Rate-Distortion Optimized Hybrid Error Control for Real-Time Packetized Video Transmission

Fan Zhai, *Member, IEEE*, Yiftach Eisenberg, *Member, IEEE*, Thrasyvoulos N. Pappas, *Senior Member, IEEE*, Randall Berry, *Member, IEEE*, and Aggelos K. Katsaggelos, *Fellow, IEEE*

***Abstract*—The problem of application-layer error control for real-time video transmission over packet lossy networks is commonly addressed via joint source-channel coding (JSCC), where source coding and forward error correction (FEC) are jointly designed to compensate for packet losses. In this paper, we consider hybrid application-layer error correction consisting of FEC and retransmissions. The study is carried out in an integrated joint source-channel coding (IJSCC) framework, where error resilient source coding, channel coding, and error concealment are jointly considered in order to achieve the best video delivery quality. We first show the advantage of the proposed IJSCC framework as compared to a sequential JSCC approach, where error resilient source coding and channel coding are not fully integrated. In the IJSCC framework, we also study the performance of different error control scenarios, such as pure FEC, pure retransmission, and their combination. Pure FEC and application layer retransmissions are shown to each achieve optimal results depending on the packet loss rates and the round-trip time. A hybrid of FEC and retransmissions is shown to outperform each component individually due to its greater flexibility.**

***Index Terms*—Error concealment, error control, error resilience, hybrid error control, multimedia streaming, quality of service (QoS), resource allocation, unequal error protection (UEP).**

I. Introduction

REAL-TIME video streaming, videophone and videoconferencing, have gained increased popularity. However, it is well known that the best effort design of the current Internet makes it difficult to provide the quality of service (QoS) needed by these applications. A direct approach dealing with the lack of QoS is to use error control, where different error control components can be implemented at different network layers. In this paper, we consider a combination of common error control approaches. Specifically, at the sender side, we consider error resilient source coding, hybrid forward error correction (FEC) and application-layer retransmission; at the receiver side, we consider error concealment. We present an integrated joint source channel coding (IJSCC) framework to jointly optimize these application-layer error control components for real-time video transmissions over packet lossy networks.

Each of the above-mentioned error control approaches is designed to deal with a lossy packet channel. Error resilient source coding accomplishes error control by adding redundancy at the source coding level to prevent error propagation and to limit the distortion caused by packet losses. For packet-switched networks, error resilience may be achieved through the selection of encoding mode for each packet, the use of scalable video coding, or multiple description coding [1]–[4]. In this paper, we focus on optimal mode selection (including prediction mode and quantizer) for nonscalable (single layer) video. Another way to deal with packet loss is to use error correction techniques at the application/transport layer. Two basic techniques are used: FEC and automatic repeat request (ARQ). Each has its own benefits with regard to error robustness and network traffic load [5], [6]; we consider both approaches in the IJSCC framework. Finally, error concealment refers to post-processing techniques employed by the decoder to handle packet loss by utilizing the spatial and temporal correlation of the video sequence.

Of the two error correction techniques, ARQ is not widely used in real-time streaming applications because it cannot accommodate the delay requirements of these applications. Also, such approaches may not be appropriate for multicast scenarios due to their inherent scalability problems [7], [8]. FEC-based techniques are usually preferred for such applications and are currently under consideration by the Internet Engineering Task Force (IETF) as a proposed standard in supporting error resilience [9]. However, FEC cannot completely avoid packet loss due to its limits on the block-size as dictated by the application's delay constraints. FEC also incurs constant overhead even when there are no losses in the channel, and the appropriate level of FEC will depend on the accurate estimation of the channel's behavior. On the other hand, ARQ can automatically adapt to the channel loss characteristics by transmitting only as many redundant packets as are lost. Thus, if the application has a relatively loose end-to-end delay constraint (e.g., on-demand video streaming), ARQ may be preferable. Even for real-time applications, delay constrained application-layer ARQ has been shown to be useful in some situations [5], [10], [11]. In our previous work [12], we studied different error control scenarios including pure FEC, pure retransmission, and hybrid FEC/retransmission. Our goal was to identify the optimal video transmission policy based on network conditions (such as packet loss probability and network round-trip time) and application requirements (such as end-to-end delay). In this paper, we extend the above work

Manuscript received May 17, 2004; revised December 23, 2004. This work was presented in part at the IEEE International Conference on Communications (ICC), Paris, France, June 2004. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Bruno Carpintieri.

F. Zhai is with Texas Instruments, Dallas, TX 75243 USA (e-mail: fzhai@ti.com).

Y. Eisenberg is with BAE SYSTEMS, Nashua, NH 03061 USA (e-mail: yiftach.eisenberg@baesystems.com)

T. N. Pappas, R. Berry, and A. K. Kastaggelos are with the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL 60208 USA (e-mail: pappas@ece.northwestern.edu; rberry@ece.northwestern.edu; aggk@ece.northwestern.edu).

