Video-on-Demand Over ATM: Constant-Rate Transmission and Transport

Jean M. McManus, *Member, IEEE*, and Keith W. Ross, *Senior Member, IEEE*

***Abstract*— We introduce a specific transport and transmission scheme for video-on-demand (VoD) called constant-rate transmission and transport (CRTT). CRTT establishes a constant bit-rate (CBR) virtual channel between the video provider and the viewer's set-top box (STB) and then transmits cells from the provider into this channel at a constant rate. Since we assume that the number of cells in a frame is variable, CRTT requires that some number of cells be built up in an STB buffer before the commencement of playback. The build up, cell transmission rate, and the set-top memory size must be chosen so that there is no starvation or overflow at the STB. We develop fundamental relationships between these parameters for viable CRTT. We then apply the theory to an MPEG encoding of *Star Wars* and find that the minimal STB memory for CRTT is 23 Mbytes. We also consider varying the constant rate over a small number of intervals. We find, for example, that for *Star Wars* approximately 2 Mbytes of set-top memory suffices with 32 constant-rate intervals.**

I. INTRODUCTION

VIDEO-on-demand (VoD) is a technology which enables a viewer to choose a video from a large selection, specify the video's start time, and have the video sent over a telecommunication network to his home. To achieve this, a VoD system can offer a wide range of services. At one extreme there is the ultra-deluxe VoD service—the video instantaneously begins to play upon request, has a high quality image on a large screen with stereo sound, and can be paused, rewound, and fast forwarded by the viewer (so-called VCR control). At the other extreme, there is the VoD economy service—the video does not begin instantaneously, but instead the viewer must schedule the start of the video from one to twenty four hours in advance; the image is VCR quality, but the viewer has no VCR control. Most likely, VoD technologies will evolve so that the viewer not only chooses from a large selection of videos, but also from a multitude of VoD services. For each session, the viewer will choose a video and a VoD service—and will pay a commensurate price. For overviews of VoD, see [1], [2], [18], and [23].

The range of services available to the viewer depends on the characteristics of the key components of the end-to-end VoD architecture. The key components, which range from physical devices to data, are the following: 1) the encoded videos, 2) the video server, 3) the ATM network, and 4) the viewer's set-top box (STB) (see Fig. 1). The video server must retrieve, upon request, streams of encoded videos from its storage. These streams are multiplexed and transmitted by the server into the ATM network. The ATM network transports the video streams to the viewers' STB's. The STB decodes the video and eventually forwards it to the viewer's video display. Kwok [16] provides examples of ATM-to-the-home architectures that utilize telco and CATV infrastructure and can support VoD.

There are two broadly defined regulatory environments for an end-to-end VoD system. In the first, a telecommunications company and a video provider team to form one company. In a given geographical region, there may be many such telecommunication/video-provider companies, but a viewer subscribes to only one of them. The viewer leases from its chosen company a proprietary STB which conforms with the company's transport, video server, and storage technologies. In the second regulatory environment, the telecommunications companies are required to provide equal access to a number of competing video providers; the viewer "surfs" over the providers and chooses a video from any of them. For this environment, the STB's are purchased in retail stores and have unique, model-dependent features (for example, size of local memory); but these STB's also have common characteristics which conform to industry standards (for example, a standard for the decoding algorithm). In both environments, the video provider owns the video server and the encoded videos. Although our observations and results are applicable to both environments, for the sake of clarity and presentation, our discussion focuses on the "open-system" environment, i.e., on the second regulatory environment.

A VoD service provided to a viewer can be either constant quality or variable quality. With constant-quality service, the video quality remains constant throughout playback; with variable-quality service, the quality of the image may occasionally drop below the nominal quality during playback. This degradation can be due to a variety of factors, including to ATM cell loss in the network and to modification of the MPEG quantization matrix during encoding. *In this paper we only consider end-to-end architectures which provide constant-quality service.* Although variable-quality service can be provided at lower cost, we briefly note that viewers have been spoiled with constant-quality service since the beginning of the motion picture era, and they may be reluctant to pay for anything less.

The ATM network can transport a video with constant bit-rate (CBR) transport, variable bit-rate transport (VBR), or best-effort transport. We introduce a new traffic management

---

Manuscript received April 27, 1995; revised October 3, 1995. This work was supported in part by the National Science Foundation under Grant NCR93-04601 and the U.S. Department of Education under the Patricia Roberts Harris Fellowship.

The authors are with the Department of Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (email: mcmanusj@eniac.seas.upenn.edu and ross@eniac.seas.upenn.edu).

Publisher Item Identifier S 0733-8716(96)04885-8.

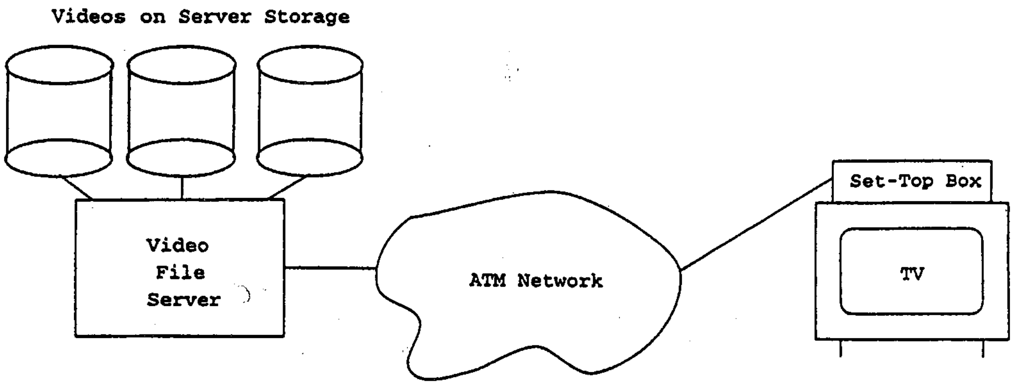


Fig. 1. A VoD system

scheme—constant-rate transmission and transport (CRTT). We shall argue that CRTT is attractive from both cost and implementation perspectives. This scheme establishes a CBR virtual channel between the video provider and the viewer's STB, and then transmits cells from the provider into this channel at a constant rate. Since the number of cells in a frame is variable, CRTT requires that a large number of cells be transmitted and stored in a STB buffer before the commencement of playback. We refer to the number of frames stored in the STB as the *build up*. In order for CRTT to be viable, the build up, cell transmission rate, and the set-top memory must be chosen so that there is no starvation or overflow at the STB. We shall develop fundamental relationships between the build up, the cell transmission rate, and the size of the set-top memory for viable CRTT. We shall then apply the theory to an MPEG encoding of *Star Wars*. In particular, the theory will enable us to determine the minimal memory for *Star Wars* to be viewed without starvation or overflow. Furthermore, we note that the fundamental reslationships developed in this paper also apply to non-ATM networks that can guarantee a CBR channel for transmission.

We also introduce piecewise-constant-rate transmission and transport (PCRTT). PCRTT is a variation of CRTT, whereby the constant transmission and transport rate is allowed to vary over a small number of time intervals. We develop fundamental relationships between the PCRTT transmission rates, the STB memory size, and the preplay build-up, and apply the theory to *Star Wars*.

Issues related to the transport of real-time or live video through packet-switched and ATM networks have received significant attention in recent years (see [3], [5], [6], [14], [17], and [21]). Prerecorded video, the focus of this paper, is different from live video—the traffic of a prerecorded source is completely known *a priori*.

Before discussing CRTT in greater detail, in Section II we examine the key components of the end-to-end architecture. In Section III, we focus on one of these components, namely, the memory in the STB and how its size impacts the choice of an ATM transport scheme. In Sections IV–VI, we study CRTT in some depth. In Section VII, we introduce PCRTT.

II. KEY SYSTEM COMPONENTS

Throughout this paper we assume that i) the videos have F frames per second, ii) the videos are digitally encoded with an MPEG standard and stored on mass storage devices at the video provider, iii) at the time of transmission, each MPEG encoded frame is segmented and encapsulated into ATM cells, and iv) the telecommunications network transports the ATM cells over a virtual channel (VC) from the provider's video server to the viewer's STB.

