Journal of VLSI Signal Processing 27, 81–97, 2001

Soft ARQ for Layered Streaming Media

MATTHEW G. PODOLSKY AND STEVEN McCANNE
Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, USA

MARTIN VETTERLI
Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, USA;
Laboratoire de Communications Audiovisuelles, École Polytechnique Fédérale de Lausanne,
Lausanne, Switzerland

Received July 27, 1999; Revised November 17, 1999

Abstract. A growing and important class of traffic in the Internet is so-called "streaming media," in which a server transmits a packetized multimedia signal to a receiver that buffers the packets for playback. This playback buffer, if adequately sized, counteracts the adverse impact of delay jitter and reordering suffered by packets as they traverse the network, and if large enough also allows lost packets to be retransmitted before their playback deadline expires. We call this framework for retransmitting lost streaming-media packets "soft ARQ" since it represents a relaxed form of Automatic Repeat reQuest (ARQ). While state-of-the-art media servers employ such strategies, no work to date has proposed an optimal strategy for delay-constrained retransmissions of streaming media—specifically, one which determines what is the optimal packet to transmit at any given point in time. In this paper, we address this issue and present a framework for streaming media retransmission based on *layered* media representations, in which a signal is decomposed into a discrete number of layers and each successive layer provides enhanced quality. In our approach, the source chooses between transmitting (1) newer but critical coarse information (e.g., a first approximation of the media signal) and (2) older but less important refinement information (e.g., added details) using a decision process that minimizes the expected signal distortion at the receiver. To arrive at the proper mix of these two extreme strategies, we derive an optimal strategy for transmitting layered data over a binary erasure channel with instantaneous feedback. To provide a quantitative performance comparison of different transmission policies, we conduct a Markov-chain analysis, which shows that the best transmission policy is time-invariant and thus does not change as the frames' layers approach their expiration times.

Keywords: streaming media, layering, retransmission, soft ARQ

1. Introduction

A common class of traffic on the Internet is so-called "streaming media," where real-time signals like audio and video are delivered from a server somewhere in the network to a human user that interactively views the material. Unlike human-to-human communication, which requires relatively tight and consistent end-to-end delays for good interactive performance [1], server-to-human communication can afford a certain level of artificial delay. As a result, streaming media applications often have sufficient time to recover from lost packets through retransmission and thereby avoid unnecessary degradation in reconstructed signal quality. We refer to this delay-constrained Automatic Repeat/ reQuest system as "soft ARQ," because it represents a relaxed form of ARQ in which the successful on-time delivery of every packet is not guaranteed.

Soft ARQ has been exploited in commercial products like RealNetworks and Microsoft streaming media systems and in research protocols like STORM [2] and MESH [3]. In these approaches, the sender determines if a lost packet will arrive in time for playout; if so the sender retransmits it; if not, the sender drops it. Prior works have focused on how to choose the playout delay and how to decide if retransmissions will arrive in time. However, these issues are only part of the retransmission puzzle. When the sender is subjected to transmission rate constraints, it may have to retransmit a packet *at the expense* of a different unsent packet. As a result, the sender must consider not only whether the retransmitted packet will arrive at the receiver in time, but also if that packet is more beneficial than one it displaces—i.e., will transmitting packet A improve the reconstructed signal quality more (on average) than if packet B had been transmitted instead?

One way to flexibly accommodate this packet displacement is to use a layered, or multi-resolution, signal representation. A fundamental attribute of layered encodings is that they represent a signal in a hierarchical fashion. Packetizing each layer separately allows finer-grained control of the transmission process than is possible with a non-layered representation. An example of layering is splitting a video signal into a coarse-image "base" layer and a second "enhancement" layer containing added details. Another example of layering is simply splitting a sequence of unencoded 8-bit audio samples into each sample's 4 most signficant bits (the base layer) and 4 least significant bits (the enhancement layer). In both cases the enhancement layer when *combined* with the base layer produces a better quality signal than the base layer alone. However, due to the hierarchical structure of the encoding, the enhancement layer does not provide a useful representation without the base layer. An alternative to this type of ordered, multi-resolution coding is multiple description coding, which does not need to distinguish among different types of packets or layers [4]. The price of this flexibility is reduced coding efficiency, however.

Layering has been exploited by other applications for finer grained control of streaming multimedia; for example, congestion control schemes can adjust the rate of multimedia streams by changing the number of layers transmitted [5, 6]. Layering benefits soft ARQ systems by allowing the sender to choose between sending *part* of one time-frame or another, rather than choosing between entire frames. For example, the sender may choose to transmit a packet containing the base layer of one frame instead of retransmitting one containing a less critical enhancement layer of an older frame. However, layering the signal in and of itself does not solve the problem of determining what the optimal packet is to transmit at any given time. The goal of this work is to find the optimal transmission policy.

The paper is organized as follows. We introduce our streaming media transmission model and its parameters in Section 2. Then, in Section 3, we define a state space for the transmission process and detail a Markov-chain analysis which is used to find the expected distortion resulting from applying a particular transmission policy. Section 4 contains our analysis results, first focusing in detail on the case that there are only two layers and two frames' lifetimes overlapping in Section 4.1, and then looking at more general cases in Section 4.2. A key result for all cases is that the best transmission policy is time-invariant and thus does not change as the frames' layers approach their expiration times. Section 5 describes areas of related and future work, and concluding remarks are given in Section 6.

2. An Analytical Model for Streaming Media

In this section we introduce the transmission model we use for a streaming media session. After describing the model we illustrate it with an example transmission sequence and show the need for a transmission policy governing decisions of what information should be (re-)transmitted at any point in time.

2.1. Transmission System Model

Figure 1 illustrates our model for a streaming layered multimedia transmission system. The transmission process begins with a multimedia signal X at the sender. We assume that the entire signal is not available prior to the start of transmission—in other words, it is either dynamically generated or retrieved from storage concurrently while the transmission process is ongoing. The signal is segmented in time into equal length segments or "frames"; these frames are produced periodically as the signal is generated. We denote frame n by X^n. The signal is also encoded into a hierarchy of N layers $\{X_1, X_2, \ldots, X_N\}$, where X_1 is the most "important" layer and X_N is the least "important" layer, and we assume that all layers have the same bit-rate. We also assume that the importance of a layer can be quantified, so that receiving a more important layer of a frame results in a measurably greater benefit (e.g., a greater increase in signal quality or decrease in distortion) than a less important layer of that frame. We denote the ith

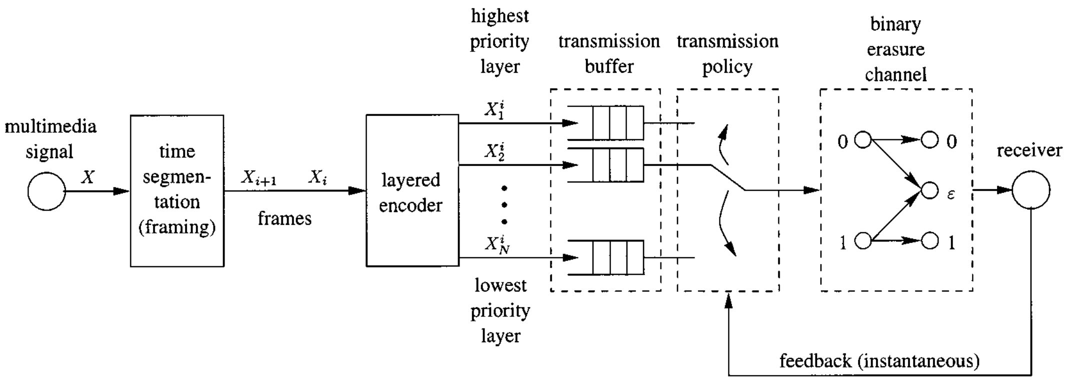

Figure 1. System diagram of layered transmission over a binary erasure channel with feedback.

frame of layer l by X_l^i. These layer/frame segments form the basic transmission units or "messages" that are sent across the network (e.g., contained in packets). The sender operates under a transmission rate constraint, which manifests itself as a lower bound on the minimum time between message transmissions.

To model network packet loss, each message passes through a binary erasure channel (BEC) on its way to the receiver. The BEC will either erase (drop) a message (with probability ε) or successfully transmit it (with probability $1 - \varepsilon$). The BEC, in conjunction with an instantaneous feedback path, serves as an idealized model for the network. Messages which successfully reach the receiver are used to reconstruct the signal. Because we have assumed a "streaming" multimedia scenario, the receiver starts playback of the signal as it is still being generated and transmitted at the source. At some fixed time after frame i is produced at the source, it is reconstructed from whatever layers X_l^i have arrived at the receiver and played back.

An important component of our model is the transmission policy, located at the sender. This policy dictates which message (frame and layer) the source should transmit (or retransmit) for any possible situation. For every feasible set of unsent (or sent but dropped) messages and their corresponding playback deadlines (i.e., the latest time they can be sent before they are no longer useful to the receiver), the transmission policy contains a rule indicating which message the sender should choose to transmit next. The need for a policy stems from the fact that messages can have both different priorities (due to the layering) and different time constraints (due to the framing and streaming playback). We illustrate this by means of an example.

