The Effects of Priority Levels and Buffering on the Statistical Multiplexing of Single-Layer H.264/AVC and SVC Encoded Video Streams

Sudhir Kumar Srinivasan, Jonathan Vahabzadeh-Hagh, and Martin Reisslein

***Abstract*—H.264/Advanced Video Coding (AVC) employs classical bi-directional encoded (B) frames that depend only on intra-coded (I) and predictive encoded (P) frames. In contrast, H.264 Scalable Video Coding (SVC) employs hierarchical B frames that depend on other B frames. A fundamental question is how many priority levels single-layer H.264 video encodings require when the encoded frames are statistically multiplexed in transport networks. We conduct extensive simulation experiments with a modular statistical multiplexing structure to uncover the impact of priority levels for a wide range of multiplexing policies. For the bufferless statistical multiplexing of both H.264/AVC and SVC we find that prioritizing the frames according to the number of dependent frames can increase the number of supported streams up to approximately 8%. In contrast, for buffered statistical multiplexing with a relatively small buffer size, frame prioritization does generally not increase the number of supported streams.**

***Index Terms*—Frame dependencies, H.264/AVC, H.264 SVC, multiplexing policy, statistical multiplexing.**

I. Introduction

The advanced coding mechanisms in H.264/Advanced Video Coding (AVC) achieve higher rate-distortion (RD) efficiency compared to earlier MPEG video coding, while the additional enhancements in H.264 Scalable Video Coding (SVC) further improve the RD efficiency over H.264/AVC [1]–[4]. H.264/AVC employs by default the classical prediction structure for bi-directional encoded (B) frames, whereby B frames are encoded with bi-directional predictive encoding from intra-coded (I) frames and forward-predictive encoded (P) frames. With the classical B frame prediction structure, B frames are not predictive encoded from other B frames. In contrast, H.264 SVC employs a hierarchical B frame prediction structure where some B frames are bi-directionally predictive encoded from other B frames according to a B frame hierarchy, as detailed in Section III. Whereas a loss of a B frame during network transport does not affect other frames in an H.264/AVC encoding, the loss of a B frame in an H.264 SVC encoding may hinder the decoding of other dependent B frames that are predictive encoded from the lost B frame. Generally, during network transport, video frames with many dependent frames may be transmitted with higher priority to increase the chances of their intact delivery. Since the frame dependency structures with classical and hierarchical B frame prediction are fundamentally different, it is important to investigate how many priority levels are needed for efficient network transport.

In this study, we consider a wide range of elementary statistical multiplexing policies. We evaluate the maximum number of video streams that can be supported with given link capacities for a prescribed limit on the fraction of lost video encoding bits. Throughout, we consider single-layer (non-scalable) encodings with fixed quantization scales. The resulting video encodings have 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 statistical multiplexing characteristics of the H.264 SVC and H.264/AVC video encodings, whose standards do not specify a normative rate control mechanism. Additionally, the statistical multiplexing gains achieved with variable bit rate streams improve the efficiency of video network transport [5].

We find for both encodings with classical B frames and with hierarchical B frames that more priority levels increase the number of supported streams in a bufferless statistical multiplexer. On the other hand, the number of supported streams is not increased by more priority levels in a buffered statistical multiplexer. We also find that for both classical and hierarchical B frames, a small multiplexer buffer significantly increases (in many scenarios doubles) the number of supported streams compared to bufferless statistical multiplexing.

II. Related Work

In this section we briefly review related work on multiplexing and prioritization during the network transport of video encoded with classical and hierarchical B frames. The network transport of H.264 encoded video has received significant attention recently, whereby a focus has been on exploiting the SVC scalability features to adapt to specific layers of the network protocol stack. For instance, adaptations for the transport layer using bandwidth estimation and congestion control mechanisms have been explored in [6]–[9]. Adaptations to the wireless channel through intelligent scheduling policies have

Manuscript received February 09, 2010; revised April 12, 2010; accepted April 14, 2010. Date of publication May 27, 2010; date of current version August 20, 2010. This work was supported in part by the National Science Foundation under Grant CRI-0750927.

S. K. Srinivasan is with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85287 USA (e-mail: sudhir.srinivasan@asu.edu).

J. Vahabzadeh-Hagh and M. Reisslein are with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ 85287-5706 USA (e-mail: jonathan.vahabzadeh@asu.edu; reisslein@asu.edu).