A. The Encoded Videos

An MPEG encoded video can be encoded using VBR or CBR encoding [13]. When the video is VBR encoded, the number of bytes in each encoded frame varies as a function of the complexity of the scene, the extent of the scene change from frame to frame, and the type of MPEG frame used (I, B, or P). For CBR encoding, by varying the quantization matrix and hence the image quality, each MPEG-encoded frame has a constant number of bits. Since we are limiting our study to constant quality VoD services, throughout this paper we assume that all videos are VBR encoded.

Unlike real-time video services, encoded videos for VoD are stored as files on the server storage. For a given video, let N denote its number of frames (encoded or decoded) and let x_n denote the number of cells in its nth encoded frame. We find it convenient at this juncture to collect some notation that we will use throughout the paper

$$C = \sum_{n=1}^{N} x_n$$

$$T = \frac{N}{F}$$

$$x_{\max} = \max_{1\leq n\leq N} x_n$$

$$x_{\text{avg}} = \frac{C}{N}.$$

Note that C is the total number of cells in the movie, T is the length in seconds of the movie, $x_{\max}$ is the maximum number of cells in a frame, and x_{avg} is the average number of cells in a frame.

B. The Video Server

In addition to the encoded videos, the video provider owns (at least) one video server with associated storage. The server is responsible for simultaneously reading multiple encoded videos from mass storage devices, segmenting and encapsulating the frames into ATM cells, and transmitting the cells into the ATM network.

The video server will typically be directing cells from multiple videos into the access link over extended intervals of time. Thus, the server must schedule cell slots for each VC in progress. The complexity of this scheduling policy depends on the ATM transport service employed by the VC's.

There are also a number of issues associated with retrieving streams of encoded video from the multimedia server's storage devices, such as block location and disk latency [9], [20]. These issues are beyond the scope of this paper. We shall simply assume that the server is capable of retrieving encoded video from its storage at adequate speeds.

C. The ATM Network

We expect the network company to offer a variety of transport types to the video provider. Before sending a video to a STB, the video provider chooses a specific transport type from the networking company. The networking company then establishes a VC from the provider to a STB and delivers the video with the chosen transport type. The transport types broadly fall into three categories. The first category is CBR transport, whereby the ATM network establishes a VC that emulates a CBR circuit between the video server and the viewer's STB. This service guarantees no or negligible network cell loss, delay, and jitter. It is fully characterized by a constant bandwidth, denoted by b, which is negotiated at VC establishment. Note that CBR transport does *not* require the video server to transmit cells into the VC at a constant cell rate; it only requires that the peak cell transmission rate be no more than the negotiated constant bandwidth.

The second category of transport types is VBR transport. If a VC uses VBR transport, its cells are statistically multiplexed with cells of other VC's in the network links along the VC's route. This transport type also guarantees that stringent quality of service (QoS) requirements are met. Compared with CBR transport, VC's with VBR transport are more difficult to manage, both in the network and at the video provider, due to the traffic variability that the scheduling algorithms and the call admission controls must accommodate.

It is important to note that transport of a prerecorded video is in many ways different from transport of real-time video, such as video conferences or the live broadcast of a sporting event. For real-time video, the exact dynamics of the VBR traffic are unknown to the network. On the other hand, in VoD the cell dynamics of the prerecorded VBR traffic (i.e., x_n, $n = 1, \cdots, N$) are fully known before the beginning of server transmission and network transport. The network and video provider should exploit this information in order to efficiently manage their resources.

The third category of transport types is best-effort transport. When a VC uses best-effort transport, the network transports its cells to the STB with little if any guarantees on cell delay or jitter. Best-effort transport is only feasible for STB's with large memory and for viewers who are willing to request a start time in the distant future. Best-effort transport, however, should be relatively inexpensive, particularly if the requested start time is sufficiently in the future to allow for cell transport during off-peak hours.

TABLE I
SET-TOP BOX MEMORY OPTIONS

set-top box	storage feature	memory requirement
minimal memory	buffers less than a second of a two-hour movie	few dozen frames
moderate memory	can buffer a portion of a two-hour movie	1 MB - 1GB
full memory	can buffer an entire two-hour movie	> 1GB

D. The STB

At the very least, the STB will have to assemble the arriving ATM cells into MPEG frames, decode the MPEG frames, and eventually forward the decoded frames to the video display. We expect some models to have minimal memory, being capable of buffering at most a few dozen frames. Due to the absence of buffering, these models require that frames arrive to the STB at a constant rate of F frames per second. We expect other models to be capable of buffering an entire encoded movie in RAM, hard-disk, or some tertiary medium such as VHS tape. (Copyright laws may restrict the sale of STB's that are capable of storing an entire movie. We ignore such legal issues, as they are beyond the scope of this paper.) An encoded movie can require several gigabytes of memory. We also envision models with one megabyte to one gigabyte of memory, capable of storing a portion but not all of an encoded movie. Thus, we classify STB's as having either minimal memory, moderate memory, or full memory; see Table I.

III. SET-TOP BOX MEMORY AND ITS IMPACT ON TRANSPORT SERVICES

VoD transport cost may reflect the start time of playback, the time the movie is requested, and the transport type actually used (that is CBR, VBR, or best-effort). In this section, we explore how the size of the memory in the STB impacts the range of transport types that are available. In all of our examples, we assume that the movie is T seconds long.

A. Minimal Memory

Suppose the viewer owns a STB with minimal memory. Further suppose the viewer indicates a specific time for its commencement, say at 9 p.m. In order for the video provider to satisfy this request, it must begin transmission of the video at 9 p.m., and its video server must transmit the MPEG frames at a constant rate of F frames per second. Each of these frames contains a variable number of cells, so that the server must transmit VBR traffic into the network. To guarantee negligible cell loss, delay, and jitter for this VBR source, the video provider must select an appropriate network transport type—one which reserves substantial transmission and switching resources for the duration of the video during the busy hours. The network company will charge the video provider a premium for this session, a premium that the viewer will ultimately pay.

In this scenario, both CBR and VBR transport can be used to deliver the video to the STB. For *both* transport schemes, at 9:00 p.m. the server transmits x_1 cells in the first $1/F$ second interval, x_2 cells in the second $1/F$ second interval, $\cdots$, x_N cells in the Nth $1/F$ second interval. For CBR transport, we need $b = Fx_{\max}$ cells/second in order to guarantee no cell loss at the server/network interface. With VBR transport, although a VC's cells are statistically multiplexed in the network with those of other VC's in progress, they are transported with low loss and jitter.

Because CBR transport reserves network resources at peak rate, we expect it to be more expensive than VBR transport, for which it may be possible to reserve resources closer to the average rate if there is substantial statistical multiplexing in the network. An effective bandwidth attempts to quantify the amount of bandwidth that must be allocated to a VBR source. Due to statistical multiplexing that occurs in ATM networks, the effective bandwidth for a VBR source will typically be less than the source's peak rate and often approach its average rate. Generally speaking, given a specific probability of cell loss and the particular statistics of a source, the effective bandwidth can be approximated (see [4], [10], [11], and [15]). How close the effective bandwidth is to the average rate for entertainment video depends on the link capacity, and remains a subject of debate.

For both CBR and VBR transport, we assume the following pricing structure for a VC

$$\text{Transport Cost } = Kyt. \tag{1}$$

Here, y is the VC's effective bandwidth (in cells per second), t is the VC's duration, and K is a constant which depends on the time of day during which the VC is transported over the network. (We expect K to be highest in the early evening and lowest in the early morning.) Denote by K_P, the K value corresponding to transport from 8–11 p.m.

Returning to our scenario, for a CBR VC, $Fx_{\max}$ cells/second are reserved for the duration of the movie; thus, $y = Fx_{\max}$ and $t = T$, giving

$$\text{CBR Transport Cost } = K_P F x_{\max} T. \tag{2}$$

For a VBR VC, we have

$$\text{VBR Transport Cost } = K_P y T. \tag{3}$$

Therefore, the relative cost of CBR transport to VBR transport is $Fx_{\max}/y$, which is what some authors call the statistical multiplexing gain. We note that it is also possible to smooth over an MPEG group of pictures (GOP's). Let $x_{\max}^{\text{GOP}}$ be the maximum of the average number of cells in a GOP. With such smoothing, the statistical multiplexing gain is reduced to $Fx_{\max}^{\text{GOP}}/y$.