2.2. *Example Transmission Sequence*

Figure 2 shows a message transmission sequence for two different transmission policies for a two-layered media signal. We denote time in terms of transmission units; one unit is equal to the minimum time between message transmissions stemming from the system's transmission rate constraint. Frames of the media signal are generated periodically every T units, and messages containing a layer of a frame must be transmitted within L units after that frame has been generated in order to reach the receiver in time. In this example T is 3 and the lifetime L is 5.

Decisions of what to transmit so as to maximize signal quality are simple when the choice is restricted to messages from within a single frame: the sender should (re-)transmit the most important layer of that frame that has not been successfully received yet. This is seen at times 0 and 1 in Fig. 2, when only the first frame is "alive" and so the sender chooses the high-priority message, X_0^1. Similarly, when choosing among all of the messages of a single layer, the sender should decide to (re-)transmit the oldest message of that layer which will still arrive in time for playback. However, the decision is not necessarily clear when choosing between messages from different frames *and* different layers. Specifically, how do you decide between sending an older, lower priority message and a newer, high priority message? In this example at any time at most two frames' lifetimes overlap ($T < L \leq 2T$), so at most four messages will be available for transmission: the two layers of frame i and $i + 1$. In order to emphasize the priority and age of these four messages, we introduce the following variable names:

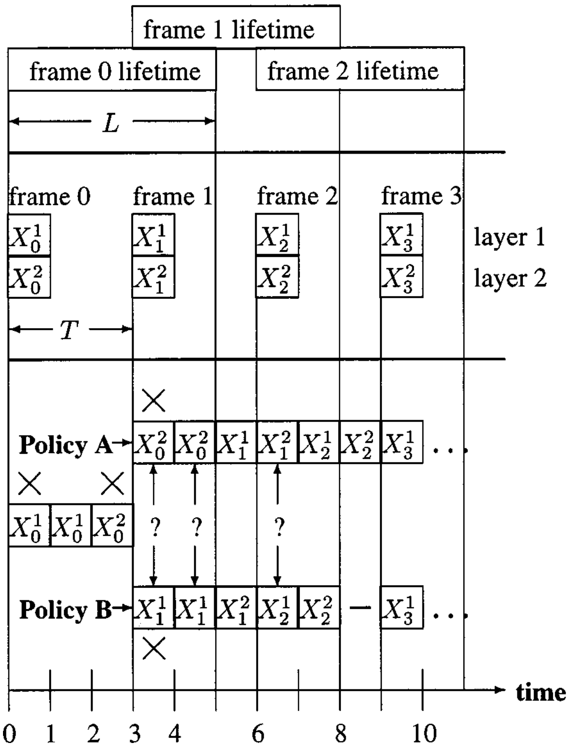

Figure 2. Example transmission sequence for a 2-layered signal ($N = 2$) with a data lifetime of $L = 5$ and a inter-frame period of $T = 3$. Two different transmission policies are shown. Erased transmissions are denoted with ×'s.

- $O_H = X_1^i$: the high priority layer of the older frame
- $O_L = X_2^i$: the low priority layer of the older frame
- $N_H = X_1^{i+1}$: the high priority layer of the newer frame
- $N_L = X_2^{i+1}$: the low priority layer of the newer frame.

We will use the above notation throughout this work whenever discussing the two layer, two overlapping-frame case.

At time 3 in Fig. 2, we have a choice between $O_L = X_2^0$ and $N_H = X_1^1$. There are fundamental tradeoffs between the data's importance and its time-constraints. One reason to favor O_L over N_H is its earlier playback (time 5 as opposed to 8), so there is less time and hence fewer opportunities in which to successfully transmit it. However, a conflicting reason to choose N_H over O_L is its greater distortion reduction. Sending the less important O_L leaves fewer transmission opportunities for N_H, and if the loss rate is high it may take all of those opportunities to successfully transmit N_H. It is not obvious which choice is better—i.e., which choice results in a higher average signal quality.

As a result, the sender relies on the transmission policy to tell it what choice to make. Two policies are shown in Fig. 2: in policy A the sender chooses O_L each time the decision arises, and in policy B it chooses N_H. In this particular segment policy B performs worse, because by the time it successfully transmits N_H, O_L has expired. But if more erasures had occurred then the more conservative policy B may have outperformed A. In the next section we develop a framework for mathematically analyzing the performance of any transmission policy. This will allow us to compare the long-term performance of one policy to another.

3. Analysis

We now present a formal analysis for the layered transmission system described above and illustrated by Fig. 1. This analysis will allow us to objectively evaluate the performance of any transmission policy for a given set of known parameters, such as the packet erasure probability and data lifetime.

We begin by formalizing the parameters for the transmission model introduced in the previous section. Next we define a state space which captures the model's dynamics—what layers of what frames have already been transmitted, how long before each frame expires, etc. We then apply Markov chain analysis to find the steady-state behavior of the transmission system. From the steady-state analysis we obtain a distribution on the number of layers per frame that are successfully received before the frame expires. We finally combine this information with a cost function (e.g., a rate-distortion curve) to find the average cost associated with a specific transmission policy.

3.1. Model Parameters

The multimedia signal X is segmented both into time frames and signal layers. One layer of one frame forms a message which is the basic unit of transmission. Messages can be transmitted at a maximum rate of one message per time unit. For convenience we will use seconds to denote time units. The signal is segmented in time into frames that are generated periodically every T time units, and each frame is further encoded into a hierarchy of N layers. We assume $T \geq N$, so that there is at least one chance to transmit each layer of every frame. Each frame has a lifetime at the *sender* of L seconds; any messages sent more than L seconds after the frame is produced will arrive too late for playback at the receiver. We say that a frame produced at time t "expires" at the sender at time $t + L$. Because we have assumed there is no network delay, this lifetime is solely a function of delays at the receiver: specifically, L is the playback delay less any processing delays. We will assume that $L > T$, so that there is at least some overlap in the lifetimes of consecutive frames. This leads to situations requiring a non-obvious decision between transmitting a less important message of a older frames and a more important message of a newer frame.

Finally, because all of the frames of the multimedia signal are not available to the sender at the start of the transmission (the signal's frames are produced periodically), and because each frame only has L seconds after it is produced to be sent to the receiver, there is a finite limit on the number of frames whose layers can be considered valid candidates for transmission. We use K to denote the maximum number of frames alive at any given time by. K is a function of how long frames live (L) and how frequently they are produced (T), and is given by:

$$K = \left[\frac{L}{T} \right]. \tag{1}$$

Table 1 summarizes the definitions of the above parameters.

3.2. *State Space Definitions*

Having identified the parameters of our transmission model, we now set about defining a state space to track the transmission process' behavior. After the initial K frames of the signal have been produced, a new frame is produced and an old frame expires once every T seconds. Let ϕ be the phase (position) within a T-length cycle, so that $\phi \in \{0, 1, \ldots, T - 1\}$, and let the cycle start at $\phi = 0$ when a new frame is produced. Note that if the lifetime L is not an exact multiple of T, the oldest frame will expire at phase $\phi = L - (K - 1)T$, *before* the next new frame is produced. In this case there will be only $K - 1$ frames alive during the last $(KT - L)$ seconds of a cycle.

Table 1. Summary of transmission model parameters.

Variable	Meaning
L	frame lifetime
T	period of frame production
N	number of layers per frame
K	maximum number of frames "alive"

In deciding which message to transmit next at any given time t, the sender must consider not only which messages of the K current live frames have been transmitted, but how much time remains before each of these frames expires. However, the sender does *not* need to consider (and hence, remember) any information about the older expired frames in order to make its decision. Because these frames have expired, there is no point in sending any of their untransmitted messages, and thus there is no need to remember their specific expiration times. Also, although we may be able to infer the channel erasure rate through knowledge of how many layers of these frames were successfully transmitted, we have assumed that we already know the erasure rate and hence this knowledge is not needed to make the current transmission decision.

We can now define a state S_t that summarizes the information the sender needs to make a transmission choice at time t. Let S_t be defined as:

$$S_t = \left(\phi^{(t)}, \mathbf{n}^{(t)}\right), \tag{2}$$

$$\phi^{(t)} = t \bmod T, \tag{3}$$

$$\mathbf{n}^{(t)} = \left[n_1^{(t)}, n_2^{(t)}, \ldots, n_K^{(t)}\right], \tag{4}$$

where $n_i^{(t)}$ is the number of successfully transmitted layers of the ith-oldest live frame at time t (i.e., frame 1 is the oldest, frame K is the newest). We omit the t superscript from ϕ, $\mathbf{n}$, and n_i when its context is clear. Because there are N layers, $0 \leq n^{(t)} \leq N$. These state space components are summarized in Table 2.

Table 2. Summary of the state space variables' definitions and relevant equation numbers.