Digital Object Identifier 10.1109/TBC.2010.2049610

been studied in [10]–[15]. Network coding techniques that divide the video packets into separate channels and apply unequal error protection have, for instance, been examined in [16], [17]. Traffic splitting techniques based on SVC layer information and dynamic frame-priority based dropping techniques have also received interest, see for instance [18]–[22]. Complementary to these existing studies, we examine the fundamental statistical multiplexing behaviors of both video encoded with classical and hierarchical B frames with a varying number of priority levels.

While many studies on video network transport neglect the frame dependencies due to the predictive encoding, e.g., [1], [23]–[25], several other studies have explored issues surrounding the frame dependencies. For instance, a rate shaping method for streaming of H.263 with consideration of frame dependencies is developed in [26]. A packet scheduling scheme for layered MPEG-4 video which considers the frame type is developed in [27]. New GoP structures that reduce inter-frame dependency coding overheads in H.264/AVC are proposed in [28]. In contrast to the existing studies, we investigate the impact of the frame encoding dependencies of both classical and hierarchical B frames on the statistical multiplexing performance.

Statistical multiplexing of encoded video streams can be conducted with or without coordinating the encoders of the multiplexed streams. The studies [29], [30] explore statistical multiplexing with coordinated encoders whereby the video qualities (encoding quantization parameters) of the individual streams are adapted such that the aggregate video traffic fits into the available network bandwidth. In contrast, we study statistical multiplexing without encoder coordination, where the encoding parameters are kept constant. A section of the study [30] explores multiplexing without encoder coordination, whereby P-frames are dropped randomly. The study [11] examined statistical multiplexing with a fixed number of priority levels without encoder coordination for two 300-frame video test sequences in the context of an 801.11e wireless network. In contrast, in this study, we examine statistical multiplexing without encoder coordination with five long (over 15 000 frames) video sequences for different numbers of priority levels for general bufferless and buffered multiplexing systems.

III. Frame Dependencies With Classical and Hierarchical B Frames

In this section we give an overview of the dependencies of video frames encoded with classical B frames (used by default in H.264/AVC) and hierarchical B frames (used in H.264 SVC).

With classical B frame encoding, the frames in the example depicted in Fig. 1 are encoded in the order *IPBBBPBBBPBBBIBBB*. The first I frame is used for the predictive encoding of all the P and B frames of the depicted GoP. In addition, the I frame on the right in Fig. 1 is used for the prediction of the three rightmost B frames (and hence is encoded before these three B frames). Generally, in a GoP with a total of g frames, and with b B frames between successive I and P frames, a given I frame is used as a prediction reference for the b B frames preceding the I frame in the display order as well as the $g-1$ P and B frames succeeding the I frame in the display order. Thus, the loss of an I frame results in the loss of all these $b+g-1$ dependent frames.

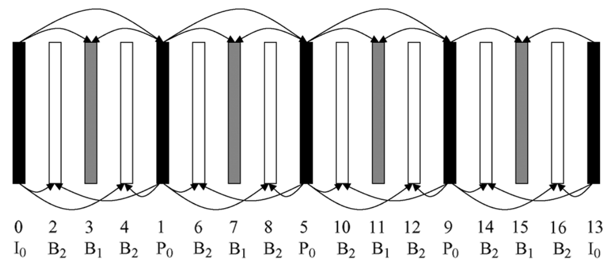

Fig. 1. Illustration of frame dependencies with classical B frames for an example with $g = 16$ frames in a GoP and $b = 3$ B frames between successive I and P frames. The frames are depicted in the display order. The encoding order is given by the top row of numbers. and the bottom row indicates the frame type in the display order.

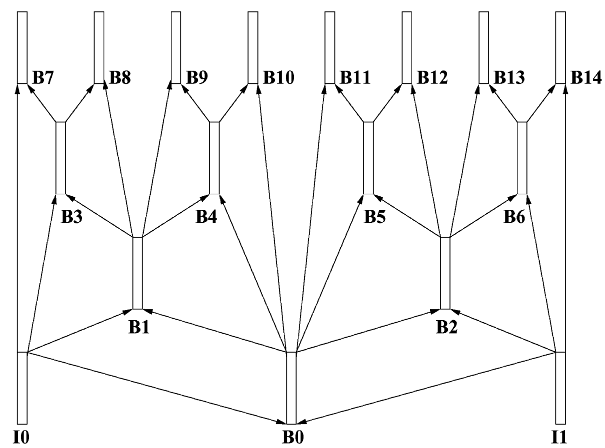

Fig. 2. Hierarchical B frames in encoding order with GoP structure with 15 B frames and no P frames.

A given P frame is used for the (backward) predictive encoding of the immediately preceding b B frames as well as the (forward) predictive encoding of all the following P and B frames in the GoP. For example, the loss of the middle P frame in Fig. 1 affects a total of one other P frame and nine B frames. Finally, with classical B frame encoding, B frames are not used for the predictive encoding of other B frames. Thus, the loss of a B frame does not have any impact on other frames.