ATM network management with CBR VC's is significantly less complicated than with VBR VC's. For example, call admission based on peak rates is well understood [22], whereas determining effective bandwidths for complex VBR sources remains a challenging problem. Furthermore, VBR transmission can complicate call admission and cell scheduling algorithms at the server.

B. Full Memory

Now suppose the viewer's STB can buffer an entire VBR-encoded movie. Further suppose that the viewer specifies 9 p.m. for commencement of the movie, but he makes this request 24 h in advance. The video provider can then begin transmission at an off-peak hour (say 2 a.m.) and can use a network transport scheme that offers minimal guarantees for cell delay and jitter, such as best-effort. In this case, we expect the transport cost to be proportional to the number of cells in the video, with the proportionality constant depending on the time of day during which the video is transported. More explicitly, we expect

$$\text{Best-Effort Transport Cost } = K_A C = K_A F x_{\text{avg}} T \tag{4}$$

where the off-peak hour constant K_A is much less than the peak-hour constant K_P used for STB's with minimal memory. Comparing (3) and (4), we see that our cost estimates predict great savings for best-effort transport. These savings, however, require that the viewer have full memory and the planning skills that are required to order his movie well in advance.

C. Moderate Memory

Suppose that viewer's STB has moderate memory. Further suppose that the viewer requests the movie with a specific start time that is in the near future, say at a time from one to ten minutes from the present time. As with full memory, the STB can build up frames in its buffer prior to play.

We do not believe that best-effort transport is feasible in this scenario, because there may be insufficient time or buffer space to build up enough frames and preclude starvation at the STB. Thus, moderate memory requires either CBR or VBR transport.

For minimal memory, we stated that the transport cost of VBR would be less than that of CBR, because CBR requires a constant cell rate, b, equal to the peak cell rate of the video. But with moderate to full memory, CBR transport can employ a lower constant cell rate by building up frames in the memory before playback. This approach could make CBR transport less expensive than VBR transport. In the next section we explore the design of CBR transport for STB's with moderate to full memory.

IV. CONSTANT-RATE TRANSMISSION AND TRANSPORT

In this section, we introduce a new scheme for providing VoD to the viewer: CRTT. With CRTT, the network establishes a CBR VC of constant bandwidth b between the server and the STB. To fully utilize the bandwidth of this VC, the server *transmits* the cells at the same constant rate b into this VC. To ensure constant quality, the cells of the first d frames are built-up in the STB prior to the video beginning play. The server transmits the remaining $N - d$ frames at constant rate b while the video is playing.

CRTT, although not as inexpensive as best-effort transport during off-peak hours, could offer considerable savings over

VBR transport. More specifically, using the pricing structure of Section III-A, we have $y = b$, $t = C/b$, and

$$\text{CRTT Transport Cost } = K_P b \frac{C}{b} = K_P F x_{\text{avg}} T \quad (5)$$

where we assumed that the transport occurs at the peak hours. Comparing (5) with (3), we see that CRTT is less expensive than VBR transport. Furthermore, the file server and the network can manage CRTT more easily than schemes that use VBR transport.

A. Feasibility of CRTT for a Given Video

When using a CRTT-based VoD service, the CBR VC bandwidth b and the pre-play frame build-up d must be determined such that the STB does not experience starvation or buffer overflow during play. Starvation occurs when a frame is scheduled for play and the cells needed to decode the frame have not yet arrived. Buffer overflow occurs when the cells arrive to a STB whose buffer is already full with cells.

Given a specific cell stream x_n, $n = 1, \cdots, N$, and set-top memory B in cells, how can we efficiently determine the set of feasible (b, d) pairs? For which of these pairs is the pre-play build-up delay minimized? What is the smallest STB memory B such that there exits a feasible pair (b, d)? We now answer these questions. Throughout our analysis we use the following convention: As soon as $\sum_{i=1}^{d} x_i$ cells arrive and are stored in the STB, the first frame is instantaneously removed and displayed; subsequent frames are removed and displayed every $1/F$ seconds. This convention implies that at n/F seconds from the start of playback, frame $(n+1)$'s cells are instantaneously removed from the STB and displayed. For a fixed d, let $b_{\min}(d)$ be the minimum VC transport rate which guarantees no starvation during playback.

Lemma 1: For all $d = 1, \cdots, N$

$$b_{\min}(d) = \max_{d \le n \le N-1} \frac{F}{n} \left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i \right).$$

Furthermore, $b_{\min}(d)$ is decreasing in d.

Proof: The number of cells that have been received by the STB at time n/F, $0 \le n \le N-1$, is

$$\text{cells received } = \min \left\{ \sum_{i=1}^{d} x_i + \frac{bn}{F}, C \right\}.$$

The two terms added together are the pre-play frame build-up and the number of cells received since the start of play. Since this sum cannot exceed the total number of cells in the video, the number of cells received never exceeds C. The number of cells that have been removed from the STB at time n/F is

$$\text{cells removed } = \sum_{i=1}^{n+1} x_i.$$

The difference between cells received and cells removed is

$$f(n) = \min \left\{ \frac{bn}{F} + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i, C - \sum_{i=1}^{n+1} x_i \right\}. \quad (6)$$

Note that $f(n)$ is negative if and only if starvation occurs at time n/F. Thus, for a given d and x_n, $n = 1, \cdots, N$, we must find a b such that $f(n) \ge 0$ for all $n = 0, \ldots, N-1$. Since $C - \sum_{i=1}^{n+1} x_i$ is always nonnegative, and since $bn/F + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i$ is always nonnegative for $n \le d-1$, it follows from (6) that $f(n) \ge 0$ for $n = 0, \ldots, N-1$, if and only if

$$\frac{bn}{F} + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i \ge 0 \quad \text{for all } n = d, \cdots, N-1$$

or, equivalently, if and only if

$$b \ge \frac{F}{n} \left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i \right) \quad \text{for all } n = d, \cdots, N-1$$

from which the lemma directly follows. □

For a fixed d and set-top memory B, let $b_{\max}(d, B)$ be the maximum VC transport rate b to guarantee no overflow at the STB. Note that for a fixed memory size B, the amount of build up is constrained by $d \le \bar{d}(B)$, where

$$\bar{d}(B) := \max \left\{ d\colon \sum_{i=1}^{d} x_i \le B \right\}.$$

Also define

$$N(B) := \max \left\{ n\colon \sum_{i=1}^{n} x_i + B < C \right\}.$$

Lemma 2: For $B > 0$ and $d \le \bar{d}(B)$

$$b_{\max}(d, B) = \min_{0 \le n \le N(B)-1} \frac{F}{n+1} \left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i + B \right).$$

Moreover, $b_{\max}(d, B)$ is decreasing in d.