Variable	Meaning	Eqn
S_t	transmission state at time t	2
$\phi^{(t)}$	phase within a T-length cycle	3
$\mathbf{n}^{(t)}$	K-tuple of the transmission state of the currently live frames $n_i^{(t)}$	4
$n_i^{(t)}$	number of frame i's layers successfully sent by time t	–

The phase ϕ tells us how much time is left before each frame expires. For example, at the beginning of a cycle ($\phi = 0$) the frame K is produced, and so we know it expires in L seconds. More generally, let ttl_i be frame i's "time-to-live," i.e., how much time remains before it expires. It is calculated as:

$$\mathrm{ttl}_i = L - \phi - iT. \qquad (5)$$

The K-tuple $\mathbf{n}$ tells us exactly what layers of the K frames have already been transmitted, and, conversely, which layers remain for each frame. At any time t, the $n_i^{(t)}$-most important layers of frame i have been transmitted, and so there are $N - n_i^{(t)}$ layers remaining. At the beginning of a cycle ($\phi = 0$) a new frame is produced, so $n_K = 0$. Also at this time, all of the frames "age" one position in the K-tuple $\mathbf{n}$. To illustrate this, suppose that, at some time t which is the start of one cycle ($t \mod T = 0$), we have a state

$$S_t = \left(0, \mathbf{n}^{(t)} = \left[n_1^{(t)}, n_2^{(t)}, \ldots, n_K^{(t)}\right]\right).$$

Now suppose that the next T transmission attempts are all erased, so that no frame gets any more messages across. For this case the next T states are independent of our transmission policy—regardless of which messages the policy dictated we attempt to (re-)transmit, those messages were all erased—and our state evolves with time as:

$$\begin{aligned} S_{t+1} &= \left(1, \mathbf{n}^{(t)}\right) \\ S_{t+2} &= \left(2, \mathbf{n}^{(t)}\right), \\ &\vdots \\ S_{t+T-1} &= \left(T-1, \mathbf{n}^{(t)}\right). \end{aligned}$$

At time $t + T$, immediately following the Tth erasure, a new frame arrives and a new cycle begins. Because the oldest frame of the previous cycle has expired by this time, we no longer track its state. The new state at time $t + T$ is $S_{t+T} = (0, \widetilde{\mathbf{n}^{(t)}})$, where

$$\widetilde{\mathbf{n}^{(t)}} \equiv \left[n_2^{(t)}, n_3^{(t)}, \ldots, n_K^{(t)}, 0\right]. \qquad (6)$$

The $\widetilde{\ }$ operator left shifts each frame's state one position to reflect how each frame ages one position per cycle as one frame expires and a new frame arrives. In this simple example, the values $n_i^{(t)}$ did not change (except for the position shifts) because all of the transmission attempts were failures, but this will not be the case in general. In the next section we consider how to analyze the state evolution for the more general case when some transmissions succeed and some are erased.

3.3. Markov Chain Analysis

In this section we present an analysis of the process $\mathcal{S} = \{S_0, S_1, S_2, \ldots\}$, which illustrates how the state space evolves with time. We perform this analysis so we can find the steady-state behavior of $\mathcal{S}$; with this knowledge we can calculate the expected distortion of a particular policy π. The steady-state behavior of $\mathcal{S}$ depends on both the erasure rate of the channel, which determines the chance of a successful transmission, and our policy π, which dictates what layers of which frames should be transmitted, or retransmitted, at any given time. To illustrate this dependency we first examine how $\mathcal{S}$ can change in a single time step.

Consider the possible transitions from a state S_t to S_{t+1}. The transition of the phase component ϕ of the state is completely deterministic:

$$\phi^{(t+1)} = \left(\phi^{(t)} + 1\right) \bmod T. \qquad (7)$$

As a result, we focus our attention on the transitions of the transmission state vector $\mathbf{n}$. There are K components n_i of $\mathbf{n}$, each of which can take on any of $N + 1$ values ($0 \le n_i \le N$), so the maximum number of possible values $\mathbf{n}$ may take on is

$$M = (N+1)^K. \qquad (8)$$

However, there are only two values that $\mathbf{n}^{(t+1)}$ may take on for a given value of $\mathbf{n}^{(t)}$. To see this, first assume (for simplicity of discussion) that at time t we are not at the end of a cycle: $\phi^{(t)} \neq T - 1$. The transmission policy π contains a rule for every state $S_t = (\phi^{(t)}, \mathbf{n}^{(t)})$ which dictates what frame's layer should next be transmitted, or retransmitted if a previous attempt has failed. If $\pi(S)$ is the frame that the policy dictates be chosen for a state S, then at time t we would send the most important layer of frame $\pi(S_t)$ not yet successfully transmitted. This layer is $(n_{\pi(S_t)}^{(t)} + 1)$-most important layer, since the first $n_{\pi(S_t)}^{(t)}$ layers of frame $\pi(S_t)$ have already been transmitted. This transmission can either succeed or be erased. If it is erased then $\mathbf{n}$ does not change; if the transmission succeeds then $\mathbf{n}^{(t+1)}$ differs from $\mathbf{n}^{(t)}$ in only one component:

$$n_{\pi(S_t)}^{(t+1)} = n_{\pi(S_t)}^{(t)} + 1. \qquad (9)$$

Because the probability of an erasure is ε, the one-step transition probability is:

$$P\left(\mathbf{n}^{(t+1)} \,\middle|\, \mathbf{n}^{(t)}\right) = \begin{cases} \varepsilon & \text{if } \mathbf{n}^{(t+1)} = \mathbf{n}^{(t)} \\ 1-\varepsilon & \text{if } n^{(t+1)}_{\pi(S_t)} = n^{(t)}_{\pi(S_t)} + 1, \\ & \quad n^{(t+1)}_j = n^{(t)}_j, \quad j \neq \pi(S_t) \\ 0 & \text{else.} \end{cases} \tag{10}$$

We encapsulate the one-step transition probabilities of all M-possible values of $\mathbf{n}^{(t)}$ in an $M \times M$ state transition matrix P_ϕ, where $\phi = t \bmod T$. Assume that we have a function f which maps each possible value of $\mathbf{n}$ to a unique index $i \in 1, \ldots, M$; for example, $f([1, 0, 0]) = 2$ and $f^{-1}(2) = [1, 0, 0]$. With this mapping function, the components of P_ϕ are defined as:

$$[P_\phi]_{i,j} = P\left(\mathbf{n}^{(t+1)} = f^{-1}(j) \,\middle|\, \mathbf{n}^{(t)} = f^{-1}(i)\right), \tag{11}$$

where the conditional probability can be found using Eq. (10). Each row i of P_ϕ contains two non-zero elements: ε in column i, and $1-\varepsilon$ in column j, where j is determined by the policy π.

In our analysis so far we assumed that we were not at the end of a cycle at time t. However, if we are at the end of a cycle ($\phi^{(t)} = T-1$) the state transition matrix given by Eq. (11) is not quite correct. It fails to account for the arrival of a new frame and the aging of each frame of the previous cycle by one position. This is corrected by right-multiplying the matrix P_{T-1} by a matrix P_a, which left-shifts each state by one position. If we let $\widetilde{\mathbf{n}}$ once again denote a state $\mathbf{n}$ shifted left by one (see Eq. (6)), then the elements of P_a can be defined as:

$$[P_a]_{i,j} = \begin{cases} 1 & \text{if } f^{-1}(j) = \widetilde{f^{-1}(i)} \\ 0 & \text{else.} \end{cases} \tag{12}$$

A new state transition matrix $P'_{T-1} = P_\phi P_a$ now properly describes transitions from $\mathbf{n}^{(t)}$ to $\mathbf{n}^{(t+1)}$ for which $\phi^{(t)} = T-1$.

We can now use the one-step state transition probabilities in order to find the steady-state behavior of $\mathcal{S} = \{S_0, S_1, S_2, \ldots\}$. Because erasures are independent, $\mathcal{S}$ is a discrete-time Markov chain. In other words, the probability of being at some state S_{t+t_1} in the future does not depend on any past knowledge of the process S_{t-t_2} if we know the current state S_t. The only factors that affect transitions from S_t to S_{t+t_1} are the transmission policy and the erasure rate. Their influence can be summarized as follows: ε affects the chance that $\mathbf{n}$ will change, and π determines how it changes.

Because S_t includes phase information, $\mathcal{S}$ is cyclostationary with period T. This stems from the fact that it is not possible to go from a state $S_{t_1} = (\phi^{(t_1)}, \mathbf{n}^{(t_1)})$ at time t_1 to the same state $S_{t_2} = S_{t_1}$ in less than T steps. The process $\mathcal{S}_\phi = \{S_\phi, S_{\phi+T}, S_{\phi+2T}, \ldots\}$, $\phi \in \{0, \ldots, T-1\}$, is a stationary process, however. Its $M \times M$ state transition matrix $P^{(\phi)}$ can be derived from Eqs. (11) and (12):

$$P^{(\phi)} = P_\phi P_{\phi+1} \cdots P_{T-1} P_a P_0 P_1 \cdots P_{\phi-1}. \tag{13}$$

A stationary distribution $\{S_\phi, S_{\phi+T}, S_{\phi+2T}, \ldots\}$ can be found analyzing the matrix of Eq. (13). Let η be the stationary distribution when the oldest live frame expires, i.e., $\phi = L - (K-1)T$. The probability ν_i of transmitting the i most important layers of a frame by its expiration time is calculated by summing out the possible states of the other $K-1$ frames:

$$\nu_i = \sum_{n_2=0}^{N} \sum_{n_3=0}^{N} \cdots \sum_{n_K=0}^{N} \eta_{f(i, n_2, n_3, \ldots, n_K)} \tag{14}$$

Note that although there are $M = (N+1)^K$ possible values of the K-tuple $\mathbf{n}$, the number of *feasible* or *reachable* values may actually be lower. Which states are unfeasible will depend on the policy π. For example, if π dictates that the most important layer of the oldest frame alive is always chosen, then it is not possible to have $n_3 \neq 0$ if $n_2 < N$, since transmission of any message of the third oldest frame would not commence until all messages from the second oldest frame had been sent. The transmission policy will have no rule associated with these states. To get around this problem, we can remove each unfeasible state $\mathbf{n}_u$ from the analysis, and thus have $M' < M$ states. Alternatively, we can still keep M states and assign a probability of 1 to the $[f(\mathbf{n}_u), f(\mathbf{n}_u)]$ entries of each P_ϕ matrix defined by Eq. (10). The stationary probability of these states $\nu_{f(\mathbf{n}_u)}$ will then 0 because they are null-recurrent, and so their presence will not change the result of Eq. (14).

