Implications of Smoothing on Statistical Multiplexing of H.264/AVC and SVC Video Streams

Geert Van der Auwera and Martin Reisslein

Abstract—While the hierarchical B frames based Scalable Video Coding (SVC) extension of the H.264/AVC standard achieves significantly improved compression over the initial H.264/AVC codec, the SVC video traffic is significantly more variable than H.264/AVC traffic. The higher traffic variability of the SVC encoder can lead to smaller numbers of streams supported with bufferless statistical multiplexing than with the H.264/AVC encoder (and even less streams than with the MPEG-4 Part 2 encoder) for prescribed link capacities and loss constraints. In this paper we examine the implications of video traffic smoothing on the numbers of statistically multiplexed H.264 SVC, H.264/AVC, and MPEG-4 Part 2 streams, the bandwidth requirements for streaming, and the introduced delay. We identify the levels of smoothing that ensure that more H.264 SVC streams than H.264/AVC streams can be supported. For a basic low-complexity smoothing technique that is readily applicable to both live and prerecorded streams, we identify the levels of smoothing that give (bufferless) statistical multiplexing performance close to an optimal off-line smoothing technique. We thus characterize the trade-offs between increased smoothing delay and increased statistical multiplexing performance for both H.264/AVC, which employs classical B frames, and H.264 SVC, which employs hierarchical B frames. We similarly identify the buffer sizes for the buffered multiplexing of unsmoothed H.264 SVC, H.264/AVC, and MPEG-4 Part 2 streams that give close to optimal performance.

Index Terms—Delay, H.264/AVC, hierarchical B frames, smoothing, statistical multiplexing, SVC, video traffic.

I. INTRODUCTION

THE recently standardized Scalable Video Coding extension (SVC) of the H.264/AVC standard [1]–[3] with its hierarchical B-frames compresses single-layer (non-scalable) video significantly more efficiently than the underlying H.264/MPEG-4 Advanced Video Coding standard [4] (H.264/AVC for brevity), which is also known as H.264/MPEG-4 Part 10. H.264/AVC in turn compresses video significantly more efficiently than MPEG-4 Part 2 (typically only half the average bit rate with H.264/AVC for same video quality). H.264/AVC and H.264 SVC video encoding are expected to be widely adopted for wired and wireless network

video transport due to their increased compression efficiency compared to MPEG-4 Part 2 and their widespread inclusion in application standards and industry consortia specifications, e.g., DVB, 3GPP2, and MediaFLO.

The compression efficiency of a video codec is generally characterized with a so-called rate-distortion (RD) curve that shows the bit rate of the compressed video stream as a function of the video quality (distortion), which is typically measured in terms of the Peak Signal to Noise Ratio (PSNR). For a given video quality, the lower the compressed bit rate, the more efficient is the compression. The improvements in rate-distortion (RD) compression efficiency with H.264 SVC and H.264/AVC come at the expense of significantly increased variabilities of the encoded frame sizes (in bits) [5]. Highly variable video frame sizes, i.e., highly variable video traffic, generally poses a challenge for efficient network transport [6]–[8]. When the video frame sizes are highly variable, i.e., when the largest frames are much larger than the average frame size, then provisioning network bandwidth according to the largest frames results in inefficient bandwidth usage. The basic idea of statistical multiplexing is that the largest frames of some video streams collude with average (or smaller than average sized) frames of other streams during network transport. With this statistical multiplexing, the bandwidth requirement is typically dramatically less than the sum of the peak bit rates of the supported streams, and may approach the sum of the mean bit rates of the supported streams. Consequently, statistical multiplexing is of great interest for network systems transporting video with variable frame sizes.

However, it was found in [9] that the H.264/AVC encoder can outperform the H.264 SVC encoder and that even the MPEG-4 Part 2 encoder can outperform both the H.264/AVC and H.264 SVC encoders when multiplexing a small number of video streams in an elementary bufferless statistical multiplexing setting. This is due to significantly higher traffic variabilities of H.264 SVC encoded video streams compared to H.264/AVC encoded streams, as well as the significantly higher traffic variabilities of both H.264 SVC and H.264/AVC encoded video streams compared to MPEG-4 Part 2 encoded streams. The higher traffic variabilities can compensate the lower average bit rates achieved with H.264 SVC encoding compared to H.264/AVC encoding, as well as the lower average bit rates achieved by both H.264 SVC and H.264/AVC compared to MPEG-4 Part 2.

In this paper we examine the effectiveness of two elementary techniques for mitigating high traffic variability, namely (*i*) video traffic smoothing, i.e., the averaging of several successive frame sizes before sending them into the bufferless multiplexer, and (*ii*) buffered multiplexing of unsmoothed video streams.

From the wide spectrum of video traffic smoothing techniques we consider two extreme approaches: optimal smoothing [10], [11], which minimizes the traffic variabilities, and basic smoothing, which simply averages (aggregates) the sizes of a prescribed number of successive video frames, whereby the number of averaged video frames is denoted by the aggregation level a. Optimal smoothing achieves the minimal traffic variability subject to given smoothing (receiver) buffer and start-up delays by computing offline the transmission schedule that delivers each video frame by its playout deadline while avoiding overflows of the smoothing buffer and minimizing transmission rate changes. Optimal smoothing has a computational complexity of $O(M)$, whereby M denotes the number of frames in the sequence and can not be directly applied to live streams. In contrast, basic smoothing is computationally very simple (has complexity $O(1)$) and can directly be applied to live streams. For a range of numbers of statistically multiplexed streams and video (texture/motion) complexities, we provide guidelines for (i) setting the aggregation levels a of basic smoothing that ensure that more H.264 SVC streams than H.264/AVC streams are supported, and (ii) setting the aggregation levels a that provide similar statistical multiplexing performance with basic smoothing as with optimal smoothing. We find that generally SVC requires larger aggregation levels a to overcome its higher traffic variabilities. We also examine the delay introduced by the hierarchical B frame predictions in H.264 SVC in conjunction with the aggregation levels for the traffic smoothing and compare with the corresponding delays for H.264/AVC.

We also examine elementary taildrop buffered statistical multiplexing of unsmoothed video streams. We identify the multiplexer buffer sizes required to support close to the maximum number of streams (given by the link capacity divided by the average stream bit rate). We find that H.264 SVC streams require roughly twice the buffer size of H.264/AVC streams, while in turn H.264/AVC streams require approximately twice the buffer size of MPEG-4 Part 2 streams.

This paper is structured as follows. In Section II, we review related work. In Section III, we present our evaluation set-up, including the examined H.264 SVC, H.264/AVC, and MPEG-4 Part 2 encoders and their settings, as well as the video sequences used for the evaluations. In Section IV, we first describe the employed basic and optimal smoothing techniques and the considered bufferless statistical multiplexing setting. We then present simulation results for optimal smoothing, followed by simulation results for basic smoothing. In Section V, we first describe the examined elementary buffered statistical multiplexing scenario, and then present simulation results. We summarize our conclusions in Section VI and analyze the delays for smoothed transmission of video encoded with classical and hierarchical B frames in the Appendix.

II. RELATED WORK

For MPEG-4 Part 2, H.263, and preceding codecs, the bit rate-distortion characteristics and rate variability characteristics have been extensively studied, see for instance [12]–[14] and references therein. Similarly, the video traffic of these codecs has been extensively studied, see for instance [15]–[19], and

they have been used as a basis for the existing studies on video traffic smoothing, as reviewed in Section IV-A, and buffer management, as reviewed in Section V.

The bit rate-distortion characteristics of H.264/AVC and H.264 SVC have been examined in a few studies [3], [4], [20] and the rate variability characteristics of H.264/AVC and H.264 SVC have been investigated in [5], [9], [21]. The study of network transport mechanisms for H.264/AVC and H.264 SVC has just begun to attract interest, see for instance the studies [22]–[27], all of which are complementary to our study examining the fundamental statistical multiplexing characteristics.

We note that the traffic characteristics of individual smoothed H.264/AVC and H.264 SVC streams have been studied in [9]; furthermore, the bufferless statistical multiplexing of unsmoothed H.264/AVC and H.264 SVC streams has been examined in [9]. To the best of our knowledge, the fundamental bufferless statistical multiplexing characteristics of smoothed H.264/AVC and H.264 SVC video and buffered multiplexing characteristics of unsmoothed H.264/AVC and H.264 SVC video are for the first time examined in this paper.

III. EVALUATION SET-UP

A. Video Encoding Set-Up

We employ the H.264/AVC encoder [4], [20], [28]–[30] in the Main profile with all compression tools enabled, including spatial intra frame prediction, variable block sizes, three reference frames for the past and the future, referenced B frames, P and B frame weighted prediction, Context Adaptive Binary Arithmetic Coding (CABAC), and Lagrangian based rate-distortion optimization (RDO). In particular, we employ the JM reference software (version 10.2), which is the official MPEG and ITU reference implementation for the H.264/AVC Main profile. For the H.264 SVC encodings, we used the SVC reference software named JSVM (version 5.9), and similar settings as for H.264/AVC.

Throughout, we employ H.264/AVC with classical B frame prediction, where a B frame is predicted only from the preceding I or P frame and from the subsequent I or P frame; other B frames are not referenced. In contrast, H.264 SVC [1]–[3] employs the hierarchical B frame structure which uses B frames for the prediction of B frames, as illustrated in the Appendix. More specifically, with the employed dyadic B frame hierarchy, the number of B frames β between successive key pictures (I or P frames) is

$$\beta = 2^k - 1 \quad \text{for an integer number } k, \ k \geq 0, \quad (1)$$

of so-called temporal layers of B frames.

We use the MPEG-4 Part 2 encoder [31], specifically the MPEG-4 Part 2 Microsoft v2.3.0 software, in the Advanced Simple profile (ASP), which includes B frames. We employ half pixel motion compensated prediction; RDO is not supported by the reference encoder implementation. The MPEG-4 Part 2 encoder uses one reference frame for the past and one for the future, and 16×16 blocks for motion estimation that can be split into 8×8 blocks.

For the H.264/AVC encodings and the MPEG-4 Part 2 encodings, which are both based on classical B frames, we employ GoP structure *IBBBPBBBPBBBPBBB* (16 frames, with 3 B frames per I/P frame) denoted by *G16-B3*. For the H.264 SVC encodings (hierarchical B frames), we employ GoP structure *IBBBBBBBBBBBBBBB* (16 frames, with 15 B frames per I frame) denoted by *G16-B15*. The statistical video traffic analysis in [5], [9] demonstrated that these encoding parameter settings and GoP structures result overall in very good rate-distortion (RD) efficiencies for the respective encoders. The analysis in [5] also indicated that encoding parameter settings that result in lower RD efficiency generally reduce the traffic variability; conversely, settings that further increase the RD efficiency generally increase the traffic variability (further increasing the need for traffic smoothing). In addition, the considered GoP structures provide identical random access functionalities (I frame period). We consider quantization parameters that correspond to the range of average PSNR qualities from either 30/32 dB (acceptable quality) or 35 dB (good quality) to at least 40 dB (high quality).

Throughout this study, we consider single-layer (non-scalable) encoding and encode the video with fixed quantization scales, which results in nearly constant video quality and variable video traffic bit rates. By considering variable bit rate encoding without the use of rate control mechanisms we are able to examine the fundamental traffic characteristics of the H.264 SVC and H.264/AVC video coding standards, which do not specify a normative rate control mechanism. An additional motivation for the focus on variable bit rate video encoded with fixed quantization scales is that the variable bit rate streams allow for statistical multiplexing gains that have the potential to improve the efficiency of video transport over communication networks [6].

B. Video Sequences

The five CIF (352 × 288 pixels) resolution video sequences employed in the statistical multiplexing simulations presented in this study are the ten minute *Sony Digital Video Camera Recorder* demo sequence (17,682 frames at 30 frames/sec), which we refer to as *Sony Demo* sequence, the first half hour of the *Silence of the Lambs* movie (54,000 frames at 30 frames/sec), the first half hour of the *Star Wars IV* movie (54,000 frames at 30 frames/sec), and the first hour of the *Tokyo Olympics* video (133,128 frames at 30 frames/sec). We also use about 30 minutes of the *NBC 12 News* (49,523 frames at 30 frames/sec), including the commercials. These sequences were obtained with the MEncoder tool through decoding the original DVD sequences into the uncompressed YUV format and subsampling to CIF resolution. The video sequences *Silence of the Lambs*, *Star Wars IV*, *Tokyo Olympics*, and *NBC 12 News* can respectively be described as drama/thriller, science fiction/action, sports, and news. The *Sony Demo* sequence is documentary style, and is a mixture of detailed scenes (textures) and various motion activities. The *NBC 12 News* and *Sony Demo* videos have relatively higher motion and texture complexity than the other three videos and pose more

challenges for statistical multiplexing as we demonstrate in Section IV-C-1.

In order to facilitate further research on network transport of H.264 SVC, H.264/AVC, and MPEG-4 Part 2 encoded video, all encodings presented in this study are publicly available as video traces from the video trace library at: http://trace.eas.asu.edu. Frame size video traces [32] are files mainly containing video frame time stamps, frame types (e.g., I, P, or B), encoded frame sizes (in bits), and frame qualities (PSNR). Video traces are employed in simulation studies of the transport of video over communication networks, see e.g., [33]–[37], and as a basis for video traffic models, as for instance in [12], [15], [16], [19], [38]–[41]. Traffic modeling of H.264/AVC and H.264 SVC video traffic is a nascent research area, see e.g., [21], [42]–[44], and we directly employ the video traces for a realistic representation of H.264 video traffic in our simulations. Generally, advantages of using video traces over using regular encoded bit streams in simulations are the availability of a large number of traces of long and real video sequences, the fact that video traces are not copyrighted, and that only knowledge of basic concepts of video encoding are required.