With hierarchical B frame encoding [31], B frames are used for the predictive encoding of other B frames resulting in a dyadic hierarchy of B frames. For the example GoP structure with 15 B frames and no P frames, we can represent the hierarchical B frame dependencies in the form of a tree, as illustrated in Fig. 2. The B0 frame in the middle of the GoP forms the root of the tree, i.e., all other B frames in the GoP depend on this B0 frame. Generally, a given B frame is used for the predictive encoding of all its dependent frames in the tree structure. For instance, a loss of frame B4 affects frames B9 and B10.

IV. Statistical Multiplexing System

In order to obtain fundamental insights into the statistical multiplexing behavior we consider a modular generic multiplexing system consisting of a drop module, a priority module, a multiplexing module, and a receiver module. Throughout the multiplexing system, time is slotted with one time slot equal to the duration of one video frame period T, which is $T = 1/30$ s for the NTSC frame rate of 30 frames/s. At the beginning of each frame period (time slot), each of the J multiplexed streams presents one frame to the multiplexing system. The drop module may instantaneously drop video frames that depend on frames that have been lost. The priority module instantaneously orders the J frames and instantaneously places them in the established order into the buffer of the multiplexing module. The priority module does not store any video frames. Any frame (or part of a frame) that does not fit into the multiplexing buffer is lost. The multiplexing module transmits the frames from the buffer onto the channel.

A. Video Stream Model

Each video stream j, $j = 1, \ldots, J$, is characterized by a sequence of frame sizes $X_n(j)$ [bit] with n, $n = 1, \ldots, N$, denoting the frame number. Note that the bit rate required to transmit frame n of stream j during one frame period of length T is $X_n(j)/T$. Note further that the average frame size is $X_{\mathrm{avg}}(j) = (1/N)\sum_{n=1}^{N} X_n(j)$. To model the random starting times (offsets) of the ongoing J video streams, we let $\theta_m(j)$, $j = 1, \ldots, J$, be random variables denoting the frame numbers of the J streams that are transmitted during a given frame period (time slot) m. In each frame period, each video stream feeds its next video frame into the multiplexing system. More specifically, for a given multiplexing experiment, we stream J identical video sequences in encoding order, whereby the starting phase for each stream is randomly selected according to a uniform distribution over all M frames of the sequence [32], [33]. The streams are wrapped around to obtain streams of equal lengths.

B. Drop Module

We consider *(i)* a no-drop policy (ND), which does not drop any frames, not even a frame that depends on a lost frame, and *(ii)* a drop policy (DP), which drops frames that are predictive encoded with respect to a frame that has been lost, i.e., did not fit fully into the multiplexing buffer.

C. Priority Module

The priority module instantaneously orders the J frames of the ongoing streams and feeds them into the multiplexing module.

1) No Priorities (NP): With the NP policy, the J frames are fed in random order into the multiplexing module.

2) Frame Type Priority (TP): The TP policy orders the frames such that I frames are placed first, i.e., with highest priority in the multiplexer buffer, followed by the P frames, which in turn are followed by the B frames. Within a given frame type, the frames are ordered randomly.

3) Full Priority (FP): The FP policy arranges the frames in decreasing order of the number of dependent frames, i.e., the frames with more dependent frames enter the multiplexer buffer before frames with fewer dependent frames. In particular, for the classical B frame encoding example with $g = 16$ frames in the GoP and $b = 3$ B frames between I and P frames (see Fig. 1), the I frames have top priority, followed by the P frames that are first in their respective GoPs, followed by the P frames that are in the middle of the GoP, followed by the P frames that are last in the GoP, followed by the B frames. For the hierarchical B frame example with $b = 15$ B frames (see Fig. 2), I frames have top priority, followed by the "middle" B frame B0, followed by the B1 and B2 frames, followed by the B3, B4, B5, and B6 frames, followed by the remaining B frames.

D. Multiplexing Module

The multiplexing module transmits the frames in its buffer in first-come-first-served manner onto the channel with capacity C [bit/s], which can drain CT [bit] from the multiplexer buffer in one video frame period (time slot) of duration T. Note that the stability limit, i.e., the absolute maximum number of streams of video j that the channel can support is given by $CT/X_{\mathrm{avg}}(j)$.

1) Bufferless Multiplexing: We first consider a "bufferless" multiplexer [33]–[36], which has no buffer beyond the CT [bit] buffer space needed to hold the bits transmitted in one time slot.