Proof: We can express the occurrences of buffer overflow at times n/F, $n = 0, \cdots, N-1$ by

$$\max_{0 \le n \le N-1} f(n) \ge B.$$

It is, however, insufficient to only check the discrete play instants since cells arrive between these points. The number of cells that arrive during such a $1/F$ second interval is b/F. Thus, there is no overflow if $g(n) \le B$ for all $n = 0, \cdots, N-1$, where

$$g(n) = \min \left\{ \frac{(n+1)b}{F} + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i, C - \sum_{i=1}^{n+1} x_i \right\}.$$

Note that $g(n)$ is nothing but $f(n)$ except for an extra b/F term in the first argument of the minimum operation. Also note that

$$g(n) - B = \min \left\{ \frac{(n+1)b}{F} + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i - B, \; C - \sum_{i=1}^{n+1} x_i - B \right\}.$$

We are assured that $g(n) - B \leq 0$ for $n > N(B) - 1$. Thus, there is no overflow if and only if

$$\frac{(n+1)b}{F} + \sum_{i=1}^{d} x_i - \sum_{i=1}^{n+1} x_i - B \leq 0 \qquad \text{for all } 0 \leq n \leq N(B) - 1$$

or, equivalently, if and only if

$$b \leq \frac{F}{n+1}\left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i + B\right) \qquad \text{for all } 0 \leq n \leq N(B) - 1$$

from which the lemma follows. □

For a given B and d, we can use the formulas in Lemmas 1 and 2 to calculate $b_{\min}(d)$ and $b_{\max}(d, B)$ in $O(N)$ time. The total effort required to calculate these quantities for all $d = 1, \cdots, N$ is $O(N^2)$. Let

$$\mathcal{D}(B) := \{d\colon d \leq \bar{d}(B),\ b_{\min}(d) \leq b_{\max}(d, B)\}$$

be the set of all feasible frame build ups. Let

$$d_{\min}(B) := \min\{d\colon d \in \mathcal{D}(B)\}$$

be the minimum frame build up for viable CRTT for fixed B. Let

$$B_{\min} = \min\{B\colon \mathcal{D}(B) \neq \phi\}$$

be the absolute minimum set-top memory requirement for viable CRTT. Note for a given frame build up d and transport rate b, $\sum_{i=1}^{d} x_i/b$ is the build-up delay in seconds perceived by the viewer. For fixed d, the minimum build-up delay is $\sum_{i=1}^{d} x_i/b_{\max}(d, B)$. As an immediate consequence of Lemmas 1 and 2 we have

Theorem 1:

1) For a given B and $d \leq \bar{d}(B)$, there exists a feasible constant transmission rate b if and only if

$$\max_{d \leq n \leq N-1} \frac{F}{n}\left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i\right) \leq \min_{0 \leq n \leq N(B)-1} \frac{F}{n+1}\left(\sum_{i=1}^{n+1} x_i - \sum_{i=1}^{d} x_i + B\right).$$

2) Fix B such that $\mathcal{D}(B) \neq \phi$. The transmission rate b and frame build-up d that minimize the build-up delay $\sum_{i=1}^{d} x_i/b$ are $d = d_{\min}(B)$ and $b = b_{\max}(d_{\min}(B), B)$.

To find $B_{\min}$, we note that $\mathcal{D}(B)$ is an increasing set as B increases, that is, $\mathcal{D}(B_1) \subseteq \mathcal{D}(B_2)$ for $B_1 \leq B_2$. Therefore, we can find the smallest B such that $\mathcal{D}(B) \neq \phi$ with a binary search over $[0, C]$. The overall worst-case effort to find $B_{\min}$ is $O(N^2 \ln C)$.

B. An Upper Bound for Memory Requirements

Because N is typically quite large, the $O(N^2 \ln C)$ complexity for calculating $B_{\min}$ translates to long run times on modern workstations. In this section, we present an $O(N)$ algorithm for calculating an upper bound of $B_{\min}$. Our computational tests seem to indicate that the bound is rather tight.

Let d' be the minimum buildup to guarantee no starvation when the constant transmission rate is b_{avg}. An analysis similar to that for Lemma 1 gives $d' \leq \bar{d}$, where

$$\bar{d} = \min_{1 \leq d \leq N}\left\{d\colon \sum_{i=1}^{d} x_i \geq \delta\right\}$$

with

$$\delta := \max_{0 \leq n \leq N-1}\left\{\sum_{i=1}^{n+1} x_i - \frac{n b_{\text{avg}}}{F}\right\}.$$

Note that $\bar{d}$ can be calculated in $O(N)$ time. Also if we use the constant transmission rate b_{avg} with the buildup $\bar{d}$, there is no starvation. Now let B' be the minimum memory to guarantee no buffer overflow when the constant transmission rate is b_{avg} and the buildup is d'. An analysis similar to that for Lemma 2 gives $B' \leq \bar{B}$, where

$$\bar{B} = \max_{0 \leq n \leq N-1}\left\{\frac{(n+1)b_{\text{avg}}}{F} - \sum_{i=1}^{n+1} x_i\right\} + \sum_{i=1}^{\bar{d}} x_i.$$

Note that $\bar{B}$ can be calculated in $O(N)$ time. Also, if we use the constant transmission rate b_{avg}, buildup $\bar{d}$, and memory $\bar{B}$, there is no overflow. Since there is also no starvation with these parameters, it follows that these three parameters give viable CRTT and that $B_{\min} \leq \bar{B}$. We can calculate this upper bound in $O(N)$ time by first calculating δ, then $\bar{d}$, and then $\bar{B}$. In the next section, we shall explore the accuracy of the upper bound.

V. NUMERICAL RESULTS

We now assess whether CRTT can be used to transport a specific full-length MPEG1 encoded movie. We apply the results of Section IV to determine 1) the set of feasible (b, d) pairs, 2) the minimum STB memory, $B_{\min}$, and 3) the bound $\bar{B}$ on $B_{\min}$.

A. MPEG Trace

The trace used in our analysis is a *Star Wars* MPEG1 bandwidth trace available via anonymous FTP [7]. The trace gives the number of bits in each video frame. Each frame engenders a number of ATM cells. We have assumed that all 48 bytes of the payload can be used to transport the frames.

The trace consists of 174 136 frames. Since the movie frame rate is 24 frames/s, the movie is 2 h and 56 s in length. As described in [8], the original video was captured as 408 lines by 508 pels. It was interpolated and filtered to standard common intermediate format (CIF) frame size. There are 12 frames in each MPEG GOP with an I, B, P frame pattern of IBBPBBPBBPBB. There are an average of 41.12 cells per frame with a peak and a minimum of 483 cells per frame and

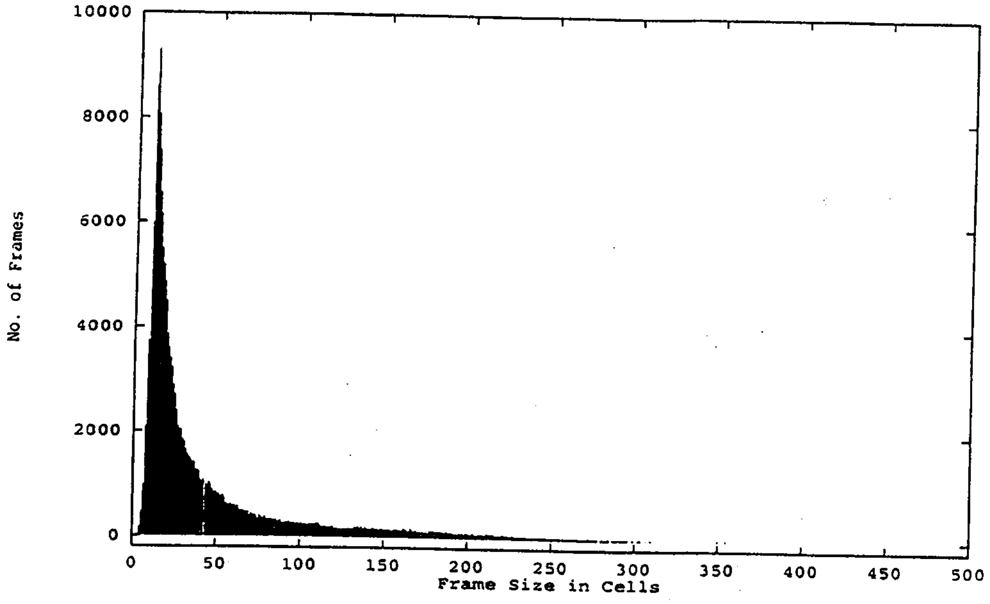


Fig. 2. Cells per frame distribution.