IV. BUFFERLESS STATISTICAL MULTIPLEXING OF SMOOTHED VIDEO TRAFFIC

A. Frame Size Smoothing

A wide variety of frame size smoothing mechanisms have been developed and studied in the context of the MPEG-4 Part 2, H.263, and preceding video standards. Broadly, these smoothing mechanisms can be classified into non-collaborative mechanisms that smooth a single video stream, see for instance [10], [45]–[56], and collaborative mechanisms that jointly smooth several streams sharing networking resources, see for instance [33], [57]–[64]. We focus on non-collaborative smoothing in this study and leave evaluations of collaborative smoothing for H.264 SVC and H.264/AVC for future work.

Among the non-collaborative smoothing mechanisms, we first consider *basic smoothing* of the sizes (in bit) of the video frames over non-overlapping blocks of a frames each. More specifically, for the aggregation level a, the sizes of a consecutive frames are averaged and transmitted at the corresponding average bit rate. Given the original (unsmoothed) frame size sequence $X_m, m = 1, \ldots, M$, we obtain the smoothed frame sizes

$$Y_n = \frac{1}{a} \sum_{m=(n-1)a+1}^{na} X_m \qquad (2)$$

for $n = 1, \ldots, M/a$. The aggregation level a can be varied, with larger a values resulting in lower video traffic variabilities at the expense of increased delay, which is analyzed in the Appendix.

We also consider *optimal smoothing* [10], [11], which is optimal in the sense that it minimizes the bit rate variability and the peak bit rate of the video traffic subject to prescribed smoothing (receiver) buffer and start-up delays. Optimal smoothing ensures that the given receiver buffer does not underflow nor overflow, while sending video frame bits ahead of the decoding

times of the corresponding video frames. The optimization algorithm computes the transmission schedule of the video frame bits in piecewise constant bit rate segments that are as long as possible and have the smallest rate changes possible, without overflowing the client buffer, and while delivering the video frames by their playout deadlines. Optimal smoothing takes as input the frame sizes of the pre-encoded video stream and computes the transmission schedule off-line. With M denoting the number of video frames in a pre-encoded video sequence, the computational complexity of a basic implementation of optimal smoothing is $O(M^2)$ (whereby a complexity reduction to $O(M)$ is possible with a more involved implementation) [10]. For our simulations we set the client buffer size to 48 KB and set the (additional) start-up delay (see Appendix) to $s = 0$. The 48 KB buffer ensures that for the highest quality streams in our experiments (approximately 40dB), the largest frames can fit into the client buffer.

Although many more video traffic smoothing techniques are available, we focus on *basic smoothing* and *optimal smoothing*, because these two techniques represent extreme situations, i.e., lowest computational complexity ($O(1)$ with basic smoothing) and lowest achievable rate variability (with optimal smoothing).

B. Bufferless Statistical Multiplexing

In the real-time frame-based video streaming scenario based on a bufferless statistical multiplexer [56], [65]–[67], a channel with bandwidth capacity C [bit/s] connects a streaming video server with a bufferless statistical multiplexer to J receivers. Each video frame is transmitted during one frame period T (e.g., 33 ms for a frame rate of 30 frames/s). Let $X_m(j)$ [bit] denote the frame size of frame m, $m = 1, \ldots, M$, of stream j, $j = 1, \ldots, J$. Then, the bit rate required to transmit frame m of stream j during one frame period of length T is given by $X_m(j)/T$. Let $\theta_m(j)$ be a random variable denoting the index of the frame of stream j transmitted during frame period m. Then, the aggregated bit rate in frame period m when statistically multiplexing all J streams is given

$$R_m = \sum_{j=1}^{J} X_{\theta_m(j)}(j)/T. \tag{3}$$

If the aggregate bit rate R_m exceeds the link capacity C, then loss occurs, which we measure as the information loss probability [66], [67], i.e., long-run fraction of lost video bits:

$$P_{\text{loss}}^{\text{info}} = \frac{E\left[(R_m - C)^+ \times T\right]}{E[R_m \times T]} = \frac{E\left[(R_m - C)^+\right]}{E[R_m]}, \tag{4}$$

where $[x]^+ = \max(0, x)$. For a given experiment, we stream J identical video sequences, whereby the starting phase for each stream is randomly selected according to a uniform distribution over all M frames of the sequence [32], [66]. The streams are wrapped around to obtain streams of equal lengths.

Aside from providing an appropriate model for low-delay, low-buffer transmission systems [65], [67], bufferless statistical multiplexing provides a "ground truth" for studying the fundamental implications of the bit rate variabilities associated with

the H.264 SVC, H.264/AVC, and MPEG-4 Part 2 video encoders and with the video content. By considering the outlined elementary bufferless statistical multiplexing scenario, we avoid introducing confounding parameters, such as network buffers, cross traffic, and network topology. Only the video encoder (and its encoding settings), the video content, and the link capacity C (along with the number of streams J) influence the outcome of the experiment and we are thus able to uncover the fundamental statistical multiplexing characteristics of the smoothed H.264/AVC and H.264 SVC streams.

We note that predicting the loss probability of statistical multiplexing from statistical descriptors of the video traffic has been extensively studied for MPEG encoded videos and verified through simulations with traces of MPEG encoded videos, see e.g., [56], [65]–[70]. Generally, such prediction works relatively well when the number of multiplexed streams is high and the streams are relatively smooth. Predicting the loss probability when multiplexing few streams as well as for the new H.264/AVC and H.264 SVC encodings with their high variability is a largely open research area. In this study we conduct extensive simulations with traces of H.264/AVC and H.264 SVC videos for a wide range of numbers of multiplexed streams, which can be used as a baseline for assessing the accuracy of novel prediction mechanisms.

C. Simulation Results

In the first set of simulations we estimate the maximum number of video streams $J_{\max}$ that can be accommodated by the given link capacity C, while constraining the information loss probability $P_{\text{loss}}^{\text{info}}$ to a value smaller than a prescribed small constant ϵ. Many independent replications of each simulation were run until the 90\% confidence interval of the information loss probability estimate was less than 10\% of the corresponding sample mean. In the second set of simulations, we estimate the minimum link capacity $C_{\min}$ that accommodates a prescribed number of streams J subject to $P_{\text{loss}}^{\text{info}} < \epsilon$. For each $C_{\min}$ estimate we perform 500 runs, each consisting of 1000 independent video streaming simulations. We do not include the 90\% confidence intervals in the $C_{\min}$ plots, because the confidence intervals are very small ($<$1\% of sample mean) and would clutter the figures.

1) $J_{\max}$ Simulations With Optimal Smoothing: Fig. 1 gives the RD curves and $J_{\max}$ simulation curves, obtained with $\epsilon = 10^{-5}$ and $C = 20$ Mbps, for the five video sequences. The $J_{\max}$ simulation curves (SIM) are, respectively, named as *SIM-G16B3-H.264-unsm* for unsmoothed H.264/AVC streams with GoP structure *G16-B3*, *SIM-G16B15-SVC-unsm* for unsmoothed H.264 SVC streams with GoP structure *G16-B15*, *SIM-G16B3-MP4-unsm* for unsmoothed MPEG-4 Part 2 streams with GoP structure *G16-B3*, *SIM-G16B3-H.264-48KB* for optimally smoothed H.264/AVC streams, *SIM-G16B15-SVC-48KB* for optimally smoothed H.264 SVC streams, and finally *SIM-G16B3-MP4-48KB* for optimally smoothed MPEG-4 Part 2 streams. For reference, we plot the $J_{\max}$ curves corresponding to the multiplexing of
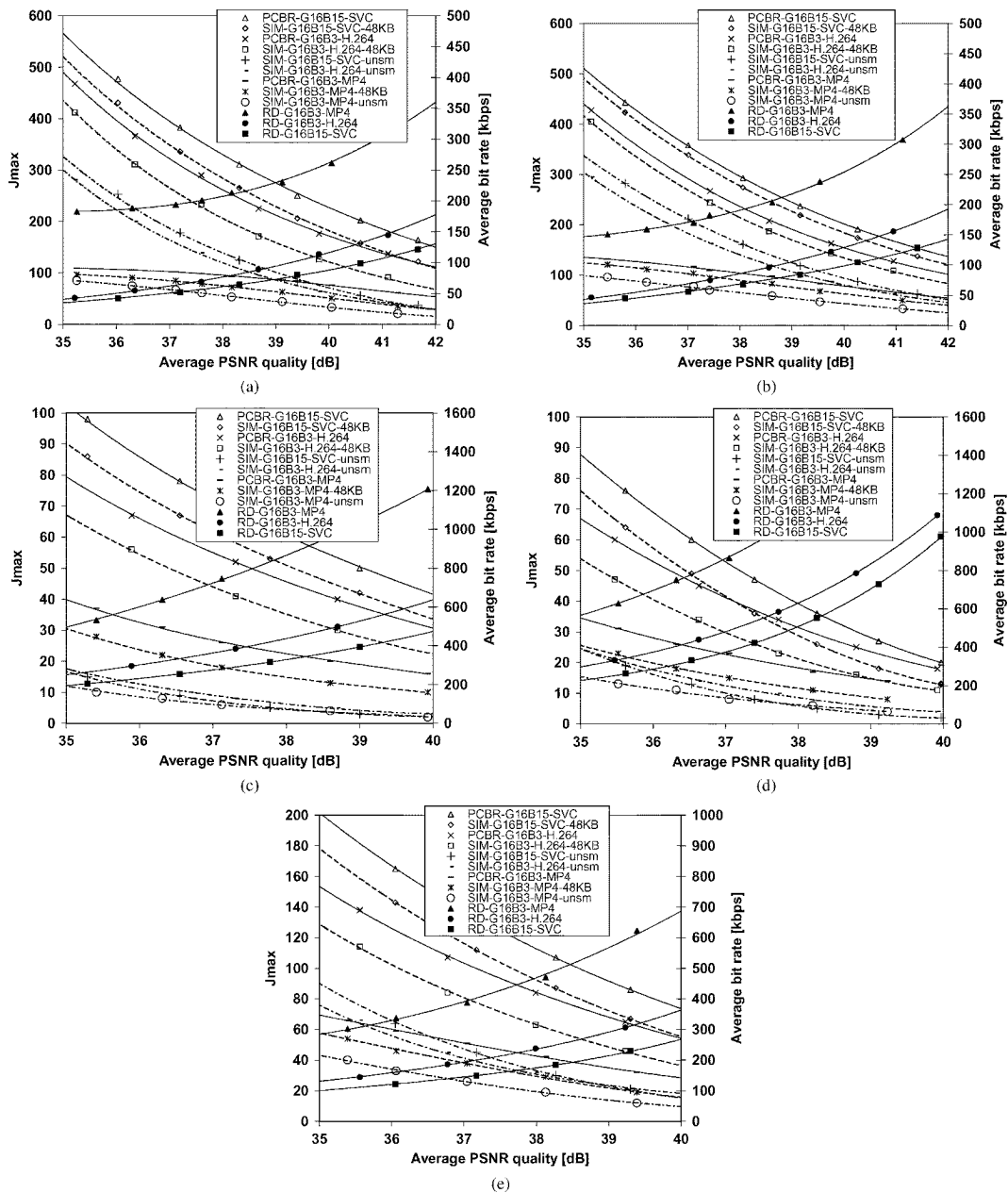
Fig. 1. J_{max} simulation (SIM) and RD curves for five long CIF sequences encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-B15*), and MPEG-4 Part 2 (*G16-B3*). The channel capacity is $C = 20~\mathrm{Mbps}$ and the bit loss probability is $\epsilon = 10^{-5}$. J_{max} curves are provided for unsmoothed (unsm) and optimally smoothed traffic with client buffer size 48 KB (48 KB). Perfect CBR (PCBR) J_{max} curves are included for comparison: (a) *Silence of the Lambs*; (b) *Star Wars IV*; (c) *Sony Demo*; (d) *NBC 12 News*; (e) *Tokyo Olympics*.

perfect constant bit rate traffic, denoted by PCBR. We define *PCBR* video traffic as the sequence of frame size values that are equal to the average frame size of the video stream. Hence, the rate variability of a *PCBR* video stream is zero and J_{max} is computed by dividing C by the stream's average bit rate, resulting in the theoretical maximum value for J_{max}.

The J_{max} values for the unsmoothed streams are strongly affected by the rate variability of the video traffic. To illustrate this effect, we compare the J_{max} curves of the unsmoothed traffic with those of the *PCBR* video traffic. The unsmoothed traffic clearly results in fewer supported streams than the *PCBR* video traffic, which is only attributable to the rate variability. In addition, the gap between the *PCBR* J_{max} curves of the H.264

SVC and the H.264/AVC encodings is much wider than the gap between the corresponding unsmoothed traffic curves, e.g., see Fig. 1(a) and (b). This is also evidence of the profound impact of the rate variability increase of H.264 SVC traffic on J_{max} compared to H.264/AVC traffic.

