

# Effective Bandwidth Based Scheduling for Streaming Media

Sang H. Kang, *Member, IEEE*, and Avidesh Zakhor, *Fellow, IEEE*

**Abstract**—We propose a class of rate-distortion optimized packet scheduling algorithms for streaming media by generating a number of nested substreams, with more important streams embedding less important ones in a progressive manner. Our goal is to determine the optimum substream to send at any moment in time, using feedback information from the receiver and statistical characteristics of the video. To do so, we model the streaming system as a queueing system, compute the run-time decoding failure probability of a group of picture in each substream based on *effective bandwidth* approach, and determine the optimum substream to be sent at that moment in time. We evaluate our scheduling scheme with various video traffic models featuring short-range dependency (SRD), long-range dependency (LRD), and/or multifractal properties. From experiments with real video data, we show that our proposed scheduling scheme outperforms the conventional sequential sending scheme.

**Index Terms**—Multimedia, rate-distortion optimization, scalability, scheduling, streaming.

## I. INTRODUCTION

STREAMING media applications are becoming increasingly more popular on the Internet. Properties of streaming media include the following. First, the media is pre-encoded without the knowledge of run-time channel status during transmission. For this reason, the encoding is desired to be flexible, e.g. using scalability, in order to achieve graceful degradation during channel fluctuations. Second, compared to interactive applications, streaming media applications typically have a pre-roll buffer, e.g., 5–20 s, allowing for rate-distortion optimization and data packet scheduling to adaptively select data to be transmitted, as the channel status varies.

Encoded media data consist of packets with different levels of importance in terms of base/enhancement layers, frame types, and motion/texture fields. In a sense, there is a complex dependency relationship across the different video packets, and lost or late data packets have different impact on video quality. Therefore, it is highly desirable to apply unequal error protection (UEP) to different types of data packets in a rate-distortion optimized manner. As a means of UEP for packet streaming of

stored media, various types of hybrid ARQ algorithms including type-I and type-II for video transmission are investigated in [1] and [2]. These studies improve overall packet throughput, but they do not address UEP and/or rate-distortion optimization problem for streaming media. In rate-distortion optimization context, a heuristic formulation and cost function is proposed for suboptimal video packet scheduling in [3].

In this paper, we address the problem of packet scheduling of streaming media over a lossy network, in a rate-distortion optimized way, assuming reverse channel availability, and no contraction or expansion of the original display duration of clips. Generating a number of nested substreams, with more important streams embedding less important ones in a progressive manner, our scheduling algorithm determines the optimum substream to send at any moment in time, using feedback information from the receiver and statistical characteristics of the video. In our approach, we incorporate a stochastic process to model the nested substreams. Video traffic streams have been shown to have both short-range (SRD) and long-range dependency (LRD) [4]–[8]. Recent research shows video traffic may be modeled as multifractal cascade accounting for the multiscaling property of the traffic [9]–[12]. Our proposed scheduling scheme is based on *effective bandwidth* or *equivalent bandwidth* calculated from various video traffic models including SRD models [13], fractional Brownian motion (FBM) [12], [14], fractional autoregressive integrated moving average (FARIMA) [4], [6], and multifractal cascade [10].

The rest of this paper is organized as follows. In Section II, we analyze rate-distortion optimization problem for error-free channels with restricted bandwidth. In Section III, we take into account the video bit rate fluctuation and channel bandwidth fluctuation to propose a packet-scheduling scheme based on the concept of effective bandwidth. In Section IV, our proposed scheme is evaluated in actual wireline and wireless scenarios. Finally, we conclude in Section V.

## II. RATE-DISTORTION ANALYSIS

In this section, we analyze distortion in streaming over fixed-bandwidth channels, taking into account error concealment techniques. For the sake of simplicity, in this section, we assume that the video is encoded without scalability or MTD.

Assume a video frame sequence  $F_n$  ( $n = 0, 1, 2, \dots$ ) with frame rate  $f$  in frames/s. Suppose that the receiver begins to display  $F_0$  at time  $t = 0$  s.; then the display time or *deadline* of  $F_n$  is  $t = n/f$  s. Assume that the video sequence is divided into GOPs, where a GOP has  $N$  frames consisting of one I frame and  $(N - 1)$  P frames. Letting  $k = (n \bmod N)$ , we have  $F_n = I$  frame if  $k = 0$ , and  $F_n = P_k$  if  $k \neq 0$ . Within each GOP, the  $j$ th

Manuscript received April 8, 2004; revised May 14, 2004. This work was supported by the National Science Foundation under Grant ANI-9905799 and by the Air Force Office of Scientific Research under Contract F49620-00-1-0327. The work of S. H. Kang was supported in part by the IT Research Center (ITRC) Project. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nelson L. S. Fonseca.

S. H. Kang is with the Department of Electrical and Computer Engineering, University of Seoul, Seoul 130-743, Korea (e-mail: shkang@uos.ac.kr).

A. Zakhor is with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720 USA (e-mail: avz@eecs.berkeley.edu).

Digital Object Identifier 10.1109/TMM.2005.858401

P frame is labeled as  $P_j$  ( $j = 1, 2, \dots, N - 1$ ). We divide the data in a GOP into  $H$  importance levels,  $h = 0, 1, \dots, H - 1$ , where highest level  $h = 0$  is assigned to the most important data part, and lowest level  $h = H - 1$ , to the least important data part. In this section, we assign  $h = 0$  to I frames and  $h = j$  to  $P_j$  ( $j = 1, \dots, N - 1$ ).

Let  $S^{(I)}$  and  $S^{(P)}$  denote the size in bits of an I and P frames, respectively, which in this section are assumed to be fixed. For the sake of simplicity, assume frame rate of  $f = N$  frames/s, i.e. a GOP duration is 1 second. The average video rate is then given by  $S^{(I)} + (N - 1)S^{(P)}$  bits/s.

In this paper, we assume video streaming under unstable channel condition in which the channel bandwidth is temporarily less than the average video bit rate. Consider an *error-free restricted-bandwidth* channel with bandwidth of  $C$  bits/s, which is smaller than the average video rate, and greater than half of the average video rate. Error-free channel assumption implies channel throughput is equal to the channel bandwidth,  $C$ , because the sender does not need to retransmit packets.

In this section, we assume that  $F_n$  can be *successfully* decoded at its display time,  $t = n/f$ , only if (a) all its preceding frames within the GOP, i.e.,  $\{F_{n-k}, F_{n-k+1}, \dots, F_{n-1}\}$ , have been successfully decoded, and (b) the data for  $F_n$  is available at the receiver by its deadline. In this context, a frame is either a *decoding success* or *decoding failure*, and hence will be referred to as a successful or failed, decoded frame. Define  $g_0$  as the *decoding success probability* of I frames, and  $g_1, \dots, g_{N-1}$ , the *decoding success probabilities* of  $P_1, \dots, P_{N-1}$ , respectively. Obviously, the number of successfully decoded  $P_j$  frames cannot be greater than that of preceding  $P_{j-1}$  frames, resulting in

$$1 \geq g_0 \geq g_1 \geq \dots \geq g_{N-1} \geq 0. \quad (1)$$

In this section, we assume zero round trip time and a scheduling scheme in which, for a frame missing its deadline, the remaining of the GOP is dropped. Thus, no arriving packet is useless, i.e., the throughput is equal to the goodput defined as traffic rate delivered to the receiver, excluding duplicate packets and useless packets due to any reason. Recalling that the throughput is also equal to the channel bandwidth, we get