If the drop module does not drop any frames (ND), then the aggregated bit rate in time slot m for the statistically multiplexed J streams is given by

$$R_m = \sum_{j=1}^{J} \frac{X_{\theta_m(j)}(j)}{T}. \quad (1)$$

If the aggregate bit rate R_m exceeds the link capacity C, then loss occurs at the bufferless multiplexer, which we measure as the information loss probability [33], [36], i.e., the long-run fraction of lost video bits

$$P_{\mathrm{loss}}^{\mathrm{info}} = \frac{E\left[(R_m - C)^+ \times T\right]}{E[R_m \times T]} = \frac{E\left[(R_m - C)^+\right]}{E[R_m]}, \quad (2)$$

where $[x]^+ = \max(0, x)$. If the drop module drops frames (DP), then we define the information loss probability $P_{\mathrm{loss}}^{\mathrm{info}}$ as the long-run ratio of the number of bits dropped in the drop module plus the number of bits lost due to buffer overflow to the total aggregate number of bits entering the multiplexing system.

2) Buffered Multiplexing: With bufferless multiplexing, i.e., a multiplexer buffer of size CT [bit], the multiplexer buffer is completely emptied at the end of each time slot. If the multiplexer buffer is larger than CT, the multiplexer buffer may not completely drain by the end of a time slot, carrying bits over from one time slot to the next. We refer to the multiplexing with a buffer size larger than CT as *buffered multiplexing*.

If the drop module does not drop any frames, R_m given in (1) denotes the aggregate bit rate [in bit/s] of the J ongoing video streams in time slot m entering the multiplexing module. Further, let b_{m-1} denote 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 note that traffic is served at bit rate C. Then, the amount of buffered video traffic at the end of frame period m is obtained as

$$b_m = \min\left\{\left[b_{m-1} + (R_m - C)T\right]^+, B\right\}, \quad (3)$$

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 $P_{\mathrm{loss}}^{\mathrm{info}}$. If the drop module drops video frames, then both the dropped frames and the video frame bits lost due to buffer overflow are included in the evaluation of $P_{\mathrm{loss}}^{\mathrm{info}}$.

TABLE I
$J_{\max}$ VALUES FOR *Sony Demo* WITH $C = 20$ Mbps

QP	PSNR	bufferless mux.							buffered mux., $B = 192$ kByte							Stab.
	[dB]	DD NP	DD TP	DD FP	ND-NP RD	ND-FP RD	ND-NP RC	ND-FP RC	DD NP	DD TP	DD FP	ND-NP RD	ND-FP RD	ND-NP RC	ND-FP RC	lim. $\frac{CT}{X_{\text{avg}}}$
H.264 SVC, $\epsilon = 10^{-5}$																
32	37.8	4	4	4	4	4	5	5	49	49	49	48	50	50	50	63
38	34.0	22	22	22	22	22	24	24	108	108	108	108	108	108	108	123
42	31.6	59	59	59	58	59	63	64	178	178	178	178	178	179	179	196
H.264/AVC, $\epsilon = 10^{-5}$																
28	37.3	6	6	6	6	6	8	8	41	41	41	41	41	41	41	52
34	33.2	30	30	30	29	30	33	34	99	99	99	99	99	99	99	114
38	30.7	77	77	78	76	78	81	81	177	177	177	177	177	177	177	197
H.264 SVC, $\epsilon = 10^{-3}$																
32	37.8	13	13	13	13	13	15	15	56	56	56	56	57	56	57	63
38	34.0	45	45	45	44	45	47	47	115	115	115	115	115	115	115	123
42	31.6	95	95	95	95	95	98	99	187	187	187	187	187	187	187	196
H.264/AVC, $\epsilon = 10^{-3}$																
28	37.3	15	15	15	15	15	15	15	45	45	45	45	45	45	45	52
34	33.2	50	51	52	49	51	54	54	106	106	106	106	106	106	106	114
38	30.7	111	113	115	111	114	119	119	187	187	187	187	187	187	187	197

E. Receiver Module

For streams transmitted with a no-drop policy (ND), see Section IV-B, frames that are predictive encoded with respect to a frame that had been lost in the multiplexing module may arrive at the receiver. For this fundamental evaluation, we consider two extreme types of receivers: a receiver that does not perform error concealment and drops such frames whose reference frames have not been received (denoted by RD), and a receiver that performs perfect error concealment and displays frames whose reference frames have not been received (RC). More specifically, for ND transmission to an RD receiver, both the video frame bits lost in the multiplexer and the video frames dropped in the receiver module are included in the evaluation of $P_{\text{loss}}^{\text{info}}$. On the other hand, for ND transmission to an RC receiver, there are no frame losses at the receiver and only the video frame bits lost in the multiplexer are included in the evaluation of $P_{\text{loss}}^{\text{info}}$.