Digital Object Identifier 10.1109/TIP.2005.860353

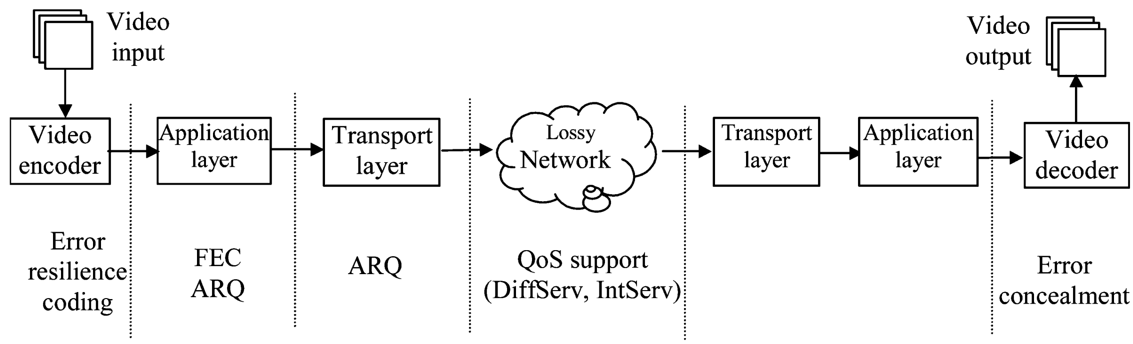


Fig. 1. Error control components in a real-time video streaming system.

by more rigorously developing an optimization framework for studying various hybrid error control schemes.

Optimal error control is often studied in a joint source-channel coding (JSCC) framework, e.g., [7], [8], and [13]–[24]. In general, JSCC is accomplished by designing the quantizer and entropy coder for given channel errors, as in [13]. For image and video applications, JSCC consists of three tasks: finding an optimal bit allocation between source coding and channel coding for given channel loss characteristics; designing the source coding to achieve the target source rate; and designing the channel coding to achieve the required robustness [3], [7]. Joint source coding and FEC has been extensively studied in the literature [14]–[24]. In [14], [21], and [23], JSCC was studied for wavelet image and video transmission over the Internet. The authors in [17], [19], [20], and [22] have studied this problem in the context of wireless channels for scalable video, and error resilience was achieved through optimized transport prioritization for layered video. In [16], JSCC with error resilient source coding and FEC for Internet scalable video is studied. These previous works do not consider all possible error control components at the application layer. In this paper, we introduce the IJSCC framework, where error resilient source coding, channel coding, and error concealment are all addressed in a tractable optimization setting. Furthermore, the approaches in the above-mentioned reports do not appear to be fully "integrated," even though "joint," while our work fully considers the interaction between source coding and channel coding.

With regard to related work on hybrid FEC/retransmission, in [11], a general cost-distortion framework was proposed to study several scenarios such as DiffServ, sender-driven retransmission and receiver-driven retransmission. In our approach, we take into account source coding and error concealment, which are not considered in [11]. As to wireless IP networks, a link-layer hybrid FEC/ARQ scheme is considered in [25] and an application-layer hybrid FEC/ARQ technique based on heuristics is proposed for video transmission in [5]. A sender-driven application layer hybrid FEC/ARQ scheme has been presented for video transmission over a hybrid wired/wireless network in [26] and [27], where parity packets are generated and sent only if a negative acknowledgment (NAK) is received. The approach in [27] may be more useful for streaming applications, where the end-to-end delay constraint is usually long and the sender can rely on ARQ more than on FEC. A receiver-driven hybrid FEC/pseudo-ARQ mechanism is proposed for Internet multimedia multicast in [6]. Another related work is [10], which considers scalable video, with pure ARQ used for the base layer and pure FEC used to protect the enhancement layer. Our work differs from the above in that we consider application-layer sender-driven retransmission, where lost packets are selectively retransmitted according to a rate-distortion optimized policy. In addition, we jointly consider the FEC/ARQ parameters with source coding and error concealment.

The first contribution of this paper is a general and flexible IJSCC framework for real-time packetized video transmission; the IJSCC framework allows for the comparison of different error control scenarios such as FEC, retransmission, and hybrid FEC/retransmission. The second contribution is the study of hybrid error control consisting of both FEC and retransmission. FEC and application layer retransmission each can achieve optimal results depending on the packet loss rates and round-trip-time (RTT). When the two are jointly employed as in the proposed hybrid technique, improved results are obtained.

The rest of this paper is organized as follows. We first introduce some preliminaries in Section II. Next, in Section III, the IJSCC problem formulation is presented, followed by the study of hybrid application-layer error control in Section IV. Experimental results are discussed in Section V. Finally, Section VI contains our conclusions.