$$S^{(I)}g_0 + S^{(P)} \sum_{j=1}^{N-1} g_j = C. \quad (2)$$

We now consider simple error concealment of a failed, decoded frame by copying the latest, successfully decoded frame. Let  $\delta_j$  ( $j = 1, 2, \dots, N$ ) denote distortion caused by the decoding failure of frame  $F_n$  caused by the loss of  $F_{n-j+1}$ , i.e., the receiver displays  $F_{n-j}$  at the display time of  $F_n$ ,  $t = n/f$ . For example, if  $P_2$  is successfully decoded and the following  $P_3$  is lost, then the distortion for  $P_3$  is  $\delta_1$  and for the following  $P_4$  is  $\delta_2$ . Specifically, we define

$$\delta_j = \text{MSE}(F_n, F_{n+j}), \quad j = 1, 2, \dots, N \quad (3)$$

where  $\text{MSE}(F_k, F_{k+j})$  denotes the mean squared error between  $F_k$  and  $F_{k+j}$ . Obviously we have  $0 < \delta_1 < \delta_2 < \dots < \delta_N$ . In this section, we assume that dropping an I frame at the sender occurs only when all frames in previous GOP have been successfully transmitted. Then the distortion caused by loss of an I frame is  $\delta_1$ .

Let  $e_j \equiv 1 - g_j$  denote the decoding failure probability for a  $P_j$ . We can write the distortion of a GOP as the summation of the distortion of each type of frames as follows:

$$\begin{aligned} D &= e_0\delta_1 + e_0\delta_2 + (e_1 - e_0)\delta_1 \\ &\quad + e_0\delta_3 + (e_1 - e_0)\delta_2 + (e_2 - e_1)\delta_1 \\ &\quad \vdots \\ &\quad + e_0\delta_N + (e_1 - e_0)\delta_{N-1} + \dots + (e_{N-1} - e_{N-2})\delta_1 \\ &= \sum_{j=1}^N \delta_j - \sum_{j=0}^{N-1} g_j \delta_{N-j}. \end{aligned} \quad (4)$$

Our goal is to find  $g_j$ 's to minimize  $D$  with constraints (1) and (2). Referring to [15], candidates of optimized condition are given by either (a)

$$\begin{aligned} g_0 &= g_1 = \dots = g_L = 1, \\ g_{L+1} &= \frac{C - S^{(I)} - LS^{(P)}}{L}, \\ \text{and } g_{L+2} &= \dots = g_{N-1} = 0 \end{aligned} \quad (5)$$

where

$$L = \left\lfloor \frac{C - S^{(I)}}{S^{(P)}} \right\rfloor \quad (6)$$

under which we have

$$D = \left(1 - \frac{C - S^{(I)} - LS^{(P)}}{L}\right) \delta_{N-L-1} + \sum_{j=1}^{N-L-2} \delta_j \quad (7)$$

or (b) one of the following  $(N - L - 1)$  conditions

$$\begin{aligned} g_0 &= g_1 = \dots = g_M = \frac{C}{S^{(I)} + MS^{(P)}}, \\ \text{and } g_{M+1} &= \dots = g_{N-1} = 0 \end{aligned} \quad (8)$$

where  $M = L + 1, L + 2, \dots, N - 1$ , and  $\lfloor x \rfloor$  denotes the greatest integer less than or equal to  $x$ , and we have

$$D = \sum_{j=1}^N \delta_j - \frac{C}{S^{(I)} + MS^{(P)}} \sum_{j=0}^M \delta_{N-j}. \quad (9)$$

The meaning of conditions (5) and (8) is as follows. Condition (5) implies sending all I frames and  $P_1$  through  $P_L$  frames with probability one, sending  $P_{L+1}$  with probability  $(C - S^{(I)} - LS^{(P)})/L$ , and dropping  $P_{L+2}$  through  $P_{N-1}$ . Here, we refer to  $L$  as the *protection scope*. Conditions in (8) imply sending all I frames and  $P_1$  through  $P_M$  frames with probability  $C/(S^{(I)} + MS^{(P)})$ , and dropping  $P_{M+1}$  through  $P_{N-1}$ . In this case,  $M$  is

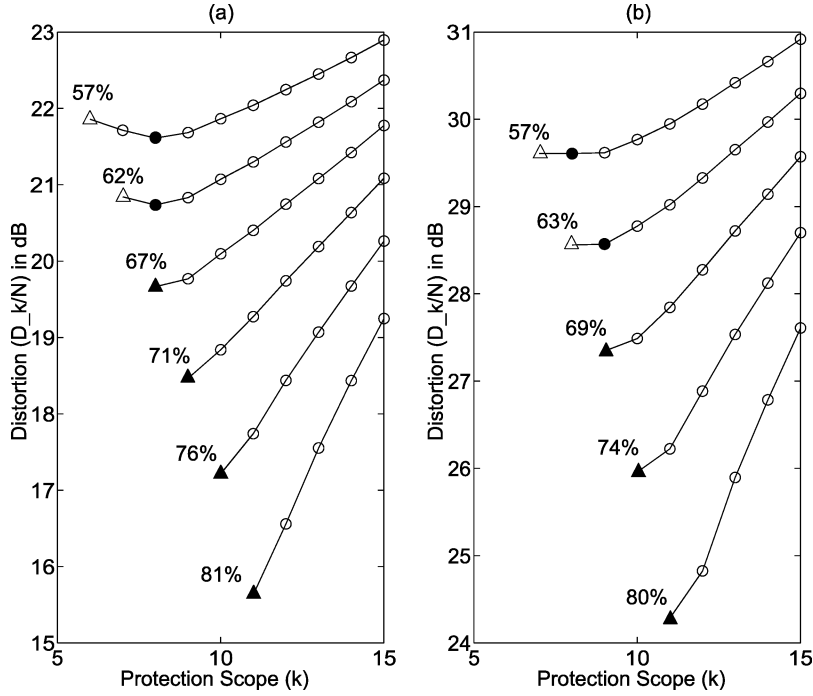


Fig. 1. Normalized GOP distortion as a function of protection scope. (a) Talk show; (b) movie trailer.

the *protection scope*. Note that the protection scope in the latter case,  $M = L + 1, \dots, N - 1$ , is larger than the former case,  $L$ .

Given  $C$ ,  $S^{(I)}$ ,  $S^{(P)}$ ,  $N$ , and  $\delta_j$ 's, we choose the optimum condition to be the one that results in lower distortion as follows. Combining (7) and (9), let  $D_k$  ( $k = L, L+1, \dots, N-1$ ) denote the distortion of a GOP with protection scope of  $k$  according to (5) and (8):

$$D_k = \begin{cases} \sum_{j=1}^{N-k-2} \delta_j \\ + \left(1 - \frac{C - S^{(I)} - kS^{(P)}}{k}\right) \delta_{N-k-1}, & \text{for } k = L, \\ \sum_{j=1}^N \delta_j - \frac{C}{S^{(I)} + kS^{(P)}} \\ \times \sum_{j=0}^k \delta_{N-j}, & \text{for } L+1 \leq k \leq N-1. \end{cases} \quad (10)$$

We compute the optimum  $k_{opt}$  to minimize the distortion,

$$k_{opt} = \arg \min_k D_k, \quad k = L, L+1, \dots, N-1. \quad (11)$$

If  $k_{opt} = L$ , the optimum condition is given by (5), else it is given by

$$g_0 = g_1 = \dots = g_{k_{opt}} = \frac{C}{S^{(I)} + k_{opt}S^{(P)}},$$

and  $g_{k_{opt}+1} = \dots = g_{N-1} = 0. \quad (12)$

Let us call parameter  $k_{opt}$  in (11) the optimum importance level; thus high priority corresponds to frames I through  $P_{k_{opt}}$ .

