Journal on Selected Areas in Communications*, 1(3):329–343, June 1993.

[2] V. S. Frost and B. Melamed. Traffic modeling for telecommunications networks. *IEEE Communications Magazine*, 32(3):70–81, Mar. 1994.

[3] 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.

[4] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE Journal on Selected Areas in Communications*, 9(7):968–981, Sept. 1991.

[5] 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.

[6] 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.

[7] M. Krunz, S. Tripathi, and H. Hughes. A source model for MPEG-coded video movies. In *First IEEE Workshop on ATM Networks*, Washington, DC, Oct. 1995.

[8] 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.