THIRD QUARTER 2004, VOLUME 6, NO. 3

IEEE COMMUNICATIONS SURVEYS
The Electronic Magazine of Original Peer-Reviewed Survey Articles

www.comsoc.org/pubs/surveys

NETWORK PERFORMANCE EVALUATION USING FRAME SIZE AND QUALITY TRACES OF SINGLE-LAYER AND TWO-LAYER VIDEO: A TUTORIAL

PATRICK SEELING, MARTIN REISSLEIN, AND BESHAN KULAPALA
ARIZONA STATE UNIVERSITY

ABSTRACT

Video traffic is widely expected to account for a large portion of the traffic in future wireline and wireless networks, as multimedia applications are becoming increasingly popular. Consequently, the performance evaluation of networking architectures, protocols, and mechanisms for video traffic becomes increasingly important. Video traces, which give the sizes, deadlines, and qualities of the individual video frames in a video sequence, have been emerging as convenient video characterizations for networking studies. In this tutorial we give an introduction to the use of video traces in networking studies. First we give a brief overview of digital video and its encoding and playout. Then we present a library of traces of single- and two-layer encoded video. We discuss the statistical properties of the traces and the resulting implications for the transport of video over networks. Finally we discuss the factors that need to be considered when using video traces in network performance evaluations. In particular, we introduce performance metrics that quantify the quality of the delivered video. We outline a procedure for generating video load for network simulations from the traces, and discuss how to meaningfully analyze the outcomes of these simulations.

With the increasing popularity of networked multimedia applications, video data is expected to account for a large portion of the traffic in the Internet of the future and next-generation wireless systems. For transport over networks, video is typically encoded (i.e., compressed) to reduce the bandwidth requirements. Even compressed video, however, requires large bandwidths of the order of hundreds of kb/s or Mb/s, as will be shown later in this tutorial. In addition, compressed video streams typically exhibit highly variable bit rates (VBR) as well as long range dependence (LRD) properties, as will be demonstrated later in this tutorial. This, in conjunction with the stringent Quality of Service (QoS) requirements (loss and delay) of video traffic, makes the transport of video traffic over communication networks a challenging problem. As a consequence, in the last decade the networking research community has witnessed an explosion in research on all aspects of video transport. The characteristics of video traffic, video traffic modeling, as well as protocols and mechanisms for the efficient transport of video streams, have received a great deal of interest among networking researchers and network operators.

Significant research effort has gone into the development of coding schemes that are tailored for video transport over networks and heterogeneous receiver-oriented display. Networks provide variable bit rates for video streams and may drop packets carrying video data (especially when wireless links are involved). The devices used for video display (e.g., TV sets, laptop computers, PDAs, cell phones) vary widely in

This article was recommended for publication after undergoing the standard IEEE Communications Surveys and Tutorials review process, which was managed by John N. Daigle, Associate EiC.

Supported in part by the National Science Foundation under grant no. Career ANI-0133252 and grant no. ANI-0136774. Supported in part by the State of Arizona through the IT301 initiative. Supported in part by two matching grants from Sun Microsystems.

their display formats (screen sizes), and processing capabilities. Also, users may want different display formats and qualities for different application scenarios.

Clearly, one way to provide the different video formats and qualities is to encode each video into different *versions*, each a single layer encoding with a fixed format and quality. The main drawbacks of versions are the increased storage requirement at the origin video server and the video proxy caches distributed throughout the network, and the need to stream multiple versions into the network to be able to quickly adapt to variations in the available bandwidth at a downstream link, for example, on a wireless last hop. *Scalable encoded video* overcomes these drawbacks and can provide the different video formats and qualities with one encoding. With conventional scalable encoding, the video is encoded into a base layer and one or multiple enhancement layers. The base layer provides a basic video quality, and each additional enhancement layer provides quality improvement. With these layered (hierarchical) encodings, the video quality (and required bit rate for transport) can be adjusted at the granularity of layers.

Given these developments in video coding it is widely expected that the encoded video carried over the Internet of the future and next-generation wireless systems will be heterogeneous in several aspects. First, future networks will carry video coded using a wide variety of encoding schemes, such as H.263, H.263+, MPEG-2, MPEG-4, divx, RealVideo, and WindowsMedia. Second, future networks will carry video of different quality levels, such as video coded with different spatial resolutions and/or signal to noise ratios (SNR). Third, and perhaps most importantly, the video carried in future networks will be to a large extent scalable encoded video since this type of video facilitates heterogeneous multimedia services over heterogeneous wire-line and wireless networks, as noted above.

Typically, studies on the network transport of video use video traces. Video frame size traces — the simplest form of video traces — give the sizes of each individual encoded video frame. Single layer MPEG-1 encoded videos have been available since the mid 1990s [1–5]. More elaborate video traces containing frame sizes as well as frame qualities have recently become available [6]. These more elaborate traces have become available for single-layer encoded video of different video formats and quality levels, as well as scalable encoded video. In this tutorial we explain how to conduct meaningful network studies with video traces, covering the spectrum from single frame size traces of single-layer MPEG-1 encoded video to elaborate traces of scalable MPEG-4 encodings.

This tutorial serves three main objectives:

- The communications and networking generalist who has no specific knowledge of video signal processing is introduced to the basic concepts of digital video and the characterization of encoded video for networking studies. In particular, we explain the basic principles that are employed in the common video coding standards and describe how these principles are employed to generate scalable (layered) video. We also explain the timing of the playout process of the digital video on the screen of the client device.
- We provide the reader with an overview of the main statistical characteristics of the video traffic and quality for single-layer and two-layer encoded video. We present the average traffic rates and qualities and the traffic and quality variabilities for encodings with different levels of video quality. We summarize the main insights from the statistical analysis of the traces in recommendations for the use of video traces in networking studies.
- We introduce the reader who is familiar with basic network performance analysis and network simulation to the unique issues that arise when using video traces in network simulations. In particular, we explain how to estimate the starvation probabilities and the video quality from simulations with video traces. We discuss the factors that need to be considered when generating a video traffic workload from video traces for the simulation of a network. We finally explain how to meaningfully analyze and interpret the outcomes of these simulations.

Overall, the objective of this tutorial is to enable networking generalists to design networking protocols and mechanisms for the transport of encoded video that take the properties of the video traffic into consideration. Furthermore, the goal is to enable the networking generalist to design and carry out simulations to evaluate performance using video traces.

Overview of Video Coding and Playout

In this section we give an introduction to the basic principles employed in video compression (coding). We first introduce digital video, which is the input to the video coder. We also discuss the implications of video coding on the playout of the video after network transport. The issues relating to the evaluation of the network transport, that is, the evaluation of network metrics such as packet delay and loss and link utilization, as well as the evaluation of the quality of the received video, are discussed.

Overview of Digital Video

Digital video consists of video frames (images) that are displayed at a prescribed *frame rate*; a frame rate of 30 frames/sec is used in the National Television Standards Committee (NTSC) video. The reciprocal of the frame rate gives the display time of a frame on the screen and is commonly referred to as *frame period*. Each individual video frame consists of picture elements (usually referred to as pixels or pels). The *frame format* specifies the size of the individual frames in terms of pixels. The ITU-R/CCIR-601 format (the common TV format) has 720×480 pixels (i.e., 720 pixels in the horizontal direction and 480 pixels in the vertical direction), while the Common Intermediate Format (CIF) format has 352×288 pixels, and the Quarter CIF (QCIF) format has 176×144 pixels. The CIF and QCIF formats are typically considered in network related studies. Each pixel is represented by three components: the luminance component (Y), and the two chrominance components, hue (U) and intensity (V). (An alternative representation is the RGB (red, green, and blue) representation, which can be converted to (and from) YUV with a fixed conversion matrix. We focus on the YUV representation, which is typically used in video encoder studies.) Since the human visual system is less sensitive to the color information than to the luminance information, the chrominance components are typically sub-sampled to one set of U and V samples per four Y samples. Thus, with chroma subsampling there are 352×288 Y samples, 176×144 U samples, and 176×144 V samples in each CIF video frame. Each sample is typically quantized into 8 bits, resulting in a frame size of 152,064 bytes for an uncompressed CIF video frame (and a corresponding bit rate of 36.5 Mb/s).

Principles of Non-Scalable Video Encoding

In this section we give a brief overview of the main principles of non-scalable (single-layer) video encoding (compression), we refer the interested reader to [7, 8] for more details. We

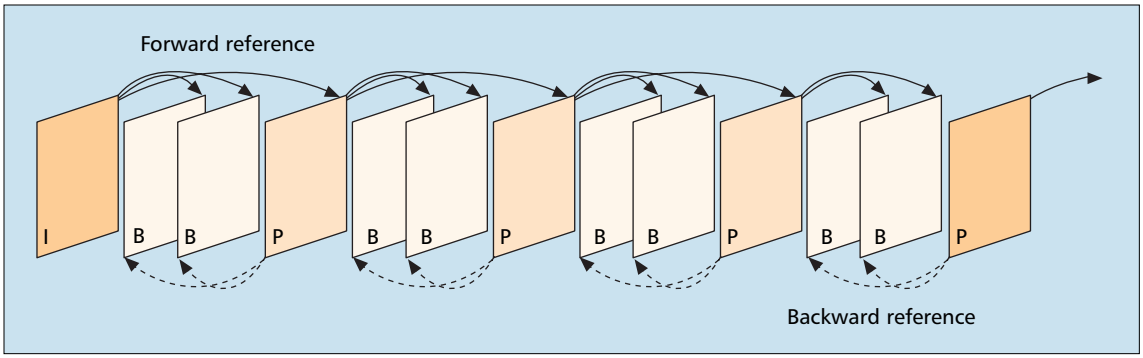

FIGURE 1. *Typical MPEG group of pictures (GoP) pattern with references used for predictive coding of P and B frames.*

focus in this overview on the principles employed in the MPEG and H.26x standards and note that most commercial codecs, such as RealVideo and WindowsMedia, are derived from these standards. The two main principles in MPEG and H.26x video coding are intra-frame coding using the discrete cosine transform (DCT), and inter-frame coding using motion estimation and compensation between successive video frames.

In intra-frame coding each video frame is divided into blocks of 8×8 samples of Y samples, U samples, and V samples. Each block is transformed using the DCT into a block of 8×8 transform coefficients, which represent the spatial frequency components in the original block. These transform coefficients are then quantized by an 8×8 quantization matrix that contains the quantization step size for each coefficient. The quantization step sizes in the quantization matrix are obtained by multiplying a base matrix by a quantization scale. This quantization scale is typically used to control the video encoding. A larger quantization scale gives a coarser quantization, resulting in a smaller size (in bits) of the encoded video frame as well as a lower quality. The quantized coefficients are then zigzag scanned, run-level coded, and variable-length coded to achieve further compression.

In inter-frame coding, MPEG introduced the frame types intra-coded (I), inter-coded (P), and bidirectional coded (B); similar frame types exist in H.26x video coding. These different frame types are organized into so called groups of pictures (GoPs). More specifically, the sequence of frames from a given I frame up to and including the frame preceding the next I frame is referred to as one GoP. The pattern of I, P, and B frames that make up a GoP is commonly referred to as *GoP pattern* or *GoP structure*. A typical GoP pattern with three P frames in a GoP and two B frames before and after each P frame is illustrated in Fig. 1. The different frame types are encoded as follows. In an I frame all blocks are intra-coded as outlined above. In a P frame the macroblocks (whereby a macroblock consists of four blocks of 8×8 samples) are inter-coded (as explained shortly) with reference to the preceding I or P frame, that is, the preceding I or P frame serves as a forward reference, as illustrated by the solid arrows in Fig. 1. In a B frame the macroblocks are inter-coded with reference to the preceding I or P frame, which serves as forward reference, and the succeeding I or P frame, which serves as backward reference, as illustrated by the dashed arrows in Fig. 1.

To intercode a given macroblock the best matching macroblock in the reference frame(s) is determined and identified by a motion vector; this process is commonly referred to as *motion estimation*. Any (typically small) difference between the block to be encoded and the best matching block is transformed using the DCT, quantized, and coded as outlined above; this process is commonly referred to as *motion compensation*. If a good match cannot be found in the reference frame(s), then the macroblock is intra coded. (In the optional 4MV mode the above processes are applied to blocks instead of macroblocks.)

PRINCIPLES OF SCALABLE VIDEO ENCODING

With conventional layered encoding the video is encoded hierarchically into a base layer and one (or more) enhancement layer(s). Decoding the base layer provides a basic video quality, while decoding the base layer together with the enhancement layer(s) provides an enhanced video quality. MPEG has standardized the following scalability modes: data partitioning, temporal, spatial, and signal-to-noise (SNR). We briefly review the temporal and spatial scalability modes as they are considered in the later discussion of the trace statistics.

With temporal scalable encoding the enhancement layer frames are interleaved between base layer frames. Each enhancement layer frame is inter-coded with reference to the immediately preceding base layer frame and the immediately succeeding base layer frame (as illustrated in Fig. 2) for a scenario where I and P frames form the base layer and B frames form the enhancement layer.

The base layer of the temporal scalable encoding provides a basic video quality with a low frame rate. Adding the enhancement layer to the base layer increases the frame rate. Note that the base layer can be decoded independently of the enhancement layer since each base layer frame is only encoded with reference to another base layer frame. On the other hand, the enhancement layer requires the base layer for decoding since the enhancement layer frames are encoded with reference to base layer frames.