Finally, we wish to convert the stationary distribution of a policy into an objective assessment of that policy's performance. We will use average distortion as our performance metric. Given a rate-distortion function $D(R)$ such that $D(i)$ is the distortion incurred in reconstructing a frame from its i highest priority layers, $0 \le i \le N$, we can compute the average distortion per frame for a transmission policy π as

$$D_\pi = \sum_{i=0}^{N} \nu_i D(i). \tag{15}$$

Equation 15 can be interpreted as a weighted sum: the distortion $D(i)$ incurred by reconstructing a frame from its i most important layers is weighted by the probability ν_i that only these i layers of the frame are successfully sent in time for playback. One constraint on these weights is that the expected number of layers transmitted, N_{avg}, can not exceed either the channel capacity C or the raw transmission rate R:

$$\begin{aligned} N_{\text{avg}} &= \sum_{i=1}^{N} \nu_i i \le \min(C, R) \\ &= \min((1-\varepsilon)T, N), \end{aligned} \tag{16}$$

where $C = (1-\varepsilon)T$ is a basic information theoretic result on the capacity of a binary erasure channel [7]. Even when the rate R is less than the channel capacity C, the bound of Eq. (16) may still be unachievable because the data is time-constrained. Thus although *on average* there may be enough channel capacity to send the entire multimedia signal, in the short term there may be a sequence of many consecutive erasures so that data expires before it is successfully transmitted. The choice of policy can affect both N_{avg} and the distribution of the rate-distortion weights (ν_i). For example, a policy always favoring the most important message of the oldest frame alive will maximize the average number of expected layers N_{avg} and hence the chance of sending all layers across (ν_N). Comparatively, a policy that sends any messages belonging to the most important layer ahead of all others will maximize the chance of sending at least one message in a frame (ν_1) by reducing the chances of both getting none (ν_0) and getting all of them (ν_N). Which policy is better will depend not only how much they can affect ν (which is also dependent on T and L), but also on the shape of the rate distortion curve $D(R)$.

4. Results

Having used our Markov chain model to calculate the average distortion D_π of a policy π, we now formulate an optimization that computes the best transmission policy:

$$\pi^* = \arg\min_{\pi \in \Pi} D_\pi, \tag{17}$$

where Π is the set of all possible policies.

In this section we use our Markov chain analysis to determine π^* for a given set of static network conditions. Because our analysis depends on the erasure rate ε, the relative importance of different priority layers (determined by a rate-distortion function $D(R)$), the lifetime L, and the inter-frame period T, the optimal policy π^* will depend on these factors as well. The steps for finding π^* can be summarized as follows:

1. Fix the four aforementioned parameters: ε, L, T, and $D(R)$.
2. For every possible policy, use the Markov analysis to calculate its average distortion.
3. Determine which policy which produced the minimum distortion.

In the results of our analysis presented below we first focus on the most basic yet interesting case, when there are two layers ($N = 2$) and T and L are valued such that there is a maximum of two frames alive at any time ($K = 2$). This is done to simplify the discussion and interpretation of our results. After analyzing the $N = 2$, $K = 2$ case we then look at more general cases and discuss what aspects of the results differ as the number of layers and overlapping frames increases.

4.1. A Basic Case: $N = 2$, $K = 2$

With two layers ($N = 2$) and at most two frames alive at any given time ($K = 2$), there are at most 4 messages to choose from at any time: the two layers X_1^i and X_2^i of an older frame i, and the two layers X_1^{i+1} and X_2^{i+1} of the next, newer frame $i+1$. We will once again use the O_L-N_H notation introduced in Section 2.2.

In Section 2 we explained that non-obvious transmission decisions arise when we must choose between older, less important messages and newer, more important messages. For the two-layer, two-frame case, this situation arises in only one of the 9 possible values of $\mathbf{n}$: $\mathbf{n} = [1, 0]$. This is the case that O_H was successfully transmitted, and so either O_L or N_H must be chosen next. Each policy that we consider in this section consists of a distinct choice of O_L or N_H for each phase in the cycle such that the older frame has not yet expired.

Because $0 \leq \phi < L - T$, there are 2^{L-T} possible policies to consider.

We first present results that illustrate the average distortion of various policies as a function of the erasure rate ε, when other parameters T, L, and the rate-distortion function $D(R)$ are all held fixed. A non-intuitive result is that the optimal policy π^* always belongs to a subset consisting of two of the possible policies, and these two do not change their message choice for a state **n** as the frames get closer to their expiration times, i.e., as ϕ changes. There is a threshold value of ε at which π^* switches from one of these policies to the other. Another key result is that the best policy on one side of this threshold was also the worst policy on the other side. Next, we illustrate how the shape of $D(R)$ can affect the value of this threshold; this tells us the best policy as a function of both $D(R)$ and ε. Finally, we examine how changing the values of L and T can also affect the best policy.

4.1.1. Effect of the Erasure Rate ε. The erasure rate ε affects the probability of successfully transmitting a message. In this section we examine how it affects the choice of π^*. For fixed values of L and T, the stationary distribution ν of a particular policy depends only on the erasure rate ε. However, the *optimal* policy depends not only ν, and hence ε, but also on the rate-distortion function $D(R)$.

D_π, the distortion associated with a policy given by Eq. (15), is a linear function of $D(R)$. As a result, translating and/or positively scaling $D(R)$ does not change which policy is optimal (i.e., has minimum distortion), since by Eq. (15) all policies' average distortions will be equally scaled and translated. Therefore, we can find and apply a scaling $a > 0$ and translation b to any N-layer rate-distortion function to normalize it so that the resulting $D'(R) = aD(R) + b$ satisfies $D'(0) = 1$ and $D'(N) = 0$. This new distortion function can be completely characterized by the $N - 1$ values of $d_i = D'(i)$, $0 < i < N$, subject to the convexity constraints

$$d_i - d_{i+1} \geq d_{i+1} - d_{i+2}, \quad 0 \leq i \leq N - 2. \tag{18}$$

For the two-layer case, this constraint means that the form of any rate-distortion function can be completely summarized by d_1, and for this case we will refer to d_1 as the "layer gap." The convexity constraint $0 \leq d_1 \leq 0.5$ is necessary so that the high priority layer actually is more important than the low priority layer (or, at the minimum, equally important). If d_1 is close to 0.5 then both layers are of near equal importance; if d_1 is near 0 the high priority layer has much more benefit (distortion reduction) than the low priority layer.

To illustrate the effect of the erasure rate on the distortion of different policies, we fix the layer gap at $d_1 = 0.1$, the inter-frame period at $T = 4$, and the frame lifetime at $L = 8$. In this case the overlap in consecutive frames' lifetimes lasts $L - T = 4$ seconds, and so there are a total of $2^4 = 16$ possible transmission policies. We found that of all possible policies, π^* is always one of the two "phase-invariant" policies, which either always choose O_L or always choose N_H throughout the entire 4-second overlap window, regardless of the phase. In other words, if the chosen message (O_L or N_H) is erased, these two policies always retransmit it until it succeeds or it expires, whichever comes first. The optimality of the phase-invariant policies can be seen in Fig. 3, which displays the average distortion as a function of ε incurred by these two transmission policies and a third, phase-varying "hybrid" policy. The hybrid policy shown favors O_L for the first two transmissions in a cycle ($\phi = \{0, 1\}$) and then switches to favor N_H for the last two transmissions ($\phi = \{2, 3\}$). The scale of the y-axis can be interpreted as follows: assuming that the distortion function is mean squared error, a one-decade decrease in distortion corresponds to a 10 dB increase in signal-to-noise ratio.