V. STATISTICAL MULTIPLEXING EVALUATION

A. Evaluation Set-Up

1) Video Sequences: We consider the same five CIF resolution (352 × 288 pixel) 30 frames/s video sequences as used in [1], [23]; namely, the *Sony Digital Video Camera Recorder Demo* sequence with 17 682 frames, the first half hour of *Silence of the Lambs* with 54 000 frames, the first half hour of *Star Wars IV* with 54 000 frames, approximately 30 minutes of *NBC 12 News* with 49 523 frames, and the first hour of *Tokyo Olympics* with 133 128 frames.

2) Video Encoding Set-Up: We employ the same H.264/AVC and H.264 SVC encoders and settings as in [1], [23]. In summary, we employ the H.264/AVC encoder [37] in the Main profile with all compression tools enabled, including variable block sizes, three reference frames for the past and the future, Context Adaptive Binary Arithmetic Coding (CABAC), and Lagrangian based rate-distortion optimization (RDO). For H.264/AVC, we employ classical B frame prediction with the GoP structure *IBBBPBBBPBBBPBBB* (16 frames, with 3 B frames per I/P frame) denoted by *G16-B3*, which was found to achieve very good rate-distortion (RD) efficiency for H.264/AVC in [1].

We employ the H.264 SVC encoder [31] with a dyadic B frame hierarchy, whereby the number of B frames β between successive key pictures (I or P frames) is $\beta = 2^k - 1$ for an integer number k, $k \geq 0$. For the H.264 SVC encodings (hierarchical B frames), we employ the GoP structure *IBBBBBBBBBBBBBBB* (16 frames, with 15 B frames per I frame) denoted by *G16-B15*, which gave very good RD efficiency in [1].

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).

3) Multiplexer Set-Up: From among the wide range of buffer management and scheduling policies, see e.g. [38]–[41], we consider the elementary taildrop policy with first-come-first-served scheduling, to assess the fundamental impact of the multiplexer buffer. For the buffered multiplexing experiments, we set the buffer capacity to $B = 192$ kByte, which was identified as the upper end of a recommended buffer size range for multiplexing H.264 SVC encoded video in [23].

4) Simulation Structure: For a given link bit rate C and given video encoding, we determine the maximum number of streams $J_{\max}$ that can be simultaneously supported while meeting the constraint that the information loss probability is less than a small constant ϵ. For each simulation for a given number of streams J we run many independent replications, each with a new independent random set of offsets $\theta_m(j)$ (see Section IV-A) until the 90% confidence interval of the information loss probability is less than 10% of the corresponding sample mean.

B. Simulation Results

In this section, we present illustrative results from simulations for the five video sequences. We refer to [42] for the full set of results, which we can not include here due to space constraints. We present in Table I results for *Sony Demo* for the full range of considered multiplexing policies. We present illustrative sample results for the other four video sequences for the multiplexing

TABLE II
$J_{\max}$ VALUES FOR H.264 ENCODINGS FOR $C = 20$ Mbps AND $\epsilon = 10^{-5}$

Video	Enc.	QP	PSNR [dB]	bufferless mux. DD NP	DD TP	DD FP	ND-NP RC	buffered mux., $B = 192$ kByte DD NP	DD TP	DD FP	ND-NP RC	Stab. lim. $\frac{CT}{X_{\text{avg}}}$
Star Wars	SVC	34	40.3	80	80	83	84	170	170	170	170	191
Star Wars	AVC	28	41.0	58	62	63	63	106	106	106	107	128
Olympics	SVC	42	33.8	120	122	129	131	226	227	227	227	254
Olympics	AVC	34	35.6	59	65	66	66	112	113	113	113	138
NBC News	SVC	42	33.8	77	77	78	81	168	168	168	168	183
NBC News	AVC	34	33.1	40	43	43	44	86	86	86	86	101
Silence	SVC	42	36.0	240	247	251	253	426	426	426	426	477
Silence	AVC	34	37.6	128	128	138	139	230	231	231	232	290

policies with dependency drop (DD) and the ND-NP-RC policy in Table II. We observe from these tables that the number of priority levels has a relatively small effect on the maximum number of supported streams $J_{\max}$ for bufferless multiplexing. More specifically, for a small number of multiplexed streams, all multiplexing policies and numbers of priority levels give the same $J_{\max}$. For scenarios with moderate to large numbers of multiplexed streams, more priority levels can slightly increase $J_{\max}$. However, even with full priority, which requires five priority levels for the considered AVC and SVC encodings, the increase in $J_{\max}$ is typically less than 6–8% of the $J_{\max}$ achieved with one priority level (NP). The largest increase in our extensive experiments [42] was 12% and occurred for *Olympics* for the AVC encoding with $\mathrm{QP} = 34$, which is included in Table II. On the other hand, for buffered multiplexing there are generally no increases in $J_{\max}$ with increasing number of priority levels (except for a few instances of one added stream in Table II).