In Fig. 1, we show frame distortion,  $D_k/N$ , as a function of  $k$  with various channel bandwidth as a percentage of the average video traffic rate, where  $N = 16$ . We use real video traces: MPEG-4 encoded talk show and movie trailer video clips. In each curve, the left most point with a triangle corresponds to condition (5) with  $k = \lfloor (C - S^{(I)})/S^{(P)} \rfloor$ . The optimum  $k_{opt}$  is denoted by filled triangle or circle in each curve. With percentage channel bandwidth from 100% down to 65% of the

average video data rate, the distortion is minimized by setting  $k_{opt} = L$ . With channel bandwidth of 62% and 57%, in talk show video, the optimum conditions are achieved by setting  $k_{opt} = L + 1$  and  $k_{opt} = L + 2$ , respectively, as in (12). With heavily restricted channel bandwidth, sacrificing I frames with large size allows increasing high priority scope significantly; for talk show and movie trailer videos, the ratios are 5.3 and 2.5, respectively, and we observe a more notable effect in the talk show video in Fig. 1. A conclusion in this section is that we achieve optimum condition when  $k_{opt}$  is either equal to or near  $L$ .

### III. PACKET SCHEDULING BASED ON EFFECTIVE BANDWIDTH

In practical situations, it is necessary to consider video rate fluctuations, time-varying queue length in the receiver buffer, and channel bandwidth fluctuations due to congestion or bit errors. In this case, the channel bandwidth, throughput, and goodput are no longer the same. These realistic conditions are taken into account in this section to develop a real-time implementable scheduling algorithm for media packet streaming. Based on the results from the previous section, we take (5) as an optimal condition resulting in minimum distortion in most cases; in addition, (5) yields smaller variability in visual quality across frames, than (8)—an important factor in human perception.

#### A. Effective Bandwidth of Video Data

We assign importance levels to encoded data taking into account MTD and temporal scalability as in [16]. Once data in a GOP is divided into  $H$  parts according to the importance levels, each part is segmented into fixed-size *packets* for transmission. From these  $H$  parts, we construct  $H$  nested substreams, with the ones containing more importance levels embedding the ones with fewer importance levels in a progressive manner as shown

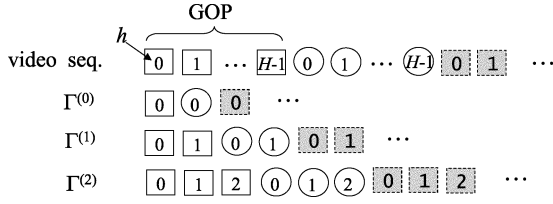


Fig. 2. Examples of substreams and the video rates.

in Fig. 2. Define a random process  $\{X_i^{(h)}; i = 0, 1, \dots\}$  to be the sequence of the size, in packets, of data with importance level  $h$  in the  $i$ th GOP. Also define  $S_i^{(\theta)}$  ( $\theta = 0, 1, \dots, H-1$ ) to be the partial sum such that

$$S_i^{(\theta)} = \sum_{h=0}^{\theta} X_i^{(h)}. \quad (13)$$

Autocovariance of  $S_i^{(\theta)}$  is defined as

$$v^{(\theta)}[\tau] = E \left[ \left( S_i^{(\theta)} - E[S^{(\theta)}] \right) \times \left( S_{i+\tau}^{(\theta)} - E[S^{(\theta)}] \right) \right], \quad \tau = 0, 1, \dots \quad (14)$$

For a threshold  $\theta$ , let  $\Gamma^{(\theta)}$  denote nested substream containing importance levels from  $h = 0$  through  $h = \theta$ . Assuming GOP duration of 1 s, the average data rate of  $\Gamma^{(\theta)}$  is  $E[S^{(\theta)}]$  in packets/s.

For substream  $\Gamma^{(\theta)}$ , we introduce a *virtual* queueing model, with  $\Gamma^{(\theta)}$  as the stochastic input process, an output service rate equal to the channel bandwidth  $C$  packets/s, and a buffer of size  $B^{(\theta)}$ . In our scheduling scheme, we assume VBR video data, which is transmitted at a constant rate of average video rate. Given  $B^{(\theta)}$ , the decoding failure probability for  $\Gamma^{(\theta)}$  is obtained as the probability that the *real* receiver buffer becomes exhaustive, i.e., some data with importance level from zero to  $\theta$  is not available for current frame on display. We take the overflow probability of the *virtual* queueing system as an estimation of the probability that the *real* receiver buffer is empty for  $\Gamma^{(\theta)}$ ; though, the former is overestimation of the latter. In order to make the queueing model effective, the buffer size  $B^{(\theta)}$  of the queueing model needs to be determined carefully by taking into account the receiver buffer status. Specifically, the buffer size,  $B^{(\theta)}$ , in packets, is determined from the pre-stored video duration for substream  $\Gamma^{(\theta)}$ , in the receiver buffer, according to the average data rate of  $\Gamma^{(\theta)}$ . In our approach, we relate  $B^{(\theta)}$  to the number of GOP's for the data with importance levels of  $h = 0$  through  $h = \theta$  as follows. The server has acknowledgment status of every packet. Suppose that the receiver is currently decoding a frame in GOP number  $k$ , and the largest GOP number, for which all packets with importance  $\theta$  have so far been acknowledged, is measured to be  $m_\theta$  ( $m_\theta \geq k$ ). To represent the buffer size corresponding to  $(m_\theta - k)$  seconds in terms of the number of packets in  $\Gamma^{(\theta)}$  whose average rate is  $E[S^{(\theta)}]$  packets/s, we propose to determine  $B^{(\theta)}$  as

$$B^{(\theta)} = (m_\theta - k)E[S^{(\theta)}] \quad [\text{packets}]. \quad (15)$$

In Fig. 3, we show an example of determining of  $B^{(\theta)}$  from a given status of receiver buffer with  $N = 4$ . In each GOP, the frames for which no data is available in the receiver buffer are denoted by tick marks. This is done in order to relate  $B^{(\theta)}$  to

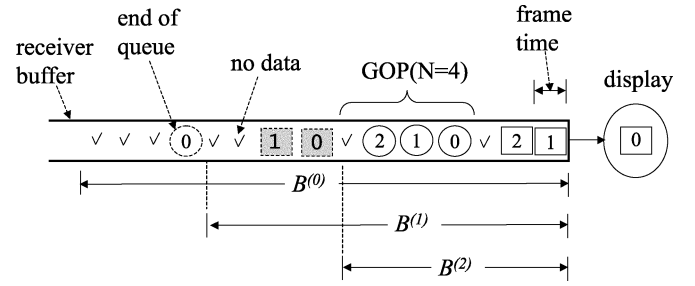


Fig. 3. Example of buffer size of the queueing model from receiver buffer status.

time. Specifically, if the marked *no data* part is not available by its deadline, the receiver displays the last decoded frame.<sup>1</sup> Note that  $E[S^{(\theta)}]$  is a constant which is obtained off-line, whereas  $m_\theta$  and  $k$  are time-varying parameters.<sup>2</sup>