With spatial scalability the base layer provides a small video format (e.g., QCIF); adding the enhancement layer increases the video format (e.g., to CIF). The base layer of the spatial

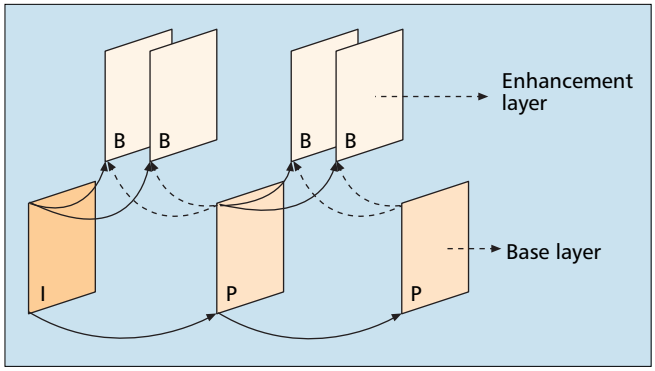

FIGURE 2. *Example for temporal scalable encoding: I and P frames form the base layer and B-frames form the enhancement layer.*

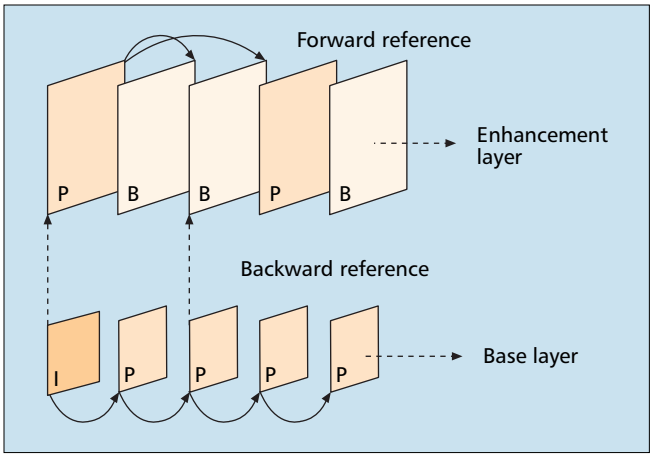

FIGURE 3. *Example for spatial scalable encoding. The downsampled video is encoded into a base layer stream consisting of I and P frames. The difference between the decoded and upsampled base layer and the original video is encoded into the P and B frames in the enhancement layer.*

scalable encoding can be up-sampled to give a coarse video at the larger format. To generate a spatial scalable encoding, the original (uncompressed) video is first downsampled to the smaller base layer format and the downsampled video is encoded employing the intra and inter coding techniques described above. A base layer consisting of only I and P frames is illustrated in Fig. 3. The encoded base layer is subsequently decoded and upsampled. The difference between a decoded and upsampled base layer frame and the corresponding uncompressed frame is then encoded using the DCT transform coding (and possibly intercoding within the enhancement layer). More specifically, a given enhancement layer frame can be encoded with reference to the corresponding base layer frame, which is referred to as backward reference in this context, and with respect to a preceding frame in the enhancement layer, which serves as forward reference. In the example illustrated in Fig. 3 the enhancement layer frames are coded as either P or B frames. A P frame in the enhancement layer is coded with reference to the corresponding I frame in the base layer. A B frame in the enhancement layer is coded with reference to the corresponding P frame in the base layer and the preceding P frame in the enhancement layer.

We close this overview of scalable encoding by noting that aside from the layered coding considered here a number of other methods to achieve scalable encoding have been developed. Fine granular scalability (FGS) [9] encodes the video into a base layer and one enhancement layer. The special property of the FGS enhancement layer is that it can be cut anywhere at the granularity of bits allowing the video stream to finely adapt to changing network bandwidths. With conventional layered coding, on the other hand, the video stream can only adapt at the granularity of complete enhancement layers. With Multiple Description Coding (MDC) [10] the video is encoded into several streams (descriptions). Each of the descriptions contributes to the decoded video quality. Decoding all the descriptions gives the high video quality while decoding an arbitrary subset of the descriptions results in lower quality. This is in contrast to conventional hierarchical layered videos where a received enhancement layer is useless if the corresponding base layer is missing. With wavelet transform coding [11] a video frame is not divided into blocks, as with the DCT-based MPEG coding. Instead, the entire frame is coded into several subbands using the wavelet transform. We note that these methodss to achieve scalable video coding are beyond the scope of this article. This article is focused on the network performance evaluation for conventional non-scalable (single-layer) and layered (hierarchical) encoded video, for which traces are currently publicly available.

VIDEO PLAYOUT

The intercoding of the video frames has important implications for the video playout at the receiver, which we explain in this section, as these implications affect the structure of video traces and video traffic simulations. Recall that a P frame is encoded with reference to the preceding I or P frame and that a B frame is encoded with reference to the preceding I(P) frame and the succeeding P(I) frame. In any case, the reference frame(s) must be decoded before the decoding of the intercoded P or B frame can commence. Consider, for instance, the GoP pattern IBBPBBPBBPBBIBBP..., with three P frames between two successive I frames and two B frames between successive I(P) and P(I) frames. With the considered GoP pattern, the decoder needs both the preceding I (or P) and the succeeding P (or I) frame for decoding a B frame. Therefore, the encoder emits the frames in the order IPBBPBBPBBIBBP..., which we refer to as the *codec sequence*. In contrast, we refer to the frame order IBBPBBPBBPBBIBBP... as the *display sequence* since the video frames are displayed in that order on the screen.

To better understand the start of the playout process consider the scenario in Fig. 4, in which the frames are received in the coded sequence. In the depicted scenario the reception of the first I frame commences at time zero and is completed at time T, which denotes the frame period of the video. Each subsequent frame takes T seconds for reception. The decoding of the first B frame commences at time $3T$, and we sup-

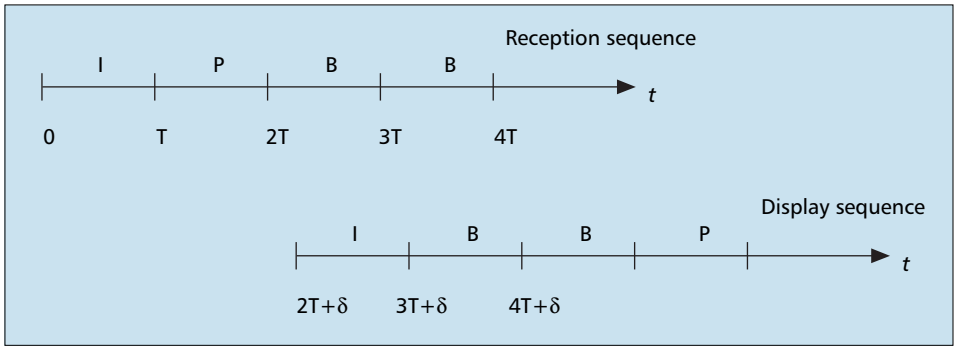

FIGURE 4. *Start of video playout. The first I and P frame are required to decode the first B frame. If the video is received in the codec sequence, as illustrated here, the playback can commence* $2T + \delta$ *after the first I frame begins to arrive.*

pose for illustration that the decoding of a frame takes δ seconds. Thus, the first B frame is available for display at time $3T + \delta$, allowing us to commence the playback by displaying the first I frame at time $2T + \delta$. Next consider the scenario in which the encoded frames are received in the display sequence. For this scenario it is straightforward to verify with a similar argument that the playback can commence at time $3T + \delta$.

We briefly note that the difference between the codec sequence and the display sequence can be exploited to relax the delivery deadlines of the I and P frames [12]. In the scenario illustrated in Fig. 4 the I frame is not needed at the decoding client until time $2T$ to ensure that it is decoded and ready for display at time $2T + \delta$. Similarly, the P frame is not needed until the time $3T$, assuming that both the P and the first B frame can be decoded within δ seconds to ensure that the B frame is available for display at time $3T + \delta$.

Different Types of Video Characterization for Network Performance Evaluation

Generally, there are three different methods to characterize encoded video for the purpose of networking research: *video bit stream*, *video traffic trace*, and *video traffic model*.

The video bit stream, which is generated using the encoding mechanisms presented in the preceding section, contains the complete video information. The traffic characterization (e.g., the frame size) can be obtained by measuring the traffic or by parsing the bit stream. The video quality can be determined by subjective (viewing) evaluation [13] or objective methods [14–16]. The advantage of the bit stream is that it allows for networking experiments where the quality of the video — after suffering losses in the network — is evaluated, as in [17–21]. One limitation of the bit stream is that it is very large in size: several GBytes for one hour of compressed video or several tens of GBytes for one hour of uncompressed video. Another limitation of bit streams is that they are usually proprietary and/or protected by copyright. This limits the access of networking researchers to bit streams, and also limits the exchange of bit streams among research groups.

Video traces are an alternative to bit streams. While the bit streams give the actual bits carrying the video information, the traces only give the number of bits used for the encoding of the individual video frames, as described in the following section in more detail. Thus, there are no copyright issues.

Video traffic models, which can be derived from video traces, have received a great deal of attention in the literature (see, for example, [22–32]). The goal of a traffic model is to capture the essential properties of the real traffic in an accurate, computationally efficient, and preferably mathematically tractable description that should also be parsimonious, that is, require only a small number of parameters. A traffic model is typically developed based on the statistical properties of a set of video trace samples of the real video traffic. The developed traffic model is verified by comparing the traffic it generates with the video traces. If the traffic model is deemed sufficiently accurate, it can be used for the mathematical analysis of networks, for model driven simulations, and also for generating so called virtual (synthetic) video traces.

Structure of Video Traces

In this section we give a general overview of video trace structures and define the quantities in the traces. First we introduce the notation for the traffic and quality characterization of the video. Let N denote the number of video frames in a given trace. Let t_n, $n = 0, \ldots, N-1$, denote the frame period (display time) of frame n. Let T_n, $n = 1, \ldots, N$, denote the cumulative display time up to (and including) frame $n-1$, that is, $T_n = \Sigma_{k=0}^{n-1} t_k$ (and define $T_0 = 0$). Let X_n, $n = 0, \ldots, N-1$, denote the frame size (number of bit or byte) of the encoded (compressed) video frame n. Let Q_n^Y, $n = 0, \ldots, N-1$, denote the quality of the luminance component of the encoded (and subsequently decoded) video frame n (in dB). Similarly, let Q_n^U and Q_n^V, $n = 0, \ldots, N-1$, denote the qualities of the two chrominance components hue (U) and saturation(V) of the encoded video frame n (in dB).

A video trace gives these defined quantities typically in an ASCII file with one line per frame. Some traces give only the frame sizes X_n; these traces are often referred to as *terse*. *Verbose* traces, on the other hand, give several of the defined quantities. For example, a line of a verbose trace may give frame number n, cumulative display time T_n, frame type (I, P, or B), frame size X_n (in bit), and luminance quality Q_n^Y (in dB) for frame n.

Generally, for layered encodings the base layer trace gives the frame sizes of the base layer and the quality values for the decoded base layer, while the enhancement layer traces give the sizes of the encoded video frames in the enhancement layer and the *improvement* in the quality obtained by adding the enhancement layer to the base layer (i.e., the difference in quality between the aggregate (base + enhancement layer) video stream and base layer video stream). In other words, the base layer traces give the traffic and quality of the base layer video stream. The enhancement layer traces give the enhancement layer traffic and the quality improvement obtained by adding the enhancement layer to the base layer.

A subtlety in the traces is the order of the frames, which may depend on the GoP pattern. In particular, some video traces give the frames in the display sequence, while others give the frames in the codec sequence, which we introduced earlier. The frame index n, $n = 0, \ldots, N-1$, however, always refers to the position of the corresponding frame in the display sequence. As an example consider the GoP pattern IBBPBBPBBPBBIBBP…, with three P frames between two successive I frames and two B frames between successive I(P) and P(I) frames. If the frames are ordered in the display sequence in the trace, then frame n, $n = 0, 1, \ldots, N-1$, is on line n of the trace. On the other hand, if the frames are ordered in the codec sequence in the trace, then frame $n = 0$ is on line 0, frame number $n = 3$ is on line 1, frames 1 and 2 are on lines 2 and 3, frame 6 on line 4, and frames 4 and 5 on lines 5 and 6, and so on. This subtlety must be considered when using traces in networking studies, as elaborated above.

In summary, in this section we have provided a general overview of the different structures of video traces. The various available collections of video traces can be categorized according to the structure used in the traces. The collections [1–5], for instance, have adopted the terse format and give the frames in the display sequence. The collection [6], which we study in the next section, provides both verbose traces with frames in the codec sequence as well as terse traces with frames in the display sequence.

Video Trace Statistics

In this section we present a publicly available library of traces of heterogeneous and scalable encoded video. The traces have been generated from more than 15 videos of one hour each, which have been encoded into a single layer at heterogeneous

Class	Video	Genre	Quantization scale settings (from Table 2)
Movies	*Citizen Kane*	Drama	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Die Hard I	Action	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Jurassic Park I	Action	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)
	Silence of the Lambs	Drama	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Star Wars IV	Sci-fi	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)
	Star Wars V	Sci-fi	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	The Firm	Drama	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)
	The Terminator I	Action	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Total Recall	Action	(30, 30, 30); (10, 14, 16); (4, 4, 4)
Cartoons	*Aladdin*	Cartoon	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Cinderella	Cartoon	(30, 30, 30); (10, 14, 16); (4, 4, 4)
Sports	*Baseball*	Game 7 of the 2001 World Series	(30, 30, 30); (10, 14, 16); (4, 4, 4)
	Snowboarding	Snowboarding Competition	(30, 30, 30); (10, 14, 16); (4, 4, 4)
TV sequences	*Tonight Show*	Late Night Show	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)

Table 1. *Overview of studied video sequences in QCIF format.*

qualities and into two layers using the temporal scalability and spatial scalability modes of MPEG-4. Due to space constraints we include here only a brief overview of the trace library and the trace statistics and refer the interested reader to [6] for details.

Videos and Encoder Modes

We consider the traces of the videos in Table 1. All considered videos are 60 minutes long, corresponding to 108,000 frames, and are in the QCIF format. For spatial scalable encoding only 30 minutes (54,000 frames) of the videos in the CIF format are considered. We consider the encodings without rate control with the fixed quantization scales in Table 2, where we use the abbreviation (x, y, z) to refer to the quantization scales for I, P, and B frames, as is common in video encoding studies. For the rate control encodings we consider TM5 [33] rate control with the target bit rate settings summarized in Table 3.

The base layer of the considered temporal scalable encoding gives a basic video quality by providing a frame rate of 10 frames per second. Adding the enhancement layer improves the video quality by providing the (original) frame rate of 30 frames per second. With the considered spatial scalable encoding, the base layer provides video frames that are one fourth of the original size (at the original frame rate), that is, the number of pixels in the video frames is cut in half in both the horizontal and vertical direction. (These quarter-size frames can be up-sampled to give a coarse grained video with the original size.) Adding the enhancement layer to the base layer gives the video frames in the original size (format).

For each video and scalability mode we have generated traces for videos encoded without rate control and for videos encoded with rate control. For the encodings without rate control we keep the quantization parameters fixed, which produces nearly constant quality video (for both the base layer and the aggregate (base + enhancement layer) stream, respectively) but highly variable video traffic. For the encodings with rate control we employ the TM5 rate control, which strives to keep the bit rate around a target bit rate by varying the quantization parameters, and thus the video quality. We apply rate control only to the base layer of scalable encodings and encode the enhancement layer with fixed quantization parameters. Thus, the bit rate of the base layer is close to a constant bit rate, while the bit rate of the enhancement layer is highly variable. This approach is motivated by networking schemes that provide constant bit rate transport with very stringent Quality of Service for the base layer, and variable bit rate transport with less stringent Quality of Service for the enhancement layer.