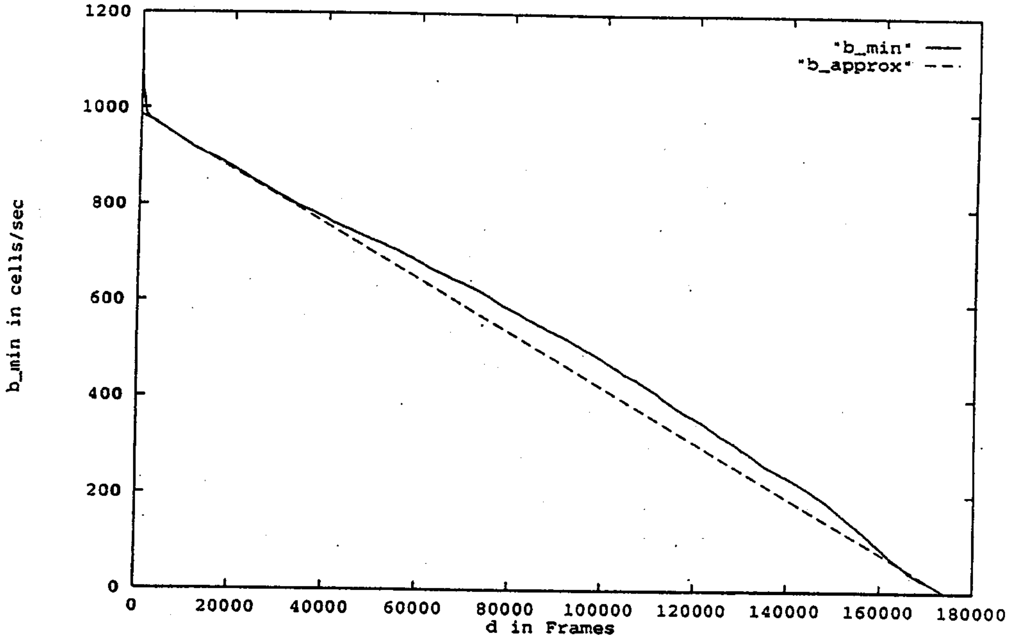


Fig. 3. $b_{\min}(d)$ for $d = 1, \cdots, 174\,136$.

two cells per frame, respectively. Fig. 2 shows the cells per frame distribution.

B. Results Summary

The relationship between $b_{\min}(d)$ and $b_{\max}(d, B)$ determine whether a feasible (b, d) pair exists for a particular movie. We have calculated $b_{\min}(d)$ and $b_{\max}(d, B)$ for $B = 895\,070$ cells (equal to 12.5% of the movie) to explore this relationship for *Star Wars*.

Figs. 3 and 4 display $b_{\min}(d)$ for $d \in [1, N]$ and for $d \in [1, 3000]$, respectively. If the traffic were perfectly smooth so that $x_n = x_{\rm avg}$ for all $n = 1, \cdots, N$, then a simple analysis gives $b_{\min}(d) = \tilde{b}_{\min}(d)$, where

$$\tilde{b}_{\min}(d) := b_{\rm avg} - \frac{d}{N} b_{\rm avg}$$

and where $b_{\rm avg} := Fx_{\rm avg}$ (= 987 cells/s for *Star Wars*). The plot of the approximation $\tilde{b}_{\min}(d)$ is also included on Figs. 3 and 4.

Note that $b_{\min}(d)$ begins with a value larger than $b_{\rm avg}$, precipitously drops in the range $d \in [1, 1200]$, then roughly follows the linear approximation $\tilde{b}_{\min}(d)$. Insight into the behavior of $b_{\min}(d)$ can be gleaned from Fig. 5, which plots the x_n's over the first few minutes of the movie. Fig. 5 shows that there are three bursts over this time period, with the third burst (frames 1000 to 2500) being particularly menacing. For $d = 1$, a high transmission rate is needed to ensure that there will be sufficient cell accumulation in the STB buffer at the onset of a burst. As d increases from 1 to 1000, $b_{\min}(d)$ remains significantly above $\tilde{b}_{\min}(d)$ because the latter transmission rate fails to provide enough cells to surmount the

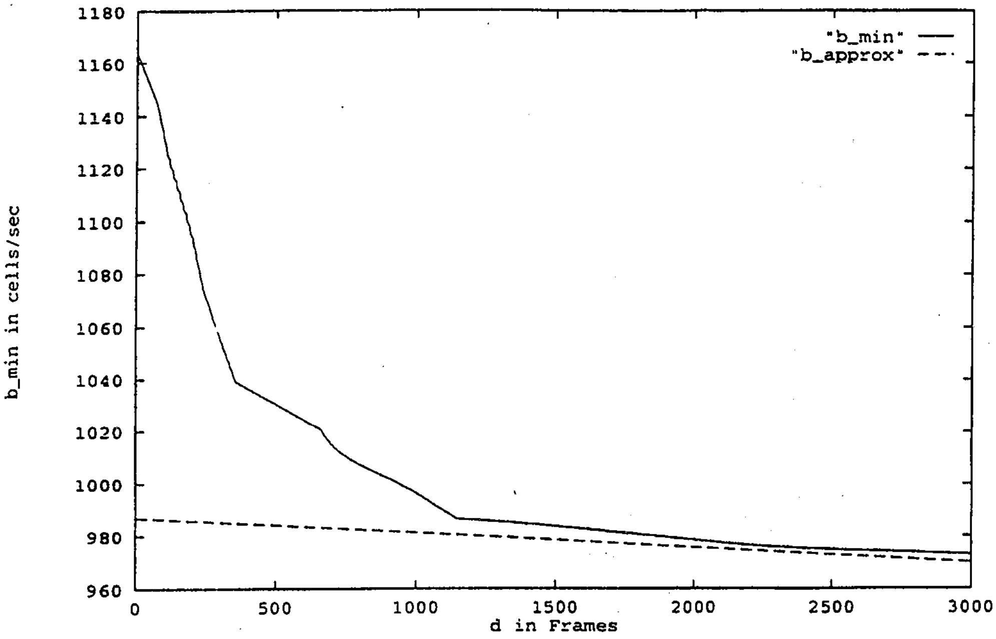


Fig. 4. $b_{\min}(d)$ for $d = 1, \cdots, 3000$.

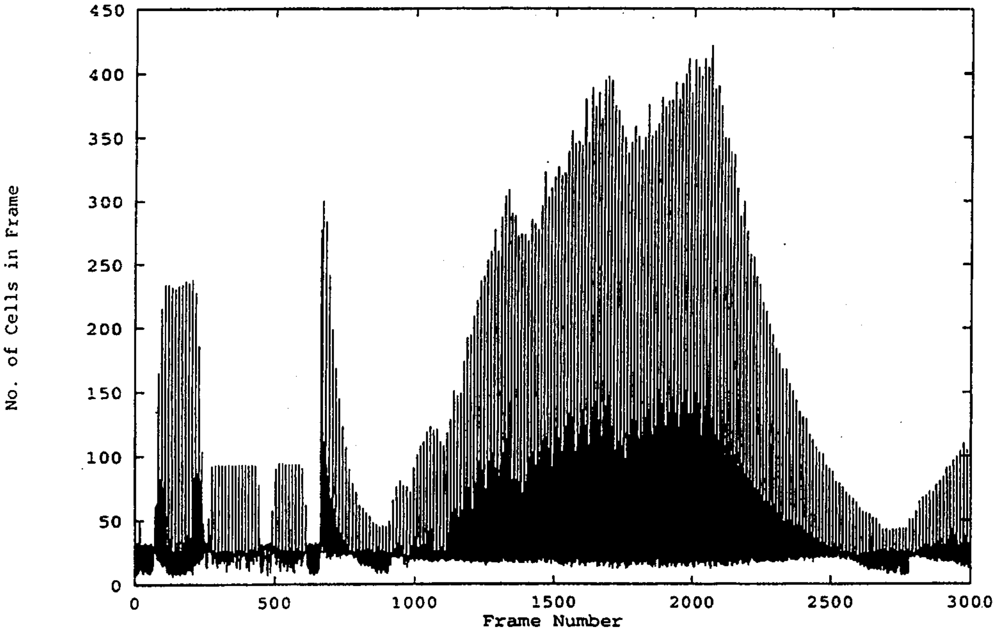


Fig. 5. *Star Wars* frame numbers 1 to 3000.

bursts. But for $d \geq 1200$, the build up is sufficient to allow for transmission rates in vicinity of $\tilde{b}_{\min}(d)$.