Figure 3 shows that for low values of ε, the best policy always favors the O_L, for high values of ε, the best policy always favors the N_H; and there is a value of ε where the two policies have equal distortion (≈0.44 here). When the erasure rate is low, sending O_L instead

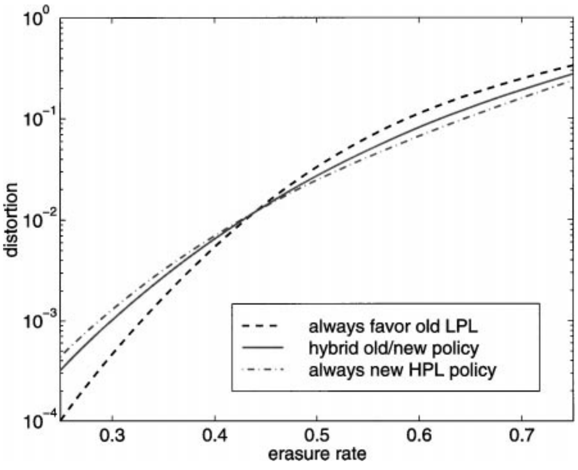

Figure 3. Distortion versus erasure rate for 3 different decision policies, for $T = 4$, $L = 8$, and $d_1 = 0.1$.

of N_H is better because O_L will expire sooner, and although this choice reduces the number of possible transmission attempts for the more important N_H, it is unlikely to need them all. However, when the erasure increases it increases the average number of attempts needed to get N_H to the receiver, and hence sending N_H before O_L becomes more beneficial, even though O_L may expire before the sender succeeds with N_H.

An equally important finding is that for values of ε in which the O_L phase-invariant policy is optimal, the N_H phase-invariant policy is not only suboptimal, but it is also the *worst* possible policy. The converse holds true as well. In general, the two optimal policy distortion curves form the upper and lower boundaries of a performance envelope between which all other policies' performance curves must lie. Note that although we have only shown results from 3 of the 16 possible policies, we did find that the distortion curves of other 13 do all lie between the envelope formed by the curves of the two phase-invariant policies. Also, although we have not proven this optimal/worst nature of the two phase-invariant policies (it was identified through exhaustive search of all possible policies), we found that this property held true for all other combinations of d_1, T, and L that we examined.

4.1.2. Effect of the Layer Gap d_1. In the preceding section we found that when all parameters except the erasure rate were fixed, the optimal policy could be characterized by the threshold value of the erasure rate: if the erasure rate is below this threshold, π^* always chooses O_L; if above, it chooses N_H. In this section we examine how the layer gap affects the value of this threshold. Figure 4 illustrates the location of this threshold (shown on the x-axis) as the layer gap is varied between 0 and 0.5 (y-axis), for an inter-frame period of 3 and a frame lifetime of 5 ($T = 3$, $L = 5$). Area A to the left of the curve indicates when the phase-invariant policy favoring O_L is optimal; area B to the curve's right indicates that the N_H phase-invariant policy is optimal. The curve was obtained by analytically solving for the average distortions D_π of the two policies as a function of ε and d_1, setting them equal and numerically solving for ε as d_1 was varied. We verified the correctness of the curve by sampling the $\varepsilon \times d_1$ plane, finding π^* through exhaustive search, and confirming that the O_L-favoring policy was indeed optimal for all points lying in area A, and likewise for area B.

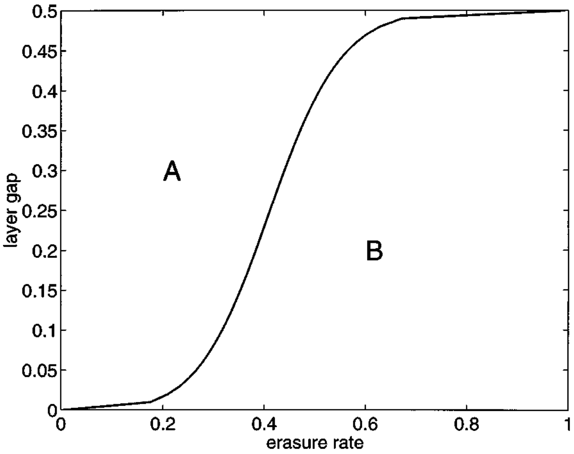

Figure 4. Optimal decision policy as a function of ε and d, for $T = 3$ and $L = 5$.

Figure 4 shows that as the layers become more equal in importance (d_1 increases), the erasure rate threshold moves to the right. This makes intuitive sense: if there is a small disparity between the layers' importance, then O_L is almost as beneficial as N_H, and thus unless the erasure rate is high it is better to send O_L because it expires sooner. Conversely, if the high priority layer is much more important, then the erasure rate does not have to be as high before it makes sense to start favoring N_H over O_L; this increases the chance of successfully transmitting this more important message.

4.1.3. Effect of the Inter-Frame Period T. We also examined the effect of the inter-frame period on the erasure rate threshold, and found that increasing T tends to increase the threshold. Figure 5 illustrates this effect; it shows threshold curves (as described in the preceding

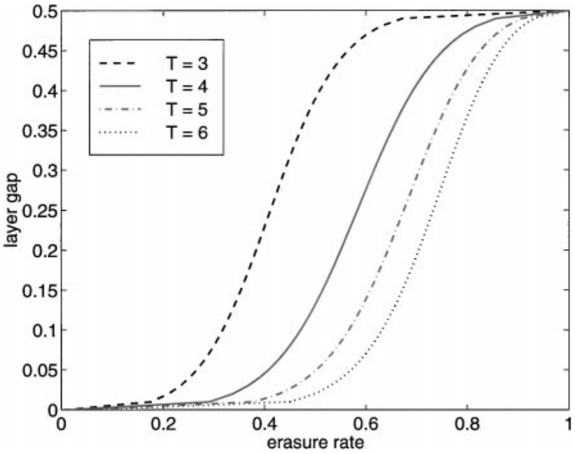

Figure 5. Effect of T on the optimal decision policy.

section) for each value of T between 3 to 6; L is set to $T+2$ for all cases. Increasing T causes the curve's knee to move further to the right; because increases in T provide more transmission opportunities per frame, the erasure rate must also increase before sending N_H over O_L becomes more beneficial. We also see that as T increases, the spacing between the curves becomes smaller, but the slope of the knees remains relatively fixed. This indicates that changing T does not change the amount of influence the layer gap has on the erasure rate threshold value.

4.1.4. Effect of the Frame Lifetime L. Finally, we examined how the frame lifetime L affects the location of the erasure rate threshold, and hence the optimal policy π^*. In general, we found that the lifetime has the following effects:

- Increasing the lifetime moves the threshold curve to the *left*, so that it becomes more beneficial to send the N_H over the O_L at even lower erasure rates. We hypothesize that this is because if the sender chooses to send the more important N_H before the older O_L, increasing the lifetime increases the chance that N_H is successfully transmitted *before* O_L expires, thereby increasing the chance that O_L can be sent as well.
- Increasing the lifetime decreases the impact of the layer gap on the choice of π^*. This is reflected by a steeper threshold curve.
- The magnitude of the difference in average distortion between the best and worst policies increases as the lifetime increases. This is because a longer lifetime results in a longer overlap between consecutive frames' lifetimes, and hence the transmission policy can influence a the sender's behavior over a larger fraction of the T second transmission cycle, for better or worse.

The first two properties are illustrated Figs. 6 and 7, which show the threshold curves for various L-values when T equals 3 and 4, respectively. In each figure L was varied between $T+1$ and $2T$. The latter property was confirmed by examining $D_\pi(\varepsilon)$ graphs (like Fig. 3) for various lifetimes.

4.2. Generalized Results

We now turn our attention to finding the best policy for more general cases than the $N=2$, $K=2$ case studied above. We begin by looking at what changes for the three overlapping-frame case ($K=3$). We then study the case when the signal is encoded into 3 layers ($N=3$). Both of these changes result in an increased number of states that require non-obvious policy decisions. Recall that these decisions arise for cases when one must choose between transmitting a less important message of an older frame and a more important message of a newer frame. We can express this relationship in terms of the state $S=[\phi, \mathbf{n}=[n_1, n_2, \ldots, n_K]$, where frame 1 is the oldest frame and frame K is the newest one. The state $\mathbf{n}^{(t)}$ requires a non-obvious transmission choice if there exists at least one pair of indices

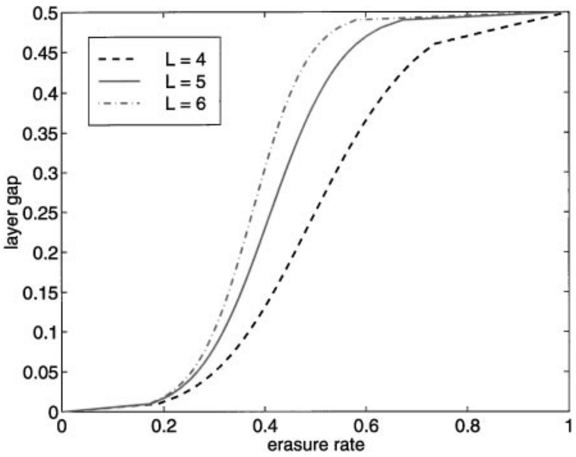

Figure 6. Effect of L on the optimal decision policy. $T=3$ for all curves.