Abbreviation	Quantization scale setting		
	I Frame	P Frame	B Frame
(30, 30, 30)	30	30	30
(24, 24, 24)	24	24	24
(10, 14, 16)	10	14	16
(10, 10, 10)	10	10	10
(4, 4, 4)	4	4	4

Table 2. *Quantization scale settings for encodings without rate control.*

Single-Layer Encoded Video

In this section we give an overview of the video traffic and quality statistics of the single-layer encodings, which are studied in detail in [34]. In Table 4 we give an overview of the elementary frame size and bit rate statistics. We consider the average frame size $\overline{X}$, the coefficient of variation CoV_X (defined as the standard deviation of the frame size normalized by the mean frame size), the peak-to-mean ratio of the frame size $X_{\max}/\overline{X}$, and the mean and peak bit rates, as well as

	Encoding mode		
	Single	Temporal/spatial	
		Base	Enhanced
No RC	All Table 2	All Table 2	All Table 2
RC	64kb/s 128 kb/s 256 kb/s	64kb/s 128 kb/s 256 kb/s	(10, 14, 16) (10, 14, 16) (10, 14, 16)

Table 3. *Overview of studied encoding modes.*

Encoding mode		Frame size			Bit rate		GoP size		Frame quality	
		Mean $\overline{X}$ [kbyte]	CoV CoV_X	Peak/M. $X_{max}/\overline{X}$	Mean $\overline{X}/T$ [Mb/s]	Peak X_{max}/T [Mb/s]	CoV CoV_Y	Peak/M. $Y_{max}/\overline{Y}$	Mean $\overline{Q}$ [dB]	CoV $CoQV$
(4, 4, 4)	Min	1.881	0.399	4.115	0.451	3.108	0.284	2.606	25.052	0.162
	Mean	3.204	0.604	6.348	0.769	4.609	0.425	4.136	36.798	0.326
	Max	5.483	0.881	8.735	1.316	6.31	0.709	7.367	37.674	0.67
(10, 10, 10)	Min	0.613	1.017	9.345	0.147	1.93	0.536	6.087	30.782	0.353
	Mean	0.738	1.146	12.819	0.177	2.202	0.645	6.754	31.705	0.56
	Max	0.949	1.36	16.303	0.228	2.398	0.803	7.902	32.453	0.907
(10, 14, 16)	Min	0.333	1.173	10.688	0.08	1.586	0.438	3.642	28.887	0.465
	Mean	0.55	1.489	16.453	0.132	2.045	0.547	6.03	30.29	1.017
	Max	0.874	2.128	25.386	0.21	2.708	0.77	12.268	31.888	3.685
(24, 24, 24)	Min	0.23	1.033	11.466	0.055	0.775	0.447	4.498	26.535	0.438
	Mean	0.273	1.206	15.438	0.065	0.992	0.546	5.405	27.539	0.824
	Max	0.327	1.547	19.468	0.078	1.272	0.747	6.148	28.745	1.099
(30, 30, 30)	Min	0.194	0.82	7.67	0.047	0.522	0.383	3.02	25.177	0.434
	Mean	0.282	0.943	11.357	0.067	0.742	0.441	4.642	26.584	0.712
	Max	0.392	1.374	17.289	0.094	1.104	0.671	8.35	28.446	1.618
64 kb/s	Min	0.267	0.806	8.398	0.064	0.774	0.354	2.991	25.052	0.446
	Mean	0.297	1.022	48.328	0.0714	3.353	0.411	9.563	26.624	0.746
	Max	0.384	1.494	82.72	0.092	5.488	0.46	18.51	28.926	1.585
128 kb/s	Min	0.534	1.066	17.749	0.128	2.274	0.089	2.626	26.12	0.641
	Mean	0.534	1.189	28.135	0.128	3.606	0.143	4.776	28.998	1.197
	Max	0.535	1.401	50.883	0.128	6.52	0.277	9.691	31.795	3.021
256 kb/s	Min	1.067	0.904	6.89	0.256	1.765	0.03	1.395	28.461	0.639
	Mean	1.067	1.000	9.841	0.256	2.521	0.0431	1.65	31.414	1.432
	Max	1.067	1.106	13.086	0.256	3.352	0.072	2.387	33.824	5.307

Table 4. *Overview of frame statistics of single-layer traces (QCIF).*

the average PSNR quality $\overline{Q}$ and the coefficient of the quality variation *CoQV*. We note that the PSNR does not completely capture the many facets of video quality. However, analyzing a large number of videos subjectively becomes impractical. Moreover, recent studies have found that the PSNR is as good a measure of video quality as other more sophisticated objective quality metrics [35]. As the PSNR is well defined only for the luminance (Y) component [36], and since the human visual system is more sensitive to small changes in the luminance, we focus on the luminance PSNR values.

For a compact presentation we report for each metric the minimum, mean, and maximum of the set of videos given in Table 1. This presentation, which we adopt for most tables in this article, conveys the main characteristics of the different encoding and scalability modes. However, it does not convey the impact of the different video genres and content features on the video traffic and quality, for which we refer to [6].

Focusing for now on the encodings without rate control, we observe that the coefficient of variation CoV_X and the peak-to-mean ratio $X_{max}/\overline{X}$ increase as the quantization scales increase (i.e., as the video quality decreases), indicating that the video traffic becomes more variable. As the quality decreases further, the coefficient of variation and peak-to-mean ratio decrease. In other words, we observe a concave shape of the coefficient of variation and the peak-to-mean ratio of the frame sizes as a function of the encoded video quality, with a maximum of the coefficient of variation and the peak to mean ratio for intermediate video quality. This concavity has important implications for resource allocation for video traffic in networks. The maximum in the peak-to-mean frame size ratio for the intermediate video quality, for instance, results in a small mean network utilization for this quality level when allocating network resources according to the peak rate.

Next we examine the GoP sizes. Recall that a GoP consists of the group of frames from an I frame up to and including the frame preceding the next I frame. We refer to the sum of the sizes of the frames in a GoP as the *GoP size* (in bit) and denote it by *Y*. From Table 4 we observe that the coefficient of variation and the peak-to-mean ratios of the GoP size also exhibit a concave shape with a maximum at an intermediate quality level. These observations build on earlier studies [37] which considered a smaller range of the quantization scale and uncovered only an increasing trend in the coefficient of variation and the peak-to-mean ratio for increasing quantization scales (i.e., decreasing video quality). While the origins of this concave shape of the coefficient of variation and the peak to mean ratio of the frame sizes are under investigation in ongoing work, we can draw some immediate guidelines for networking studies, which are detailed later.

Next we observe that the encodings with rate control with target bit rates of 64 kb/s and 128 kb/s tend to have significantly larger coefficients of variation than the encodings without rate control. This is primarily because the employed TM5 rate control algorithm allocates target bit rates to each of the frame types (I, P, and B) and thus provides effective rate control at the GoP time scale, with potentially large variations of the individual frame sizes. Even with TM5 rate control, however, there are some small variations in the GoP sizes (see Table 4). These variations are mostly due to relatively few outliers, resulting in the quite significant peak-to-mean ratio, yet very small coefficient of variation. (As a side note, we remark that the 128 kb/s and 256 kb/s target bit rates are met perfectly (in the long run average), while the 64 kb/s is not always met. This is because the employed encoder does not

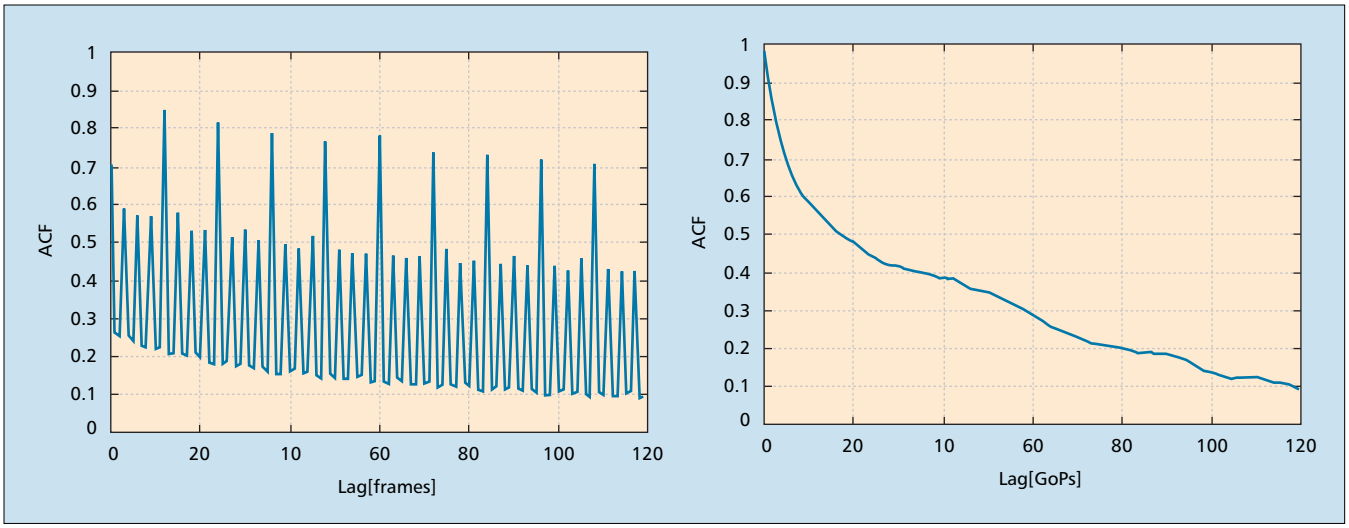

FIGURE 5. *Autocorrelation function for frame sizes (left) and GoP sizes (right) of single-layer encoding with quantization scales (4, 4, 4) of the* Star Wars IV *video.*

allow for quantization scales smaller than (30, 30, 30), which gives average bit rate above 64 kb/s for some videos.) Both the typically very large frame size variations with rate control, and the residual variation at the larger GoP time scale, need to be taken into consideration in networking studies.

An important aspect of video traffic is its correlation over time [38]. For a first assessment of the correlations in video traffic we consider the autocorrelation function of the frame sizes, that is, of the series $\{X_0, X_1, \ldots, X_{N-1}\}$, and the autocorrelation function of the GoP sizes. In Fig. 5 we plot these autocorrelation functions for the (4, 4, 4) encoding of the *Star Wars* video. We observe from Fig. 5 (left) that the autocorrelation function of the frame sizes consists of a periodic "spike" pattern that is superimposed on a decaying curve. The periodic spike pattern is due to the MPEG frame types in a GoP. In particular, I frames are typically the largest frames, resulting in the high spikes that are spaced 12 frames apart in the autocorrelation function. The P frames are typically smaller than the I frames but larger than the B frames, resulting in the three intermediate spikes. The B frames are typically the smallest frames, resulting in the two small autocorrelation values between successive spikes due to I and P frames. The autocorrelation function of the GoP sizes plotted in Fig. 5 (right) gives a better picture of the underlying decay of the autocorrelation function. We observe that this autocorrelation function decays relatively slowly. For a lag of 100 GoPs, which correspond to approximately 40 seconds, the autocorrelation coefficient is approximately 0.15. This indicates fairly significant correlations over relatively long time periods, which are mainly due to the correlations in the content of the video (e.g., scenes of persistent high motion or a high level of detail). In more extensive investigations we have found that these behaviors of the frame size and GoP size autocorrelation functions are typical for encodings without rate control, whereby there are typically only minor differences between the autocorrelation functions for encodings with different quality levels. For the encodings with rate control the autocorrelation of the GoP sizes drops immediately to zero, and the frame size autocorrelation function exhibits the periodic spike pattern due to the different MPEG frame types around zero. This behavior of the autocorrelation function is a consequence of the rate control, which adjusts the quantization scales to keep the bit rate averaged over a GoP close to the specified target bit rate, independent of the video content.

To assess the long-range dependence properties of the encoded videos, we determined the Hurst parameter of the frame size traces using the R/S plot, the periodogram, the variance-time plot, and the logscale diagram (see [39] for details). We have found that the encodings without rate control typically exhibit long-range dependence with the Hurst parameter typically ranging between 0.75 and 0.95. The encodings with rate control do not typically exhibit long-range dependence (except for the cases where the 64 kb/s target bit rate could not be reached due to the quantization scale being limited to at most 30). We have also found that the Hurst parameter estimates are roughly the same when comparing different quality levels.

We have also investigated the multifractal scaling characteristic of the video traffic using the wavelet-based multiscale diagram [40]. We found that the linear multiscale diagram generally does not differ significantly from a horizontal line. This indicates that the video traffic is mono-fractal, that is, does not exhibit a significant multi-fractal behavior.

Temporal Scalable Encoded Video

Base Layer — Table 5 summarizes the frame size and quality statistics of the base layer of the temporal scalable encoded video. Recall that in the considered temporal scalable encodings, the I and P frames constitute the base layer and the B frames constitute the enhancement layer. With the IBBPBBPBBPBBPBBIBB…GoP structure, the frame sizes X^b_{3k+1} and X^b_{3k+2}, $k = 0, \ldots, N/3 - 1$ are zero as these correspond to gaps in the base layer frame sequence. We observe for the encodings without rate control that the temporal base layer traffic is significantly more variable than the corresponding single-layer traffic. The peak-to-mean ratio $X^b_{\max}/\overline{X}_b$ of the base layer frame sizes is roughly 1.5 to 2 times larger than the corresponding $X_{\max}/\overline{X}$ of the single-layer traces (from Table 4). This larger variability of the base layer of the temporal scalable encoding is due to the fact that the frames missing in the base layer are counted as zeros in the frame size analysis, that is, the frame size analysis considers a scenario where each frame is transmitted during its frame period of 33 msec and nothing is transmitted during the periods of the skipped frames. To overcome the large variabilities of the base layer we consider averaging three base layer frames, that is, an I or P frame and the subsequent two missing frames of size zero, and denote the averaged base layer frame size by $X^{b(3)}$. For example, consider the base layer trace segment X_I, 0, 0, X_P, 0, 0, where X_I and X_P denote the size of an I and P frame, respectively. With three-frame smoothing this trace segment