Recall that the $b_{\max}(d, B)$ function bounds b so that blocking does not occur at the STB. Fig. 6 includes $b_{\max}(d, B)$ for $B = 895\,070$ cells. As shown, $b_{\max}(d, B)$ has a relatively small negative slope, and it abruptly drops to zero at $\bar{d}(B)$. Additionally, by graphing $b_{\min}(d)$ and $b_{\max}(d, B)$ on the same graph, the set of feasible (b, d) pairs can easily be identified. In Fig. 6 we also see that the feasible region is in the range $\mathcal{D}(B) \in [225, 22\,486]$. Thus, for $B = 895\,070$ cells, a d exists such that *Star Wars* can be transported using CRTT with $b \in [865, 1081]$ cells/s. Furthermore, we note that *Star Wars* can be supported using CRTT with $b \approx b_{\text{avg}}$.

We now determine the minimum STB memory, $B_{\min}$, needed to transport *Star Wars* with CRTT. In other words, we wish to find the smallest B such that at least one feasible (b, d) pair exists. To this end, we have used a divide and conquer algorithm to find $B_{\min}$. In this algorithm we start with $B = \bar{B}$, where $\bar{B}$ is the upper bound described in the previous section. We then search for a $d \in \mathcal{D}(B)$, beginning with $d = \bar{d}(B)$ and then decrementing d until a $d \in \mathcal{D}(B)$ is found. As soon

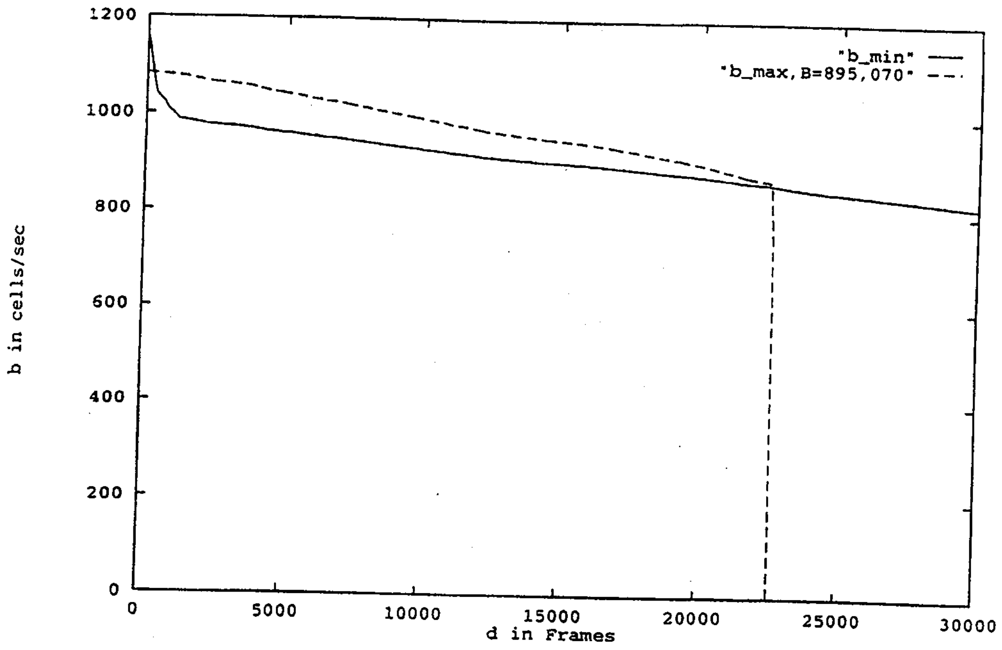


Fig. 6. *Star Wars* CRTT feasible region (enlarged).

as $d \in \mathcal{D}(B)$ is found, we decrease B appropriately; if $\mathcal{D}(B)$ is empty, we increase B appropriately.

The final result from the algorithmic search was that $B_{\min} =$ 464 029 cells = 22.3 Mbytes, which corresponds to a set-top memory size approximately equal to 6% of the size of the entire encoded video. In order for CRTT to work with a STB memory possessing 464 029 cells worth of memory, d must be 1146 frames and $b \in [986.477, 986.478]$ cells/s. This corresponds to a build-up delay of 37 s. Our upper bound $\bar{B}$, calculated in 4 s on a Sun SPARCstation2 workstation, is 465 533 cells, which is very close to $B_{\min}$.

We also calculated $B_{\min}$ and $\bar{B}$ for a permuted trace; whereby the first half and second half of the trace were interchanged. We found $B_{\min} = 433\,631$ cells and $\bar{B} =$ 465 633 cells; note that $\bar{B}$ is again very close to $B_{\min}$. Finally, we calculated $\bar{B}$ for 24 permutations of the original trace. These permutations were created by partitioning the original trace into four equal parts and then generating all $4! = 24$ combinations. For these 24 permutations we found the lowest $\bar{B}$ to be 366 750 cells, the highest $\bar{B}$ to be 465 633 cells, and the average $\bar{B}$ to be 413 340 cells. Note that the difference between the highest $\bar{B}$ and the lowest $\bar{B}$ is rather small.

In summary, our numerical results show that CRTT can be used to support this particular MPEG1 encoding of *Star Wars*. We found that CRTT allows a wide range of set-top memory sizes to be used with CRTT. The smallest amount of memory needed, $B_{\min}$, for CRTT to be feasible is approximately 6% of the encoded video size. When this minimum memory is present, CRTT requires an ATM connection with $b \approx b_{\text{avg}}$ and a build-up delay of 37 s.

VI. IMPLEMENTATION PERSPECTIVE

Two costs must be examined to determine the viability of using CRTT in commercial VoD system implementations. The first is the recurring transport cost. This was discussed in Sections III and IV, where it was shown that CRTT is less expensive than CBR and VBR transport. The second is the fixed STB cost.

The major components in a basic STB are memory, an MPEG decoder chip, an ATM adapter, a processor (for connection management), a frame buffer, and other "glue" logic. We assume that the cost of the MPEG decoder chip, the ATM adapter, the processor, the frame buffer, and the other logic remains relatively constant across different box implementations.[1] The major component influencing cost differences in STB's is then the memory where the compressed data is stored until it is decoded. With CRTT, a memory size of 25 Mbytes suffices for all 24 permutations of *Star Wars*.

Broadly speaking, there are two possible approaches for including the additional CRTT memory in a STB. The first approach is to implement all the memory with dynamic random access memory (DRAM). With 1995 memory prices, the cost of 25 Mbytes of DRAM is over $600; however, DRAM prices have traditionally dropped at a rate of 40% per year [12] and are expected to continue to drop in the upcoming years.

The second approach is to implement the CRTT memory in magnetic disk. A hard disk of sufficient size to support *Star Wars* with CRTT costs approximately $25. However, a hard disk cannot read and write simultaneously as required by CRTT. This problem can be resolved by including a small amount of DRAM at the input and output of the disk. Such a design should significantly reduce the cost of the STB. On the other hand, a hard disk has a higher failure rate than DRAM and can be the source of undesirable background noise.

We note that the above analysis is strictly based on our numerical results from *Star Wars*. Other movies may require

---

[1] A rough estimate for the current cost of these components is approximately $300.

considerably more or less memory depending on the MPEG encoding parameters used and the actual movie.

VII. PIECEWISE CONSTANT RATE TRANSMISSION AND TRANSPORT

CRTT uses a constant bandwidth, b, over the life of the VC connection. Since $b \approx b_{\text{avg}}$, the STB memory contents will grow and diminish in accordance with the variable consumption rate. This variable rate is influenced by factors including scene type; with more complex scenes consuming, on average, more cells.

We can further reduce the STB memory requirement using a refinement of CRTT called PCRTT. In PCRTT we allow b to vary over a small number of time intervals, where each interval is on the order of several minutes. In this manner, the cell arrival rate can better reflect the STB consumption rate, thereby reducing the STB memory contents. For example, suppose a movie is being transmitted at rate $b_1 \geq b_{\text{avg}}$ when a scene of low complexity begins. On average, the consumption rate of cells during this scene will be less than b_{avg}. Thus, the STB memory contents will grow while the new scene is played. But if we adjust to a new transmission rate, b_2, where $b_2 < b_{\text{avg}}$, we can prevent excessive growth of memory contents. However, we must also ensure when switching to rate b_2 that starvation is not triggered in subsequent scenes of high complexity.