Turning to the comparison of the different multiplexing policies for bufferless multiplexing, we observe from Table I that the ND-NP-RD policy has a slight tendency to support a smaller $J_{\max}$ compared to the DD-NP policy. This is because frames whose reference frames have already been lost may still be transmitted with the ND-NP-RD policy, consuming bandwidth and thus increasing the chance that other frames are dropped. The DD-NP policy avoids this waste of bandwidth. However, the DD-NP policy requires that information about frames dropped in the multiplexing module is fed back to the drop module. The ND-FP-RD policy largely overcomes the drawback of the ND-NP-RD policy, achieving almost the same $J_{\max}$ as the DD-FP policy.

We further observe from Table I for bufferless multiplexing that the policies with error concealment at the receiver (RC) achieve slightly higher $J_{\max}$ than the policies with frame dropping at the drop module (DD) or receiver (RD). (Only the ND-NP-RC policy is considered in [23].) With the RC policies, only the losses due to buffer overflow in the multiplexing module are considered; the RC policies do not consider any losses due to frame encoding dependencies. For bufferless multiplexing, neglecting the frame encoding dependencies with the RC policy typically leads to $J_{\max}$ values that exceed the $J_{\max}$ for the other policies by no more than around 10% when the number of multiplexed streams is moderately large. Interestingly, with buffered multiplexing there are generally no differences between considering frame dependencies (in the DD and RD policies) and neglecting frame dependencies (with the RC policies).

Comparing bufferless with buffered multiplexing, we observe for both H.264/AVC and SVC that the relatively modest buffer of $B = 192$ kByte, which is less than three times the buffer space $CT = 83.3$ kByte of the bufferless multiplexer, significantly increases the number of supported streams $J_{\max}$. For relatively small to modest numbers of multiplexed streams, buffered multiplexing increases $J_{\max}$ by a factor of two, three, or larger in some instances; whereas $J_{\max}$ is increased by a factor of roughly 1.5 for large numbers of multiplexed streams. With buffered multiplexing, the number of supported streams comes typically within about 20% percent of the stability limit.

Examining the impact of the bit loss criterion ϵ, we observe that ϵ has a rather significant impact on the number of multiplexed streams for bufferless multiplexing, whereas the effect for buffered multiplexing is relatively weak. In additional experiments, we have examined the relationship between the information loss probability, which is required to be less than a prescribed ϵ, and the resulting reduction in the average video frame PSNR (in dB) using the offset distortion approach [43]. The offset distortion approach corresponds to frame-based receiver error concealment that replaces a frame with some lost bits or missing reference frame by the preceding completely received frame. We found that for $\epsilon = 10^{-5}$ the reduction in PSNR is less than 0.02 dB, for $\epsilon = 10^{-3}$ typically less than 0.06 dB, and for $\epsilon = 10^{-2}$ about 0.5 dB.

VI. Conclusions and Future Work

We have conducted an extensive simulation study of statistical multiplexing of single-layer variable bit rate video encoded with H.264 with classical B frames (AVC) and hierarchical B frames (SVC) with different numbers of priority levels. For both classical and hierarchical B frames we have found that the increases in the numbers of supported streams achieved by introducing different priority levels for frames with different numbers of dependent frames are relatively small. For bufferless multiplexing the number of supported streams was typically increased by up to 8% (in some instances up to 12%). For buffered multiplexing, added priority levels did not increase the number of supported streams. On the other hand, buffered multiplexing achieved substantially higher numbers of supported streams than bufferless multiplexing.

One interesting direction for future research is to examine improvements to buffered multiplexing through active buffer management policies [41], i.e., to manipulate the multiplexer buffer contents after the video frames have been placed in the multiplexer buffer.

REFERENCES

[1] G. Auwera, P. 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. Broadcast.*, vol. 54, no. 3, pp. 698–718, Sep. 2008.

[2] M. Pinson, S. Wolf, and G. Cermak, "HDTV subjective quality of H.264 vs. MPEG-2, with and without packet loss," *IEEE Trans. Broadcast.*, vol. 56, no. 1, Jun. 2010.