Very interesting is that for the *Sony Demo* (Fig. 1(c)) and *NBC 12 News* (Fig. 1(d)) sequences, which have relatively high texture and motion complexity, the J_{max} curve of the unsmoothed H.264 SVC traffic is below the curve of the H.264/AVC traffic. This is a very important observation, since this means that the RD efficiency gain of H.264 SVC is completely canceled out by the associated increased rate variability. For very high quality (>38 dB), the H.264 SVC J_{max} curve for the *Sony Demo*
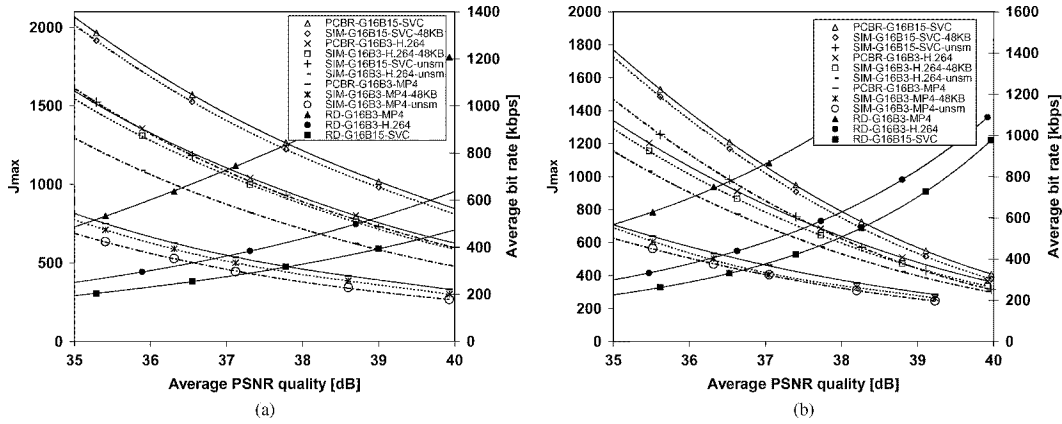
Fig. 2. $J_{\max}$ simulation (SIM) and RD curves for the *Sony Demo* and *NBC 12 News* sequences encoded with H.264 SVC (*G16-B15*), H.264/AVC (*G16-B3*), and MPEG-4 Part 2 (*G16-B3*) for unsmoothed traffic (unsm) and for optimally smoothed traffic (48 KB). The channel capacity is $C = 400$ Mbps, the bit loss probability is $\epsilon = 10^{-5}$. Perfect CBR (PCBR) $J_{\max}$ curves are included for comparison: (a) *Sony Demo*; (b) *NBC 12 News*.
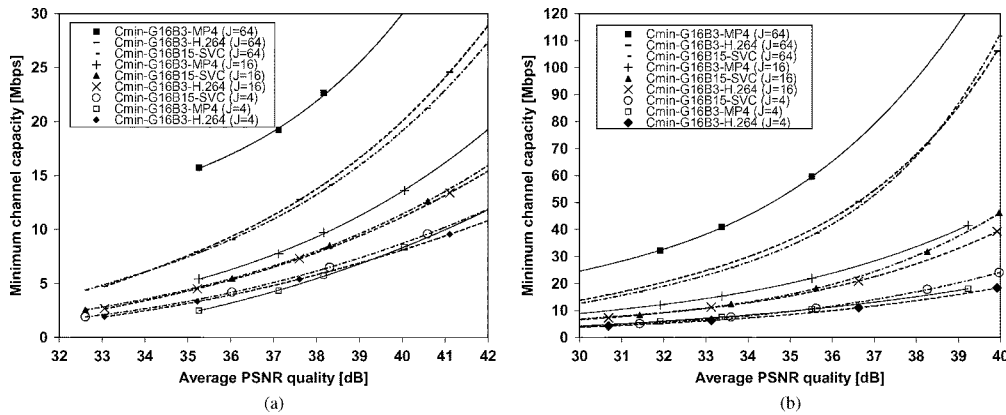

Fig. 3. Minimum channel capacity $C_{\min}$ simulation results for the *Silence of the Lambs* and *NBC 12 News* sequences encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-B15*), and MPEG-4 Part 2 (*G16-B3*) for unsmoothed video traffic. The bit loss probability is $\epsilon = 10^{-5}$ and the numbers of streams are $J = 4$, 16, and 64: (a) *Silence of the Lambs*; (b) *NBC 12 News*.

sequence even approaches the MPEG-4 Part 2 $J_{\max}$ curve, and surprisingly, for the *NBC 12 News* sequence the H.264 SVC $J_{\max}$ curve is below the MPEG-4 Part 2 $J_{\max}$ curve. The reason is that for these two relatively complex sequences, the number of streams that can be supported by the link is small (<20 streams) and as a result the statistical multiplexing effect that copes with the rate variability of the streams is reduced.

Next, we study whether traffic smoothing would bring out the gains in the number of supported streams $J_{\max}$ that one would expect from the RD efficiency gains of H.264 SVC over H.264/AVC. We initially employ optimal smoothing with a client buffer size of 48 KB. We observe that all $J_{\max}$ curves for the optimally smoothed traffic in Fig. 1 have significantly increased values compared to the values for the unsmoothed traffic, and that they are much closer to the theoretical maximum values given by the *PCBR* curves. (In additional experiments with the *Sony* sequence, we found that optimal smoothing with a larger, 128 KB buffer increases $J_{\max}$ by one to five streams; generally, for very large smoothing buffers the PCBR is approached [71].) When examining the gaps between the $J_{\max}$ curves of H.264 SVC and H.264/AVC, we notice that the gaps have increased and approach the theoretical maximum gaps of the *PCBR* curves or equivalently the maximum gain in number of supported streams. We conclude from this

initial analysis that optimal smoothing effectively mitigates the effects of the increased variability of H.264 SVC traffic on the maximum number of streams supported in a bufferless statistical multiplexer. Interesting is that for the relatively lower complexity (texture, motion) *Silence of the Lambs*, *Star Wars 4*, and *Tokyo Olympics* sequences, the $J_{\max}$ curves of the smoothed MPEG-4 Part 2 traffic approach the $J_{\max}$ curves of the unsmoothed H.264 SVC and H.264/AVC traffic in the very high quality region. For the relatively higher complexity *Sony Demo* and *NBC 12 News* sequences, the $J_{\max}$ curves of the unsmoothed H.264 SVC and H.264/AVC traffic are considerably below the curves of the smoothed MPEG-4 Part 2 traffic.

The above observations are clearly dependent on the video content, but also on the chosen link capacity C. Clearly, if the link can only support a small number of streams, then the statistical multiplexing effect is small, resulting in a strong impact of the rate variability on the number of multiplexed streams. The impact is particularly significant when multiplexing high quality H.264 SVC encodings of the relatively complex *Sony Demo* and *NBC 12 News* sequences in the scenario with $C = 20$ Mbps considered in Fig. 1. In order to examine the statistical multiplexing of these two sequences with a higher link capacity, we plot in Fig. 2 $J_{\max}$ curves for $C = 400$ Mbps and $\epsilon = 10^{-5}$. First, we observe that the $J_{\max}$ values are much larger than for
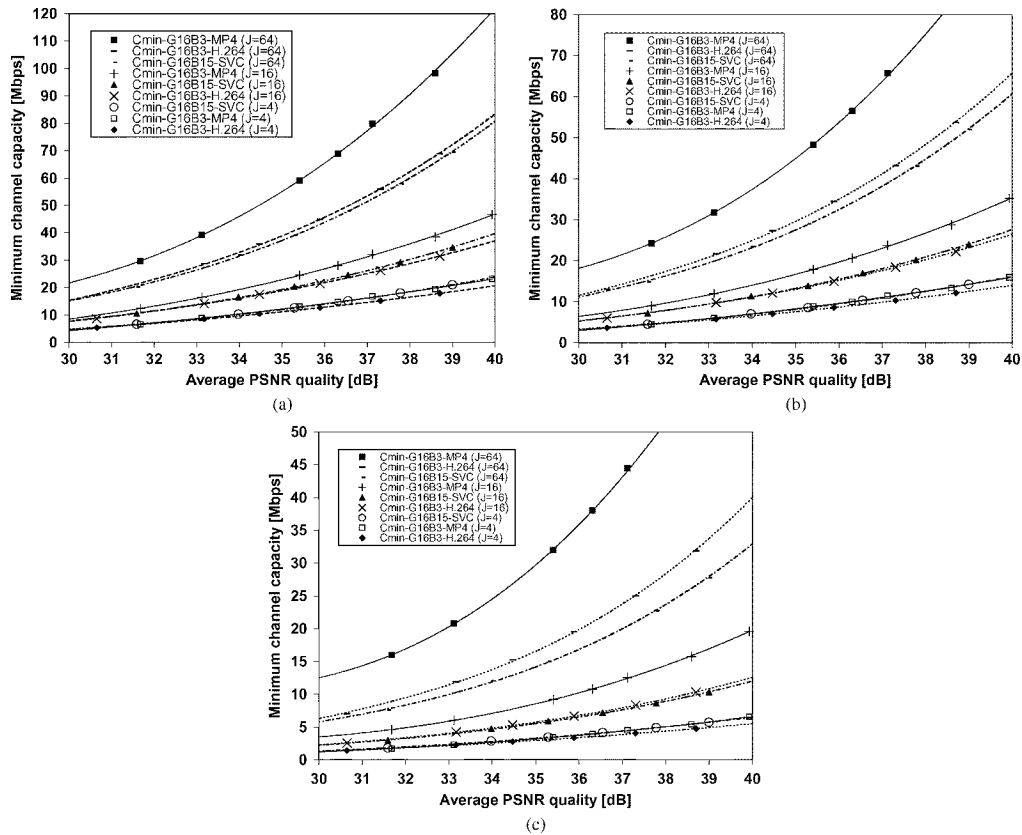
Fig. 4. Minimum channel capacity $C_{\min}$ simulation results for the *Sony Demo* sequence encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-B15*), and MPEG-4 Part 2 (*G16-B3*) for unsmoothed video traffic. The bit loss probabilities are $\epsilon = 10^{-5}, 10^{-3}$, and 10^{-1} and the number of streams are $J = 4$, 16, and 64: (a) *Sony Demo*, $\epsilon = 10^{-5}$; (b) *Sony Demo*, $\epsilon = 10^{-3}$; (c) *Sony Demo*, $\epsilon = 10^{-1}$.

the $C = 20$ Mbps experiments, as we expected. Second, the $J_{\max}$ curves for the unsmoothed traffic are closer to the theoretical upper boundary given by the *PCBR* curves. The optimally smoothed traffic is particularly close to this theoretical upper limit, again illustrating that even for large C there is still a significant impact of smoothing on the $J_{\max}$ values. Nevertheless, in both cases, unsmoothed and smoothed, H.264 SVC clearly allows for more statistically multiplexed streams than H.264/AVC and MPEG-4 Part 2.

Although the $J_{\max}$ simulations provide insight into the significant effects of the increased rate variability of H.264 SVC, they are dependent on the prescribed link capacity C and result in varying numbers of multiplexed streams (i.e., varying levels of statistical multiplexing) across the range of average PSNR video qualities. Therefore, in the next section we perform a second set of simulations that estimate the minimum link capacity $C_{\min}$ required for supporting a prescribed number of streams J. These $C_{\min}$ simulations allow us to study the effects of the rate variability for a fixed number of multiplexed streams across the range of PSNR video qualities.

2) $C_{\min}$ Simulations With Optimal Smoothing: Fig. 3 depicts the $C_{\min}$ curves for unsmoothed traffic of the sequences *Silence of the Lambs* and *NBC 12 News* for $J = 4, 16$, and 64 multiplexed streams for $\epsilon = 10^{-5}$. In general, for $J = 64$, we observe that the $C_{\min}$ values are somewhat lower for the H.264 SVC streams than for H.264/AVC streams. This link capacity

difference is particularly significant for *Silence of the Lambs* in the high quality range (>35 dB), otherwise the $C_{\min}$ differences become relatively small. However, both encoders have a clear advantage over MPEG-4 Part 2. For $J = 16$, the statistical multiplexing effect is less able to compensate for the bit rate variabilities. Overall, the H.264/AVC streams are accommodated by $C_{\min}$ values that are smaller than or nearly equal to the values for the H.264 SVC streams, despite the higher average bit rates of the H.264/AVC streams. H.264 SVC still outperforms MPEG-4 Part 2 over the entire quality range.

For $J = 4$, the increased rate variability of H.264 SVC results in $C_{\min}$ values that are overall comparable to those of multiplexed MPEG-4 Part 2 streams. For the *Silence of the Lambs* sequence, we observe the surprising result that H.264 SVC requires the highest $C_{\min}$ values over the entire quality range and MPEG-4 Part 2 even outperforms H.264/AVC below 38 dB. For the *NBC 12 News* sequence, H.264 SVC has worst performance in the quality range above 35 dB. The conclusion is that for a relatively small number of multiplexed streams (<16), H.264/AVC generally results in lower $C_{\min}$ requirements, while depending on the video sequence, H.264 SVC can even be outperformed by MPEG-4 Part 2 streams.