Encoding mode		Frame size			Bit rate		Aggregated (3)		GoP size		Frame quality	
		Mean $\overline{X}^b$ [kbyte]	CoV CoV_X^b	Peak/M. $X^b_{max}/\overline{X}^b$	Mean $\overline{X}^b/T$ [Mb/s]	Peak X^b_{max}/T [Mb/s]	CoV $CoV_X^{b(3)}$	Peak/M. $X^{b(3)}_{max}/\overline{X}^b$	CoV CoV_Y^b	Peak/M. $Y^b_{max}/\overline{Y}^b$	Mean $\overline{Q}^b$ [dB]	CoV $CoQV^b$
(4, 4, 4)	Min	0.895	1.54	9.68	0.215	3.124	0.351	3.227	0.281	2.437	20.944	2.292
	Mean	1.458	1.6878	12.897	0.35	4.363	0.522	4.3	0.395	3.536	24.28	3.167
	Max	2.316	1.994	18.463	0.556	6.285	0.812	6.154	0.668	5.762	27.623	4.731
(10, 10, 10)	Min	0.349	1.96	16.47	0.084	1.918	0.783	5.49	0.486	4.596	24.437	2.406
	Mean	0.4245	2.135	22.033	0.102	2.179	0.919	7.345	0.57	5.513	25.386	2.865
	Max	0.539	2.405	28.651	0.129	2.398	1.123	9.551	0.708	7.532	26.809	3.402
(10, 14, 16)	Min	0.224	2.038	16.478	0.054	1.586	0.848	5.493	0.375	3.138	20.797	2.172
	Mean	0.3727	2.292	23.818	0.089	2.0349	1.037	7.939	0.49	4.837	23.804	2.76
	Max	0.567	2.872	37.791	0.136	2.708	1.443	12.597	0.686	8.617	27.047	3.85
(24, 24, 24)	Min	0.146	1.987	19.051	0.035	0.784	0.806	6.351	0.414	3.896	23.422	0.848
	Mean	0.16425	2.163	25.88	0.0393	1.002	0.939	8.627	0.500	4.989	24.264	1.805
	Max	0.197	2.533	33.329	0.047	1.272	1.213	11.111	0.665	6.776	25.067	2.859
(30, 30, 30)	Min	0.11	1.797	13.74	0.026	0.556	0.64	4.58	0.352	2.639	20.279	1.494
	Mean	0.1574	1.912	20.058	0.038	0.736	0.743	6.687	0.418	4.152	22.842	2.157
	Max	0.211	2.37	30.309	0.051	1.104	1.098	10.104	0.622	7.139	25.828	2.673
64 kb/s	Min	0.267	1.782	24.886	0.064	1.594	0.626	8.296	0.138	3.286	20.35	1.875
	Mean	0.267	1.883	42.52	0.064	2.723	0.716	14.173	0.209	6.016	23.364	2.473
	Max	0.267	2.051	70.436	0.064	4.511	0.857	23.479	0.338	12.126	26.853	3.434
128 kb/s	Min	0.534	1.645	10.29	0.128	1.318	0.486	3.43	0.045	1.417	20.688	2.102
	Mean	0.534	1.705	12.629	0.128	1.617	0.548	4.21	0.082	1.737	23.842	2.796
	Max	0.534	1.819	18.772	0.128	2.404	0.661	6.257	0.138	2.613	27.292	4.127
256 kb/s	Min	1.067	1.518	8.504	0.256	2.177	0.318	2.835	0.021	1.231	20.842	2.218
	Mean	1.067	1.546	10.125	0.256	2.593	0.359	3.375	0.038	1.397	24.088	2.992
	Max	1.067	1.617	11.664	0.256	2.987	0.453	3.888	0.064	1.722	27.508	4.577

Table 5. *Overview of frame statistics for the base layer of temporal scalability (QCIF).*

becomes $X_I/3$, $X_I/3$, $X_I/3$, $X_P/3$, $X_P/3$, $X_P/3$. We observe from Table 5 that with this averaging (smoothing), which is equivalent to spreading the transmission of each base layer frame over three frame periods (100 msec), the variability of the base layer traffic is dramatically reduced. We also observe that the $X^{b(3)}_{max}/\overline{X}_b$ is typically one half to two thirds of the corresponding $X_{max}/\overline{X}$ in Table 4. Noting that the peak-to-mean ratio of the time series $X_I/3$, $X_I/3$, $X_I/3$, $X_P/3$, $X_P/3$, $X_P/3$,… is equal to the peak-to-mean ratio of the time series X_I, X_P, …, that is, the time series containing only the sizes of the I and P frames, we may conclude from this observation that the I and P frames are relatively less variable in size compared to the B frames. This has been confirmed in more extensive studies [6] and is intuitive as B frames can cover the entire range from being completely intra-coded (e.g., when a scene change occurs at that frame) to being completely inter-coded.

For the encodings with rate control, we observe from Table 5 in comparison with Table 4 that the smoothed (over three frames or GoP) base layers are significantly less variable than the corresponding single layer encodings. This is again primarily due to the generally smaller variability of the I and P frames in the base layer. The peak bit rates of the 128 kb/s and 256 kb/s base layers with GoP smoothing are typically less than 200 kb/s and 300 kb/s, respectively. This enables the transport of the base layer with rate control over reliable constant bit rate network "pipes," provisioned, for instance, using the guaranteed services paradigm [41]. We note, however, that even the rate controlled base layers smoothed over GoPs require some over-provisioning since the peak rates are larger than the average bit rates. In more detailed studies [42] we have found that the excursions above (and below) the average bit rate are typically short-lived. Therefore, any of the common smoothing algorithms (e.g., [43, 44]) should result in a reduction of the peak rates of the GoP streams to rates very close to the mean bit rate with a moderately sized smoothing buffer. In addition, we note that the TM5 rate control employed in our encodings is a basic rate control scheme that is standardized and widely used. More sophisticated and refined rate control schemes (e.g., [45]) may further reduce the variability of the traffic. In summary, we recommend using our traces obtained with TM5 rate control in scenarios where the video traffic is smoothed over the individual frames in a GoP (which incurs a delay of approximately 0.4 sec) or use some other smoothing algorithm.

Now turning to the video frame PSNR quality, we observe that the average quality $\overline{Q}$ is significantly lower and the variability in the quality significantly larger compared to the single-layer encoding. This severe drop in quality and increase in quality variation are due to decoding only every third frame and displaying it in place of the missing two B frames. The reduction in quality with respect to the single-layer encoding is not as severe for the rate controlled encodings, which now can allocate the full target bit rate to the I and P frames.

Enhancement Layer — The main observations from the enhancement layer traffic statistics in Table 6 are a very pronounced maximum in the variability and a relatively large variability, even when smoothing the two B frames over three frame periods or over a GoP. For the enhancement layers corresponding to the base layers with rate control, we observe that the average enhancement layer bit rate decreases as the target bit rate of the base layer increases. This is to be expected as the higher bit rate base layer contains a more accurate encoding of the video, leaving less

Encoding mode		Frame size			Bit rate		Aggregated (3)		GoP Size	
		Mean $\overline{X}^e$ [kbyte]	CoV CoV_X^e	Peak/M. $X_{max}^e/\overline{X}^e$	Mean $\overline{X}^e/T$ [Mb/s]	Peak X_{max}^e/T [Mb/s]	CoV $CoV_X^{e(3)}$	Peak/M. $X_{max}^{e(3)}/\overline{X}^e$	CoV CoV_Y^e	Peak/M $Y_{max}^e/\overline{Y}^e$
(4, 4, 4)	Min	0.914	0.801	4.885	0.219	2.368	0.305	3.219	0.291	2.759
	Mean	1.748	0.951	9.872	0.42	3.92	0.491	6.096	0.462	4.83
	Max	3.172	1.175	15.765	0.761	6.138	0.765	9.738	0.757	8.831
(10, 10, 10)	Min	0.262	1.13	15.736	0.063	1.05	0.687	10.238	0.62	7.907
	Mean	0.311	1.277	20.121	0.075	1.484	0.841	12.562	0.793	9.234
	Max	0.407	1.439	23.71	0.098	1.738	1.018	15.07	0.992	10.166
(10, 14, 16)	Min	0.101	1.093	14.714	0.024	0.531	0.669	9.688	0.619	5.223
	Mean	0.176	1.361	25.136	0.042	1.035	0.905	14.976	0.811	9.977
	Max	0.317	1.773	37.224	0.076	1.778	1.319	22.732	1.258	20.066
(24, 24, 24)	Min	0.082	1.103	12.393	0.02	0.31	0.669	7.15	0.556	5.74
	Mean	0.106	1.233	20.181	0.026	0.5	0.804	10.825	0.715	7.251
	Max	0.127	1.486	28.648	0.031	0.594	1.061	14.029	0.986	8.683
(30, 30, 30)	Min	0.073	0.978	9.637	0.018	0.226	0.544	5.639	0.49	3.905
	Mean	0.122	1.096	17.295	0.03	0.511	0.665	9.86	0.562	6.057
	Max	0.183	1.353	24.727	0.044	0.829	0.937	15.923	0.828	11.155
64 kb/s	Min	0.153	0.985	9.535	0.037	0.678	0.557	6.129	0.53	4.521
	Mean	0.293	1.269	16.185	0.07	1.078	0.848	9.879	0.817	7.831
	Max	0.547	1.601	26.351	0.131	1.801	1.166	17.543	1.142	16.43
128 kb/s	Min	0.119	1.088	13.012	0.029	0.616	0.669	8.544	0.634	5.998
	Mean	0.208	1.323	21.845	0.05	1.059	0.886	13.295	0.833	10.288
	Max	0.388	1.547	31.076	0.093	1.804	1.103	20.409	1.062	19.154
256 kb/s	Min	0.11	1.078	14.599	0.026	0.561	0.652	9.672	0.608	5.131
	Mean	0.181	1.276	24.153	0.043	1.037	0.823	14.683	0.746	10.168
	Max	0.32	1.53	35.494	0.077	1.807	1.063	22.745	0.995	18.692

Table 6. *Overview of frame statistics of the enhancement layers of temporal scalability.*

information to be encoded in the enhancement layer. We also observe that the enhancement layers of the rate controlled layers tend to have a somewhat higher variability than the (10, 14, 16) single-layer encoding, which uses the same quantization parameters as the enhancement layer of the rate-controlled base layer.

Aggregate (Base + Enhancement Layer) Stream — Table 7 gives the traffic and quality statistics of the aggregate (base+enhancement layer) streams with temporal scalability. We observe that for the encodings without rate control, the aggregate stream statistics are approximately equal to the corresponding statistics of the single layer encodings (in Table 4). Indeed, we have verified that for encodings without rate control, extracting the I and P frames out of a single-layer encoding is equivalent to the base layer of a temporal scalable encoding. Extracting the B frames out of a single-layer encoding gives a stream equivalent to the enhancement layer of a temporal scalable encoding. This is to be expected since temporal scalable encoding adds essentially no overhead. The situation is fundamentally different for the temporal scalable encodings with rate control, where the rate-controlled base layer and the open-loop encoded enhancement layer are aggregated. If rate control is employed for the base layer encoding, the obtained base layer is very different from the I and P frame sequence of a single-layer encoding (both when the single layer is encoded with and without rate control). Similarly, the enhancement layer obtained from an actual temporal scalable encoding with a rate controlled base layer is quite different from the B frame sequence of a single-layer encoding, even though the enhancement layer of the temporal scalable encoding is coded with fixed quantization parameters.

Spatial Scalable Encoded Video

In this section we give an overview of the video traffic and quality statistics of spatial scalable encoded video, which are studied in detail in [46]. In the considered spatial scalable encoding the base layer provides the video in QCIF format. Adding the enhancement layer to the base layer gives the video in the CIF format. Table 8 gives an overview of the videos that have been studied for spatial scalability.

Base Layer — Table 9 gives an overview of the frame size and quality statistics of the base layers of the spatial scalable encodings. Focusing for now on the encodings without rate control, we observe again a concave shape of the coefficients of variation and the peak-to-mean ratios of both the frame sizes and (somewhat less pronounced) the GoP sizes with maxima at intermediate quality levels. Comparing these base layers which provide the video in the QCIF format with the single layer QCIF video in Table 4, we observe that the frame size, bit rate, and GoP size statistics are roughly the same. The observed differences are primarily due to considering a different set of videos in the spatial scalability study. A comparison for the individual videos [6] reveals that the traffic statistics of the QCIF base layer are typically almost identical to the corresponding statistics of the single-layer QCIF encodings.

Next consider the frame qualities of the base layer in Table 9. These qualities are obtained by up-sampling the QCIF base layer frames to CIF format and comparing these CIF frames with the original CIF frames. We observe that the PSNR qualities of these up-sampled base layer frames are

Encoding mode		Frame size			Bit rate		GoP		Frame quality	
		Mean $\overline{X}^{b+e}$ [kbyte]	CoV CoV_X^{b+e}	Peak/M. $X_{max}^{b+e}/\overline{X}^{b+e}$	Mean $\overline{X}^{b+e}/T$ [Mb/s]	Peak X_{max}^{b+e}/T [Mb/s]	CoV CoV_Y^{b+e}	Peak/M. $Y_{max}^{b+e}/\overline{Y}^{b+e}$	Mean $\overline{Q}^{b+e}$ [dB]	CoV $CoQV^{b+e}$
(4, 4, 4)	Min	1.881	0.399	4.097	0.451	3.606	0.284	2.707	35.996	0.162
	Mean	3.163	0.626	6.493	0.759	4.575	0.443	4.319	36.803	0.321
	Max	5.488	0.881	8.884	1.317	6.174	0.709	7.372	37.676	0.620
(10, 10, 10)	Min	0.61	1.021	9.382	0.146	1.918	0.538	6.072	30.786	0.353
	Mean	0.735	1.15	12.728	0.176	2.179	0.646	6.783	31.709	0.561
	Max	0.946	1.363	16.371	0.227	2.398	0.802	7.928	32.459	0.914
(10, 14, 16)	Min	0.332	1.174	10.659	0.08	1.586	0.445	3.731	28.893	0.418
	Mean	0.549	1.497	16.498	0.132	2.045	0.550	6.07	30.302	0.614
	Max	0.877	2.139	25.477	0.21	2.708	0.77	12.348	31.892	1.207
(24, 24, 24)	Min	0.228	1.044	11.569	0.055	0.784	0.455	4.443	26.538	0.438
	Mean	0.270	1.219	15.753	0.065	1.002	0.552	5.434	27.542	0.832
	Max	0.324	1.565	19.627	0.078	1.272	0.749	6.13	28.748	1.127
(30, 30, 30)	Min	0.191	0.833	8.208	0.046	0.556	0.395	3.076	25.17	0.394
	Mean	0.28	0.954	11.585	0.067	0.753	0.449	4.685	26.586	0.564
	Max	0.391	1.39	17.926	0.094	1.104	0.673	8.442	28.438	1.033
64 kb/s	Min	0.42	0.583	13.026	0.101	1.594	0.359	3.031	26.655	0.566
	Mean	0.56	0.893	20.469	0.134	2.723	0.422	4.807	28.713	0.783
	Max	0.814	1.229	32.596	0.195	4.511	0.473	7.982	31.351	1.439
128 kb/s	Min	0.652	0.817	7.495	0.157	1.357	0.176	2.18	28.207	0.572
	Mean	0.742	1.131	9.304	0.178	1.656	0.228	3.569	30.56	0.77
	Max	0.921	1.394	11.642	0.221	2.404	0.319	6.43	32.973	1.126
256 kb/s	Min	1.176	1.049	6.561	0.282	2.177	0.076	1.552	29.695	0.507
	Mean	1.248	1.245	8.698	0.3	2.593	0.109	2.356	32.196	0.713
	Max	1.387	1.391	10.578	0.333	2.987	0.168	4.032	34.316	0.954

Table 7. *Overview of frame statistics of the aggregate (base + enhancement layer) stream with temporal scalability.*

Class	Video	Genre	Quantization scale settings (from Table 2)
Movies	*Silence of the Lambs* *The Terminator I*	Drama Action	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4) (30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)
Sports	Snowboarding	Snowboarding competition	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)
Lecture and surveillance	*Lecture Martin Reisslein* *Parking Lot Cam*	Lecture Surveillance	(30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4) (30, 30, 30); (24, 24, 24); (10, 14, 16); (10, 10, 10); (4, 4, 4)

Table 8. *Overview of video sequences in CIF format considered in spatial scalability study.*

quite low compared to the single-layer QCIF frames. In fact, the mean frame qualities are quite similar to the PSNR qualities of the temporal base layer.