Given a specific movie, how do we determine what transmission rates to use? How much set-top memory is required for PCRTT? How much preplay build-up is needed? We now give a formal definition of PCRTT and address these questions. We defer the issue of finding the optimal rate-switch times to a subsequent paper [19]. We also discuss the support of limited VCR features in [19].

PCRTT divides time into $K + 1$ intervals, requires the transmission rate to be constant over each interval, but allows the rates to vary across intervals. We formally define PCRTT as follows. Fix positive integers K, $C(0), C(1), \cdots, C(K-1)$, with $C(0) < C(1) < \cdots < C(K-1) < C$ and positive numbers $b(0), b(1), \cdots, b(K)$. We shall refer to these parameters as the PCRTT parameters. PCRTT transmits and transports cells 1 through $C(0)$ at rate $b(0)$, cells $C(0)+1$ through $C(1)$ at rate $b(1), \cdots$, cells $C(K-1)+1$ through C at rate $b(K)$. As soon as the first $C(0)$ cells arrive and are stored at the STB, the first frame is instantaneously removed; subsequent frames are removed and displayed every $1/F$ seconds.

If K is not very large, then PCRTT should be easy to manage at the file server and in the network. Its transport cost should be comparable to that of CRTT.

A natural optimization problem is for fixed K to choose $C(0), C(1), \cdots, C(K-1)$ and $b(0), b(1), \cdots, b(K)$ such that the memory at the STB is minimized with no starvation or overflow. In this preliminary study, we provide a heuristic for choosing PCRTT parameters. Although these parameters are suboptimal, they seem to give good results.

In this heuristic, we first select a positive integer J and integers $0 < L(1) < \cdots < L(J-1) < N$. Set $L(0) = 0$ and $L(J) = N$. Define $b_{\text{avg}}(j)$ as

$$b_{\text{avg}}(j) = \frac{F \sum_{i=L(j-1)+1}^{L(j)} x_i}{L(j) - L(j-1)}, \quad j = 1, \cdots, J.$$

After a small build up, we would like to transmit, for each $j = 1, \cdots, J$, at the average rate $b_{\text{avg}}(j)$ for an interval of length $[L(j) - L(j-1)]/F$ seconds. Such a scheme should keep the buffer contents at the STB small.

To formally define the heuristic, let

$$j_n := \min\{j \colon n \leq L(j)\}$$

and note that $L(j_n - 1) < n \leq L(j_n)$. Let

$$\delta := \max_{0 \leq n \leq N-1} \left\{ \sum_{i=1}^{n+1} x_i - \sum_{m=1}^{j_n - 1} \frac{b_{\text{avg}}(m)}{F} [L(m) - L(m-1)] - \frac{b_{\text{avg}}(j_n)}{F} [n - L(j_n - 1)] \right\}$$

and

$$\bar{d} := \min_{1 \leq d \leq N} \left\{ d \colon \sum_{i=1}^{d} x_i \geq \delta \right\}.$$

The heuristic sets the PCRTT parameters as follows:

$$K = \min \left\{ j \colon \sum_{i=1}^{\bar{d}} x_i + \sum_{i=1}^{L(j)} x_i \geq C \right\}$$

$$C(0) = \sum_{i=1}^{\bar{d}} x_i$$

$$C(k) = C(0) + \sum_{i=1}^{L(k)} x_i, \quad k = 1, \cdots, K-1$$

$$b(k) = b_{\text{avg}}(k), \qquad k = 1, \cdots, K$$

and $b(0)$ is arbitrary.

Note that this heuristic first transmits cells at rate $b(0)$ for $C(0)/b(0)$ seconds, then at rate $b_{\text{avg}}(1)$ for $L(1)/F$ seconds, then at rate $b_{\text{avg}}(2)$ for $[L(2) - L(1)]/F$ seconds, ... , then at rate $b_{\text{avg}}(K-1)$ for $[L(K-1) - L(K-2)]/F$ seconds, and finally at rate $b_{\text{avg}}(K)$ for

$$\frac{C - \sum_{i=1}^{L(K-1)} x_i - \sum_{i=1}^{\bar{d}} x_i}{b_{\text{avg}}(K)} \text{ s.}$$

Therefore, when the $(n+1)$th frame is removed from the STB, which occurs at time n/F seconds from the start of playback, cells are arriving at the STB at rate $b_{\text{avg}}(j_n)$.

We claim that this heuristic ensures that there is no starvation at the STB. The argument is similar to that of Lemma 1. The number of cells that have been removed from the STB at time n/F is

$$\sum_{i=1}^{n+1} x_i. \tag{7}$$

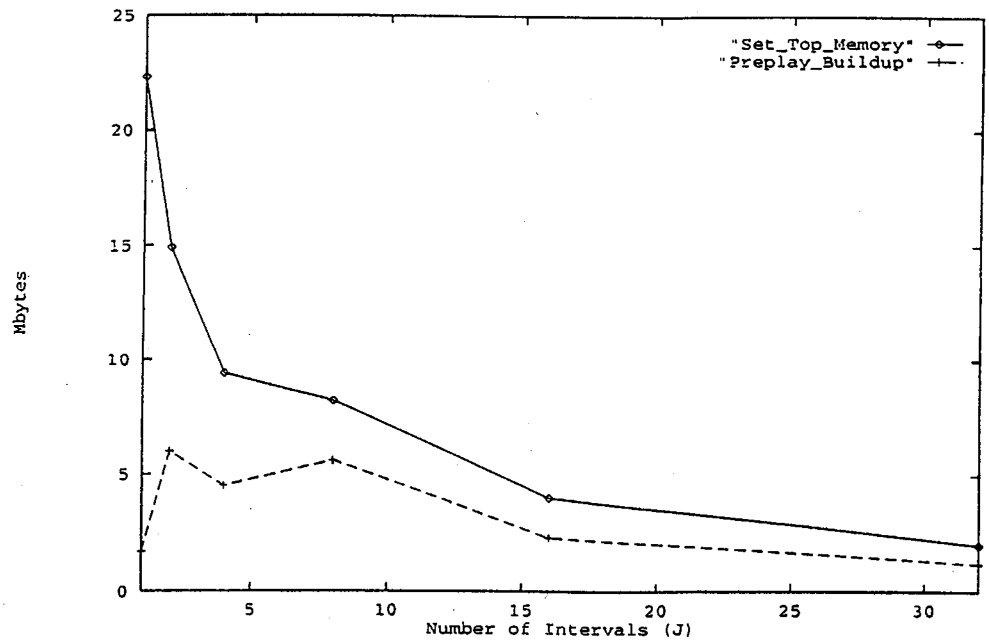


Fig. 7. $\bar{B}$ and $\bar{d}$ in Mbytes for $J = 1, 2, 4, 8, 16$, and 32.

The number of cells that have been received by the STB at time n/F, $n = 0, \cdots, N-1$, is no more than

$$\sum_{i=1}^{\bar{d}} x_i + \sum_{m=1}^{j_n - 1} \frac{\bar{b}_{\text{avg}}(m)}{F}[L(m) - L(m-1)] + \frac{\bar{b}_{\text{avg}}(j_n)}{F}[n - L(j_n - 1)]. \quad (8)$$

Since the definition of $\bar{d}$ implies that (8) minus (7) is never negative, there is no starvation at the STB. A similar argument shows that there is no buffer overflow when $B = \bar{B}$, where $\bar{B}$ is given by the equation at the bottom of the page. It is straightforward to code the calculation for $\bar{d}$ and $\bar{B}$ so that the overall computational effort is $O(N)$. Furthermore, since the transmission rate change times are precalculated off-line, we know precisely how much network bandwidth is needed for each interval. This knowledge allows for the reservation of necessary network resources in advance when a play request is received.