Next, we examine the impact of the information loss probability on $C_{\min}$ in Fig. 4. The unsmoothed *Sony Demo* streams are multiplexed with maximum losses $\epsilon = 10^{-5}, 10^{-3}$, and 10^{-1}, respectively, for $J = 4$, 16,
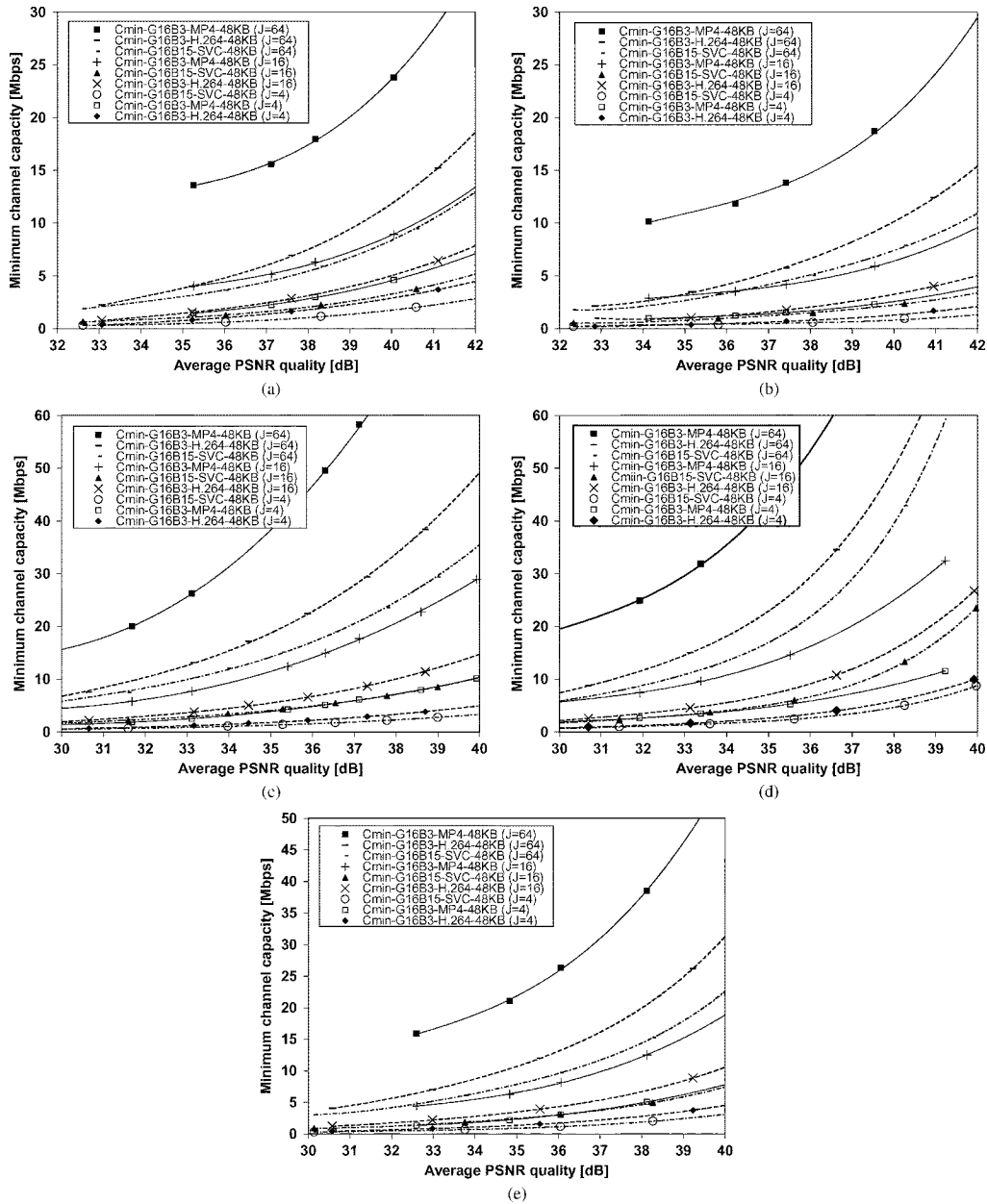
Fig. 5. $C_{\min}$ simulation results for five long CIF sequences encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-B15*), and MPEG-4 Part 2 (*G16-B3*) for optimally smoothed (48 KB) video traffic with a 48KB client buffer. The bit loss probability is $\epsilon = 10^{-5}$ and the number of streams are $J = 4$, 16, and 64: (a) *Silence of the Lambs*; (b) *Star Wars IV*; (c) *Sony Demo*; (d) *NBC 12 News*; (e) *Tokyo Olympics*.

and 64 streams. The $C_{\min}$ values are significantly lower when the allowable losses are larger, as we expected, and this is the case for all encoders and numbers of streams J. Interesting is that overall the relative order of the $C_{\min}$ curves, corresponding to the different encoders for each value of J, is preserved.

In Fig. 5, we examine the $C_{\min}$ values for optimally smoothed streams (client buffer size 48 KB). Overall, optimally smoothed H.264 SVC traffic has lower $C_{\min}$ values for $J = 4$, 16, and 64 over the entire quality range. The quality range above 35 dB is particularly favorable for optimally smoothed H.264 SVC over H.264/AVC. Optimally smoothed MPEG-4 Part 2 traffic clearly requires substantially more network bandwidth resources.

In summary, we conclude from the $J_{\max}$ and $C_{\min}$ simulations with optimally smoothed traffic that optimally smoothed

H.264 SVC streams clearly have an advantage over optimally smoothed H.264/AVC and MPEG-4 Part 2 streams. In particular, the $J_{\max}$ simulations indicate that close to optimal results (*PCBR*) are achievable with optimally smoothed traffic. Optimal smoothing [10], [11] is an off-line technique designed for prerecorded video streams. Optimal smoothing can been adapted for live video through appropriate traffic descriptors and predictors, which have so far only been examined for MPEG-4 Part 2 and preceding MPEG codecs [72]. Researching appropriate traffic descriptors and predictors for the new H.264/AVC and H.264 SVC encoders with their more bursty traffic is an open problem. On the other hand, basic smoothing, which is computationally significantly less complex than optimal smoothing, can easily be implemented for live video. We are therefore motivated to com-
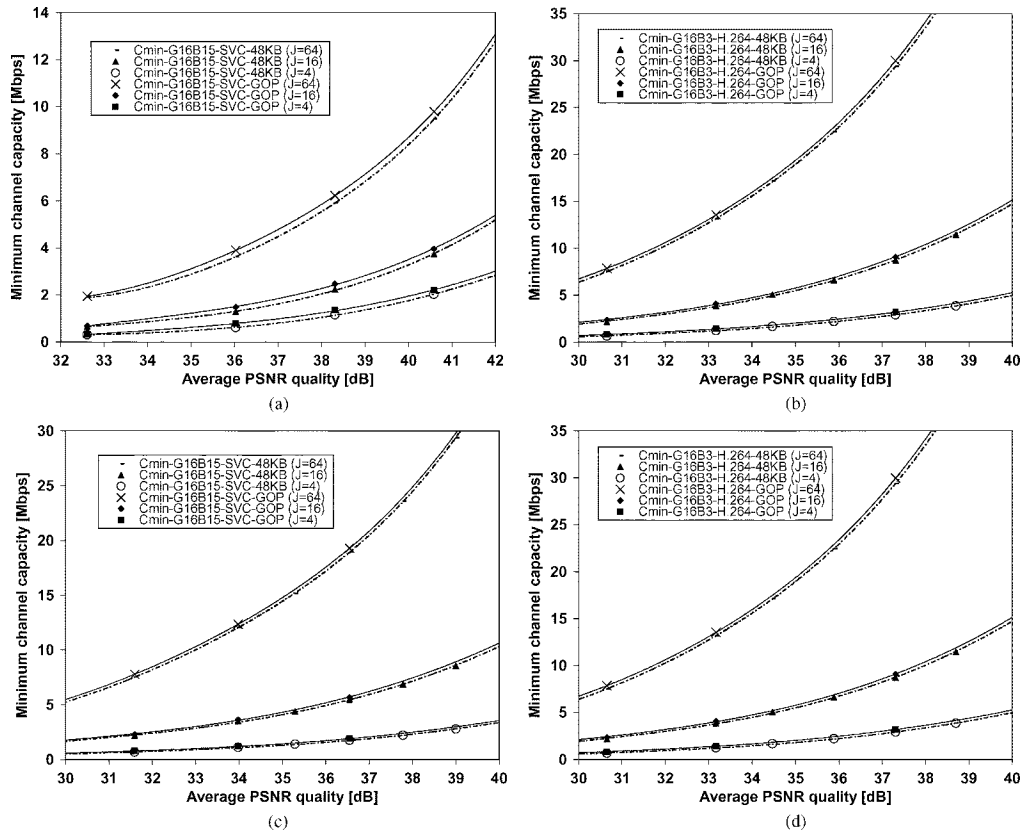
Fig. 6. $C_{\min}$ simulation results for the *Silence of the Lambs* and *Sony Demo* sequences encoded with H.264 SVC (*G16-B15*) and H.264/AVC (*G16-B3*) for basic smoothed traffic with aggregation level $a = 16$ (GOP) and for optimally smoothed traffic (48 KB). The bit loss probability is $\epsilon = 10^{-5}$ and the number of streams are $J = 4$, 16, and 64: (a) *Silence of the Lambs*, H.264 SVC (*G16-B15*); (b) *Silence of the Lambs*, H.264/AVC (*G16-B3*); (c) *Sony Demo*, H.264 SVC (*G16-B15*); (d) *Sony Demo*, H.264/AVC (*G16-B3*).

pare the bufferless statistical multiplexing performance ($C_{\min}$) of basic smoothing with optimal smoothing.

3) $C_{\min}$ Simulations With Basic Smoothing: Fig. 6 depicts the $C_{\min}$ curves for the *Silence of the Lambs* and *Sony Demo* H.264 SVC video traffic (*G16-B15*) that is smoothed with aggregation level $a = 16$ (GoP size), and for H.264/AVC video traffic (*G16-B3*) smoothed with $a = 16$. We also include the results obtained for optimal smoothing. The basic smoothing $C_{\min}$ curves are only very slightly above the $C_{\min}$ curves for optimally smoothed traffic. This indicates that basic smoothing with $a = 16$ is almost as effective as optimal smoothing in reducing the rate variability for efficient bufferless statistical multiplexing.

4) Basic Smoothing Delay Implications: The simulation results in the preceding sections together with the delay analysis in the Appendix establish a reference framework for evaluating the traffic smoothing versus delay trade-off. In this section, we investigate the choice of the basic smoothing parameters a that ensure that (*i*) the link capacity requirements for H.264 SVC traffic (hierarchical B frames) are reduced compared to H.264/AVC traffic (classical B frames), and (*ii*) the link capacity required with basic smoothing closely approaches the link capacity required with optimally smoothed traffic.

Fig. 7 depicts $C_{\min}$ simulation curves for unsmoothed and smoothed (basic) traffic with aggregation levels $a = 2, 4, 8$, and 16. The experiments cover the five sequences that are en-

coded with H.264 SVC (*G16-B15*) and H.264/AVC (*G16-B3*). We present illustrative results for $J = 4$, 16, and 64 streams, while the bit loss probability is restricted to $\epsilon = 10^{-5}$; we have also analyzed identical experiments with $\epsilon = 10^{-3}$, which we can not include due to space constraints. Fig. 7(a) and (b) present the case with $J = 64$ multiplexed streams, Fig. 7(c) and (d)for $J = 16$ streams, and Fig. 7(e) and (f) for $J = 4$ streams. We present illustrative results for videos with relatively low texture and motion complexity in Fig. 7(a), (c), and (e), while illustrative results for videos with relatively high texture and motion complexity are presented in Fig. 7(b), (d), and (f).

For $J = 64$ streams, in general, the unsmoothed H.264 SVC streams require smaller $C_{\min}$ values than the H.264/AVC streams. This is explained by the relatively large number of streams that are statistically multiplexed. Ideally, the $C_{\min}$ values should be close to the $C_{\min}$ values for optimally smoothed streams or, equivalently, close to the $C_{\min}$ values for basic smoothed streams with aggregation $a = 16$, which is the GoP size, as we illustrated in Fig. 6. In Fig. 7(a), for example, the $C_{\min}$ simulation curves for increasing aggregation levels a approach the $C_{\min}$ curves of the traffic smoothed with $a = 16$ (which gives very close to optimal smoothing results). This observation holds for all test sequences and numbers of multiplexed streams J, although for the *Sony Demo* sequence the convergence is slower than for the other four sequences. Overall, when $J = 64$ the aggregation level a should be set
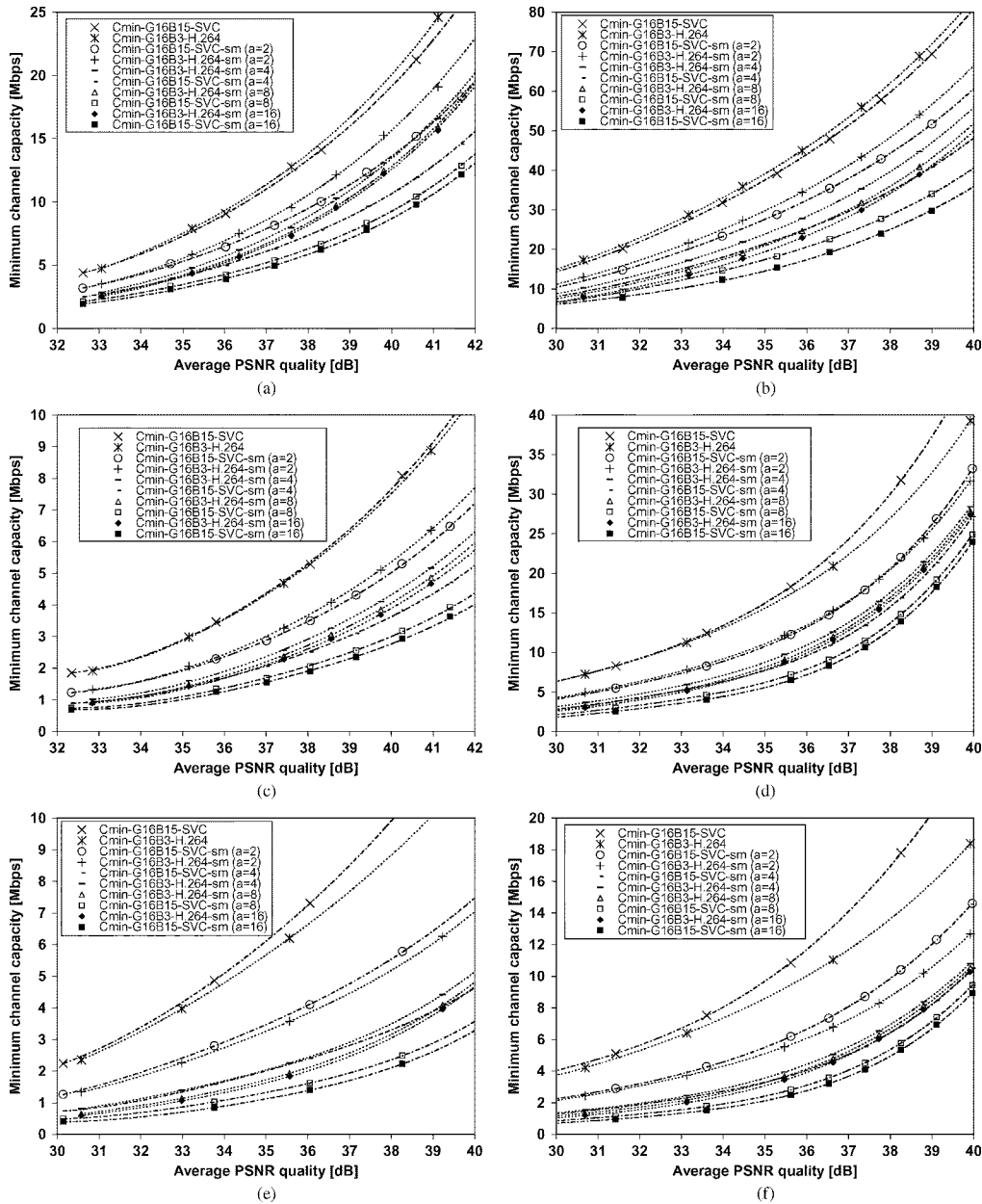
Fig. 7. $C_{\min}$ simulation results for unsmoothed and basic smoothed traffic with aggregation levels $a = 2$, 4, 8, and 16. The five sequences are encoded with H.264 SVC (*G16-B15*) and H.264/AVC (*G16-B3*). The bit loss probability is $\epsilon = 10^{-5}$ and the number of streams are $J = 4$, 16, and 64: (a) *Silence of the Lambs* (*J = 64*); (b) *Sony Demo* (*J = 64*); (c) *Star Wars 4* (*J = 16*); (d) *NBC 12 News* (*J = 16*); (e) *Tokyo Olympics* (*J = 4*); (f) *NBC 12 News* (*J = 4*).