The traffic characteristics of the base layers with rate control are generally similar to the corresponding traffic statistics of the single-layer encodings. In particular, the rate controlled base layers exhibit quite significant traffic variability even at the GoP level (and in particular for small bit rates), which may require substantial over-provisioning or smoothing to reliably transmit the base layer. This is in contrast to the base layer of the temporal scalable encoding, which exhibited smaller traffic variability at the GoP level. The primary reason for this phenomenon is that, as noted earlier, the temporal base layer dedicates the entire target bit rate to the less variable (when viewed at the GoP level) I and P frames.

Enhancement Layer — From the summary of the statistics of the enhancement layer of the spatial scalable encodings in Table 10 we first observe for the encodings with fixed quantization scales that the mean frame sizes and bit rates of the enhancement layer are roughly three times larger than the corresponding base layer frame sizes and bit rates. This is to be expected as the enhancement layer stream increases the frame format from one quarter of the CIF format to the full CIF format. Next we observe that the coefficient of variation of the frame sizes and the GoP sizes of the enhancement layer as a function of the encoded quality level exhibit a concave shape with a maximum at an intermediate quality level and decreasing coefficient of variation and peak to mean ratio for lower quality levels. The peak-to-mean ratio of the frame sizes, on the other hand, only increases with increasing quantization scales (i.e., decreasing video quality). This effect is the subject of ongoing studies. Another noteworthy observation is that the GoP size variability of the enhancement layer is significantly larger than for the base layer (or the single-layer QCIF video), especially for larger quantization scales. This indicates that the enhancement layer is typically more difficult to accommodate in packet-switched networks.

Encoding mode		Frame size			Bit rate		GoP size		Frame quality	
		Mean $\overline{X}^b$ [kbyte]	CoV CoV_X^b	Peak/M. $X^b_{max}/\overline{X}^b$	Mean $\overline{X}^b/T$ [Mb/s]	Peak X^b_{max}/T [Mb/s]	CoV CoV_Y^b	Peak/M. $Y^b_{max}/\overline{Y}^b$	Mean $\overline{Q}^b$ [dB]	CoV $CoQv^b$
(4, 4, 4)	Min	1.868	0.463	3.167	0.448	3.396	0.245	2.348	19.465	0.883
	Mean	3.589	0.629	5.632	0.861	4.186	0.421	3.512	23.557	1.055
	Max	5.962	0.831	8.849	1.431	5.468	0.658	6.820	27.858	1.258
(10, 10, 10)	Min	0.494	0.782	4.523	0.119	1.670	0.322	3.197	19.414	0.890
	Mean	1.089	1.044	9.473	0.262	1.999	0.563	5.041	23.383	1.063
	Max	1.957	1.390	15.602	0.470	2.486	0.922	11.549	27.507	1.268
(10, 14, 16)	Min	0.338	1.216	6.620	0.081	1.608	0.299	3.279	19.385	0.895
	Mean	0.687	1.541	13.798	0.165	1.852	0.530	4.966	23.301	1.067
	Max	1.196	2.183	22.825	0.287	2.034	0.819	11.032	27.386	1.274
(24, 24, 24)	Min	0.233	0.859	5.702	0.056	0.708	0.252	2.989	19.105	0.914
	Mean	0.391	1.139	10.251	0.094	0.830	0.470	4.496	22.829	1.085
	Max	0.612	1.615	15.354	0.147	0.917	0.638	8.880	26.678	1.301
(30, 30, 30)	Min	0.201	0.786	5.922	0.048	0.553	0.212	2.753	18.940	0.924
	Mean	0.321	1.045	9.278	0.077	0.646	0.417	4.006	22.591	1.093
	Max	0.461	1.423	13.817	0.111	0.717	0.551	7.032	26.384	1.313
64 kb/s	Min	0.267	0.773	5.888	0.064	0.543	0.144	2.704	18.902	0.925
	Mean	0.340	1.224	15.823	0.082	1.160	0.371	4.146	22.686	1.086
	Max	0.446	2.107	32.089	0.107	2.088	0.545	7.036	26.659	1.315
128 kb/s	Min	0.534	1.042	11.533	0.128	1.478	0.039	1.427	18.959	0.904
	Mean	0.534	1.308	23.467	0.128	3.009	0.217	3.741	23.060	1.074
	Max	0.535	1.772	46.579	0.128	5.977	0.515	4.754	27.360	1.309
256 kb/s	Min	1.067	0.890	9.256	0.256	2.371	0.033	1.300	19.310	0.891
	Mean	1.067	1.122	11.051	0.256	2.831	0.049	1.607	23.367	1.063
	Max	1.067	1.494	14.739	0.256	3.775	0.081	2.410	27.641	1.279

Table 9. *Overview of frame statistics for the base layer of spatial scalability (CIF).*

Next, we turn to the enhancement layers corresponding to the base layers encoded with rate control. These enhancement layers are encoded with the fixed quantization scales (10, 14, 16). Similar to the encodings with temporal scalability, we observe that the average enhancement layer traffic decreases as the target bit rate for the base layer increases. We also observe that the variability of the enhancement layers corresponding to the rate controlled base layers is slightly higher than the variability of the enhancement layer of the encoding with fixed (10, 14, 16) quantization scales.

Aggregate (Base + Enhancement Layer) Stream — In Table 11 we summarize the traffic and quality statistics of the aggregate spatial scalable stream which gives the video in the CIF format. For comparison we provide in Table 12 the traffic and quality statistics of single-layer CIF format encodings of the videos. For the encodings without rate control, we observe that the aggregate spatial scalable video tends to have larger average frame and GoP sizes and bit rates as well as lower PSNR quality. This is primarily due to the overhead of spatial scalable encodings. In a more detailed study we determined this overhead by comparing the bit rates of aggregate spatial and single-layer encodings with essentially the same average PSNR quality to be approximately 20 percent [46]. Aside from this overhead, the statistics of the aggregate spatial scalable encodings and the corresponding single-layer CIF encodings are quite similar. Note, however, that the frame sizes and bit rates of the spatial scalable encodings with rate control are significantly larger than the corresponding frame sizes and bit rates of the single-layer CIF encodings. This is because the fixed target bit rate is allocated to the QCIF-sized base layer in the spatial scalable encodings, whereas it is allocated to the full CIF-sized video in the single-layer encodings.

Summary of Insights from Trace Statistics

We now briefly summarize the main insights from the video trace statistics presented in the preceding sections. We have learned that video traffic typically exhibits a concave shape of the traffic variability (in terms of the coefficient of variation or peak-to-mean ratio of the frame or GoP sizes) as a function of the encoded video quality, with a maximum in the traffic variability at an intermediate quality level. We have observed this phenomenon for single-layer encoded video as well as for the individual layers of layered encoded video. This phenomenon, which can critically affect the resource utilization in networks, requires the network engineer to pay attention to the quality level (or mix of quality levels) of the video that is to be transported over the network under study.

We have reconfirmed the fairly well known effect of long-range dependence in the video traffic. From our study of video encoded with rate control (both single-layer video and the rate controlled base layer of scalable video), we have found that rate controlled video may exhibit significant variability over short time scales (e.g., within a GoP) and typically has a small level of traffic variability over longer time scales. Rate controlled video may thus require a simple smoothing technique (e.g., traffic averaging for fixed time intervals) for the short-time scale traffic fluctuations and some additional smoothing over longer time scales to fit into constant bit rate channels. From our study of temporal scalable encoded video we have found that smoothing the transmission of the frames

Encoding mode		Frame size: Mean $\overline{X}^e$ [kbyte]	Frame size: CoV CoV_X^e	Frame size: Peak/M. $X^e_{max}/\overline{X}^e$	Bit rate: Mean $\overline{X}^e/T$ [Mb/s]	Bit rate: Peak X^e_{max}/T [Mb/s]	GoP size: CoV CoV_Y^e	GoP size: Peak/M. $Y^e_{max}/\overline{Y}^e$
(4, 4, 4)	Min	5.765	0.378	3.928	1.384	10.147	0.235	2.844
	Mean	10.451	0.506	5.965	2.508	13.210	0.402	3.555
	Max	17.793	0.757	8.654	4.270	16.773	0.658	6.182
(10, 10, 10)	Min	1.386	0.639	6.492	0.333	5.601	0.319	3.330
	Mean	2.869	0.833	12.056	0.689	6.891	0.596	5.461
	Max	5.280	1.247	16.844	1.267	8.227	1.001	10.585
(10, 14, 16)	Min	0.693	0.793	9.354	0.166	4.218	0.358	3.647
	Mean	1.480	1.001	17.652	0.355	5.114	0.671	6.436
	Max	2.698	1.423	25.621	0.647	6.056	1.085	12.425
(24, 24, 24)	Min	0.464	0.772	9.770	0.111	3.233	0.300	3.951
	Mean	0.931	0.919	20.141	0.223	3.770	0.621	6.304
	Max	1.559	1.218	29.009	0.374	4.539	0.916	10.941
(30, 30, 30)	Min	0.373	0.728	11.456	0.090	2.859	0.273	3.958
	Mean	0.729	0.881	21.918	0.175	3.294	0.589	6.228
	Max	1.152	1.103	31.906	0.276	3.910	0.819	9.969
64 kb/s	Min	0.776	0.822	8.661	0.186	4.245	0.374	3.648
	Mean	1.679	1.037	15.589	0.403	5.182	0.649	6.211
	Max	2.981	1.369	22.801	0.716	6.197	1.068	12.221
128 kb/s	Min	0.704	0.831	8.678	0.169	4.226	0.379	3.952
	Mean	1.602	1.041	16.945	0.385	5.173	0.698	6.736
	Max	2.965	1.506	25.145	0.712	6.175	1.201	13.949
256 kb/s	Min	0.676	0.815	9.142	0.162	4.204	0.355	4.249
	Mean	1.484	1.046	18.077	0.356	5.144	0.714	7.161
	Max	2.797	1.556	27.201	0.671	6.137	1.197	15.102

Table 10. *Overview of frame statistics of the enhancement layers of spatial scalability.*

of one layer over the gaps created by the frames in the other layer (e.g., the three-frame smoothing in the context of the form of temporal scalable encoding considered earlier) reduces the variability of the layer traffic dramatically.

An important aspect of the video trace statistics that was not explicitly studied in the preceding sections is the content dependency of the statistics. For a compact presentation we have reported the minimum, mean, and maximum across the videos in Table 1 for each statistical metric. It is important to keep in mind, however, that the traffic and quality statistics of the encoded video depend on the video content and generally differ according to the content quite significantly from video to video, as indicated by the Min to Max ranges in the above tables. When evaluating a network it is thus important to consider a mix of videos that is representative of the typical mix of videos supported by the network. We also note that there are additional genres of videos with content quite different from the genres considered here. One such example are lecture videos employed in distance education. These videos have quite different content dynamics, resulting in correspondingly different traffic and quality statistics from the entertainment and sports videos considered here (see [6]). This needs to be taken into consideration when designing and evaluating networks dedicated to special applications, such as distance education.

Using Video Traces in Network Performance Evaluations

In this section we discuss the issues involved in using video traces in network performance evaluation studies. We focus primarily on how to use the traces in simulations, but our discussions apply analogously for using traces as a basis for traffic modeling. Our focus throughout this section is on the aspects that are unique to simulations using video traces. For general instructions on how to conduct simulations we refer to the standard simulation textbooks, for example, [47, 48].

There are three broad areas that require special consideration when using video traces in simulations: the definition of the video-related performance metrics; the generation of the video traffic work load for the system under study; and the statistically sound estimation of the performance metrics of interest. We discuss these three areas in the following three subsections. We note that the purpose of our discussions is not to develop a specific simulation design to evaluate a specific set of performance metrics for a particular network set-up. Instead, our goal is to explain the issues and considerations involved in simulations using video traces in general terms, so as to enable the reader to design simulations that use video traces for the specific networking systems of interest to the reader and to obtain meaningful insights from such simulations.

Video Related Performance Metrics

Simulations with video traces can be used to evaluate conventional network performance metrics, such as the utilization of networking resources, the delay and delay jitter, buffer occupancies, and buffer overflow probabilities, for networks carrying video traffic. In addition, simulations with video traces can be used to evaluate performance metrics that are related to the video, namely the starvation probability and the video quality. These two metrics give some indication of the quality of the video delivered to the user over the network under study.