For  $\Gamma^{(\theta)}$ , we define the decoding failure of a GOP as loss of any motion packet in the GOP within  $\Gamma^{(\theta)}$ .<sup>3</sup> Let  $\epsilon_{GOP}^{(\theta)}$  denote GOP decoding failure probability requirement for  $\Gamma^{(\theta)}$ . Also let  $\epsilon_p^{(\theta)}$  be the packet loss probability during the transmission of  $\Gamma^{(\theta)}$ . Given  $\epsilon_{GOP}^{(\theta)}$ , we translate it into the packet loss probability requirement,  $\epsilon_p^{(\theta)}$ , by referring to Appendix:

$$\epsilon_p^{(\theta)} = \begin{cases} 1 - \left( 1 - \epsilon_{GOP}^{(\theta)} \right)^{\frac{1}{E[S^{(\theta)}]}}, & \theta \leq N-1 \\ 1 - \left( 1 - \epsilon_{GOP}^{(\theta)} \right)^{\frac{1}{E[S^{(N-1)}]}}, & \theta > N-1. \end{cases} \quad (16)$$

From queueing theory, we note that  $\epsilon_{GOP}^{(\theta)}$  is a monotonically increasing function of  $\theta$ .

Let  $W_{eff}^{(\theta)}$  denote the *effective bandwidth* of  $\Gamma^{(\theta)}$ ; assuming  $\Gamma^{(\theta)}$  to be an input process to a queueing system with finite buffer size with size  $B^{(\theta)}$ , the effective bandwidth of the data rate process  $\{S_i^{(\theta)}; i = 0, 1, \dots\}$  of  $\Gamma^{(\theta)}$  is defined as the minimum service rate of the queue in order to guarantee the loss probability less than  $\epsilon_p^{(\theta)}$ . In our scheduling scheme, the video traffic is modeled by any stochastic process as long as the effective bandwidth can be obtained in real time. In the following, we show how to calculate the effective bandwidth for well-known video traffic models of  $S^{(\theta)}$ .

1) *Short-Range Dependency (SRD) Model*: Assume that  $\{X_i^{(h)}\}$  is wide sense stationary. Suppose the sequence  $\{Z_k^{(\theta)}\}$ ,  $Z_k = [S_1^{(\theta)} + \dots + S_k^{(\theta)}]/k$ , satisfies the assumptions of the *Gärtner–Ellis theorem* [13]. This class of stationary sources includes Gaussian sources, Markov (and its variants) sources, autoregressive moving average (ARMA) sources, etc. [6], [13], [17]. The effective bandwidth is given by [13]:

$$W_{eff}^{(\theta)} = E[S^{(\theta)}] + \frac{-\log(\epsilon_p^{(\theta)})}{2B^{(\theta)}} \sum_{\tau=0}^{\infty} v^{(\theta)}[\tau]. \quad (17)$$

Here the autocovariance  $v^{(\theta)}[\tau]$  is assumed to have exponential tail, and consequently be infinitely summable corresponding to short-range dependency (SRD). The effective bandwidth

<sup>1</sup>The received packets will be packed in the receiver buffer without gaps in actual implementations.

<sup>2</sup>Here, the number of packets in the receiver buffer within GOP numbers  $k$  through  $m$  has no effect on missing deadlines of future frames.

<sup>3</sup>Due to channel conditions

of  $\Gamma^{(\theta)}$  is larger than its average bandwidth,  $E[S^{(\theta)}]$ , by a certain amount which is a function of the characteristics of input process, buffer size of the queueing system, and the loss probability. Intuitively, the difference becomes larger as the loss probability requirement is tightened and/or variability of the input process increases, and/or as the buffer size of the queueing system decreases.

2) *Fractional Brownian Motion (FBM)*: Long-term memory in video traffic implies that  $\sum_{\tau} v^{(\theta)}[\tau] \rightarrow \infty$ . A widely-used long-range dependency (LRD) model for video traffic is fractional Brownian motion (FBM) where  $S^{(\theta)}$  is modeled as a fractional Gaussian noise. In FBM, the number of arrivals in an observation interval  $(0, t)$  can be represented by a self-similar fluctuation about its expectation:

$$A(t) = mt + \sqrt{am}Z(t) \quad (18)$$

where  $A(t)$  denotes the number of arrivals up to time  $t$ ,  $m$  is the mean arrival rate,  $a$  is the variance coefficient, and  $Z(t)$  is a normalized FBM with Hurst parameter  $H$ . In our notation, it is represented as

$$A^{(\theta)}(t) = E[S^{(\theta)}]t + \sqrt{v^{(\theta)}[0]E[S^{(\theta)}]}Z(t) \quad (19)$$

where a GOP time is assumed to be 1 s. Applying the analysis result in [14] to our scheduling scheme, the effective bandwidth is given by

$$W_{eff}^{(\theta)} = E[S^{(\theta)}] + H \left( -2v^{(\theta)}[0] \log(\epsilon_p^{(\theta)}) \right)^{\frac{1}{2H}} \left( \frac{1-H}{B^{(\theta)}} \right)^{\frac{1-H}{H}}. \quad (20)$$

3) *Fractional Autoregressive Integrated Moving Average (FARIMA)*: Calculation of effective bandwidth based on a model that captures only the long-term memory property is inadequate when the traffic shows both SRD and LRD. FARIMA model is widely used to capture both SRD and LRD of video traffic [4], [6]. We invoke the results in [7], where the video traffic is modeled as log-normal FARIMA, i.e., the histogram of MPEG GOP trace is characterized by log-normal probability density function (PDF), and the effective bandwidth is calculated by transforming the log-normal random variable into the Gaussian random variable and using an iteration method based on the large-deviation theory. First characterize  $\{S^{(\theta)}\}$  as log-normal FARIMA in which the marginal PDF of  $S^{(\theta)}$  is modeled as the log-normal PDF and  $v^{(\theta)}[\tau]$  characterizes both SRD and LRD. Consider the sequence  $\{Z_k^{(\theta)}\}$ ,  $Z_k = [S_1^{(\theta)} + \dots + S_k^{(\theta)}]/k$  of which the mean is given by  $E[Z_k^{(\theta)}] = E[S^{(\theta)}]$  and its variance is given by

$$\bar{\sigma}_k^2 = \text{Var}(Z_k^{(\theta)}) = \frac{1}{k} \sum_{\tau < k} \left(1 - \frac{\tau}{k}\right) v^{(\theta)}[\tau]. \quad (21)$$

Given  $B^{(\theta)}$  and  $\epsilon_p^{(\theta)}$ , we now construct an objective function of  $c$  as transmission rate of the queueing system, as follows:

$$J_c^{(\theta)}(k) \equiv \frac{1}{2\bar{\sigma}_k^2} \left[ \log \left( \frac{c + \frac{B^{(\theta)}}{k}}{E[S^{(\theta)}]} \right) + \frac{\bar{\sigma}_k^2}{2} \right]^2 + \log(2\epsilon_p^{(\theta)}) \quad (22)$$

where

$$\bar{\sigma}_k^2 = \log \left( \frac{\bar{\sigma}_k^2}{E[S^{(\theta)}]} + 1 \right).$$

Finally, the effective bandwidth is obtained as

$$W_{eff}^{(\theta)} = \arg \min_c \left\{ \min_k J_c^{(\theta)}(k) \geq 0 \right\}. \quad (23)$$