to $a = 4$ or $a = 8$ for H.264/AVC stream multiplexing to approach the optimal performance, and to $a = 8$ or $a = 16$ for H.264 SVC streams. The choice between the two a values for each encoder depends on the content type, with the larger value meant for the most complex sequences.

Analogously, we analyzed the cases with $J = 32$, $J = 16$, and $J = 4$ multiplexed streams. Table I enumerates aggregation levels a that when applied to both H.264 SVC and H.264/AVC video streams result in lower $C_{\min}$ requirements for H.264 SVC streams (*G16-B15*) than for H.264/AVC streams (*G16-B3*) for both examined loss probabilities 10^{-5} and 10^{-3}. Table II gives basic smoothing aggregation levels a that achieve close to optimal smoothing $C_{\min}$ values for H.264/AVC and H.264 SVC, respectively. For the cases with two a values, we recommend

the higher value for sequences with relatively high texture and motion complexity. The corresponding end-to-end delays, calculated based on the delay analysis in the Appendix, are provided in Table II for live video streaming (middle two columns) and for prerecorded video streaming (right two columns).

From this analysis we conclude that the H.264 SVC streams generally require aggregation levels twice as large as the H.264/AVC streams to obtain close to optimal statistical multiplexing performance. The corresponding end-to-end-delays are approximately two to three times larger for H.264 SVC than for H.264/AVC.

The preceding analysis considers one video sequence (out of the five sequences) in a given multiplexing experiment. Next, we examine whether the recommendations for the choice of

TABLE II
AGGREGATION LEVELS a [(a FOR LOW COMPL. SEQ.) $\vee$ (a FOR HIGH COMPL. SEQ.)] FOR BASIC SMOOTHING SUCH THAT C_{min} FOR BASIC SMOOTHING VERY CLOSELY APPROACHES C_{min} FOR OPTIMAL SMOOTHING FOR H.264/AVC AND H.264 SVC, RESPECTIVELY; CORRESPONDING DELAYS [IN FRAME PERIODS] FOR PRERECORDED AND LIVE VIDEO ARE ALSO PROVIDED. THESE RESULTS APPLY FOR $J = 4, 16, 32, 64$ MULTIPLEXED STREAMS FOR BOTH $\epsilon = 10^{-5}$ AND $\epsilon = 10^{-3}$

Basic sm. → opt. sm.		Δ_c (AVC)	Δ_h (SVC)	Δ_c^{pre} (AVC)	Δ_h^{pre} (SVC)
H.264/AVC	H.264 SVC				
$(a = 4) \vee (a = 8)$	$(a = 8) \vee (a = 16)$	$13 \vee 21$	$36 \vee 52$	$6 \vee 10$	$13 \vee 21$

TABLE I
AGGREGATION LEVELS a FOR BASIC SMOOTHING SUCH THAT C_{min} FOR H.264 SVC (*G16-B15*) IS LESS THAN FOR H.264/AVC (*G16-B3*) FOR BOTH $\epsilon = 10^{-5}$ AND $\epsilon = 10^{-3}$. WE PROVIDE (a VALUE FOR LOW COMPLEXITY SEQUENCE) $\vee$ (a VALUE FOR HIGH COMPLEXITY SEQUENCE)

J	Basic sm. a such that C_{min}(SVC) < C_{min}(AVC)
64	$(a = 1)$
32	$(a = 1) \vee (a = 2)$
16	$(a = 2) \vee (a = 4)$
4	$(a = 4) \vee (a = 8)$

TABLE III
C_{min} BIT RATES FOR MIXES OF $J = 16$ VIDEO STREAMS DRAWN FROM ALL FIVE VIDEOS FOR DIFFERENT BASIC SMOOTHING LEVELS a AND OPTIMAL SMOOTHING (OPT. SM.) FOR $\epsilon = 10^{-3}$

a	H.264/AVC			H.264 SVC		
	low qu.	medium qu.	high qu.	low qu.	medium qu.	high qu.
1	6.85	10.43	16.71	7.89	11.74	18.62
2	4.62	7.21	12.48	5.10	7.66	12.79
4	3.41	5.50	10.47	3.57	5.44	9.87
8	2.94	4.91	9.96	2.75	4.27	8.62
16	2.73	4.66	9.71	2.34	3.70	8.10
Opt. Sm.	2.57	4.44	9.47	2.23	3.55	7.91

TABLE IV
C_{min} BIT RATES FOR MIXES OF $J = 16$ VIDEO STREAMS DRAWN FROM ALL FIVE VIDEOS FOR DIFFERENT BASIC SMOOTHING LEVELS a AND OPTIMAL SMOOTHING (OPT. SM.) FOR $\epsilon = 10^{-5}$

a	H.264/AVC			H.264 SVC		
	low qu.	medium qu.	high qu.	low qu.	medium qu.	high qu.
1	10.33	15.60	24.24	12.03	17.84	27.76
2	6.61	10.18	17.02	7.46	11.15	18.12
4	4.57	7.29	13.70	4.94	7.46	13.30
8	3.79	6.32	12.94	3.57	5.51	11.38
16	3.47	5.93	12.58	2.91	4.62	10.63
Opt. Sm.	3.18	5.55	12.17	2.72	4.36	10.31

the aggregation level a also hold for a heterogeneous mix of the five video sequences. We organized the H.264/AVC video streams and the H.264 SVC video streams each into three quality groups based on average PSNR values: low quality (32–34 dB), medium quality (35–37 dB), and high quality (38–40 dB). We conducted multiplexing simulations for each quality group to determine the minimum link capacities C_{min} required to achieve loss probabilities below $\epsilon = 10^{-3}$ and $\epsilon = 10^{-5}$, respectively. In each simulation, we multiplex $J = 16$ streams drawn randomly from the five video sequences (while equalizing for the different stream lengths so that each video sequence is selected with approximately equal probability). The respective estimated C_{min} values are reported in Table III for $\epsilon = 10^{-3}$, and in Table IV for $\epsilon = 10^{-5}$.

From the data in Tables III and IV, we conclude that the above recommendations for the aggregation levels a also hold for the heterogeneous mix of the video streams; furthermore, the recommendations hold across quality groups and for both $\epsilon = 10^{-3}$ and $\epsilon = 10^{-5}$. The recommended aggregation levels for approaching the optimal smoothing C_{min} value within 15\%, are $a = 4$ to $a = 8$ for H.264/AVC streams and $a = 8$ to $a = 16$ for H.264 SVC streams. This observation confirms that H.264 SVC streams require higher aggregation levels to approximate the optimal smoothing C_{min}. We also reconfirm that the aggregation level at which H.264 SVC streams achieve C_{min} link capacities below H.264/AVC capacities, is at least $a = 4$ and even as high as $a = 8$. Since these multiplexing experiments with heterogeneous video sequences reconfirm the aggregation level recommendations, we conclude that the different encoder configurations, i.e., hierarchical B frames for H.264 SVC (*G16-B15*) versus classical B frames for H.264/AVC (*G16-B3*) are the determining factors in the statistical multiplexing behavior of the respective video streams.

V. BUFFERED STATISTICAL MULTIPLEXING

Next, we study the buffered statistical multiplexing of video streams encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-*

B15), and MPEG-4 Part 2 (*G16-B3*). The video traffic is not smoothed in order to assess the direct impact of the multiplexer buffer size. The buffer serves the purpose of absorbing some of the rate variability of the video streams that are multiplexed on the link. From among the wide range of buffer management and scheduling policies, see e.g. [73]–[76], we consider the elementary taildrop policy with first-come-first-served scheduling, to assess the fundamental impact of the multiplexer buffer. Specifically, with R_m given in (3) denoting the aggregate bit rate [in bit/s] of the J ongoing video streams in frame period m, b_{m-1} denoting the buffered video traffic [in bit] at the end of the preceding frame period $m-1$ (i.e., at the beginning of frame period m), and noting that traffic is served at bit rate C, the amount of buffered video traffic at the end of frame period m is obtained as

$$b_m = \min\left\{[b_{m-1} + (R_m - C)T]^+, B\right\}, \quad (5)$$

where B denotes the buffer capacity [in bit]. The amount of lost video bits during frame period m is given by $[b_{m-1} + (R_m - C)T - B]^+$ and the expected long run fraction of lost bits gives the information loss probability, which is required to be less than ϵ.

Fig. 8 depicts J_{max} simulation results for the five CIF sequences. The channel capacity is $C = 20$ Mbps and $\epsilon = 10^{-5}$. Curves are presented for buffer sizes 24, 192, and 3840 KB. (We also examined the buffer sizes 48 and 96 KB, which
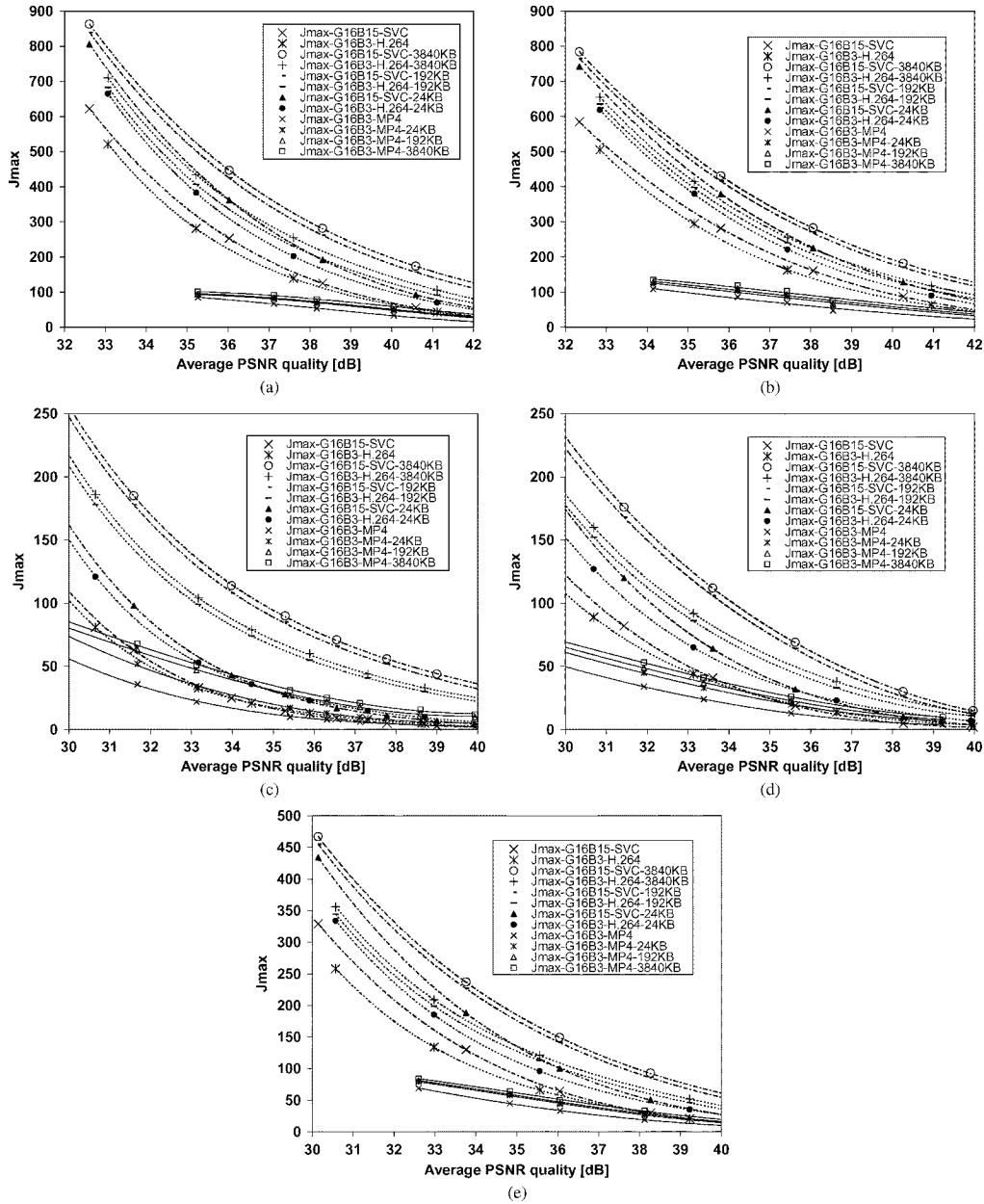
Fig. 8. $J_{\max}$ buffered multiplexing simulation results for five long CIF sequences encoded with H.264/AVC (*G16-B3*), H.264 SVC (*G16-B15*), and MPEG-4 Part 2 (*G16-B3*). The channel capacity is $C = 20$ Mbps and the bit loss probability is $\epsilon = 10^{-5}$. $J_{\max}$ curves are provided for unsmoothed traffic. Curves are presented for buffer sizes are set to 24, 192, and 3840 KB. The bufferless multiplexing results are included for reference: (a) *Silence of the Lambs*; (b) *Star Wars IV*; (c) *Sony Demo*; (d) *NBC 12 News*; (e) *Tokyo Olympics*.