We have calculated $\bar{B}$ and $C(0)$ for $J = 1, 2, 4, 8, 16$, and 32. For all these cases, the $L(j)$'s were chosen so that the intervals $[L(j-1)+1, L(j)]$ have equal length, that is, for a given J each interval has length N/J frames. The results are plotted for *Star Wars* in Fig. 7.

We first observe from Fig. 7 that significant reductions in STB memory are achieved with a small number of rates. For $J = 4$, the memory requirement is less than half that for CRTT, and for $J = 32$, the memory requirement is approximately 2 Mbytes. Also note that the required build up, $C(0)$, can actually increase as the number of constant rates increases. To explain this erratic and surprising behavior of the build up requirements, consider the cases $J = 1$ and $J = 2$. With $\bar{b}_{\text{avg}}$ denoting the average rate for $J = 1$, and $\bar{b}_{\text{avg}}(1)$ and $\bar{b}_{\text{avg}}(2)$ denoting the average rates for $J = 2$, *Star Wars* has $\bar{b}_{\text{avg}}(1) < \bar{b}_{\text{avg}}$. Since there is a burst near the beginning of the movie (frames 1000 to 2500—see Section V-B), $\bar{b}_{\text{avg}}(1) < \bar{b}_{\text{avg}}$ implies that the case $J = 2$ requires more build up to surmount the burst.

VIII. CONCLUSION

We have explored the viability of CRTT for constant-quality VoD. We have argued that CRTT has lower transport cost than VBR transport and greater playback flexibility than best-effort transport. CRTT, however, requires more memory in the STB than does VBR transport. If RAM prices continue to decline at the current rate, a pure RAM implementation may be economically feasible in the upcoming years; otherwise, the STB can use a hard disk with a small amount of RAM cache. An alternative transmission and transport scheme is PCRTT, whereby multiple constant rates are used. This scheme slightly complicates the management of the network and server resources, but its reduced memory requirements render a pure-RAM implementation for the STB economically feasible. In a recent paper [19], we have developed dynamic programming algorithms to calculate optimal PCRTT parameters and explored using additional memory to support limited VCR features.

$$\bar{B} = \max_{0 \le n \le N-1} \left\{ \sum_{m=1}^{j_{n+1}-1} \frac{\bar{b}_{\text{avg}}(m)}{F}[L(m) - L(m-1)] + \frac{\bar{b}_{\text{avg}}(j_{n+1})}{F}[n + 1 - L(j_{n+1} - 1)] - \sum_{i=1}^{n+1} x_i \right\} + \sum_{i=1}^{\bar{d}} x_i.$$

ACKNOWLEDGMENT

The authors thank M. Garrett for making the *Star Wars* trace available.

REFERENCES

[1] Y. H. Chang, D. Coggins, D. Pitt, D. Skellern, M. Thapar, and C. Venkatraman, "An open-systems approach to video-on-demand," *IEEE Commun. Mag.*, pp. 68–80, May 1994.

[2] D. Deloddere, W. Verbiest, and H. Verhille, "Interactive video-on-demand," *IEEE Commun. Mag.*, pp. 82–88, May 1994.

[3] M. El Zarki and P. Pancha, "Bandwidth-allocation schemes for variable bit rate MPEG sources in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 3, pp. 190–198, June 1993.

[4] A. I. Elwalid and D. Mitra, "Effective bandwidth of general markovian traffic sources and admission control of high-speed networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 3, pp. 329–343, June 1993.

[5] D. Ferrari and D. Verma, "A scheme for real-time channel establishment in wide-area networks," *IEEE J. Select. Areas Commun.*, vol. 8, no. 3, pp. 368–379, Apr. 1990.

[6] D. C. Verma, H. Zhang, and D. Ferrari, "Delay jitter for real-time communication in a packet switching network," *Computer Commun.*, pp. 367–373, July/Aug. 1992.

[7] MPEG.data Trace available via anonymous FTP from thumper.bellcore.com.

[8] M. W. Garrett and A. Fernandez, *Variable Bit Rate Video Bandwidth Trace Using MPEG Code*, Available via anonymous FTP from thumper.bellcore.com., Nov. 4, 1994.

[9] J. Gemmell and S. Christodoulakis, "Principles of delay sensitive multimedia data storage and retrieval," *ACM Trans. Inform. Syst.*, Jan. 1992.

[10] R. J. Gibbens and P. J. Hunt, "Effective bandwidths for the multi-type UAS channel," *Queueing Syst.*, vol. 9, pp. 17–27, 1991.

[11] R. Guerin, H. Almadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 968–981, Sept. 1991.

[12] J. L. Hennessey and D. A. Patterson, *Computer Architecture: A Quantitative Approach*. San Mateo, CA: Morgan Kaufmann, 1990.

[13] K. Joseph and D. Reininger "Source traffic smoothing for VBR video encoders," in *Proc. 6th Int. Workshop Packet Video*, Portland, OR, Sept. 26–27, 1994.

[14] H. Kanakia, P. P. Mishra, and A. Reibman, "An adaptive congestion control scheme for real-time packet video transport," in *SIGCOMM'93 Conf. Proc.*, Sept. 1993, pp. 20–31.

[15] F. P. Kelly, "Effective bandwidths at multi-class queues," *Queueing Syst*, vol. 9, pp. 5–15, 1991.

[16] T. Kwok, "A vision for residential broadband services: ATM-to-the-home," *IEEE Network*, pp. 14–28, Sept./Oct. 1995.

[17] S. S. Lam, S. Chow, and D. K. Y. Yau, "An algorithm of lossless smoothing of MPEG video," in *SIGCOMM'94 Conf. Proc.*, Aug. 1994, pp. 281–293.

[18] T. D. C. Little and D. Venkatesh, "Prospects for interactive video-on-demand," *IEEE Multimedia*, pp. 14–24, Fall 1994.

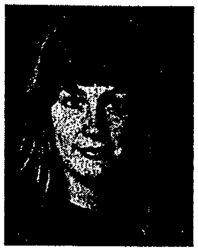
[19] J. M. McManus and K. W. Ross, "Prerecorded VBR sources in ATM networks: Piecewise-constant-rate transmission and transport," submitted for publication.

[20] P. V. Rangan, H. M. Vin, and S. Ramanathan, "Designing an on-demand multimedia service," *IEEE Commun. Mag.*, pp. 56–64, July 1992.

[21] A. R. Reibman and B. G. Haskell, "Constraints on variable bit rate video for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 361–372, Dec. 1992.

[22] K. W. Ross, *Multiservice Loss Models for Broadband Telecommunications Networks*. New York: Springer-Verlag, 1995.

[23] W. D. Sincoskie, "Video-on-demand: Is it feasible?" in *Proc. GLOBECOM'90*, pp. 201–204.



Jean M. McManus (M'88) received the B.S.E. degree from Duke University, Durham, NC, in 1987, and the M.S.E. degree from the University of Pennsylvania, Philadelphia, in 1993, both in electrical engineering. She is currently a Ph.D. candidate in systems engineering at the University of Pennsylvania, Philadelphia.

From 1987 to 1992, she was employed as a Systems Engineer at GTE (formerly Contel Federal Systems) in Chantilly, VA. Her research interests include video-on-demand and high speed networks.

Keith W. Ross (S'82–M'86–SM'90) received the B.S. degree from Tufts University, Medford, MA, 1979, the M.S. degree from Columbia University, NY, 1981, and the Ph.D. degree from the University of Michigan, Ann Arbor, in 1985.

He is an Associate Professor in the Department of Systems Engineering, University of Pennsylvania, Philadelphia, and holds secondary appointments in the Computer Information Science and in the Operations and Informations Management (Wharton) Departments. In 1980, he designed satellite radar systems as an employee of AVCO. He has been a visiting scholar at several research and academic institutions in France. His current research interest is in protocols and traffic management in high-speed telecommunication networks, including local-area networks, wide-area data networks, voice networks, and broadband integrated services digital networks. He is currently performing research in ATM and video-on-demand, and teaches courses on Internet technology and commerce, and on broadband networking.

Dr. Ross is the recipient of numerous grants from the National Science Foundation and AT&T and was the Program Chairman of the 1995 INFORMS Telecommunications Conference.