4) *Multifractal Scaling*: Recent measurement and simulation studies have revealed that wide area network traffic and VBR video traffic display complex statistical characteristics—possibly multifractal scaling—on fine timescales, in addition to the well-known property of self-similar scaling on coarser timescales in the context of LRD [10]. In [9], it is shown that unreasonable overestimation of the equivalent bandwidth of a flow can be made if scaling at small scales are not considered in the modeling process, e.g., fractional Brownian motion. If the multitime scaling property is essential, the video traffic can be modeled as a combination of fractional Brownian motion and multifractal cascade to account for the small-scale effect corresponding to subframe scale using the method described in [10]. In the above three modeling cases, we use the fixed time scale of a GOP time and the traffic is characterized by the stochastic process  $\{S_i^{(\theta)}\}$ . If we consider a time scale  $t$  larger than a frame time  $t_f = 1/f$ , then the traffic is characterized by FBM as in (19) and the  $q$ th moment of the  $A_{FBM}^{(\theta)}(t)$  is given by

$$E \left[ \left( A_{FBM}^{(\theta)}(t) \right)^q \right] = \sum_{k=0}^{\lfloor \frac{q}{2} \rfloor} \binom{q}{2k} (mt)^{q-2k} (amt^{2H})^k (2k-1)!! \quad (24)$$

where  $m = E[S^{(\theta)}]$  and  $a = v^{(\theta)}[0]$ , and  $(\cdot)!!$  means the double factorial. If we consider a time scale  $t$  less than  $t_f = 1/f$ , then the traffic is modeled as multifractal cascade (MFC) and the  $q$ th moment of the counting process  $A_{MFC}^{(\theta)}(t)$  is modeled as

$$E \left[ \left( A_{MFC}^{(\theta)}(t) \right)^q \right] = E \left[ \left( A_{FBM}^{(\theta)}(t_f) \right)^q \right] \left( \frac{t}{t_f} \right)^{1-\tau(q)} \quad (25)$$

where

$$\tau(q) = \frac{\log(p^q + (1-p)^q)}{\log 2}$$

where we have  $p = 0.6$  [10]. We next construct an objective function

$$J_c^{(\theta)} \equiv \inf_{t>0} \sup_s \left[ s \left( ct + B^{(\theta)} \right) - \Lambda^{(\theta)}(s, t) \right] + \log(\epsilon_p^{(\theta)}) \quad (26)$$

where

$$\Lambda^{(\theta)}(s, t) = \log \left[ \sum_{q=0}^{\infty} \frac{s^q E[A^q(t)]}{q!} \right].$$

In the preceding equation,  $A(t)$  is substituted by  $A_{FBM}^{(\theta)}(t)$  or  $A_{MFC}^{(\theta)}(t)$  according to the time scale. Finally, the effective bandwidth is determined by

$$W_{eff}^{(\theta)} = \arg \min_c \left\{ \min_k J_c^{(\theta)}(k) \geq 0 \right\}. \quad (27)$$

### B. Packet Scheduling

Suppose we have a target GOP decoding failure probability,  $\epsilon_{GOP}^{(\theta)} = \gamma$ . Given  $C$ , the server determines the optimum threshold of importance level  $\theta_o$  such that

$$\theta_o = \arg \max_{\theta} \left\{ W_{eff}^{(\theta)}; W_{eff}^{(\theta)} < C, B^{(\theta)} < B_{max} E[S^{(\theta)}] \right\}. \quad (28)$$

where  $B_{max}$  in second denotes the maximum buffering time of the receiver, for example, it is optionally set at less than 60 s in *Windows Media Player*. Thus the optimum importance level is basically the largest  $\theta$  such that the GOP decoding failure probability of substream  $\Gamma^{(\theta)}$  is less than the requirement  $\gamma$  and the queue length in the queueing model is less than the given maximum buffering time. If it does not exist, then we find the smallest  $\theta$  holding only the queue length condition.

Since  $\theta_o$  is constrained to be an integer, in an ideal scenario, the server must alternate between  $\theta_o$  and  $\theta_o + 1$  in order to regulate the receiver buffer, and to maximize utilization of the channel bandwidth. Specifically, the relative time spent at  $\theta_o$  versus  $\theta_o + 1$  has to do with receiver buffer fullness; when receiver buffer is full  $\theta_o + 1$  is chosen, whereas when it is empty  $\theta_o$  is chosen. To see the reason for this, consider the following example. Suppose channel throughput is 100 Kbps, average video data rate is 100 Kbps, there are 15 frames in a GOP, and frame rate is 15 frames/s. Assume the high priority data is composed of only I frames and the server sends all I frames first, whose average data rate is assumed to be 10 Kbps compared with total video rate of 100 Kbps. The decoder does not display 15 I frames within 1 s, but only one I frame in a second. During the displaying time for P frames, the sender will keep sending all I frames, whereas the receiver will keep showing the same I frame, resulting in no exhaustion of the receiver buffer until the displaying time for extended I frame.

With  $\theta_o$ , high-priority data part with importance levels from  $h = 0$  through  $h = \theta_o$  is scheduled to be sent first according to EDF. In Fig. 4, we show importance levels  $h = 0, 1, 2, 3$  to illustrate scheduling examples for  $\theta_o = 1$  and 2, where the original display order is shown on top. Assume  $h = 0$  corresponds to I frames,  $h = 1$  corresponds to  $P_1$  frames, and so on. At time  $t = t_1$ ,  $\theta_o = 1$  and thus all frames with importance levels 0 and 1 are scheduled to be sent before all the frames with importance levels 2 and 3. Suppose that the server has sent the first three frames, i.e., frames with  $h = 0$  and 1 in solid squares and 0 in a solid circle, up until  $t = t_1 + T$  at which time the threshold  $\theta_o$  is updated to 2. Then all frames with  $h = 0, 1$ , and 2 form high priority part, and frames with  $h = 3$  form low priority part, as seen in Fig. 4. The server now continues with sending the third frame in the new schedule, with  $h = 2$  in a solid square, given the first two frames labeled 0 and 1 are already sent. In this example,

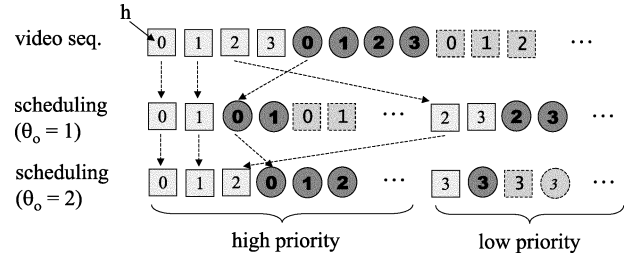


Fig. 4. Scheduling example.

compared with original video sequence, the resulting sending order, so far, is  $[h = 0, 1, 0, 2]$  instead of the original display order  $[h = 0, 1, 2, 3, 0, \dots]$ , i.e., the I frame in the second GOP was sent before the  $P_2$  in the first GOP.