are not included to avoid clutter in the plots.) The bufferless multiplexing results are depicted for comparison. Analogous to the minimum channel capacity experiments, we determine the buffer size that gives near optimal statistical multiplexing results for H.264 SVC, H.264/AVC, and MPEG-4 Part 2 streams, whereby we adopt as benchmark for optimal results the $J_{\max}$ curve for the largest buffer size 3840 KB. Comparisons of the results in Figs. 1 and 8 indicate that the $J_{\max}$ curve for 3840 KB is very close to the PCBR curve, which gives the maximum number of streams that can be supported on the link. We identify the buffer sizes that result in $J_{\max}$ values that are relatively close to the $J_{\max}$ values for buffer size 3840 KB. The recommended buffer size ranges for each encoder are summarized in Table V. We determine the buffer ranges

across the five video sequences, with the largest buffer sizes corresponding to complex sequences. The H.264 SVC streams require approximately twice the buffer size compared to the H.264/AVC streams, which in turn require about double the buffer size required for MPEG-4 Part 2 streams. With the delay analysis presented in the Appendix, we obtain a delay of 25 frame periods for transmitting unsmoothed live H.264 SVC video over a transmit path with a single buffer stage with 192 KB compared to 9 frame periods for transmitting H.264/AVC video over a transmit path with a single 96 KB buffer stage.

We similarly studied the case when $\epsilon = 10^{-3}$; the corresponding plots are not included due to space constraints. For $\epsilon = 10^{-3}$, the recommended buffer size ranges are significantly smaller (approx. half) for each encoder than for $\epsilon = 10^{-5}$.
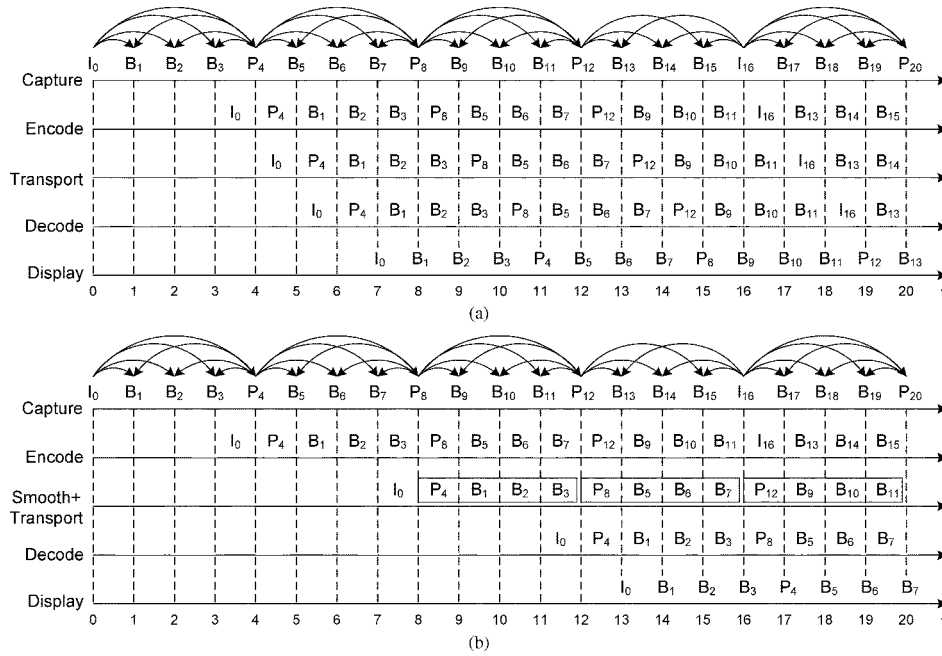
Fig. 9. Delay analysis of classical B frame H.264/AVC encoding with GoP structure *G16-B3* for no smoothing and for basic smoothing with $a = 4$: (a) no smoothing, $a = 1$; (b) basic smoothing with $a = 4$.
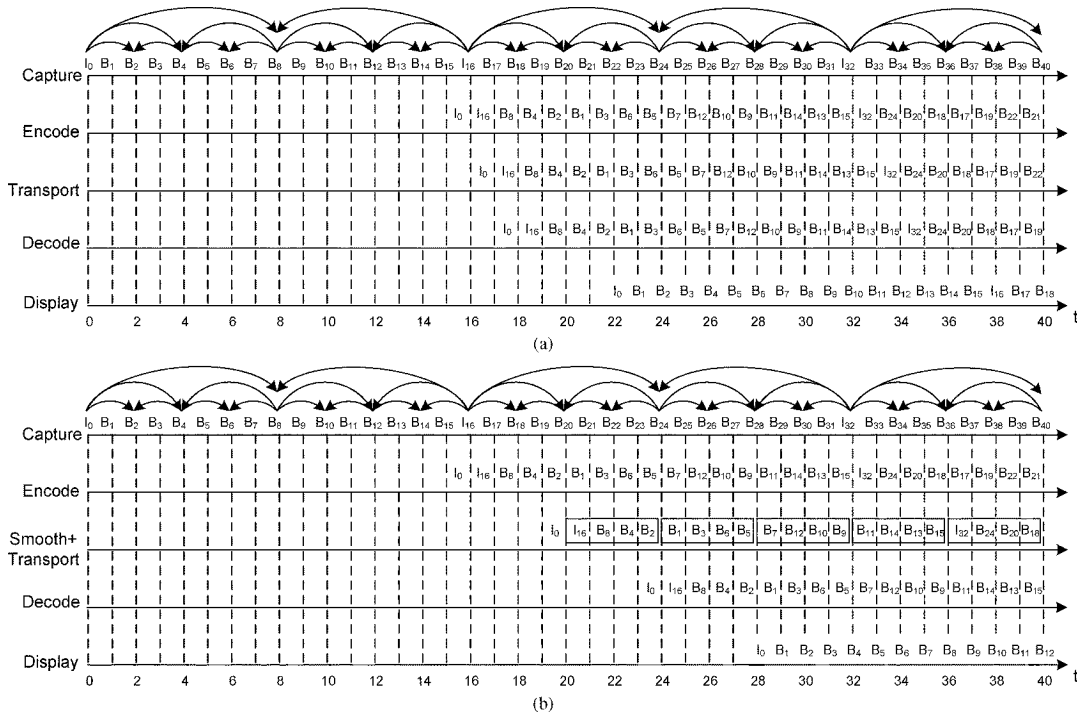

Fig. 10. Delay analysis of hierarchical B frame H.264 SVC encoding with GoP structure *G16-B15* for no smoothing, and for basic smoothing with $a = 4$: (a) no smoothing, $a = 1$; (b) basic smoothing with $a = 4$.

TABLE V
OVERVIEW OF RECOMMENDED BUFFER SIZE RANGES FOR BUFFERED STATISTICAL MULTIPLEXING WITH $\epsilon = 10^{-5}$

	H.264 SVC	H.264/AVC	MPEG-4 Part 2
Buffer size [KB]	48–192	24–96	24–48

TABLE VI
DELAYS [IN FRAME PERIODS] FOR LIVE H.264/AVC (*G16-B3*) AND H.264 SVC (*G16-B15*) STREAMS

Delay	unsm	$a = 2$	$a = 4$	$a = 8$	$a = 16$
H.264/AVC	7	9	13	21	37
H.264 SVC	22	24	28	36	52

However, the double buffer size relationship between encoders remains, as well as the corresponding delay differences. We conclude that the RD efficiency improvements between the encoders comes at the price of increased buffer sizes and corresponding delays in the buffered statistical multiplexing scenario.

TABLE VII
DELAYS [IN FRAME PERIODS] FOR PRERECORDED H.264/AVC (*G16-B3*) AND
H.264 SVC (*G16-B15*) STREAMS. THE DELAY WITH OPTIMAL SMOOTHING
(opt) WITH (ADDITIONAL) START-UP DELAY $s = 0$ IS IDENTICAL TO THE
DELAY FOR UNSMOOTHED TRAFFIC

Delay	unsm/opt	$a = 2$	$a = 4$	$a = 8$	$a = 16$
H.264/AVC	3	4	6	10	18
H.264 SVC	6	7	9	13	21

VI. CONCLUSIONS

We have examined the statistical multiplexing behavior of H.264 SVC, H.264/AVC, and MPEG-4 Part 2 encoded video with long video sequences. In particular, we have considered the bufferless statistical multiplexing of smoothed video streams and the buffered statistical multiplexing of unsmoothed video streams. We have found that off-line optimal smoothing ensures that the RD efficiency gains of H.264 SVC with hierarchical B frames over H.264/AVC with classical B frames translate into commensurate gains in the number stream supported with statistical multiplexing. (Without smoothing, the higher rate variability of H.264 SVC may actually result in fewer supported streams than with the less RD efficient H.264/AVC and in some scenarios even fewer SVC streams than with the even less RD efficient MPEG-4 Part 2.) We further examined basic smoothing which averages the sizes of blocks of a successive video frames and is thus simple to implement in on-line fashion and readily applicable to live video. We characterized the trade-off between increased delay with increased levels of smoothing (for larger a) and the resulting reduced rate variability and corresponding increased number of supported streams with statistical multiplexing. Specifically, we identified the basic smoothing levels a that ensure that (*i*) more H.264 SVC than H.264/AVC streams are supported with statistical multiplexing, and that (*ii*) the number of H.264 SVC streams and H.264/AVC streams supported with basic smoothing closely approaches the number of streams supported with optimal smoothing. Moreover, we identified the sizes of the multiplexer buffers that ensure that the numbers of supported H.264 SVC streams and H.264/AVC streams approach the theoretical maximum given by the link capacity divided by the average stream bit rate; we found that H.264 SVC requires roughly twice the multiplexer buffer of H.264/AVC, which in turn requires twice the buffer of MPEG-4 Part 2.

There are numerous directions for future research on the statistical multiplexing of H.264 SVC and H.264/AVC encoded video. One important direction is examining collaborative smoothing strategies and active buffer management strategies considering the frame playout deadlines for H.264/AVC and H.264 SVC encoded video.

APPENDIX
DELAY ANALYSIS OF SMOOTHED TRANSMISSION OF H.264 SVC AND H.264/AVC VIDEO

In this Appendix we analyze the end-to-end delay introduced by the video encoding and decoding in conjunction with the smoothing of the video frame sizes for network transport. We initially consider live video and evaluate the time shift between the capture of a frame at the sender and the display of the frame at the receiver; we subsequently examine prerecorded video. Throughout, we normalize time by the frame period (33 ms for NTSC video). (For all delays reported in units of frame periods, the corresponding delays in units of seconds are obtained by dividing the delay in units of frame periods by the frame rate F in units of frames/second, which is $F = 30$ frames/second for NTSC video.) In general, the time shift Δ between frame capture and display can be decomposed into the following components:

- $\delta_{\text{enc, dep}}$: Delay introduced due to the dependencies of the encoded frames, i.e., maximum delay a given captured frame experiences due to waiting for the capture of subsequent frames that are needed for the encoding of the given captured frame.
- $\delta_{\text{enc, comp}}$: Delay introduced by the computations needed for the encoding.
- $\delta_{\text{sm trans}}$: Delay introduced by the smoothed transmission.
- $\delta_{\text{dec, comp}}$: Delay introduced by the computations needed for the decoding of a frame.
- δ_{d}: Delay introduced by reordering of frames to ensure uninterrupted display sequence.

The total end-to-end delay is obtained by summing the delay components

$$\Delta = \delta_{\text{enc, dep}} + \delta_{\text{enc, comp}} + \delta_{\text{sm trans}} + \delta_{\text{dec, comp}} + \delta_{\text{d}}. \quad (6)$$

For each of the following delay analyzes we initially suppose that the encoding computations and the decoding computations take one frame period, i.e., $\delta_{\text{enc, comp}}^{\text{1frame}} = \delta_{\text{dec, comp}} = 1$, we subsequently consider the cases when computation times become negligible. Throughout, we suppose that it takes one frame period to transmit one (unsmoothed) frame, and a frame periods to transmit a block of a smoothed frames, as is consistent with the evaluation of the aggregate bit rate in (3). We note that the transmission of unsmoothed video is equivalent to basic smoothing with the aggregation level of one frame, i.e., $a = 1$. We let β, $\beta \geq 0$, denote the number of B frames between successive key picture (I or P frames).