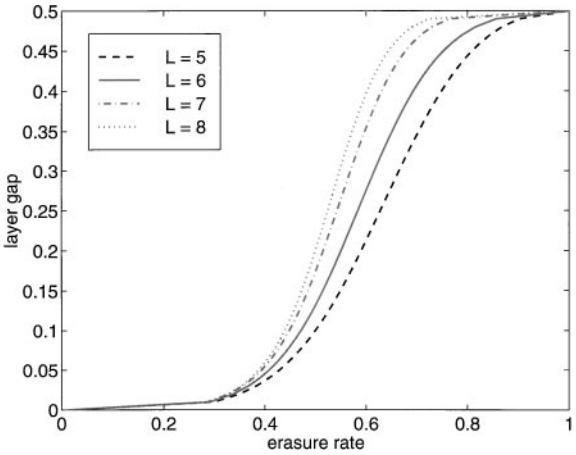

Figure 7. Effect of L on the optimal decision policy. $T=4$ for all curves.

i and j such that

$$\begin{aligned} n_j &< n_i < N, \\ i &< j, \quad \text{and} \\ 1 &< i \quad \text{if } \phi \geq L - (K-1)T. \end{aligned} \tag{19}$$

The first two conditions state that frame j must be newer than frame i but have had fewer messages transmitted, and that not all of frame i's messages have been transmitted. The last condition states that if time t falls in the latter part of a cycle (if there is one) such that the oldest frame has expired, then the oldest frame cannot be considered for transmission.

The number of unique pairs of indices satisfying Eq. (19) can be greater than the number possible messages to choose from for that state $S = [\phi, \mathbf{n}]$, because when there are ties among states (e.g., $n_i = n_{i+1}$) the older frame should be chosen. Let the number of message choices be $n_{\text{msg}}(S)$. Assuming S requires a decision (the conditions of Eq. (19) are satisfied), then $n_{\text{msg}}(S)$ is at least 2 and is upper bounded by:

$$n_{\text{msg}}(S) \leq \begin{cases} \min(K, N) & \text{if } \phi < L - (K-1)T, \\ \min(K-1, N) & \text{else.} \end{cases} \tag{20}$$

If N and K are equal then there is one state vector $\mathbf{n}$ which results in N messages to choose from:

$$\mathbf{n} = [N-1, N-2, \ldots, 1, 0]. \tag{21}$$

4.2.1. Increasing the Number of Overlapping Frames. Increasing the number of frames whose lifetimes overlap (K) increases the dimensionality of $\mathbf{n}$, and hence increases the number of states requiring decisions (the number of states satisfying Eq. (19) increases). In this section we examine the best policies for $K = 3$ and $N = 2$. There are now 3 states during the full-overlap period requiring decisions: [1, 0, 0], [1, 1, 0], [2, 1, 0]. After the oldest frame expires, we have one state requiring a decision: [x, 1, 0], where the value of x is irrelevant to the decision. Because we have two layers, for all of these states there are still only two choices: O_L, an older less important layer (layer 2), and N_H, a newer more important layer (layer 1). These choices are summarized in the first four columns of Table 3. We have used the X_j^i notation to identify messages: X_j^i corresponds to layer j of frame i. When one of the possible message choices belongs to frame i and there is a $k > i$ such that $n_k = n_i$, we have chosen $X_{n_i+1}^i$ for transmission because frame i has the earlier expiration time.

Table 3. Best policies for $N = 2$ and $K = 3$.

State		Choices		Policies	
ϕ	$\mathbf{n}$	O_L	N_H	A	B
0, 1	[1, 0, 0]	X_2^1	X_1^2	O_L	N_H
0, 1	[1, 1, 0]	X_2^1	X_1^3	O_L	O_L
0, 1	[2, 1, 0]	X_2^2	X_1^3	O_L	O_L
2	[x, 1, 0]	X_2^2	X_2^3	O_L	O_L

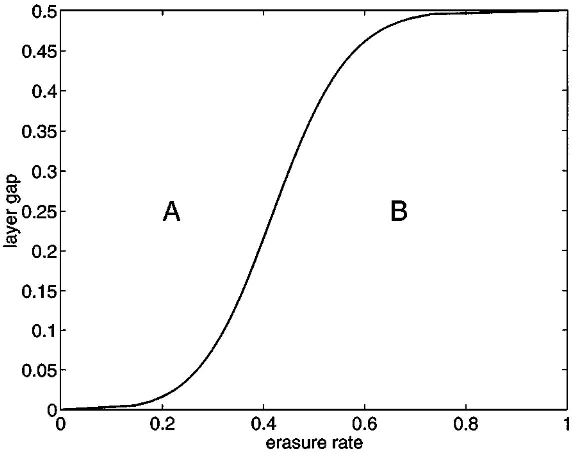

Figure 8. Crossover curve for $T = 3$, $L = 8$, and $N = 2$. Policies A and B are defined in Table 3.

For all cases that we analyzed for which $N = 2$ case and values of L and T such that $K = 3$, we once again found that the best policy belonged to a subset of size two. Figure 8 shows a crossover curve between these two policies for $T = 3$ and $L = 8$. For this case there are 128 possible policies and of these 16 are time-invariant. We found that two best policies, A and B, were also time-invariant as well, and we have listed their transmission decisions in Table 3. These policies differ only in their choice for state $[1, 0, 0]$; for all other states they choose O_L. This indicates that the sender should not be too conservative in ensuring the high priority message of future frames gets transmitted; in this instance the high priority message of the newest frame (frame 3) before transmitting the low priority message of the middle-aged frame (frame 2).

4.2.2. Increasing the Number of Layers. Increasing the number of layers N increases the number of values each component n_i of $\mathbf{n}$ can take on, and so this also increases the number of states requiring policy decisions. In this section we present results for the case that $K = 2$ and $N = 3$. Because there are two overlapping frames, we choose between a less important layer of the older frame and a more important layer of the newer frame. There are 3 states during the overlap period requiring decisions: [1, 0], [2, 0], and [2, 1] (with $K = 2$ there are no policy decisions needed after the older frame expires). These states correspond to deciding between, respectively, the older frame's "medium" priority message and the newer frame's most priority message, the older lowest priority message and the newer most priority message, and the older lowest priority message and the newer medium priority message. We summarize these choices for $T = 4$ and $L = 6$ in the first four columns of Table 4, once again using the X^i_j notation to indicate the messages to be considered for transmission. Note that the decision for state [2, 1] does not need to be made at $\phi = 0$ because at the start of a cycle the new frame has just arrived, so n_2 must be 0, and thus this state cannot be achieved.

Unlike our previous results for two layer cases, with $N = 3$ we found that the subset of optimal policies was greater than two. Of the 32 possible policies (which includes a subset of 8 phase invariant ones), we found 5 different phase invariant policies which formed the optimal subset. These five policies are labeled A through E and their decisions are listed in Table 4. Note that because there are 3 layers, we now need 2 parameters to describe the rate distortion curve: d_1 and d_2. The shaded areas of Fig. 9 correspond to the areas of the $\varepsilon \times d_2$ plane for which each policy is optimal, for d_1 fixed at 0.5. The graph was obtained by finding the best distortion of all possible policies over a 100 × 100 grid of sampled values of the ε and d_2. Note that the range of d_2 is determined by the value of d_1 and the convexity constraints of Eq. (18).

The following behavior is indicated by the results shown in Fig. 9. As we move left to right, the erasure rate increases, and the optimal policy increasingly chooses the more important N_H over O_H to compensate. We see that at low erasure rates we start out at policy A, which always favors all of the O_L's. As ε increases we next switch to B, whose only difference with A is for state [2, 0], for which B chooses the newer highest priority message over the older's lowest priority message. This state's decision is the first to change with increasing erasure rate because it is the state whose eligible messages have the greatest discrepancy in importance (distortion). As ε continues to increase the next policy we encounter is either C or D, depending on the value of d_2. If d_2 is low then the lowest priority message is much less important than the medium priority message, and so we move to policy C, which chooses the newer medium priority message over the older lowest priority one. However, increasing d_2 not only decreases the gap in importance between the two lower priority messages, but it also increases the performance gap between the highest priority message and the medium priority message. Thus for high d_2 we transition from policy B to policy D instead, which chooses the newer highest priority message over the older medium priority one. Our final transition is to policy E, which chooses all of the higher priority newer messages over the lower priority older messages. Note that when $d_2 = d_1/2 = .25$ this final transition never occurs, and D is still the best even at the highest erasure rates. This is because for this case the two lower priority messages are of identical importance, and so the older lowest priority message should always be sent instead of the newer "medium" (but equal) priority message.

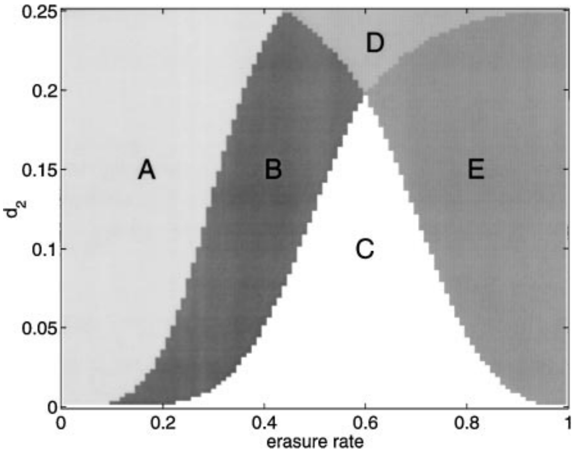

Figure 9. Optimal policies as a function of ε and d_2 for $T = 4$, $L = 6$, $N = 3$, and $d_1 = 0.5$. Policies A through E are defined in Table 4.