### C. Channel Throughput

So far, we have assumed a given channel throughput,  $C$ . However, to make the scheduling algorithm be adaptive to time varying channel fluctuations, the server needs to periodically update channel throughput. Data sending rate at the server is equal to the average video rate. The receiver measures the channel throughput every  $T$  seconds; define a random variable  $Y_m$  to be the throughput measured at time  $t = mT$ . The scenes within a video usually have correlations of order of several seconds, whereas widely used on-off error models for networks have average on and off durations of several tens or hundreds milliseconds. Hence, compared to large autocorrelation of the video data,  $Y_m$  is assumed to be i.i.d. The receiver reports the measured throughput  $Y_m$  to the sender, which then updates the average channel throughput as an exponentially weighted moving average (EWMA) at time  $t = mT$  as follows:

$$C \leftarrow \alpha C + (1 - \alpha) Y_m. \quad (29)$$

### D. Media Streaming Protocol

We assume our proposed scheduling scheme resides between the application layer, i.e., coder/decoder, and the transport layer in the protocol stack. For each received packet, the receiver sends acknowledgment. The sender then updates the packet status by dividing eligible packets into acknowledged, outstanding, and not-sent packets; eligible packets mean future packets under consideration for transmission, excluding all packets with passed deadline and dropped packets due to transmission failure of preceding frames within a GOP. Since we assume no contraction and expansion of original display duration of clips, the server is able to calculate the deadline of each frame from current time and given frame rate.

The server needs to measure the statistics of the Round Trip Time (RTT) of the channel in order to determine the eligibility of a packet in terms of its deadline. Let  $RTT_{max}$  denote the maximum RTT over time. Considering possible asymmetric traffic loads between forward and backward links, the forward trip time, i.e., from sender to receiver, is assumed to be upper bounded by  $RTT_{max}$ . Thus, the sender uses  $RTT_{max}$  to estimate whether a packet arrives at the receiver before its

TABLE I  
INFORMATION ON VIDEO DATA

video	quantiser step	length [frames]	frame rate [frames/sec]	data type	STATISTICS		
					avg. size [Kbytes]	variance	peak-to-mean ratio
Talk show (Larry King's Live) QCIF	Qp = 5	1689	15	I	4.40	1.03	1.37
				P*	0.83	0.33	7.44
				motion**	0.078	0.002	7.15
				texture**	0.75	0.28	7.56
Movie trailer (Preacher's Wife)	Qp = 5	1871	15	I	3.29	1.71	2.21
				P*	1.29	1.12	5.87
				motion**	0.15	0.014	4.18
				texture**	1.13	0.92	6.37
Animation (A Bug's Life) 384 × 240	Qp = 10	1611	24	I	15.2	18.5	1.50
				P*	3.62	6.11	6.81
				motion**	0.017	3e-5	2.20
				texture**	3.60	6.12	6.84
Music video (Elton John) 320 × 240	Qp = 5	860	15	I	14.4	75.0	1.93
				P*	6.12	15.3	3.50
				motion**	0.55	0.04	2.33
				texture**	5.57	15.1	3.72

\* There are 15 P frames per GOP. \*\* Motion and texture refer to P frames.

deadline, if it were to be sent.<sup>4</sup> Once a packet is sent, the packet is referred to be outstanding, and a timer is set to go off after  $RTT_{max}$  seconds. If the acknowledgment arrives before the timer goes off, then it is referred to as an acknowledged packet. If the timer goes off without acknowledgment, then the packet becomes a not-sent packet to be retransmitted. Note that the sender does not distinguish between retransmission packets and first-time transmission ones.

#### IV. PERFORMANCE EVALUATION

Our proposed scheduling method is evaluated by sending real video data in Table I over wireline and wireless channels. We use Microsoft MPEG-4 Visual Reference Software version 2 FDMA1-2.3-001213, to encode/decode video data. We have added error recovery/concealment features related to MTD and temporal scalability. Our proposed scheduling algorithm is implemented on top of the transport layer using UDP. For each arriving packet of 1500 Bytes, the receiver sends 3 bytes of acknowledgment including time stamp and measured throughput.

Our proposed scheduling scheme is referred to as Effective Bandwidth based Scheduling (EBS) in the remainder of this paper, and is compared with a conventional sequential sending (SS) based on EDF scheme without any optimization. Initial preroll buffering is 10-s long, i.e., the receiver begins display 10 s after it received the first packet. For our optimization scheme, we use,  $\gamma = 10^{-4}$ , as the decoding failure probability requirement for high priority part. According to (29), the average channel throughput is measured every 1 s and updated according to "exponentially weighted moving average" (EWMA) with a design parameter  $\alpha$ , which is determined by taking into account the channel characteristics. For  $\alpha = 0.9$ , the newly updated channel throughput is calculated as  $(0.9) \times (\text{last updated throughput}) + (0.1) \times (\text{currently measured throughput})$ . We have  $B_{max} = 30$  s.

<sup>4</sup>We use  $RTT_{max}$  as the criterion in WLAN case due to asymmetric loads between forward and backward links, and a half of  $RTT_{max}$  in wireline case where the forward and backward trip time is assumed to be symmetrical.

First, we transmit talk show video over the wireline Internet. The average round trip time was measured to be around 250 ms. We use talk show video with average video bit rate of 125.5 Kbps. Since the real channel bandwidth of the wireline Internet is measured to be far larger than the video bit rate, we have implemented ON/OFF type artificial error in the receiver. The duration of each state is exponentially distributed; the average duration of bad state is 50 ms; the percentage of the time bad state is on varies in the range between 0% and 33%. The parameters in this range correspond to average channel throughput of 100% to 30% of the average video bit rate, 125.5 Kbps.

In Fig. 5, we illustrate the distortion curves for conventional sequential sending (SS) and the proposed EBS with various traffic models using motion-texture discrimination (MTD), as a function of channel throughput. To begin with, the plane solid curve corresponding to SS shows rapid video quality degradation as the throughput decreases. All the other curves are from EBS with various video traffic models. The curve from SRD model is marked with legend "EBS(SRD model)". Scheduling schemes with SRD model and FARIMA show the best performance. Fractional Brownian motion (FBM) gives the worst performance among the EBS scheduling results. However, multifractal model shows good performance with very small difference compared with SRD model and FARIMA, although it is computationally complex.

The results in Fig. 5 are explained as follows. Video traffic is observed to have LRD and even multifractal properties. Specifically, it has been observed that many real encoded video traces have SRD property for correlation among frames up to several minutes apart, and LRD, among frames apart longer than that. In that context, SRD models can be successfully applied to VBR video traffic for communication systems with buffer size of practical interest [8]. In some situations, the empirical autocovariance function (ACF) of video is better captured by ACF of a form of SRD rather than LRD [5]. There is no doubt that LRD model and multifractal model can be highly valuable when analyzing the performance of modern high-speed routers (layer 3) with aggregate traffic. However, for end-to-end video streaming applications above the transport layer (layer 4), LRD

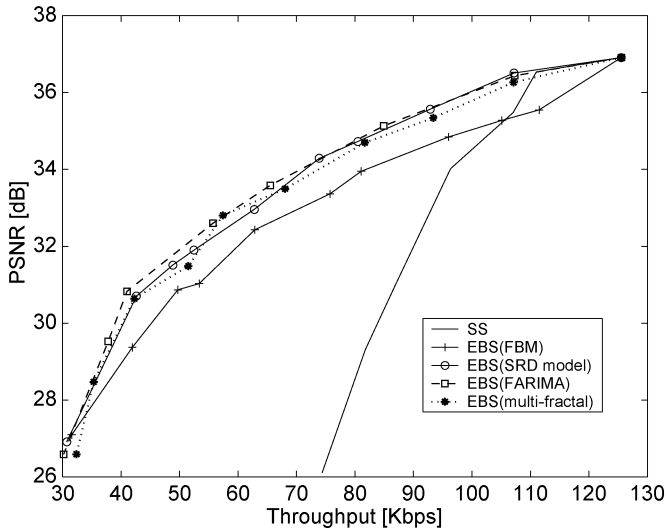


Fig. 5. Rate-distortion curves with various schemes.