[3] F. Speranza, A. Vincent, and R. Renaud, "Bit-rate efficiency of H.264 encoders measured with subjective assessment techniques," *IEEE Trans. Broadcast.*, vol. 55, no. 4, pp. 776–780, Dec. 2009.

[4] T. Wiegand, L. Noblet, and F. Rovati, "Scalable video coding for IPTV services," *IEEE Trans. Broadcast.*, vol. 55, no. 2, pp. 527–538, Jun. 2009.

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

[6] L. Bjornar and K. Oivind, "Congestion control for scalable VBR video with packet pair assistance," in *Proc. Int. Conf. Comput. Commun. and Netw.*, St. Thomas, US Virgin Islands, Aug. 2008.

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

[8] W. Sheng and H. Hsu-Feng, "TCP-friendly congestion control for layered video streaming using end-to-end bandwidth inference," in *Proc. Int. Workshop Multimedia Signal Process.*, Cairns, Qld, Australia, Oct. 2008, pp. 462–467.

[9] X. Yang and L. Lei, "End-to-end congestion control for H.264/SVC," in *Proc.Int. Conf. Netw.*, Martinique, France, Apr. 2007, pp. 497–502.

[10] C. W. Chan, N. Bambos, S. Wee, and J. Apostolopoulos, "Wireless video broadcasting to diverse users," in *Proc. Int. Conf. Commun.*, Beijing, China, May 2008.

[11] E. M. A. Fiandrotti, D. Gallucci, and E. Magli, "Traffic prioritization of H.264/SVC video over 802.11e ad hoc wireless networks," in *Proc. IEEE Int. Conf. Comput. Commun. and Netw.*, St. Thomas, US Virgin Islands, Aug. 2008.

[12] R. Haenens, J. Doggen, D. Bakker, and T. Dams, "Transmitting scalable video with unequal error protection over 802.11b/g," in *Proc. IEEE Int. Conf. Wireless and Mobile Comput., Netw. and Commun.*, Avignon, France, Oct. 2008, pp. 638–643.

[13] 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 Comput.*, vol. 5, no. 6, pp. 755–768, Jun. 2006.

[14] T. Schierl, C. Hellge, S. Mirta, K. Grüneberg, and T. Wiegand, "Using H.264/AVC-based scalable video coding (SVC) for real time streaming in wireless IP networks," in *Proc. IEEE Int. Symp. Circuits Syst.*, New Orleans, LA, May 2007.

[15] M. Shoaib and M. Waheed, "Streaming video in cellular networks using scalable video coding extension of H.264-AVC," in *Proc. IEEE Int. Conf. Wireless Commun., Netw. and Mobile Comput.*, Dalian, China, Oct. 2008.

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

[17] H. Wang and C. J. Kuo, "Apply network coding for H.264/SVC multicasting," in *Proc. SPIE—Int. Soc. Opt. Eng.*, San Diego, CA, Aug. 2008, vol. 7073.

[18] D. Bakker, D. Cromboom, T. Dams, A. Munteanu, and J. Barbarien, "Priority-based error protection for the scalable extension of H.264/SVC," in *Proc. SPIE - Int. Soc. Opt. Eng.*, Strasbourg, France, Apr. 2008.

[19] L. Chen and G. Liu, "A delivery system for scalable video streaming using the scalability extension of H.264/AVC over diffserv networks," in *Int. Conf. Intelligent Inf. Hiding and Multimedia Signal Process.*, Harbin, China, Aug. 2008.

[20] J. Chiang, H. Lo, and W. Lee, "Scalable video coding of H.264/AVC video streaming with QoS-based active dropping in 802.16e networks," in *Proc. Int. Conf. Adv. Inf. Netw. and Appl.*, Okinawa, Japan, Mar. 2008.

[21] J. Monteiro, C. Calafate, and M. Nunes, "Evaluation of the H.264 scalable video coding in error prone IP networks," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 652–659, Sep. 2008.

[22] Y. Xiaogang, L. Jiqiang, and L. Ning, "Congestion control based on priority drop for H.264/SVC," in *Proc. Int. Conf. Multimedia Ubiquitous Eng.*, Seoul, South Korea, Apr. 2007.

[23] G. Van der Auwera and M. Reisslein, "Implications of smoothing on statistical multiplexing of H.264/AVC and SVC video streams," *IEEE Trans. Broadcast.*, vol. 55, no. 3, pp. 541–558, Sep. 2009.

[24] A. Lazaris and P. Koutsakis, "Modeling video traffic from multiplexed H.264 videoconference streams," in *Proc. IEEE Globecom*, Dec. 2008, pp. 1–6.