A. Live Video With Classical B Frames

Fig. 9 illustrates the delay structure for live streaming of H.264/AVC video encoded with classical B frames for GoP structure *G16-B3*, i.e., $\beta = 3$. The capture time index axis represents the frame type (I, P, or B) that is used to encode each captured frame. Each frame is designated by its frame type and its capture time, e.g., P_4 is the frame captured at time index four and is encoded as a P frame. We suppose that the capture time itself is infinitesimally short and negligible. On the encode time axis, the frames are put in encoding order according of the motion compensated prediction frame dependencies, which are indicated by the arrows above the capture time axis. The time shift between the capture and encode axes represents the delay due to the encoding dependencies $\delta_{\text{enc, dep}}$. Specifically, we observe that $\delta_{\text{enc, dep}} = \beta = 3$, since frame B_1 needs to wait for the capture of frame P_4 before frame B_1 can be encoded. The time shift between the encode and transport time axes represents the delay due to encoding computations $\delta_{\text{enc, comp}}$.

For example, the frame P_4 is encoded in between time indices $t = 4$ and $t = 5$, followed by the encoding of the frames B_1, B_2, and B_3 that depend on P_4 and I_0. In general, with smoothed transmission, all a frames of a smoothing block need to be encoded before transmission of the block can commence, hence $\delta_{\text{enc, comp}} = a$.

Subsequently, the encoded frames are transmitted in encoding order since the decoder needs the frames in encoding order for the decoding process to run without introducing unnecessary reordering delays. In the illustrated unsmoothed example, frame P_4 is transmitted between time indices $t = 5$ and $t = 6$, while for the illustrated smoothed transmission example, the first block of $a = 4$ frames is transmitted between time indices $t = 8$ and $t = 12$. Generally, noting that the decoding can only start when the entire block of a frames is received, we obtain $\delta_{\text{sm trans}} = a$, which is represented by the time shift between the transport and decode axes in the illustration in Fig. 9. We do not consider store-and-forward transmission delays nor propagation, queueing, or processing delays in the transport network; these delays could be subsumed in $\delta_{\text{sm trans}}$ in straightforward fashion. In particular, for buffered multiplexing of unsmoothed video ($a = 1$), as considered in Section V, the transmission delay in frame periods with a single buffered multiplexing stage on the transmission path is bounded by one frame period (for the transmission by the sending host) plus the maximum buffer delay, namely the buffer capacity B [in bit] divided by the bit rate C [in bit/s] normalized with the frame rate F [in frames/s], i.e., $\delta_{\text{trans}} = 1 + \lceil FB/C \rceil$.

Next, the decoder processes frame P_4 in between time indices $t = 6$ and $t = 7$ in Fig. 9(a); generally, $\delta_{\text{dec, comp}} = 1$. In addition, the receiver needs to reorder the decoded frames into display order to ensure uninterrupted playback. This reordering introduces one frame period delay, i.e., $\delta_{\text{d}} = 1$, since frame B_1 in Fig. 9(a) is not available for display until time instant $t = 8$.

In summary, we obtain for live video with classical B frames

$$\Delta_c = \beta + a + a + 1 + 1 = \beta + 2a + 2. \tag{7}$$

We remark that we have not included the first I frame in the data blocks for basic smoothing. Alternatively, this I frame can be included and the non-overlapping blocks would shift one frame index to the left without any implications for the end-to-end delay. The advantage of not including the first I frame is that the first block already contains a large P frame. Singling out the first I frame allows for spreading its transmission over multiple frame periods if the I frame is encoded immediately when it is captured. For example, in Fig. 9(a) the first I frame can be transmitted over four frame periods, if it is immediately encoded after time index zero.

We briefly adapt the above delay analysis to scenarios with negligible encoding and/or decoding computation times as follows. We focus on scenarios where either $(\beta+1)/a$ or $a/(\beta+1)$ is an integer. If an arbitrary number of video frames can be encoded in negligible time, $\delta_{\text{enc, comp}} = 0$, then the delay due to frame encoding dependencies becomes $\delta_{\text{enc, dep}} = \max\{\beta, a - 1\}$. To see this, note that two conditions need to be met before transmitting the first block of a encoded frames: (i) the first B

frame needs to await the capture of the successive P frame, i.e., for β frame periods, and (ii) the first frame to be transmitted in a smoothing block needs to await the capture of the remaining $a-1$ frames for the block. If an arbitrary number of video frames can be decoded in negligible time, $\delta_{\text{dec, comp}} = 0$, then the display reordering delay becomes $\delta_{\text{d}} = 1_{\{((\beta+1)/a)\geq 2\}}$, where $1_{\{A\}}$ denotes the indicator function which is one if A is true, and zero otherwise.

B. Live Video With Hierarchical B Frames

We consider hierarchical B frames with a dyadic structure, i.e., $\beta = 2^k - 1$ B frames between key pictures for some integer $k \geq 0$. We do not consider low-delay or constrained delay B frame prediction structures [2].

Reasoning as above, along the illustration in Fig. 10, we find that the delay components $\delta_{\text{enc, dep}}$, $\delta_{\text{enc, comp}}$, $\delta_{\text{sm trans}}$, and $\delta_{\text{dec, comp}}$ are identical to the above case of classical B frames. Note however the hierarchical B frame dependency structure, which is indicated with arrows above the capture time axis, and the encoding order of the frames on the encode time axis, which results in minimal reordering delay for the display process [77]. Importantly, we note that due to the hierarchical dependencies between B frames, the reordering delay for achieving the display sequence depends on the number of temporal levels, i.e., $\delta_{\text{d}} = \log_2(\beta + 1)$. In summary,

$$\Delta_h = \beta + a + a + 1 + \log_2(\beta+1) = \beta + 2a + 1 + \log_2(\beta+1). \tag{8}$$

In Table VI, we summarize the delays for the H.264 SVC (*G16-B15*) and H.264/AVC (*G16-B3*) streams considered in this study. The end-to-end delays for the H.264 SVC traffic are 15 frame periods larger than for H.264/AVC, which is attributable to the hierarchical B frame prediction structure. In particular, with the *G16-B15* hierarchical B prediction structure, which results in improved RD performance, the encoder has to wait until the frame with time index 16 is captured before it can encode this frame as an I frame and start encoding all 15 preceding hierarchical B frames. In addition, the reordering delay increases to four frame periods with the considered RD efficient hierarchical B frame structure.

For scenarios with $\beta \leq 31$, as well as either $(\beta + 1)/a$ or $a/(\beta + 1)$ an integer, we adapt the preceding analysis as follows. With negligible encoding time, $\delta_{\text{enc, comp}} = 0$, the encoding dependency delay becomes $\delta_{\text{enc, dep}} = \max\{\beta, a-1\}$, similar to the case of classical B frames. For negligible decoding time, $\delta_{\text{dec, comp}} = 0$, the smoothed transmission and display reordering delay become together

$$\delta_{\text{sm trans}} + \delta_{\text{d}}$$
$$= \max \left\{ \left\lceil \frac{1 + \log_2(\beta + 1)}{a} \right\rceil a, \right.$$
$$1_{\{\beta \geq 7\}} \left(\left\lceil \frac{\frac{\beta+1}{2} + 1}{a} \right\rceil a - \frac{\beta + 1}{2} + 2 \right),$$
$$\left. 1_{\{\beta = 31, a \geq 8\}} \left(\left\lceil \frac{\frac{\beta+1}{4} + 1}{a} \right\rceil a - \frac{\beta + 1}{4} + 4 \right) \right\}.$$
$$\tag{9}$$

C. Prerecorded Video

For prerecorded video, all frames are preencoded, leaving only the smoothed transport, decoding, and display reordering delays, i.e., $\delta_{\text{enc, dep}} = \delta_{\text{enc, comp}} = 0$ and

$$\Delta = \delta_{\text{sm trans}} + \delta_{\text{dec, comp}} + \delta_{\text{d}}. \tag{10}$$

Effectively, for prerecorded video, the time index $t = 0$ is shifted to the beginning of the frame sequence on the transport axis in the illustrations in Figs. 9 and 10, e.g., to $t = 7$ in Fig. 9. Specifically, we obtain for classical B frames

$$\Delta_c^{pre} = a + 1 + 1 = a + 2, \tag{11}$$

and for hierarchical B frames

$$\Delta_h^{pre} = a + 1 + \log_2(\beta + 1). \tag{12}$$

Table VII gives the delays for prerecorded H.264/AVC (*G16-B3*) and H.264 SVC (*G16-B15*) streams. The end-to-end delays for the H.264 SVC traffic are three frame periods larger than for H.264/AVC, which is a smaller difference than for live video in Table VI.

The delays for optimal smoothing of prerecorded video with the (additional) startup delay of s frame periods (defined in [10]) are obtained by replacing a by $s + 1$ in (11) and (12). This is because optimal smoothing is designed to deliver the first frame within $s + 1$ frame periods to the decoder; and then ensure that for each subsequent frame period the next frame is available for decoding. For the examples in Table VII, the delay for optimally smoothed prerecorded traffic is one to fifteen frame periods smaller than for basic smoothed prerecorded traffic; optimal smoothing is however much more computationally demanding than basic smoothing.

ACKNOWLEDGMENT

REFERENCES

[1] H.-C. Huang, W.-H. Peng, T. Chiang, and H.-M. Hang, "Advances in the scalable amendment of H.264/AVC," *IEEE Communications Magazine*, vol. 45, no. 1, pp. 68–76, Jan. 2007.

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[3] M. Wien, H. Schwarz, and T. Oelbaum, "Performance analysis of SVC," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1194–1203, Sep. 2007.

[4] D. Marpe, T. Wiegand, and G. Sullivan, "The H.264/MPEG–4 advanced video coding standard and its applications," *IEEE Communications Magazine*, vol. 44, no. 8, pp. 134–143, Aug. 2006.

[5] G. Van der Auwera, P. T. David, and M. Reisslein, "Traffic characteristics of H.264/AVC variable bit rate video," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 164–174, Nov. 2008.

[6] T. Lakshman, A. Ortega, and A. Reibman, "VBR video: Tradeoffs and potentials," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 952–973, May 1998.

[7] A. R. Reibman and M. T. Sun, *Compressed Video over Networks*. New York: Marcel Dekker, 2000.

[8] D. Wu, Y. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha, "Streaming video over the internet: Approaches and directions," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 282–300, Mar. 2001.

[9] G. Van der Auwera, P. T. David, and M. Reisslein, "Traffic and quality characterization of single-layer video streams encoded with the H.264/MPEG–4 Advanced Video Coding standard and Scalable Video Coding extension," *IEEE Trans. Broadcasting*, vol. 54, no. 3, pp. 698–718, Sep. 2008.

[10] J. Salehi, Z.-L. Zhang, J. Kurose, and D. Towsley, "Supporting stored video: Reducing rate variability and end–to–end resource requirements through optimal smoothing," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 397–410, Aug. 1998.

[11] A. R. Reibman and A. W. Berger, "Traffic descriptors for VBR video teleconferencing over ATM networks," *IEEE/ACM Trans. Networking*, vol. 3, no. 3, pp. 329–339, Jun. 1995.

[12] G. Van der Auwera, M. Reisslein, and L. J. Karam, "Video texture and motion based modeling of rate variability-distortion (VD) curves," *IEEE Trans. Broadcasting*, vol. 53, no. 3, pp. 637–648, Sep. 2007.

[13] A. Ortega and K. Ramachandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 23–50, Nov. 1998.

[14] P. Seeling and M. Reisslein, "The rate variability-distortion (VD) curve of encoded video and its impact on statistical multiplexing," *IEEE Trans. Broadcasting*, vol. 51, no. 4, pp. 473–492, Dec. 2005.

[15] A. Alheraish, S. Alshebeili, and T. Alamri, "A GACS modeling approach for MPEG broadcast video," *IEEE Trans. Broadcasting*, vol. 50, no. 2, pp. 132–141, Jun. 2004.

[16] N. Ansari, H. Liu, Y. Q. Shi, and H. Zhao, "On modeling MPEG video traffics," *IEEE Trans. Broadcasting*, vol. 48, no. 4, pp. 337–347, Dec. 2002.

[17] D. P. Heyman and T. V. Lakshman, "Source models for VBR broadcast video traffic," *IEEE/ACM Trans. Networking*, vol. 4, no. 1, pp. 40–48, Jan. 1996.

[18] X.-D. Huang, Y.-H. Zhou, and R.-F. Zhang, "A multiscale model for MPEG-4 varied bit rate video traffic," *IEEE Trans. Broadcasting*, vol. 50, no. 3, pp. 323–334, Sep. 2004.

[19] M. M. Krunz and A. M. Makowski, "Modeling video traffic using $M/G/\infty$ input processes: A compromise between Markovian and LRD models," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 733–748, Jun. 1998.

[20] D. Marpe, T. Wiegand, and S. Gordon, "H.264/MPEG-4 AVC Fidelity Range Extensions: Tools, profiles, performance, and application areas," in *Proc. IEEE Int. Conf. on Image Proc. (ICIP)*, Sep. 2005, pp. 593–596.

[21] A. Undheim, Y. Lin, and P. Emstad, "Characterization of slice-based H.264/AVC encoded video traffic," in *Proceedings of Fourth European Conference on Universal Multiservice Networks (ECUMN)*, Feb. 2007, pp. 263–272.

[22] H.-H. Juan, H.-C. Huang, C. Huang, and T. Chiang, "Scalable video streaming over mobile WiMAX," in *Proceedings of IEEE Int. Symposium on Circuits and Systems (ISCAS)*, May 2007, pp. 3463–3466.

[23] P. Li, W. Lin, S. Rahardja, X. Lin, X. Yang, and Z. Li, "Geometrically determining the leaky bucket parameters for video streaming over constant bit-rate channels," *Signal Processing: Image Communication*, vol. 20, no. 2, pp. 193–204, Feb. 2005.

[24] D. T. Nguyen and J. Ostermann, "Congestion control for scalable video streaming using the scalability extension of H.264/AVC," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 2, pp. 246–253, Aug. 2007.