property generally has little or no impact on visual quality from a practical viewpoint. For example, *Windows Media Player* has buffering time of around 10 to 20 s with a maximum of 60 s. Since our scheme runs at the application layer, i.e., above UDP, and receiver buffering time is set to be 30 s in the experiments, our proposed scheduling scheme is insensitive to LRD properties; this is seen as little difference in performance between SRD model and FARIMA (SRD + LRD) model in Fig. 5. The relatively poor result from FBM is due to overestimation of the equivalent bandwidth. This problem is overcome by multifractal model, as expected in [9], where it is shown that the equivalent bandwidth of a flow is overestimated if scaling at small scales is not considered in the modeling process. However, the multifractal model is computationally complex and results in similar performance as SRC and FARIMA models. This can be explained considering that multifractal model deals with fine time scale at subframe level, whereas in our framework, we are primarily concerned with intactness of an entire frame; specifically, in our framework, a frame is displayed only if at least its motion part is received in its entirety.

We also compare the performance of EBS with “Expected Run-time Distortion Based Scheduling” (ERDBS) from [3], where ERDBS is shown to improve visual quality by 2 dB over SS scheme at packet loss rate of 20% or higher, which translates to 100 Kbps or lower in the experiment in Fig. 5. Comparing EBS(FARIMA) with ERDBS, we observe that our scheduling scheme outperforms ERDBS by 1.2 dB at throughput of 90 Kbps (25% packet loss) and by 2 dB at throughput of 80 Kbps (35% packet loss).

We next show that our proposed scheduling scheme further improves video quality by incorporating temporal scalability (TS). Fig. 6 exhibits rate-distortion curves from SS, EBS(FARIMA) with MTD, EBS(FARIMA) with MTD + TS. Two-layer temporal scalability is observed to increase the encoded video size by 10%. Therefore, at channel throughput of 125.5 Kbps, MTD + TS scheme shows degradation of about 0.5 dB as compared with MTD-only scheme. However, at channel throughput below 95 Kbps, MTD + TS scheme

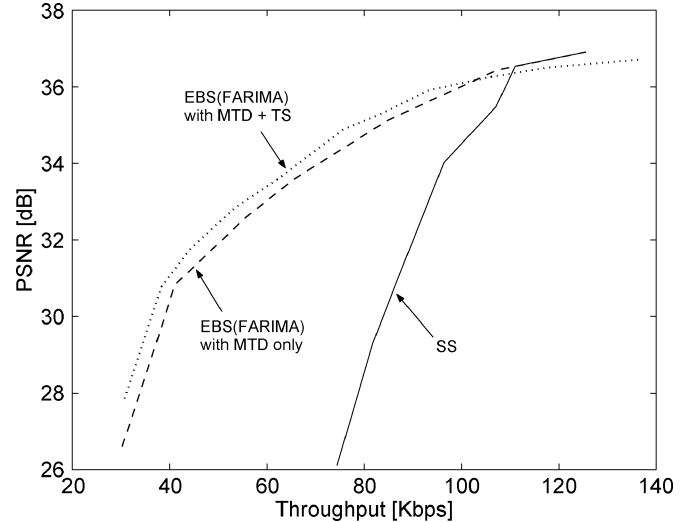


Fig. 6. Rate-distortion curves for two-layer temporal scalability.

outperforms MTD-only scheme by 0.5 dB, showing more graceful degradation.

We now consider wireless channels such as wireless LAN. In Fig. 7, we show PSNR performance for streaming videos over IEEE 802.11b wireless LAN. Referring to original video, PSNR of a frame in an encoded video is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} \text{ [dB]} \quad (30)$$

where MSE is mean squared error between the two corresponding frames in original and encoded videos. We evaluate scheduling schemes in terms of average PSNR over 5 s. Generating background cross traffic, the server opens two sessions with the client at the same time, one session with SS and the other with EBS(SRD model) using MTD; this way, channel congestion and/or interference have the same effect on both schemes. The amount of background traffic is such that the total packet loss rate to be 15%–30%. As seen in Fig. 7, the scheduling scheme shows significantly larger improvement over SS than in the previous wireline network case. The average PSNR is in parentheses. In each plot, video quality in the beginning is good due to initial pre-roll buffering. However, as time goes on, SS scheme shows degradation as the receiver buffer remains empty most of the time, whereas EBS shows consistently high video quality.

Finally, we run the streaming using Microsoft *Windows Media 9* under the same condition as in Fig. 7(c) with the animation video. To be fair, we choose the streaming protocol as UDP in the *Windows Media* menu option. With five minutes of animation video, we observe more than 200 freezes; each time the display freezes for about 1 to 4 s, and the total display time is expanded to over 6 min. On the other hand, with the assumption of no expansion of display time, our proposed EBS never freezes.

## V. CONCLUSION

We have developed a class of rate distortion optimized packet scheduling algorithms, taking into account both video bit rate



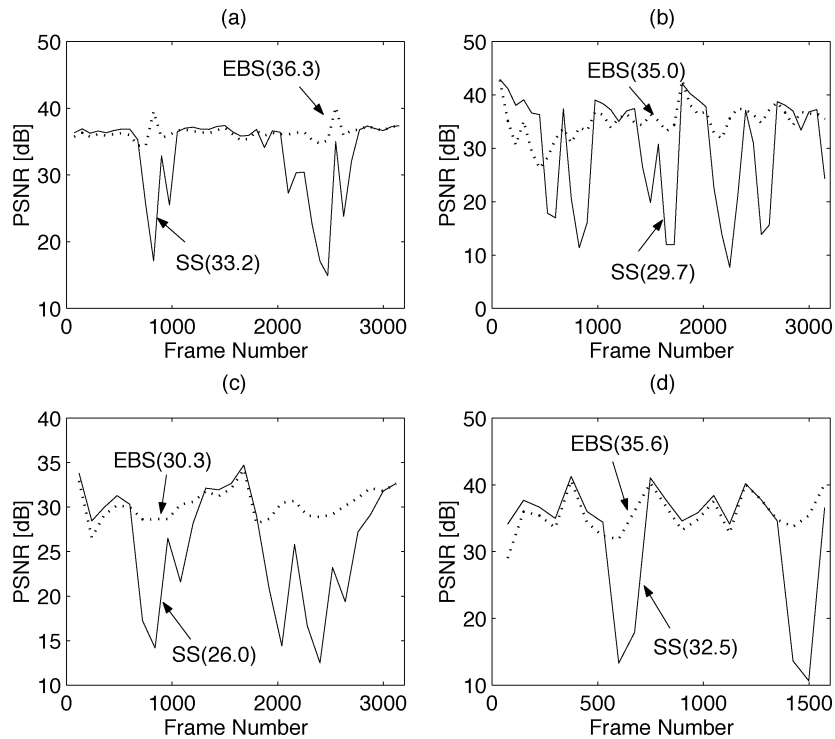


Fig. 7. Performance on wireless LAN. (a) Talk show; (b) movie trailer; (c) animation; (d) music video.

fluctuations as well as channel fluctuations. In doing so, we take advantage of temporal scalability and MTD in order to arrive at a set of nested substreams whereby the more important substreams embed the less important ones. We apply the notion of effective bandwidth from queueing theory to determine the optimum substream to send at any moment in time. We have shown the effectiveness of our approach over sequential sending scheme for actual video sequences over both the wireline Internet and WLAN. Future work involves optimization over combined networks from wireline and wireless links, optimization with VCR functionality, and applying EBS to multicast networks.