Encoding mode		Frame size			Bit rate		GoP		Frame quality	
		Mean $\overline{X}^{b+e}$ [kbyte]	CoV CoV_X^{b+e}	Peak/M. $X_{max}^{b+e}/\overline{X}^{b+e}$	Mean $\overline{X}^{b+e}/T$ [Mb/s]	Peak X_{max}^{b+e}/T [Mb/s]	CoV CoV_Y^{b+e}	Peak/M. $Y_{max}^{b+e}/\overline{Y}^{b+e}$	Mean $\overline{Q}^{b+e}$ [dB]	CoV $CoQV^{b+e}$
(4, 4, 4)	Min	7.633	0.394	3.681	1.832	10.585	0.235	2.747	30.679	0.913
	Mean	14.040	0.509	5.403	3.370	16.286	0.404	3.507	35.994	1.170
	Max	23.754	0.730	8.573	5.701	20.983	0.656	6.338	37.846	1.307
(10, 10, 10)	Min	1.880	0.653	5.626	0.451	5.986	0.318	3.196	30.553	1.105
	Mean	3.958	0.836	10.041	0.950	7.954	0.582	5.287	32.493	1.174
	Max	7.237	1.165	16.070	1.737	9.771	0.975	10.839	33.990	1.278
(10, 14, 16)	Min	1.058	0.837	8.134	0.254	4.264	0.330	3.465	27.840	1.072
	Mean	2.167	1.068	13.540	0.520	5.911	0.618	5.911	30.350	1.155
	Max	3.893	1.370	22.175	0.934	7.601	0.992	11.981	32.398	1.268
(24, 24, 24)	Min	0.698	0.756	8.340	0.167	3.303	0.281	3.489	25.216	1.058
	Mean	1.322	0.903	14.732	0.317	4.078	0.569	5.704	28.116	1.151
	Max	2.171	1.045	19.949	0.521	4.742	0.823	10.299	30.571	1.280
(30, 30, 30)	Min	0.575	0.728	8.906	0.138	2.920	0.248	3.483	24.007	1.050
	Mean	1.051	0.845	15.610	0.252	3.507	0.529	5.426	27.080	1.149
	Max	1.613	0.913	21.171	0.387	4.173	0.714	8.990	29.744	1.286
64 kb/s	Min	1.043	0.805	8.139	0.250	4.308	0.291	3.494	27.752	1.071
	Mean	2.020	1.012	12.892	0.485	5.448	0.577	5.614	30.231	1.157
	Max	3.428	1.338	17.688	0.823	6.696	0.875	10.565	32.299	1.273
128 kb/s	Min	1.238	0.773	8.260	0.297	4.311	0.217	3.243	27.762	1.057
	Mean	2.136	0.957	11.802	0.513	5.504	0.507	5.163	30.375	1.149
	Max	3.500	1.263	15.463	0.840	6.937	0.712	9.236	32.645	1.271
256 kb/s	Min	1.743	0.704	8.300	0.418	4.283	0.140	2.407	27.868	1.049
	Mean	2.551	0.846	9.900	0.612	5.921	0.381	4.217	30.580	1.143
	Max	3.864	1.069	11.251	0.927	8.434	0.481	6.710	32.988	1.261

Table 11. *Overview of frame statistics of the aggregate (base + enhancement layer) stream with spatial scalability (CIF).*

Starvation Probability — Starvation (loss) probability comes in two main forms. The *frame starvation probability* is the long-run fraction of video frames that miss their decoding (playout) deadline, that is, those frames that are not completely delivered to the receiver by the time the receiver needs them to start the decoding. The frame starvation probability may be estimated for individual clients or for the complete system under study.

The *information loss probability* is the long-run fraction of encoding information (bits) that misses its decoding (playout) deadline. The information loss probability has a finer granularity than the frame loss probability because a partially delivered frame is considered as one lost frame toward the frame loss probability (irrespective of how much of the frame was delivered/not delivered in time), whereas the information loss probability counts only the fraction of the frame's information bits that were not delivered in time. As an illustrative example consider the transmission of 10 frames each of size 240 bits to a client, and suppose only 120 bits of the first frame are delivered on time (and the other 120 bits arrive after the decoding deadline). Also suppose the remaining nine frames are all completely delivered ahead of their respective decoding deadlines. In this scenario the frame loss probability is 1/10 = 10 percent, whereas the information loss probability is 120/(10 · 240) = 5 percent. We note that in this example and throughout this discussion so far on the loss probability, we have ignored the dependencies between the encoded video frames. Specifically, in an MPEG encoding, the I frame in a GoP is required to decode all other P and B frames in the GoP (as well as the B frames in the preceding GoP encoded with reference to the I frame starting the next GoP). Thus, the loss of an I frame is essentially equivalent to the loss of all the frames in the GoP (as well as some frames in the preceding GoP). Similarly, a given P frame is required to decode all the successive P frames in the same GoP as well as the B frames encoded with respect to these P frames. Thus, the loss of a P frame is equivalent to the loss of all these dependent frames.

The information loss probability is mainly motivated by error concealment and error resilience techniques [49] that allow for the decoding of partially received video frames. Error resilience techniques are currently a subject of intense research efforts, and more advances in this area are to be expected. The deployment of these techniques may be affected by the required computational effort and energy, which are often limited in wireless devices.

Video Quality — The frame loss probability and information loss probability are convenient performance metrics for video networking as they can be directly obtained from network simulation with video traces. However, these loss probabilities are to a large extend still "network" metrics and provide only limited insight into the video quality perceived by the user. It is certainly true that a smaller loss probability corresponds in general to a higher video quality. However, it is difficult to quantify this relationship, because the rate-distortion curves of encoders relate only the bit rates of completely received streams (layers) to the corresponding PSNR video quality. Hence, we should keep in mind that the PSNR provides only a limited, albeit widely used, characterization of the video quality. If a part of a stream (layer) is lost, the video quality can no longer be obtained from the encoder rate-distortion curve. In general, experiments with actual encoders, decoders,

Encoding mode		Frame size			Bit rate		GoP		Frame quality	
		Mean $\overline{X}$ [kbyte]	CoV CoV_X	Peak/M. $X_{max}/\overline{X}$	Mean $\overline{X}/T$ [Mb/s]	Peak X_{max}/T [Mb/s]	CoV CoV_Y	Peak/M. $Y_{max}/\overline{Y}$	Mean $\overline{Q}$ [dB]	CoV CoQV
(4, 4, 4)	Min	6.419	0.402	4.150	1.541	9.649	0.221	2.759	37.025	1.100
	Mean	11.289	0.542	5.727	2.709	14.099	0.388	3.629	37.654	1.189
	Max	17.832	0.742	8.271	4.280	17.760	0.620	6.290	38.303	1.232
(10, 10, 10)	Min	1.596	0.710	6.422	0.383	5.434	0.311	3.664	30.989	0.935
	Mean	3.329	0.943	10.379	0.799	7.149	0.561	5.401	32.867	1.120
	Max	5.546	1.221	14.506	1.331	8.548	0.914	10.163	34.337	1.243
(10, 14, 16)	Min	1.074	0.970	8.888	0.258	4.458	0.291	3.741	29.305	0.925
	Mean	2.172	1.296	13.092	0.521	6.012	0.550	5.502	31.585	1.087
	Max	3.411	1.915	18.782	0.819	7.277	0.835	9.653	33.423	1.176
(24, 24, 24)	Min	0.706	0.790	8.647	0.170	3.079	0.252	3.580	25.975	0.978
	Mean	1.382	0.975	12.628	0.332	3.846	0.498	5.112	28.896	1.112
	Max	1.900	1.336	18.159	0.456	4.618	0.651	7.654	31.384	1.248
(30, 30, 30)	Min	0.657	0.733	9.193	0.158	2.733	0.215	3.346	24.849	1.002
	Mean	1.201	0.881	12.408	0.288	3.364	0.446	4.642	27.965	1.116
	Max	1.530	1.156	17.327	0.367	4.078	0.569	6.333	30.677	1.255
64 kb/s	Min	0.653	0.720	9.221	0.157	2.675	0.210	3.322	24.708	1.002
	Mean	1.184	0.865	12.297	0.284	3.294	0.440	4.584	27.846	1.116
	Max	1.497	1.126	17.072	0.359	3.968	0.562	6.208	30.591	1.257
128 kb/s	Min	0.653	0.720	9.221	0.157	2.674	0.211	3.322	24.708	1.002
	Mean	1.184	0.865	12.295	0.284	3.294	0.440	4.584	27.847	1.116
	Max	1.497	1.126	17.065	0.359	3.968	0.562	6.207	30.595	1.257
256 kb/s	Min	1.067	0.722	9.618	0.256	3.457	0.101	2.280	24.711	1.001
	Mean	1.303	1.024	20.731	0.313	6.095	0.401	5.098	28.642	1.093
	Max	1.497	1.741	49.493	0.359	13.131	0.562	9.908	31.626	1.256

Table 12. *Overview of frame statistics of the single layer stream (CIF).*

and video data are required to obtain the video quality after lossy network transport.

There are, however, scenarios in which it is possible to obtain the approximate PSNR video quality after lossy network transport. One such scenario is the network transport of layered encoded video with priority for the base layer, that is, the enhancement layer data is dropped before the base layer data when congestion arises. First consider temporal scalable encoded video in this context. If an enhancement layer frame is completely received (and all the frames that are used as encoding references are also completely received), then the PSNR quality of the frame is obtained by adding the base layer PSNR quality of the frame (from the base layer trace) and the enhancement layer PSNR quality improvement of the frame (from the enhancement layer trace). If all the referenced frames are completely received and a part of or all of the enhancement layer is lost, then one can (conservatively) approximate the quality of the frame by the PSNR quality of the base layer trace. If a part or all of a frame that serves as a reference frame for the encoding of other frame(s) is lost, for example, a P frame (in the base layer) of the encoding considered in Fig. 2, then all frames that depend on the (partially) lost reference frame are affected. The quantitative impact of such a loss can currently only be determined from experiments with the actual video. Note that quantitatively capturing such losses in traces would require a separate trace for each possible combination of reference frame loss (e.g., only the last P frame in the GoP is lost, only the second to last P frame is lost, etc.), in conjunction with different error concealment mechanisms. For network researchers using the currently available traces it appears reasonable to approximate the PSNR quality of the lost reference frame and all dependent frames by a very small PSNR value, e.g., less than 20 dB. In summary, the impact of losses of enhancement layer frames (without any dependent frames), for example, losses of B frames in Fig. 2, can be assessed with fairly reasonable accuracy using the PSNR values in the trace structures outlined earlier. If frames that are referenced by other frames suffer losses, then the impact is very difficult to assess and only very rough approximations can be made. Generally, when transporting video it is recommended to stay within an operating regime where losses are limited to B frames, since losses of I and P frames typically deteriorate the video quality quite significantly.

Next consider scalable encoded video where each video frame has a base layer component and an enhancement layer component, for example, the spatial scalable encoding considered earlier. If a frame is completely received, then the PSNR quality of the received frame is the PSNR quality of the base layer frame (from the base layer trace) plus the PSNR quality improvement of the enhancement layer (from the enhancement layer trace). If the base layer component of the frame is completely received but a part (or all) of the enhancement layer of the frame is lost, then one can approximate the quality of the received frame by the PSNR quality of the base layer frame. Finally, if a part (or all) of the base layer is lost, then one has to roughly approximate the quality of the received frame by a very small PSNR value. This discussion so far has ignored frame dependencies, which are illustrated in Fig. 3 for a typical spatial scalable encoding scenario. Assessing the impact of losses in a frame component that is referenced by some other frame requires experiments with actual videos. For simulations using traces, it is again recommended to stay in an operation range that completely avoids the loss of refer-

enced frame components.

Another scenario in which one can assess the video quality of the received video after lossy network transport is transcoding (also referred to as the cropping scenario [50]). In this scenario single-layer encoded video is transported through a network. Whenever congestion arises the video is transcoded [51] to a lower quality (corresponding to a larger quantization scale, so that the transcoded video fits into the available bandwidth). This scenario can be (approximately) simulated using the single-layer video traces by switching to the trace of a lower-quality encoding of the same video.

To conclude this section on video quality as a performance metric in video trace simulations, we note that the received video quality is generally maximized by maximizing the average frame quality and minimizing the quality variations. More specifically, the received video quality is maximized by maximizing the qualities of the individual video frames and minimizing the variations in quality between consecutive video frames.

Generating Video Traffic Workload from Traces

In this section we discuss how to generate a video traffic workload for a network under study from traces. When generating the video traffic workload there are a number of issues to consider. These issues range from choosing and preparing the video streams (traces) to the packetization of the video frames. We first address the issues at the stream level and then turn to the issues at the level of individual video frames and packets.

Stream-Level Considerations

Selecting the Videos (Titles) — The first consideration at the stream level is typically to select the videos (titles) to be used in the evaluation. As alluded to earlier, it is important to consider the content of the videos that will be transported over the network under study. If the network is being designed for the transport of lecture videos, for instance, then traces of lecture videos should be used in the simulations. Generally, it is advisable to select as many different videos as possible (available) from the video genre(s) that will be transported over the network. Let M denote the number of different videos selected for a given evaluation study.

Composing the Workload — Next, one needs to decide how to compose the workload from the selected set of video traces. The main consideration in composing the workload is typically whether or not the networking protocol or mechanism under evaluation exploits localities of reference. A video caching mechanism, for instance, relies on localities of reference and strives to improve the network performance by caching the most frequently requested videos. A scheduling mechanism for a router output port, on the other hand, typically does not exploit any locality of reference. Thus, for evaluations of protocols and mechanisms that exploit localities of reference the workload should be composed according to the appropriate distribution. For example, studies of streaming media servers, for example, [52], indicate that video popularity follows a Zipf distribution [53]. More specially, if there are M videos available, with video 1 being the most popular and video M being the least popular, then the probability that a given request is for the mth most popular video is

$$\frac{K}{m^{\zeta}}, \qquad m = 1,\ldots,M, \tag{1}$$

where

$$K = \frac{1}{1+\frac{1}{2^{\zeta}}+\ldots+\frac{1}{M^{\zeta}}}. \tag{2}$$

The Zipf distribution is characterized by the parameter $\zeta \geq 0$. The larger ζ, the more localized the Zipf distribution, that is, the more popular is the most popular video. In an initial measurement study requests for streaming videos were found to be distributed according to a Zipf distribution with ζ around 0.5 [52]. It has been observed that the request for movies in video rental stores and video-on-demand systems are well described by a Zipf distribution, with ζ in the vicinity of 1 [54]. Furthermore, studies of Web caches indicate that requests for HTML documents and images follow approximately a Zipf distribution, with ζ in the vicinity of 1 [55]. It is therefore reasonable to expect that requests for streaming videos generally follow a Zipf distribution with ζ in the range between 0.5 and 1.

If locality of reference plays no role in the studied network protocol it is reasonable to select the videos according to a discrete uniform distribution $U[1, M]$, that is, each video is equally likely selected with probability $1/M$ to satisfy a client request. This uniform random video selection ensures that the traffic patterns in the selected mix of M videos are roughly uniformly "experienced" by the protocol under study.

Select Encoding Mode — The next step in setting up a simulation study is typically the selection of the appropriate encoding mode(s) for the individual videos. The choice of appropriate encoding mode, that is, single-layer or scalable encoded video, with or without rate control, depends largely on the particular protocol or mechanisms under study. We provide here a few general considerations and recommendations.

Generally, one should avoid scaling the video traces. By scaling we mean multiplying the size of each individual video frame by a constant to adjust the average bit rate of the video trace to some desired level. Scaling generally does not provide valid traces for the desired average bit rate. To see this, consider scaling a trace for the single-layer (4, 4, 4) encoded video with high quality and bit rate to smaller bit rates see (Table 4). To obtain average bit rates of the trace of the (30, 30, 30) encoded video, for instance, one would need to divide the size of every frame in the (4, 4, 4) trace by approximately 10. The (4, 4, 4) trace scaled in this manner would have the average bit rate of a (30, 30, 30) trace, but the variability (*CoV* and peak-to-mean ratio) of the scaled (4, 4, 4) trace would still be the same as for the original (4, 4, 4) trace. The variability of the (4, 4, 4) trace, however, is quite different from the variability of a (30, 30, 30) trace, as is evident from Table 4. It is therefore generally recommended to avoid scaling the traces.