[25] T. Ozcelebi, A. Tekalp, and M. Civanlar, "Delay-distortion optimization for content-adaptive video streaming," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 826–836, Jun. 2007.

[26] M. van der Schaar, Y. Andreopoulos, and Z. Hu, "Optimized scalable video streaming over IEEE 802.11a/e HCCA wireless networks under delay constraints," *IEEE Trans. Mobile Computing*, vol. 5, no. 6, pp. 755–768, Jun. 2006.

[27] T. Schierl, K. Ganger, C. Hellge, T. Wiegand, and T. Stockhammer, "SVC-based multisource streaming for robust video transmission in mobile ad hoc networks," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 96–103, Oct. 2006.

[28] A. Puri, X. Chen, and A. Luthra, "Video coding using the H.264/MPEG-4 AVC compression standard," *Journal of Visual Communication and Image Representation*, vol. 19, no. 9, pp. 793–849, Oct. 2004.

[29] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: Tools, performance and complexity," *IEEE Circuits and Systems Magazine*, vol. 4, no. 1, pp. 7–28, First Quarter, 2004.

[30] G. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions," in *Proc. of SPIE 5558, Conference on Applications of Digital Image Processing XXVII, Special Session on Advances in New Emerging Standard: H.264/AVC I*, Denver, CO, Aug. 2004, pp. 454–474.

[31] *Information Technology–Generic Coding of Audio-Visual Objects–Part 2: Visual, Final Proposed Draft Amendment 1*, ISO/IEC JTC 1/SC 29/WG 11 N2802, , Geneva, Jul. 1999.

[32] P. Seeling, M. Reisslein, and B. Kulapala, "Network performance evaluation with frame size and quality traces of single-layer and two-layer video: A tutorial," *IEEE Communications Surveys and Tutorials* vol. 6, no. 3, pp. 58–78, Third Quarter, 2004 [Online]. Available: http://trace.eas.asu.edu, video traces available at

[33] S. Bakiras and V. O. K. Li, "Maximizing the number of users in an interactive video-on-demand system," *IEEE Trans. Broadcasting*, vol. 48, no. 4, pp. 281–292, Dec. 2002.

[34] P. Koutsakis and M. Paterakis, "Policing mechanisms for the transmission of videoconference traffic from MPEG-4 and H.263 video coders in wireless ATM networks," *IEEE Trans. Vehicular Technology*, vol. 53, no. 5, pp. 1525–1530, 2004.

[35] B. Nikolaus, J. Ott, C. Borrmann, and U. Borrmann, "Generalized greedy broadcasting for efficient media-on-demand transmissions," *IEEE Trans. Broadcasting*, vol. 51, no. 3, pp. 354–359, 2005.

[36] J. Roberts, "Internet traffic, QoS, and pricing," *Proceedings of the IEEE*, vol. 92, no. 9, pp. 1389–1399, 2004.

[37] Y. Xu and R. Guerin, "Individual QoS versus aggregate QoS: A loss performance study," *IEEE/ACM Trans. Networking*, vol. 13, no. 2, pp. 370–383, 2005.

[38] X.-D. Huang, Y.-H. Zhou, and R.-F. Zhang, "A multiscale model for MPEG-4 varied bit rate video traffic," *IEEE Trans. Broadcasting*, vol. 50, no. 3, pp. 323–334, Sep. 2004.

[39] C. H. Liew, C. K. Kodikara, and A. M. Kondoz, "MPEG-encoded variable bit-rate video traffic modelling," *IEE Proceedings Communications*, vol. 152, no. 5, pp. 749–756, Oct. 2005.

[40] U. K. Sarkar, S. Ramakrishnan, and D. Sarkar, "Modeling full-length video using Markov-modulated gamma-based framework," *IEEE/ACM Trans. Networking*, vol. 11, no. 4, pp. 638–649, Aug. 2003.

[41] U. K. Sarkar, S. Ramakrishnan, and D. Sarkar, "Study of long duration MPEG-trace segmentation methods for developing frame size based traffic models," *Computer Networks*, vol. 44, no. 2, pp. 177–188, 2004.

[42] M. Dai and D. Loguinov, "Analysis and modeling of MPEG-4 and H.264 multi-layer video traffic," in *Proc. of IEEE INFOCOM*, Miami, FL, Mar. 2005, pp. 2257–2267.

[43] D. Fiems, V. Inghelbrecht, B. Steyaert, and H. Bruneel, "Markovian characterization of H.264/SVC scalable video," in *Proceedings of 15th Int. Conference on Analytical and Stochastic Modeling Techniques and Applications (ASMTA)*, Jun. 2008, Lecture Notes in Computer Science 5055, pp. 1–15.

[44] S. Kempken and W. Luther, "Modeling of H.264 high definition video traffic using discrete-time semi-Markov processes," in *Proceedings of 20th Int. Teletraffic Congress (ITC)*, Jun. 2007, Lecture Notes in Computer Science 4516, pp. 42–53.

[45] C. Bewick, R. Pereira, and M. Merabti, "Network constrained smoothing: Enhanced multiplexing of MPEG-4 video," in *Proceedings of IEEE International Symposium on Computers and Communications*, Taormina, Italy, Jul. 2002, pp. 114–119.

[46] H.-C. Chao, C. L. Hung, and T. G. Tsuei, "ECVBA traffic-smoothing scheme for VBR media streams," *International Journal of Network Management*, vol. 12, pp. 179–185, 2002.

[47] W.-C. Feng and J. Rexford, "Performance evaluation of smoothing algorithms for transmitting prerecorded variable-bit-rate video," *IEEE Trans. Multimedia*, vol. 1, no. 3, pp. 302–312, Sep. 1999.

[48] T. Gan, K.-K. Ma, and L. Zhang, "Dual-plan bandwidth smoothing for layer-encoded video," *IEEE Trans. Multimedia*, vol. 7, no. 2, pp. 379–392, Apr. 2005.

[49] C.-D. Iskander and R. T. Mathiopoulos, "Online smoothing of VBR H.263 video for the CDMA2000 and IS-95B uplinks," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 647–658, Aug. 2004.

[50] M. Krunz, W. Zhao, and I. Matta, "Scheduling and bandwidth allocation for distribution of archived video in VoD systems," *Journal of Telecommunication Systems, Special Issue on Multimedia*, vol. 9, no. 3/4, pp. 335–355, Sep. 1998.

[51] M. Krunz, "Bandwidth allocation strategies for transporting variable–bit–rate video traffic," *IEEE Communications Magazine*, vol. 37, no. 1, pp. 40–46, Jan. 1999.

[52] H. Lai, J. Y. Lee, and L.-K. Chen, "A monotonic-decreasing rate scheduler for variable-bit-rate video streaming," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 221–231, Feb. 2005.

[53] A. Solleti and K. J. Christensen, "Efficient transmission of stored video for improved management of network bandwidth," *International Journal of Network Management*, vol. 10, pp. 277–288, 2000.

[54] B. Vandalore, W.-C. Feng, R. Jain, and S. Fahmy, "A survey of application layer techniques for adaptive streaming of multimedia," *Real-Time Imaging Journal*, vol. 7, no. 3, pp. 221–235, 2001.

[55] D. Ye, J. Barker, Z. Xiong, and W. Zhu, "Wavelet-based VBR video traffic smoothing," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 611–623, Aug. 2004.

[56] Z. Zhang, J. Kurose, J. Salehi, and D. Towsley, "Smoothing, statistical multiplexing and call admission control for stored video," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 1148–1166, Aug. 1997.

[57] Z. Antoniou and I. Stavrakakis, "An efficient deadline-credit-based transport scheme for prerecorded semisoft continuous media applications," *IEEE/ACM Trans. Networking*, vol. 10, no. 5, pp. 630–643, Oct. 2002.

[58] J. C. H. Yuen, E. Chan, and K.-Y. Lam, "Real time video frames allocation in mobile networks using cooperative pre-fetching," *Multimedia Tools and Applications*, vol. 32, no. 3, pp. 329–352, Mar. 2007.

[59] Y.-W. Leung and T. K. C. Chan, "Design of an interactive video-on-demand system," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 130–140, Mar. 2003.

[60] F. Li and I. Nikolaidis, "Trace-adaptive fragmentation for periodic broadcast of VBR video," in *Proceedings of 9th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, Basking Ridge, NJ, Jun. 1999, pp. 253–264.

[61] C.-S. Lin, M.-Y. Wu, and W. Shu, "Transmitting variable-bit-rate videos on clustered VOD systems," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, New York, Jul. 2000.

[62] S. Oh, Y. Huh, B. Kulapala, G. Konjevod, A. W. Richa, and M. Reisslein, "A modular algorithm-theoretic framework for the fair and efficient collaborative prefetching of continuous media," *IEEE Trans. Broadcasting*, vol. 51, no. 2, pp. 200–215, Jun. 2005.

[63] S. Oh, B. Kulapala, A. W. Richa, and M. Reisslein, "Continuous-time collaborative prefetching of continuous media," *IEEE Trans. Broadcasting*, vol. 54, no. 1, pp. 36–52, Mar. 2008.

[64] M. Reisslein and K. W. Ross, "High–performance prefetching protocols for VBR prerecorded video," *IEEE Network*, vol. 12, no. 6, pp. 46–55, Nov./Dec. 1998.

[65] S. Racz, T. Jakabfy, J. Farkas, and C. Antal, "Connection admission control for flow level QoS in bufferless models," in *Proc. IEEE INFOCOM*, 2005, pp. 1273–1282.

[66] M. Reisslein and K. W. Ross, "Call admission for prerecorded sources with packet loss," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 6, pp. 1167–1180, Aug. 1997.

[67] J. Roberts, U. Mocci, and J. Virtamo, *Broadband Network Traffic: Performance Evaluation and Design of Broadband Multiservice Networks, Final Report of Action COST 242*. New York: Springer Verlag, 1996, vol. 1155, Lecture Notes in Computer Science.

[68] A. Elwalid, D. Heyman, T. Lakshman, D. Mitra, and A. Weiss, "Fundamental bounds and approximations for ATM multiplexers with applications to video teleconferencing," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 1004–1016, Aug. 1995.

[69] F. Y.-S. Lin, "Optimal real-time admission control algorithms for the video-on-demand (VOD) service," *IEEE Trans. Broadcasting*, vol. 44, no. 4, pp. 402–408, Dec. 1998.

[70] N. Shroff and M. Schwartz, "Improved loss calculations at an ATM multiplexer," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 411–421, Aug. 1998.

[71] J. McManus and K. Ross, "Video-on-demand over ATM: Constant-rate transmission and transport," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 6, pp. 1087–1098, Aug. 1996.

[72] S. Sen, J. L. Rexford, J. K. Dey, J. F. Kurose, and D. F. Towsley, "On-line smoothing of variable-bit-rate streaming video," *IEEE Trans. Multimedia*, vol. 2, no. 1, pp. 37–48, Mar. 2000.

[73] Y. Bai and M. Ito, "Application-aware buffer management: New metrics and techniques," *IEEE Trans. Broadcasting*, vol. 51, no. 1, pp. 114–121, Mar. 2005.

[74] Y. Huang, R. Guerin, and P. Gupta, "Supporting excess real-time traffic with active drop queue," *IEEE Trans. Networking*, vol. 14, no. 5, pp. 965–977, Oct. 2006.

[75] G.-M. Muntean, P. Perry, and L. Murphy, "A new adaptive multimedia streaming system for all-IP multi-service networks," *IEEE Trans. Broadcasting*, vol. 50, no. 1, pp. 1–10, Mar. 2004.

[76] S. Ryu, C. Rump, and C. Qiao, "Advances in internet congestion control," *IEEE Communications Surveys and Tutorials*, vol. 5, no. 1, pp. 28–39, 2003.

[77] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, Toronto, Canada, Jul. 2006, pp. 1929–1932.

Geert Van der Auwera received the Ph.D. degree in Electrical Engineering from Arizona State University, Tempe, USA, in 2007, and the Belgian MSEE degree from Vrije Universiteit Brussel (VUB), Brussels, Belgium, in 1997. Presently, he is Staff Research Engineer with Samsung Electronics in Irvine, CA. His research interests are video coding, video traffic and quality characterization, video streaming mechanisms and protocols. Until the end of 2004, he was Scientific Advisor with IWT-Flanders, the Institute for the Promotion of Innovation by Science and Technology in Flanders, Belgium. In 2000, he joined IWT-Flanders after researching wavelet video coding at IMEC's Electronics and Information Processing Department (VUB-ETRO) in Brussels, Belgium. In 1998, his thesis on motion estimation in the wavelet domain received the Barco and IBM prizes by the Fund for Scientific Research of Flanders, Belgium.

Martin Reisslein is an Associate Professor in the Department of Electrical Engineering at Arizona State University (ASU), Tempe. He received the Dipl.-Ing. (FH) degree from the Fachhochschule Dieburg, Germany, in 1994, and the M.S.E. degree from the University of Pennsylvania, Philadelphia, in 1996. Both in electrical engineering. He received his Ph.D. in systems engineering from the University of Pennsylvania in 1998. During the academic year 1994-1995 he visited the University of Pennsylvania as a Fulbright scholar. From July 1998 through October 2000 he was a scientist with the German National Research Center for Information Technology (GMD FOKUS), Berlin and lecturer at the Technical University Berlin. From October 2000 through August 2005 he was an Assistant Professor at ASU.

He served as editor-in-chief of the IEEE Communications Surveys and Tutorials from January 2003 through February 2007 and has served on the Technical Program Committees of IEEE Infocom and numerous other networking conferences. He maintains an extensive library of video traces for network performance evaluation, including frame size traces of MPEG-4 and H.264 encoded video, at http://trace.eas.asu.edu. His research interests are in the areas of Internet Quality of Service, video traffic characterization, wireless networking, optical networking, and engineering education.