#### APPENDIX

In this appendix, we derive an approximation for the GOP decoding failure probability  $\epsilon_{GOP}^{(\theta)}$  when the packet loss probability for substream  $\Gamma^{(\theta)}$  is given by  $\epsilon_p^{(\theta)}$  in the queueing model. Recall that we define a successful GOP decoding as the availability of all motion packets by their deadline. Referring to [16], the motion data is related to importance levels from  $h = 0$  to  $h = N - 1$  in all layering schemes under consideration.

Suppose that the server transmits data stream  $\Gamma^{(\theta)}$  with  $\theta \leq N - 1$  over a channel with packet loss probability  $\epsilon_p^{(\theta)}$ . For  $\theta \leq N - 1$ , then every packet in  $\Gamma^{(\theta)}$  includes motion part, and thus, a loss of any packet will cause GOP decoding failure. Since the average GOP size is  $E[S^{(\theta)}]$  in packets, the GOP decoding failure probability is given by

$$\epsilon_{GOP}^{(\theta)} = 1 - \left(1 - \epsilon_p^{(\theta)}\right)^{E[S^{(\theta)}]}. \quad (31)$$

With MTD, importance levels  $h > N - 1$  correspond to texture parts. When the sender transmits stream  $\Gamma^{(\theta)}$  with  $\theta > N - 1$

over a channel with packet loss probability  $\epsilon_p^{(\theta)}$ , then among the packets in  $\Gamma^{(\theta)}$  only the packets with importance levels  $h = 0$  through  $h = N - 1$  will affect the GOP decoding failure. The average size of data with motion part in a GOP is then  $E[S^{(N-1)}]$ , and thus the GOP decoding failure probability is given by

$$\epsilon_{GOP}^{(\theta)} = 1 - \left(1 - \epsilon_p^{(\theta)}\right)^{E[S^{(N-1)}]}. \quad (32)$$

The above obtained decoding failure probability of a GOP, given substream  $\Gamma^{(\theta)}$ , is compared with a requirement in order to find an optimum  $\theta$  in our scheduling algorithm.

#### REFERENCES

- [1] H. Liu and M. E. Zarki, "Performance of H.263 video transmission over wireless channels using hybrid ARQ," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 9, pp. 1775–1786, Dec. 1997.
- [2] Q. Zhang and S. A. Kassam, "Hybrid ARQ with selective combining for fading channels," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 5, pp. 867–880, May 1999.
- [3] Z. Miao and A. Ortega, "Expected run-time distortion based scheduling for delivery of scalable media," in *Int. Packetvideo Workshop 2002*, Apr. 2002.
- [4] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 1566–1579, Feb./Mar./Apr. 1995.
- [5] M. M. Krunz and A. M. Makowski, "Modeling video traffic using M/G/ $\infty$  input processes: a compromise between Markovian and LRD models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 5, pp. 733–748, Jun. 1998.
- [6] K. Nagarajan and G. T. Zhou, "Self-similar traffic sources: modeling and real-time resources allocation," in *Statist. Sig. Process.*, 2001, pp. 74–76.
- [7] —, "A new resource allocation scheme for VBR video traffic source," in *Proc. 34th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 2000, pp. 1245–1249.
- [8] S. H. Kang, Y. H. Kim, D. K. Sung, and B. D. Choi, "An application of Markovian arrival process (MAP) to modeling superposed ATM cell streams," *IEEE Trans. Commun.*, vol. 50, no. 4, pp. 633–642, Apr. 2000.

- [9] C. A. V. Melo and N. L. S. Fonseca, "An envelope process for multifractal traffic modeling," in *Proc. IEEE Int Conf. Communication (ICC) 2004*, vol. 4, June 20–24, 2004, pp. 2168–2173.
- [10] A. Erramilli, O. Narayan, A. Neidhardt, and I. Sanjeev, "Multi-scaling models of TCP/IP and sub-frame VBR video traffic," *J. Commun. Networks*, vol. 3, no. 4, pp. 383–395, Dec. 2001.
- [11] J. Gao and I. Rubin, "Multiplicative multifractal modeling of long-range-dependent network traffic," *Int. J. Commun. Syst.*, vol. 14, pp. 783–801, 2001.
- [12] N. L. S. Fonseca, G. S. Mayor, and C. A. V. Neto, "On the equivalent bandwidth of self similar sources," *ACM Trans. Model. Comput. Simul.*, vol. 10, no. 3, pp. 104–124, 2000.
- [13] C. Courcoubetis and R. Weber, "Effective bandwidths for stationary sources," *Prob. Eng. Inf. Sci.*, vol. 9, pp. 285–296, 1995.
- [14] I. Norros, "On the use of fractional Brownian motion in the theory of connectionless networks," *IEEE J. Sel. Areas. Commun.*, vol. 13, no. 6, pp. 953–962, Aug. 1995.
- [15] ———, "Effective bandwidth based scheduling for streaming multimedia," in *Proc. IEEE ICIP 2003*, vol. 3, Barcelona, Spain, Sept. 14–17, 2003, pp. 633–636.
- [17] M. Parulekar and A. M. Makowski, " $M/G/\infty$  input process: a versatile class of models for network traffic," in *Proc. IEEE INFOCOM'97*, 1997, pp. 419–426.



**Sang H. Kang** (S'92–M'98) received the B.S., M.S. and Ph.D. degrees from the Korea Advanced Institute of Science and Technology in 1990, 1992, and 1997, respectively, all in electrical engineering.

In 1997, he joined the faculty of the Department of Electrical Engineering, University of Seoul, Korea, where he is currently Associate Professor. From January 2001 to December 2002, he was a Visiting Research Engineer with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, where he worked in the areas of multimedia streaming. He is Editor of the *Journal of Communications and Networks (JCN)*, in the division of Networks and Services. His research interests include high-speed Internet, wireless/mobile networking, multimedia streaming and digital broadcasting.

Dr. Kang is a member of the KICS.



**Avidesh Zakhor** (S'87–M'87–SM'91–F'01) received the B.S. degree from the California Institute of Technology, Pasadena, and the M.S. and Ph.D. degrees from Massachusetts Institute of Technology, Cambridge, all in electrical engineering, in 1983, 1985, and 1987 respectively.

In 1988, she joined the Faculty at the University of California, Berkeley, where she is currently Professor in the Department of Electrical Engineering and Computer Sciences. Her research interests are in the general area of image and video processing, multimedia communication, and 3-D modeling. She holds five U.S. patents and is the co-author, with Soren Hein, of the book *Sigma Delta Modulators: Nonlinear Decoding Algorithms and Stability* (Norwell, MA: Kluwer, 1993).

Prof. Zakhor, together with her students, has won a number of best paper awards, including from the IEEE Signal Processing Society in 1997, the IEEE Circuits and Systems Society in 1997 and 1999, International Conference on Image Processing in 1999, and Packet Video Workshop in 2002. She was a General Motors scholar from 1982 to 1983, was a Hertz fellow from 1984 to 1988, received the Presidential Young Investigators (PVI) award, and Office of Naval Research (ONR) young investigator award in 1992. From 1998 to 2001, she was an elected member of IEEE Signal Processing Board of Governors.