Nevertheless, for some evaluations it may be desirable and convenient to use traces for rate controlled video with a different bit rate than available. For other evaluations it may be convenient to use traces for different open-loop encoded videos, with the same average bit rate of some prespecified level. With scaling, each open-loop encoded video (title) contributes equally to the system utilization, which makes it easy to maintain a prespecified constant utilization with a mix of different videos. For these reasons it may be necessary to scale traces before using them in network simulations. In such situations it is recommended to use the trace with the average bit rate closest to the desired bit rate so that the scaling factor is as close to 1 as possible.

Constant Utilization Simulation Scenario — We conclude this discussion of the stream-level issues by outlining the trace usage in two streaming scenarios, which may be useful for the reader in setting up his/her own networking simulation study. First we outline a "constant utilization" scenario. Suppose we wish to examine the performance of a multiplexer, scheduler, or similar network system that is fed by several streams for a specific long-run average utilization level. Furthermore, suppose that we wish to examine the system performance for open-loop VBR encoded video titles and scaled the closest traces to a common average bit rate $\overline{X}/T$. Let J denote the number of simultaneous video streams required to achieve a desired level of system utilization $J \cdot \overline{X}/(C \cdot T)$, where C denotes the capacity of the system. For each of the J video streams we uniformly randomly select one of the M traces. For each selected trace we independently draw a starting (frame) phase from a discrete uniform distribution $U[1, N]$ over the N frames in the trace. The video frames are then processed according to the protocol or mechanism under study from the starting frame onward.

The next question that arises is for how long, that is, how many frames, should the mechanism under study be simulated? One option is to continue the simulation for N frames, that is, for the full length of the traces. (Note that due to the random starting frame, the end of the traces may be reached before processing all N frames. When the end of a trace is reached the trace is "wrapped around," that is, the processing continues from the beginning of the trace.) Once all N frames have been processed, we immediately randomly select a new trace and starting phase into the trace for each of the J streams. Thus there are always J streams in progress.

There are a number of variations of the outlined constant utilization simulation, which may be appropriate depending on the protocol under study. One variation is to not continue the simulation after all N frames of a trace have been processed, but to draw a random independent stream duration (bounded by N) instead. With this approach one can study the effect of new streams starting up and the stream duration (lifetime) by varying the distribution used to draw the random stream duration.

Another variation is to use the original unscaled traces to achieve a constant utilization. This is achieved by fixing the composition $J_1, J_2, \ldots, J_M$ of the streams that achieves a specific utilization,

$$\frac{\sum_{m=1}^{M} \frac{J_m \cdot \overline{X}_m}{T}}{C}.$$

With this approach the videos are not chosen randomly. Instead, there are always J_m streams with video m ongoing. For each stream a random uniform start phase into the corresponding trace is selected. When all the frames of a given trace have been processed or a stream's lifetime expires, the same video is immediately started up, but with a new independent random starting phase. Thus, with this approach the number of ongoing streams of each video title is deterministic, but the traffic is random due to the random phase profiles. The advantage of this approach is that it avoids the scaling of the videos and allows for studies with streams with heterogeneous average bit rates.

We conclude this discussion of the constant utilization approaches by noting that they are appropriate to examine performance metrics at the packet-level and burst-level time scale, such as packet loss and delay. However, the constant utilization approaches are not suitable for examining call-level metrics, such as call blocking probabilities. Therefore, we outline next a "varying utilization" simulation scenario that is appropriate for call-level evaluations, as they are required for call admission control and caching mechanisms, for instance.

Varying Utilization Simulation Scenario — To illustrate the "varying utilization" simulation scenario, suppose we wish to examine the performance of a call admission or caching mechanism that processes incoming requests for video streams. Depending on the current system load, cache contents, and traffic characteristics of the currently supported streams and the requested stream, the new request is either granted or denied.

Suppose that we have selected a set of M video traces for the evaluation. To run the simulation we need to generate requests according to some stochastic process. The Poisson process, in which the time between successive arrivals is exponentially distributed, is generally a good model for request arrivals. For each new client request we draw independently the video (e.g., according to a uniform or Zipf distribution), the starting phase, and the lifetime (duration) of the stream. Whenever the end of a stream lifetime is reached, the stream is simply removed from consideration, freeing up the system resources it occupied. The distribution of the lifetime (for which the exponential distribution is generally a good choice) and the request arrival process are adjusted to achieve the desired load level of the system. To illustrate the load-level adjustment consider a system with capacity C b/s to which requests for (scaled) video streams with an average bit rate of $\overline{X}/T$ arrive, and suppose each accepted video stream consumes the bandwidth $\overline{X}/T$ of the available bandwidth C. The stability limit of such a system is $J_{\max} = C \cdot T/\overline{X}$ streams. Let L denote the mean of the lifetime distribution in frame periods and let ρ denote the mean request arrival rate in requests per frame period. The long run average fraction of calls (requests) that can be accepted is given by

$$\frac{\frac{1}{\rho}}{\frac{L}{J_{\max}}}. \qquad (3)$$

To see this, note that $1/\rho$ is the average spacing between request arrivals in frame periods, and $L/J_{\max}$ is the average spacing in frame periods between call departures (streams reaching the end of their lifetime) when the system is fully loaded. We considered scaled video streams for this illustrative calculation of the load level, because some mechanisms may give preference to requests according to the average bit rate of the requested stream. With such a preferential granting of requests, the average of the average bit rates of the currently supported streams may be quite different from the average of the average bit rates of the stream requests.

In concluding this discussion of the "varying utilization" simulation scenario, we point out one subtle issue with the average bit rates of the streams. The average bit rate of an original or scaled trace is calculated over all N frames of the trace. When generating a video stream from a trace by drawing a starting phase from a discrete uniform distribution $U[1, N]$ over all frames in the trace, and a random lifetime, the average bit rate of a given thus generated stream may be quite different from the average bit rate of the trace. In particular, the average stream bit rate may be quite different from the average trace bit rate if the lifetime is relatively short compared to the length of the trace. This is because a short lifetime may "sample" a part of the trace that has unusual characteristics compared to the overall trace. (It should also

be noted that in the opposite extreme with a lifetime significantly longer than the trace, and wraparound whenever the end of the trace is reached, the generated stream contains duplicate traffic patterns.) One way to enforce a desired average bit rate for each individual stream generated from a trace is to scale the randomly selected video trace segment (from the starting phase onward until the end of the stream lifetime). Such per-stream scaling, however, is computationally demanding and, as noted above, may falsify the true variability characteristics. On the other hand, by generating many (short) streams from a given trace (without any per-stream scaling) the average bit rate of the streams converges to the average bit rate of the trace. It is recommended to keep these subtleties in mind when designing and evaluating a simulation study using video traces.

Frame/Packet Level Issues — In this section we discuss the issues arising at the level of individual video frames and network packets (e.g., IP packets, data link-layer frames). A key consideration at the frame level is to determine whether a frame meets its playout deadline. This is especially important when the frame or information starvation probability is one of the considered performance metrics. Recall that the decoder in the video client consumes the frames in the codec sequence and displays the frames in the display sequence on the screen. The client suffers playout starvation when it wants to start the decoding of a video frame but has not yet fully received that frame or its reference frame(s). The client may use error concealment techniques [49] to conceal the missing video information. The simplest technique is to continue displaying the last fully and on-time received frame. There is a range of more sophisticated techniques that attempt to decode partially received frames or extrapolate the missing frame from preceding frames.

A related consideration is that for many networking studies it may be preferable to simulate the transmission of frames in the IBBP… order, because the GoPs are successively transmitted with this frame order. With the IPBB… order, on the other hand, the I frame of the second GoP is transmitted before the last two B frames of the first GoP. Consequently, there is a combined total of nine P and B frames transmitted between the first two I frames and a total of 11 P and B frames between all successive I frames. This may lead to difficulties for mechanisms that smooth the video frames in individual GoPs, and also for mechanisms that exploit specific alignments of the I frames in the supported streams.

In addition, it should be noted that for many networking studies it may be appropriate to consider start-up delays introduced by the networking protocol under study in isolation from the playout commencement delay due to the MPEG encoder (discussed earlier and illustrated in Fig. 4). For such studies it may very well be appropriate to assume that the first frame (I frame) is decoded and displayed at a time governed by the network protocol and the subsequent frame (B frame, when using the IBBP ordering) is independently decoded and then displayed when the frame period of the I frame expires. With such a simulation, the playout commencement delay due to the MPEG frame encoder order is added to the network-introduced start-up delay and possibly other delay components (e.g., server delay) to give the total start-up delay experienced by the user.

Packetization — For the transport over packet-switched networks the video traffic is typically packetized, that is, the video data is packaged into packets. In general, the packetization strategy of choice can be selected from a large set of alternatives depending on the overall objective and set-up of a specific simulation. To illustrate the issues involved in the packetization of the video traffic, we discuss the packetization of the video traffic in the context of the Real Time Protocol (RTP) [56]. An RTP packet consists of the 12-byte RTP header, an 8-byte UDP header, and 20-byte IPv4 header/40-byte IPv6 header. (When TCP is used for the video transport a 20-byte TCP header is used instead of the UDP header.) The packetization of MPEG-4 encoded video into RTP packets is described in RFC 3016 [57]. This RFC recommends that a given RTP packet carries data from only one video frame, such that the loss of an RTP packet will affect only one video frame. The amount of video data in an RTP packet should be adjusted such that the complete RTP packet (consisting of video data plus headers) is no larger than the maximum transfer unit (MTU) on the path through the network to avoid fragmentation in the network (except for wireless links that may perform fragmentation of the RTP packet carried over the wired network). In case the video frames are small it is permitted to carry multiple consecutive video frames in one RTP packet.

We note that the packet headers may contribute significantly to the total traffic, especially when low bit rate video streams are transmitted with tight real-time constraints that prohibit the grouping of multiple frames into one RTP packet. Header compression schemes have been proposed to limit the waste of bandwidth due to protocol headers in such situations (see, for example, [58]).

It should also be noted that with scalable (layered) encoded video, each layer is typically packetized independently to allow for the different treatment of the layers in the network (e.g., at the IP level). Furthermore, we note that the video traces reflect only the video data; typical video display, however, consists of video and audio. The bit rate of the encoded audio is in many scenarios negligible compared to the bit rate of the encoded video (see, for example, [59]). The encoded audio stream, however, is typically packetized independently from the video. This packetized audio stream may make a significant contribution to the total (video + audio) traffic.

Packet Transmission — A final consideration at the packet level is the transmission of the individual packets. First consider the simple case in which one packet carries a complete video frame. Depending on the overall simulation setup the packet may be sent at once, which may be appropriate for a packet-level simulation that keeps track of the individual packets but not the individual bits. For a fluid traffic simulation running at the granularity of frame periods, on the other hand, it may be appropriate to transmit a packet of size S bits at the constant bit rate S/T b/s over the duration of one frame period of length T secs.

If a single video frame is packetized into multiple packets, it may be appropriate (depending on how the simulation is set up) to space out the transmission instants of the individual packets equally over one frame period in a packet-level simulation, whereas in a fluid simulation the aggregate size of all the packets could be transmitted at a constant bit rate over one frame period.

Finally, consider the case in which multiple video frames are packetized into a single packet into a fluid simulation. Depending on the simulation scenario, it may be preferable to transmit this single packet over one frame period (e.g., in a real-time scenario), or to transmit it over as many frame periods as there are video frames in the packet (e.g., in a non-real-time scenario).

Estimating Performance Metrics

In this section we discuss the analysis of the output of a simulation involving video traces in order to draw meaningful conclusions about the networking system, protocol, or mechanisms under study. As with any simulation, a key consideration when simulating a network mechanism or protocol using video traces is the statistical validity of the obtained results. We refer the reader to standard simulation texts (e.g. [47, 48]) for general instructions on how to obtain statistically meaningful simulation results and focus here primarily on the aspects unique to simulation using video traces.

Video traces, in general, and the constant utilization and varying utilization simulation scenarios outlined earlier lend themselves both to terminating simulations and steady-state simulations. In terminating simulations, as defined in [48], several independent simulation runs are performed and the estimates of the metrics of interest are obtained by averaging the metric estimates obtained from the individual runs. A terminating simulation of the constant utilization scenario can be conducted by running several simulations, as outlined above. Each simulation is started with independently randomly selected traces, starting phases (and possibly stream lifetimes). The advantage of this terminating simulation approach is that the individual simulation runs are independent and thus the classical student t or normal distribution-based statistics can be used to evaluate the confidence intervals around the estimated sample means.

The disadvantage of the terminating simulation approach is that each simulation run needs to be "warmed up" sufficiently to remove the initial transient. While this is not a problem for system simulations that do not require any warm-up (e.g., the simulation of a bufferless multiplexer for a constant utilization), the warm-up may be a significant problem for systems that need warm-up (e.g., buffered multiplexers). This problem of warming up simulations driven by self-similar input is to the best of our knowledge an open problem. We therefore only note that it is widely expected that the transient period is longer when driving simulations with self-similar input traffic and that the conventional methods (e.g., [60]), may underestimate the required warm-up period. One way to mitigate this warm-up problem is to start up the entire system in steady state (in case it is known) or at least to start up the traffic load of the system at (or close to) the steady state load.

Next we consider steady-state simulations where, as defined in [48], a single (typically very long) simulation run is performed and the metrics of interest are typically obtained by averaging metric estimates obtained during independent observation periods (usually referred to as batches). A steady-state simulation with video traces can be conducted by running one long constant utilization simulation as outlined above or one long varying utilization simulation as outlined above. The advantage of the steady-state simulation is that the warm-up period (during which the system is not observed) is incurred only once. The challenge of the steady-state simulation of systems with video traces is that due to the long-range dependence in the video traffic, the metric estimates of successive (non-overlapping) observation periods (batches) are typically somewhat correlated. The problem of estimating confidence intervals from these batches has received some initial interest (e.g., the studies [61, 62]) to which we refer for details on the estimation methods.

We note that a simple heuristic to obtain uncorrelated batches despite long-range dependent video traffic is to separate successive observation periods (batches) such that they are (approximately) independent. More specifically, the heuristic is to run the constant utilization or varying utilization simulation and to truncate the distribution of the stream duration at a specific value Δ. Then, separating successive batches by at least Δ will ensure that none of the video streams that contribute to the traffic load during a given batch contributes to the traffic load during the next batch. This ensures that the successive batches are independent, provided the system under study has only a small amount of "memory." This heuristic provides a simple method to obtain statistically meaningful performance metrics at the expense of increased simulation duration.