Table 4. Best policies for $N = 3$ and $K = 2$.

State		Choices		Policies				
ϕ	$\mathbf{n}$	O_L	N_H	A	B	C	D	E
0, 1	[1, 0]	X^1_2	X^2_1	O_L	O_L	O_L	N_H	N_H
0, 1	[2, 0]	X^1_3	X^2_1	O_L	N_H	N_H	N_H	N_H
1	[2, 1]	X^1_3	X^2_2	O_L	O_L	N_H	O_L	N_H

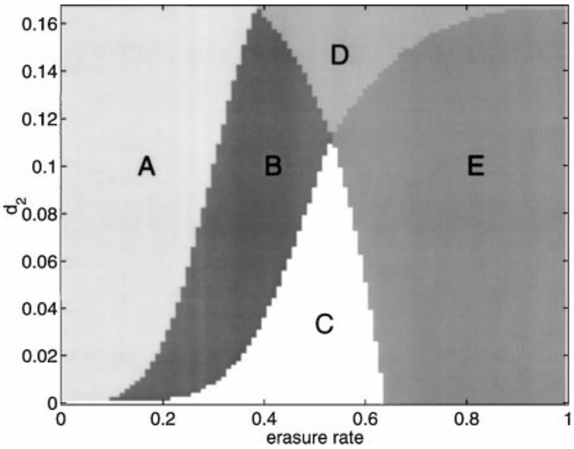

Figure 10. Optimal policies as a function of ε and d_2 for $T = 4$, $L = 6$, $N = 3$, and $d_1 = 1/3$.

Similarly, note that when $d_2 = 0$ for $d_1 = .5$, the lowest priority message has no value and the two higher priority messages are of equal importance. The best policy for all erasure rates is thus C, which always transmits either of the two newer higher priority messages over the older lowest priority one and also chooses to send the older medium priority message instead of the newer high priority one. When $d_1 < 0.5$, the highest priority message will always be more important than the medium priority one, and so for high erasure rates policy C is not necessarily the best. This is illustrated by Fig. 10, which shows the best policies when $d_1 = 1/3$.

5. Future and Related Work

The tradeoff we analyzed in choosing between messages differing in both priority and playback deadlines has analogies to delay-constrained class-based queuing, in which a switch must choose between packets of different priorities (classes) with different deadlines. Such queuing problems have been examined in [8] and [9]. An important distinction is that in these works, the arrival times of packets (i.e, production times of layers) are random and geometrically distributed; in our case, we have known deterministic and periodic arrival times of messages.

Another interesting optimization for streaming layered media is posed in [10]. This work also examines optimally choosing between a base layer and an enhancement layer of a video stream, but it employs a different network model. The model differs from our own in that it assumes the available bandwidth varies randomly (as might result when using a congestion control algorithm) and that the network is loss-free (so that retransmissions are unnecessary); our model assumes fixed available bandwidth but random losses. Other differences include the assumptions in [10] that the entire video signal is available on the server prior to the start of playback (as is the case with pre-recorded media clips) and that an unlimited amount of data can be prefetched by the server and buffered for playback at the receiver. In our model, the stream is either generated or fetched only as playback is occuring (as is the case with live media sources), and as a result the receiver's playback buffer is a fixed and finite size proportional to the chosen playback delay. These modelling differences in [10] lead to a different optimization problem—how to dynamically allocate the a priori unknown available bandwidth between the two video layers when it is known that all transmisisons will succeed.

Much of the significant body of work that has been done on streaming multimedia is oriented towards improving the performance of *interactive* multimedia streams. Because delay requirements of interactive multimedia (less than 200 ms by some measures [1]) typically preclude soft ARQ for error recovery, interactive multimedia research has explored alternative ways to improving signal quality. For example, many algorithms have been developed to automatically adjust the playback point at the receiver to compensate for variations in network delay (jitter) [11–15]. Such techniques are complementary to our work, as the playback delay should account for both the average round trip time needed for a retransmission plus a additional factor to account for variations in the delay experienced by retransmissions. Similarly, other techniques to deal with packet loss in interactive media streams, such as error concealment [16–18] and forward error correction (FEC) [18–23] can also be used in conjunction with retransmission-based loss recovery.

One disadvantage of FEC is that the error correction is *forward*; because the source does not know a priori which packets will be lost, it sends redundant information even if it is not actually needed. ARQ retransmission schemes, on the other hand, only send extra information that the sender believes has been lost (whether this belief is accurate depends on the specifics of the protocol and network). As a result they do not unnecessarily waste bandwidth when there is no packet loss, and they can easily adapt to changes in the loss rates. Thus studies have examined soft ARQ for both unicast [24, 25] and multicast [2, 26, 27] streaming multimedia. One way our work differs from all of these is that we assume there is an overall transmission rate limit, so that a retransmission of one message can come at the expense of the first transmission of another; these other works assume that enough bandwidth is available for any retransmissions the sender decides to send, and instead focus on soft ARQ protocol issues. And although it is not soft-ARQ based, [28] describes FLITT, an FEC-based scheme for transmission of layered images in a finite amount of time (this time is determined by a transmission rate that is fixed for the image).

In our analysis of the transmission model of Fig. 1, we found that the optimal transmission policy was always phase-invariant. For the two-layer ($N = 2$) case, the best policy was found to belong to a subset of two phase-invariant policies (for $K = 2$ there are only two phase-invariant policies). For $N = 3$ we found an optimal subset consisting of five phase-invariant policies. We have observed similar results for all values of L and T tested such that $K \leq 3$ when $N \leq 3$. A practical limitation of our analysis is that not only is the size of the state space exponential in N and K, but the size of the set of possible policies (Π) is also exponential in these factors and in T as well. This has limited our ability to find the optimal policy for arbitrary values of these parameters. As a result one area of future work is the exploration of other approaches to analyzing this transmission problem, such as Markov decision analysis or using approximations. Also, note that while the aforementioned properties have been strongly indicated by our results, they come from exhaustive search and hence have not been proven to be true. Such proofs remain an open area of research.

A limitation of our analysis is the zero-network-delay assumption. Because this assumption clearly does not hold in the Internet, it could be eliminated in future work. However, this assumption was made to make the analysis tractable; a difficulty in accounting for network delay is that the delay leads to an explosion in the state space, and as a result the number of potential transmission policies. The complexity introduced by non-zero delay could be reduced, to a degree, by assuming a balanced multiple description encoding of the media signal [4]. Another limitation of our work is our assumption that packet erasures are independent events; Internet losses are often very correlated. Future work could account for correlation by incorporating the network status into the state space (for example, using a 2-state Gilbert model for the loss process).

Finally, we have performed our analysis for the case that the model parameters are fixed over time. Although T is fixed for a given multimedia encoding, and $D(R)$ is arguably so, both the erasure rate ε and data lifetime L are likely to change. We have seen from our results that the best policies are phase-invariant and relatively few in number. The value(s) of ε for which the optimal policy changes thus could be pre-computed for various values of L. A protocol could then adaptively estimate the data lifetime L (which can be defined to account for variations in the network delay) and lookup crossover value(s) of ε, and finally compare the results to the current estimate of ε in order to choose the best policy. We have examined the feasibility and performance of such an adaptive protocol for the two-layer, two-overlap case in [29].

6. Conclusion

We have examined optimal transmission policies for real-time layered multimedia data, and we looked at the specific case of using limited retransmission over a BEC with instantaneous feedback. We illustrated that there are non-trivial transmission choices to be made between older, lower priority layers and newer, higher priority layers. We presented a Markov chain analysis for evaluating the efficacy of a specific transmission policy. We then examined how the optimal choice of policy varies as a function of the lifetime of the data, the period between frames, the erasure rate, and the relative importance (in a rate-distortion sense) of the layers. A key result of our analysis is that, for fixed network conditions, the best transmission policy is time-invariant and does not change as layers approach their expiration.

Acknowledgments

This work was supported in part by the Defense Advanced Research Projects Agency under Grant DARPA N66001-96-C-8508, the California State MICRO Program, and the National Science Foundation under Grant NSF CDA-9401156.

References

1. P. Brady, “Effects of Transmission Delay on Conversational Behavior on Echo-Free Telephone Circuits,” *Bell Syst. Tech. J.*, vol. 50, no. 1, 1971, pp. 115–134.