[25] C.-H. Hsu and M. Hefeeda, "On statistical multiplexing of variable-bit-rate video streams in mobile systems," in *Proc. ACM Multimedia*, Jan. 2009, pp. 411–420.

[26] T. P. Chen and T. Chen, "Rate shaping for video with frame dependency," in *Proc. IEEE Int. Conf. Multimedia Expo*, Baltimore, MD, 2003, vol. 1, pp. 201–204.

[27] S. Hong and Y. Won, "SAPS: Significance-aware packet scheduling for real-time streaming of layer encoded video," in *Proc. 10th IEEE Int. Symp. Multimedia*, Baltimore, MD, Dec. 2008, pp. 521–526.

[28] J. Loua, S. Liub, A. Vetro, and M. Suna, "Complexity and memory efficient GoP structures supporting VCR functionalities in H.264/AVC," in *Proc. IEEE Int. Symp. Circuits Syst.*, Seattle, WA, May 2008, pp. 636–639.

[29] M. Jacobs, J. Barbarien, S. Tondeur, R. V. de Walle, T. Paridaens, and P. Schelkens, "Statistical multiplexing using SVC," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast.*, Las Vegas, NV, Mar. 2008, pp. 147–152.

[30] V. Vukadinovic and J. Huschke, "Statistical multiplexing gains of H.264/AVC video in E-MBMS," in *Proc. 3rd Int. Symp. Wireless Pervasive Comput.*, Santorini, Greece, May 2008, pp. 468–474.

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

[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 Commun. Surveys Tuts.* vol. 6, no. 3, pp. 58–78, Third Quarter 2004 [Online]. Available: http://trace.eas.asu.edu

[33] M. Reisslein and K. W. Ross, "Call admission for prerecorded sources with packet loss," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 6, pp. 1167–1180, Aug. 1997.

[34] N. M. Markovich, A. Undheim, and P. J. Emstad, "Classification of slice-based VBR video traffic and estimation of link loss by exceedance," *Comput. Netw.*, vol. 53, no. 7, pp. 1137–1153, May 2009.

[35] 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.

[36] 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.

[37] 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 Syst. Mag.*, vol. 4, no. 1, pp. 7–28, First Quarter, 2004.

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

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

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

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

[42] S. Srinivasan, J. Vahabzadeh, and M. Reisslein, "The effects of priority levels and buffering on the statistical multiplexing of single-layer H.264/AVC and SVC encoded video streams (extended version)," Electrial Eng., ASU, Apr. 2010 [Online]. Available: http://mre.faculty.asu.edu/h264mux.pdf, Tech. Rep.

[43] P. Seeling, M. Reisslein, and F. H. P. Fitzek, "Offset trace-based video quality evaluation after network transport," *J. Multimedia*, vol. 1, no. 2, pp. 1–13, May 2006.

Sudhir Kumar Srinivasan received the Bachelor's Degree in electronics and communication engineering in 2004 from SJCE Mysore affiliated to Viveshwaraiah Technological University, India. He started his career in 2004 with Lucent Technologies, Bangalore, India working on developing IP, ATM, and FR protocols. He then moved onto Aricent Communications, Bangalore, in 2006 and working on the development of multimedia frameworks and RTP based wireless video delivery for mobile handsets. In 2008, Sudhir moved to the USA and received his Master's Degree in Computer Science and Engineering in 2010 from Arizona State University, Tempe, He was a research intern with RealNetworks, Seattle, WA in 2009 working on developing a next generation end-to-end media delivery architecture. Since March 2010 Sudhir is working for Qualcomm Inc. as a Senior Engineer. His research interests are video coding, wireless video communication, and scalable video.

Jonathan Vahabzadeh-Hagh is an Electrical Engineering undergraduate student at Arizona State University's School of Electrical, Computer, and Energy Engineering. His interests are in circuit design, power systems, and signal analysis. He is an Intern in Inverter Research and Development for Rogers Corporation. He is an Electrical Engineering Intern for the U.S. Army Corps of Engineers, mainly focusing on power generation and distribution. Jonathan also researches statistical multiplexing techniques for H.264 encoded video, through Arizona State University's Fulton Undergraduate Research Initiative (FURI) and National Science Foundation's Research Experience for Undergraduates (REU). In his free time he enjoys digital photography, rock-climbing, mountain biking, and tennis.

Martin Reisslein is an Associate Professor in the School of Electrical, Computer, and Energy Engineering at Arizona State University, Tempe. He received the Ph.D. in systems engineering from the University of Pennsylvania, Philadelphia, in 1998. 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. 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.