Conclusion

In this tutorial we have explained how to evaluate the network performance for single-layer and two-layer encoded video using traces. We have given an overview of a library of traces of single-layer encoded video and video encoded in two layers using the temporal and spatial scalability modes. We have outlined a procedure for conducting network simulations using the traces, and we have explored the analysis of the output of such simulations.

Throughout this tutorial we have made an effort to keep the discussions general to ensure this tutorial is relevant and useful for traces of all types of single-layer and multi-layer encoded video. In addition, we have strived to provide generally valid yet detailed instructions that enable the reader to conduct simulations for any networking architecture, protocol, or mechanism.

Acknowledgment

We are grateful to Associate Editor-in-Chief John Daigle and the anonymous reviewers for their detailed and thoughtful comments on earlier versions of this article, which have greatly improved its quality. We are grateful to Osama Lotfallah and Sethuraman Panchanathan for help with the encoder software.

References

[1] W.-C. Feng, "Video-on-Demand Services: Efficient Transportation and Decompression of Variable Bit Rate Video," Ph.D. dissertation, University of Michigan, Apr. 1996.

[2] —, *Buffering Techniques for Delivery of Compressed Video in Video-on-Demand Systems*, Kluwer Academic Publisher, 1997.

[3] M. W. Garret, "Contributions Toward Real-Time Services on Packet Networks, "Ph.D. dissertation, Columbia University, May 1993.

[4] M. Krunz, R. Sass, and H. Hughes, "Statistical Characteristics and Multiplexing of MPEG Streams," *Proc. IEEE Infocom '95*, Boston, MA, April 1995, pp. 455–62.

[5] O. Rose, "Statistical Properties of MPEG Video Traffic and their Impact on Traffic Modelling in ATM Systems," Univ. of Wuerzburg, Inst. of Computer Science, Tech. Rep. 101, Feb. 1995.

[6] M. Reisslein *et al.*, "Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study," Arizona State University, Dept. of Elect. Eng., Tech. Rep., Dec. 2003. Video traces available from http://trace.eas.asu.edu.

[7] T. Sikora, "MPEG Digital Video Coding Standards," *Digital Electronics Consumer Handbook*, McGraw Hill, 1997.

[8] M. Ghanbari, *Standard Codecs: Image Compression to Advanced Video Coding*, IEE, 2003.

[9] W. Li, "Overview of Fine Granularity Scalability in MPEG-4 Video Standard," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 11, no. 3, pp. 301–17, Mar. 2001.

[10] V. K. Goyal, "Multiple Description Coding: Compression Meets the Network," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 74–93, Sept. 2001.

[11] D. S. Taubman and M. W. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2001.

[12] W. Tan and A. Zakhor, "Packet Classification Schemes for Streaming MPEG Video over Delay and Loss Differentiated Net-

works," *Proc. Packet Video Wksp.*, Kyongju, Korea, April 2001.
[13] ITU-500-R, "Recommendation BT.500-8 — Methodology for the Subjective Assessment of the Quality of Television Pictures," 1998.
[14] A. Basso *et al.*, "Study of MPEG-2 Coding Performance Based on a Perceptual Quality Metric," *Proc. Picture Coding Symp.*, Melbourne, Australia, Mar. 1996.
[15] A. Webster *et al.*, "An Objective Video Quality Assessment System Based on Human Perception," *Proc. SPIE Human Vision, Visual Processing and Digital Display*, vol. 1913, pp. 1526, 1993.
[16] S. Winkler, "A Perceptual Distortion Metric for Digital Color Video," *Proc. SPIE Human Vision and Electronic Imaging*, vol. 3644, pp. 175–84, Jan. 1999.
[17] H. Liu and M. E. Zarki, "Performance of H.263 Video Transmission over Wireless Networks using Hybrid ARQ," *IEEE JSAC*, vol. 15, no. 9, pp. 1775–86, Dec. 1997.
[18] W. Luo and M. E. Zarki, "Analysis of Error Concealment Schemes for MPEG-2 Video Transmission over ATM Based Networks," *Proc. SPIE Visual Communications and Image Processing 1995*, Taiwan, pp. 102–8, May 1995.
[19] — , "MPEG2Tool: A Toolkit for the Study of MPEG-2 Video Transmission over ATM-Based Networks," Dept. of Electrical Engineering, Univ. of Pennsylvania, Tech. Rep., 1996.
[20] R. Puri *et al.*, "An Integrated Source Transcoding and Congestion Control Paradigm for Video Streaming in the Internet," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 18–32, Mar. 2001.
[21] J. Shin, J. W. Kim, and C.-C. J. Kuo, "Quality–of–Service Mapping Mechanism for Packet Video in Differentiated Services Network," *IEEE Trans. Multimedia*, vol. 3, no. 2, pp. 219–31, June 2001.
[22] I. Dalgic and F. A. Tobagi, "Characterization of Quality and Traffic for Various Video Encoding Schemes and Various Encoder Control Schemes," Stanford Univ., Dept. of Elec. Eng. and Comp. Science, Tech. Rep. CSL-TR-96-701, Aug. 1996.
[23] M. Frey and S. Nguyen-Quang, "A Gamma-Based Framework for Modeling Variable-Rate MPEG Video Sources: The GOP GBAR model," *IEEE/ACM Trans. Net.*, vol. 8, no. 6, pp. 710–9, Dec. 2000.
[24] D. P. Heyman, "The GBAR Source Model for VBR Video Conferencing," *IEEE/ACM Trans. Net.*, vol. 5, no. 4, pp. 554–60, Aug. 1997.
[25] M. Garrett and W. Willinger, "Analysis, Modeling, and Generation of Self-Similar VBR Video Traffic," *Proc. ACM Sigcomm*, London, UK, pp. 269–80, Sept. 1994.
[26] D. P. Heyman and T. V. Lakshman, "Source Models for VBR Broadcast Video Traffic," *IEEE/ACM Trans. Net.*, vol. 4, no. 1, pp. 40–8, Jan. 1996.
[27] M. Krunz, "A Source Model for VBR Video Traffic Based on M/G/∞ Input Processes," *Proc. IEEE Infocom*, San Francisco, CA, pp. 1441–9, Apr. 1998.
[28] M. Krunz and S. Tripathi, "On the Characterization of VBR MPEG Streams," *Proc. ACM SIGMETRICS*, Seattle, WA, pp.192–202, June 1997.
[29] D. Lucantoni, M. Neuts, and A. Reibman, "Methods for Performance Evaluation of VBR Video Traffic Models," *IEEE/ACM Trans. Net.*, vol. 2, no. 2, pp. 176–80, Apr. 1994.
[30] O. Rose, "Simple and Efficient Models for Variable Bit Rate MPEG Video Traffic," *Performance Evaluation*, vol. 30, no. 1-2, pp. 69–85, 1997.
[31] B. Ryu, "Modeling and Simulation of Broadband Satellite Networks — Part II: Traffic Modeling," *IEEE Commun. Mag.*, vol. 37, no. 7, pp. 48–56, July 1999.
[32] N. Semret, "Characterization and Modeling of MPEG Video Traffic on Multiple Timescales," Columbia University, New York, Tech. Rep., May 1995.
[33] Test Model Editing Committee, "MPEG-2 video Test Model 5, iso/iecjtc1/sc29wg11 mpeg93/457," Apr. 1993.
[34] M. Reisslein *et al.*, "Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study, Part 2: Statistical Analysis of Single-Layer Encoded Video," Arizona State University, Dept. of Electrical Engineering, Tech. Rep., Dec. 2002.
[35] A. M. Rohaly *et al.*, "Video Quality Experts Group: Current Results and Future Directions," *Proc. SPIE Visual Commun. and Image Processing*, vol. 4067, Perth, Australia, pp. 742–53, June 2000.
[36] S. Winkler, "Vision Models and Quality Metrics for Image Processing Applications," Ph.D. dissertation, EPFL, Switzerland, 2000.
[37] F. Fitzek and M. Reisslein, "MPEG-4 and H.263 Video Traces for Network Performance Evaluation," *IEEE Network*, vol. 15, no. 6, pp. 40–54, Nov./Dec. 2001.
[38] J. Beran *et al.*, "Long-Range Dependence in Variable-Bit-Rate Video Traffic," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 1566–79, Feb./Mar./Apr. 1995.
[39] M. Reisslein *et al.*, "Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study, Part 1: Overview and Definitions," Arizona State Univ., Telecommunications Research Center, Tech. Rep., Dec. 2002.
[40] D. Veitch and P. Abry, "A Wavelet-Based Joint Estimator of the Parameters of Long-Range Dependence," *IEEE Trans. Inform. Theory*, vol. 45, no. 3, pp. 878–97, Apr. 1999.
[41] S. Shenker, C. Partridge, and R. Guerin, "Request for Comments 2212: Specification of Guaranteed Quality of Service," Sept. 1997.
[42] M. Reisslein *et al.*, "Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study, Part 3: Statistical Analysis of Temporal Scalable Encoded Video," Arizona State University, Dept. of Electrical Engineering, Tech. Rep., Dec. 2002.
[43] W. Feng and J. Rexford, "A Comparison of Bandwidth Smoothing Techniques for the Transmission of Prerecorded Compressed Video," *Proc. IEEE Infocom*, Kobe, Japan, pp. 58–67, Apr. 1997.
[44] M. Krunz, "Bandwidth Allocation Strategies for Transporting Variable-Bit-Rate Video Traffic," *IEEE Commun. Mag.*, vol. 37, no. 1, pp. 40–6, Jan. 1999.
[45] Z. He and S. K. Mitra, "A Unified Rate-Distortion Analysis Framework for Transform Coding," *IEEE Trans. Circuits and Sys. for Video Tech.*, vol. 11, no. 12, pp. 1221–36, Dec. 2001.
[46] M. Reisslein *et al.*, "Traffic and Quality Characterization of Scalable Encoded Video: A Large-Scale Trace-Based Study, Part 4: Statistical Analysis of Spatial Scalable Encoded Video," Arizona State University, Dept. of Electrical Engineering, Tech. Rep., Aug. 2003.
[47] G. S. Fishman, *Principles of Discrete Event Simulation*, Wiley, 1991.
[48] A. M. Law and W. D. Kelton, Simulation, *Modeling and Analysis, 3rd Ed.*, McGraw Hill, 2000.
[49] Y. Wang and Q. Zhu, "Error Control and Concealment for Video Communication: A Review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–97, May 1998.
[50] N. Duffield, K. Ramakrishnan, and A. Reibman, "Issues of Quality and Multiplexing when Smoothing Rate Adaptive Video," *IEEE Trans. Multimedia*, vol. 1, no. 4, pp. 53-68, Dec. 1999.
[51] T. Shanableh and M. Ghanbari, "Heterogeneous Video Transcoding to Lower Patio-Temporal Resolutions and Different Encoding Formats," *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 101–10, June 2000.
[52] M. Chesire *et al.*, "Measurement and Analysis of a Streaming Media Workload," *Proc. USITS*, San Francisco, CA, Mar. 2001.
[53] G. K. Zipf, *Human Behavior and Principle of Least Effort: An Introduction to Human Ecology*, Cambridge, MA:Addison-Wesley, 1949.
[54] A. Dan, D. Sitaram, and P. Shahabuddin, "Dynamic Batching Policies for An On-Demand Video Server," *Multimedia Systems*, vol. 4, no. 3, pp. 112–21, Mar. 1996.
[55] L. Breslau *et al.*, "Web Caching and Zipf-Like Distributions: Evidence and Implications," *Proc. IEEE Infocom 1999*, New York, NY, pp. 126–34, Mar. 1999.
[56] H. Schulzrinne *et al.*, Audio-Video Transport Working Group, "RFC1889: RTP — A Transport Protocol for Real-Time Applications," 1996, work in progress.
[57] Y. Kikuchi *et al.*, "RFC3016: RTP Payload Format for MPEG-4 Audio/Visual Streams," 2000, work in progress.
[58] P. Seeling *et al.*, "Video Quality Evaluation for Wireless Transmission with Robust Header Compression," *Proc. 4th IEEE Int'l. Conf. Info., Commun. & Signal Processing and 4th Pacific-Rim Conf. Multimedia (ICICS-PCM 2003)*, Singapore, pp. 1346–50, Dec. 2003.

[59] F. Fitzek *et al.*, "Video and Audio Trace Files of Pre-Encoded Video Content for Network Performance Measurements," *Proc. IEEE Consumer Commun. and Net. Conf. (CCNC)*, Las Vegas, NV, pp. 245–50, Jan. 2004.
[60] L. Schruben, "Initialization Bias in Simulation Output," *Operations Research*, vol. 30, pp. 569–90, 1982.
[61] J. Beran, *Statistics for Long-Memory Process*, Chapman and Hall/CRC, 1994.
[62] A. Suarez-Gonzales *et al.*, "A Batch Means Procedure for Mean Value Estimation of Processes Exhibiting Long-Range Dependence," *Proc. 2002 Winter Simulation Conf.*, San Diego, CA, pp. 456–64, Dec. 2002.

BIOGRAPHIES

PATRICK SEELING received the Dipl.-Ing. degree in industrial engineering and management (specializing in electrical engineering) from the Technical University of Berlin (TUB), Germany, in 2002. Since 2003 he has been a Ph.D. student in the Department of Electrical Engineering at Arizona State University. His research interests are in the area of video communications in wired and wireless networks. He is a student member of the IEEE and the ACM.

MARTIN REISSLEIN (reisslein@asu.edu) is an assistant professor in the Department of Electrical Engineering at Arizona State University, 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. While in Berlin he was teaching courses on performance evaluation and computer networking at the Technical University Berlin. He is editor-in-chief of *IEEE Communications Surveys and Tutorials* and has served on the Technical Program Committees of IEEE INFOCOM, IEEE GLOBECOM, and the IEEE International Symposium on Computer and Communications. He has organized sessions at the IEEE Computer Communications Workshop (CCW). He maintains an extensive library of video traces for network performance evaluation, including frame size traces of MPEG-4 and H.263 encoded video, at http://trace.eas.asu.edu. He is co-recipient of the Best Paper Award of the SPIE Photonics East 2000 — Terabit Optical Networking conference. His research interests are in the areas of Internet Quality of Service, video traffic characterization, wireless networking, and optical networking.

BESHAN KULAPALA received the bachelor's degree in electrical engineering from the University of Kentucky, Lexington, in 2001, and received the master's degree in electrical engineering from Arizona State University, Tempe, in 2003. Since 2003 he has been a PhD student in the Department of Electrical Engineering at Arizona State University. His research interests are in the area of video transmission over wired and wireless networks. He is a student member of the IEEE.