2. X. Xu, A. Myers, H. Zhang, and R. Yavatkar, “Resilient Multicast Support for Continuous-Media Applications,” in *Proc. NOSSDAV*, St. Louis, MO, May 1997, pp. 183–194.
3. M. Lucas, B. Dempsey, and A. Weaver, “MESH: Distributed Error Recovery for Multimedia Streams in Wide-Area Multicast Networks,” in Proc. *IEEE Int. Conf. on Commun.*, Montreal, Quebec, June 1997, vol. 2 pp. 1127–1132.
4. V. Vaishampayan, “Design of Multiple Description Scalar Quantizers,” *IEEE-Trans. Info. Theory*, vol. 39, no. 3, 1993, pp. 821–834.
5. S. McCanne, V. Jacobson, and M. Vetterli, “Receiver-Driven Layered Multicast,” in *Proc. ACM SIGCOMM*, Stanford, CA, Aug. 1996, pp. 26–30.
6. R. Rejaie, M. Handley, and D. Estrin, “RAP: An End-to-End Rate-Based Congestion Control Mechanism for Realtime Streams in the Internet,” in *Proc. IEEE INFOCOM*, New York, NY, March 1999.
7. T. Cover and J. Thomas, *Elements of Information Theory*, New York: John Wiley and Sons, 1991.
8. P. Bhattacharya and A. Ephremides, “Optimal Scheduling with Strict Deadlines,” *IEEE Trans. Automat. Control*, vol. 34, no. 7, 1989, pp. 721–728.
9. P. Bhattacharya and A. Ephremides, “Optimal Allocation of a Server Between Two Queues with Due Times,” *IEEE Trans. Automat. Control*, vol. 36, no. 12, 1991, pp. 1417–1423.
10. D. Saparilla and K. Ross, “Optimal Streaming of Layered Video,” in *Proc. IEEE INFOCOM*, Tel Aviv, Israel, March 2000.
11. M. Lara-Barron and G. Lockhard, “Speech Encoding and Reconstruction for Packet-Based Networks,” in *IEE Colloqium on Coding for Packet Video and Speech Transmission*, 1992, vol. 199, pp. 1–4.
12. D. Clark, S. Shenker, and L. Zhang, “Supporting Realtime Applications in an Integrated Services Packet Network: Architecture and Mechanism,” in *Proceedings of SIGCOMM ’92*, Baltimore, Maryland: ACM, Aug. 1992, pp. 14–26.
13. R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, “Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks,” in *Proc. IEEE INFOCOM*, Toronto, Ontario, June 1994, pp. 680–688.
14. V. Jacobson, SIGCOMM ’94 Tutorial: Multimedia Conferencing on the Internet, Aug, 1994.
15. S. Moon, J. Kurose, and D. Towsley, “Packet Audio Playout Delay Adjustment: Performance Bounds and Algorithms,” *Multimedia Systems*, vol. 6, no. 1, 1998, pp. 17–28.
16. M. Wada, “Selective Recovery of Video Packet Loss Using Error Concealment,” *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, 1989, pp. 807–814.
17. O. Wasem, D. Goodman, C. Dvorak, and H. Page. “The Effect of Waveform Substitution on the Quality of PCM Packet Communications,” *IEEE Trans. Acoust. Speech Signal Proc.*, vol. 26, no. 3, 1998, pp. 342–348.
18. V. Hardman, M.A. Sasse, M. Handley, and A. Watson, “Reliable Audio for Use Over the Internet,” in *Proc. INET*, 1995.
19. N. Shacham and P. McKenney, “Packet Recovery in High-Speed Networks Using Coding and Buffer Management,” in *Proc. IEEE INFOCOM*, San Francisco, CA, June 1990, vol. 1, pp. 124–131.
20. E.W. Biersack, “A Simulation Study of Forward Error Correction in ATM Networks,” *Computer Communication Review*, vol. 22, no. 1, 1992, pp. 36–47.
21. I. Cidon, A. Khamisy, and M. Sidi, “Analysis of Packet Loss Processes in High-Speed Networks,” *IEEE-Trans-Info. Theory*, vol. 39, no. 1, 1993, pp. 98–108.
22. J.-C. Bolot and A.V. Garcia, “The Case for FEC-Based Error Control for Packet Audio in the Internet,” To appear in ACM Multimedia System.
23. M. Podolsky, C. Romer, and S. McCanne, “Simulation of FEC-Based Error Control for Packet Audio on the Internet,” in *Proc. IEEE INFOCOM*, San Francisco, CA, March 1998, vol. 2, pp. 505–515.
24. C. Papadopoulos and G. Parulkar, “Retransmission-Based Error Control for Continuous Media Applications,” in *Proc. NOSSDAV*, April 1996, pp. 5–12.
25. B. Dempsey, J. Liebeherr, and A. Weaver, “On Retransmission-Based Error Control for Continuous Media Traffic in Packet-Switching Networks,” *Computer Networks and ISDN System Journal*, vol. 28, no. 5, 1996, pp. 719–36.
26. S. Pejhan, M. Schwartz, and D. Anastassiou, “Error Control Using Retransmission Schemes in Multicast Transport Protocols for Real-Time Media,” *IEEE/ACM Trans. Networking.*, vol. 4, no. 3, 1996, pp. 413–427.
27. L. Xue, S. Paul, P. Pancha, and M. Ammar, “Layered Video Multicast with Retransmission (LVMR): Evaluation of Error Recovery Schemes,” in *Proc. nossdav*, St. Louis, MO, May 1997, pp. 161–172.
28. J. Danskin, G. Davis, and X. Song, “Fast Lossy Internet Image Transmission,” in *Proc. ACM Multimedia*, San Francisco, CA, Nov. 1995. pp. 321–331.
29. M. Podolsky, S. McCanne, and M. Vetterli, “Soft ARQ for Streaming Layered Multimedia,” Technical Report, UC-Berkeley/ERL, Nov. 1998.

Matthew G. Podolsky was born in Fountain Hill, PA, in 1972. He received the B.S. degree in eletrical engineering from Cornell University, Ithaca, NY, in 1994. He received the M.S. degree in electrical engineering from the University of California, Berkeley, in 1996. He was an intern at Sandia Network Laboratories, Livermore, CA, during 1992–1994. He worked at TCSI, Inc., Berkeley, CA, in 1995, and at Digital Equipment Corporation’s System Research Center, Palo Alto, CA, in 1997. He is currently completing the Ph.D. degree in the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley. His research interests include internetworking and signal processing, with an emphasis on performance enhancements for streaming multimedia. Mr. Podolsky is a member of the IEEE, Tau Beta Pi, and Eta Kappa Nu.
m.podolsky@ieee.org

Steven McCanne received the B.S. degree with high honors in Electrical Engineering and Computer Science and the Ph.D. degree in

Computer Science from the University of California, Berkeley, in 1990 and 1996 respectively. From 1988 to 1996, he was on the staff at the Lawrence Berkeley National Laboratory where he co-developed the network and application architecture and software that underlies the Internet Multicast Backbone or “MBone” tools. In 1996, he joined the faculty of the Department of Electrical Engineering and Computer Science at the University of California, Berkeley, where he is currently an Assistant Professor. His research interests include multimedia networking, multicast communication protocols, signal compression, remote collaboration technologies, and network simulation architectures. His dissertation work, which was awarded the ACM doctoral dissertation award for 1996–97, investigates methods for scalable video representation and transmission over hetergeneous multicast networks. Dr. McCanne received the Best Student Paper award at both Winter USENIX ’93 and ACM Multimedia ’95, the AT&T Graduate Scholarship, and the 1995 R&D 100 Award for the development of the MBone tools. He is a member of the IEEE, the ACM, the AAAS, and the Internet Engineering Task Force.
mccanne@eecs.berkeley.edu

Martin Vetterli received the Dipl. El.-Ing. degree from ETH Zürich (ETHZ), Switzerland, in 1981, the MS degree from Stanford University in 1982, and the Doctorat és Science degree from EPF Lausanne (EPFL), Switzerland, in 1986. HE was a Research Assistant at Stanford and EPFL, and has worked for Siemens and AT&T Bell Laboratories. In 1986, he joined Columbia University in New York where he was last an Associate Professor of Electrical Engineering and co-director of the Image and Advanced Television Laboratory. In 1993, he joined the University of California at Berkeley where he was a Professor in the Dept. of Electrical Engineering and Computer Sciences until 1997, and holds now Adjunct Professor position. Since 1995, he is a Professor of Communication Systems at EPF Lausanne, Switzerland, where he chaired the Communications Systems Division (1996/97), and heads of the Audio-Visual Communications Laboratory. He held visiting positions at ETHZ (1990) and Standford (1998). He is a fellow of the IEEE, a member of SIAM, and he was the Area Editor for Speech, Image, Video, and Signal Processing of the IEEE Transactions on Communications. He is also on the editorial boards of Annals of Telecommunications, Applied and Computational Harmonic Analysis and The Journal of Fourier Analysis and Applications. He received the Best Paper Award of EURASIP in 1984 for his paper on multidimensional subband coding, the Research Prize of the Brown Bovery Corporation (Switzerland) in 1986 for his doctoral thesis, the IEEE Signal Processing Society’s Senior Award in 1991 and in 1996 (for papers with D. LeGall and K. Ramchandran, respectively). and he is a IEEE Signal Processing Distinguished lecturer in 1999. He received the Swiss National Latsis Prize in 1996 and the SPIE Presdential award in 1999. He was a plenary speaker at various conferences (e.g. 1992 IEEE ICASSP and is the co-author, with J. Kovacevic, of the book *Wavelets and Subband Coding* (Prentice-Hall, 1995). He has published about 75 journal papers on a variety of topics in signal and image processing and holds 5 patents. His research interests include wavelets, multirate signal processing, computational complexity, signal processing for telecommunications, digital video processing and compression and wireless video communications.
martin@eecs.berkeley.edu