II. Preliminaries

A. Real-Time Video Transmission System

Fig. 1 depicts the architecture of a packet-based real-time video transmission system and indicates the error control components available at different layers. At the sender, video packets (referred to as *source packets*) are generated by a video encoder. At the application layer, parity check packets used for FEC may be generated. In addition, lost packets may be retransmitted if applicable. After passing through the network protocol stack (e.g., RTP/UDP/IP), *transport packets* are formed to be sent over a lossy network. Packets that reach the decoder in time are buffered in the decoder buffer. We define an *initial setup time* (also referred to as the maximum end-to-end delay), $T_{\max}$, as the duration between the time when the first packet is captured at the encoder and its playback at the decoder. The longer the initial setup time, the more robust the system is to channel variations. The setup time is application dependent, and is limited by how long a user is willing to wait for the video to be displayed. The video decoder reads video packets from the decoder buffer and displays the resulting video frames in real time (i.e., the video is displayed continuously without interruption at the decoder). Lost packets are concealed at the decoder.

B. Channel Model

In this paper, the network is modeled as an independent time-invariant packet erasure channel. Packet losses in the network can be modeled in various ways, e.g., a Bernoulli process, a two-state or kth order Markov chain, etc. [28]. In our simulations, packet loss in the network is modeled by a Bernoulli process, i.e., each packet is independently lost with probability ϵ. The proposed framework is general and not limited to any specific packet loss model. All that is needed is a stochastic model of the packet losses. We assume that the receiver responds to a lost or corrupt packet with a NAK, and responds to a correctly received packet with a positive acknowledgment (ACK). All acknowledgments are assumed to arrive correctly after one RTT, i.e., the feedback delay is a constant and the feedback channel is error free[1] as in [6]. In addition, we can assume that the probability an ACK is lost is included in ϵ, i.e., whenever an ACK is lost, we will assume that the corresponding packet is lost when doing the optimization. Also if an ACK arrives after its estimated frame time, then we could easily update the probability of loss, before optimizing the next frame. These parameters (probability of packet loss, RTT, etc.) can be estimated from the feedback channel at regular time instances. In addition, we consider the situation where the individual user's traffic is a small part of the overall traffic in the network and, thus, has a negligible effect on these parameters.[2]

C. FEC

For Internet applications, many researchers have considered using erasure codes to recover from packet losses [15], [16], [18]. In such approaches, a video stream is first partitioned into segments; each segment is packetized into a group of M packets. A block code is then applied to the M packets to generate additional l parity packets resulting in an N-packet block, where $N = M + l$. With such a code, the receiver can recover the original M packets if a sufficient number of packets in the block are received. The most commonly studied erasure codes are Reed–Solomon (RS) codes, which have good erasure correcting properties and are widely used in practice [15], [16], [18]. In this paper, we consider systematic RS codes, but the basic framework could easily be applied to other codes, such as Tornado codes, which have slightly worse erasure protecting properties, but can be encoded and decoded much more efficiently than RS codes [29].

An RS code is represented as $\mathrm{RS}(N,M)$, where M is the number of source symbols and $N-M$ is the number of parity symbols. The code rate of an $\mathrm{RS}(N,M)$ code is defined as M/N. For Internet applications, since the channel errors are typically in the form of packet erasure, an $\mathrm{RS}(N,M)$ code applied across packets can correct up to $(N-M)$ lost packets. The protection capability of an RS code depends on the block size and the code rate, which are limited by the extra delay introduced by FEC. For Internet applications, the block length, N, can be determined based on the end-to-end system delay constraints [30].

D. End-to-End Distortion

Due to channel losses, we use the expected end-to-end distortion to evaluate video quality. Three factors can be identified as affecting this: the source behavior (quantization and packetization), the channel characteristics, and the receiver behavior (error concealment) [2]–[4], [8], [31]. The expected distortion can be calculated at the encoder as

$$E[D_k] = (1-\rho_k)E[D_{R,k}] + \rho_k E[D_{L,k}] \tag{1}$$

where $E[D_{R,k}]$ and $E[D_{L,k}]$ are the expected distortion when the kth source packet is either received correctly or lost, respectively, and ρ_k is its loss probability. The relationship between the source packet loss probability and transport packet loss probability depends on the specific transport packetization scheme chosen. Note that both $D_{L,k}$ and $D_{R,k}$ are usually random variables. This is because, due to channel losses, the reference frames for intercoding at the decoder and the encoder may not be the same. However, based on the given feedback information, D_k could be deterministic. This occurs when packet k and all the associated packets in the previous frames used to predict packet k are acknowledged and have not been further retransmitted. Thus, the expectations in (1) are taken with respect to the probability distribution of channel losses given the available feedback and the previous FEC decisions. This is discussed in greater detail in Section IV-B.

Note that the calculation of $D_{L,k}$ depends on the specific error concealment strategy used at the decoder. Assuming the mean-squared error (MSE) criterion, the distortion measurement based on the recursive optimal per-pixel estimate (ROPE) algorithm [16] can be used to recursively calculate the overall expected distortion level of each pixel. The image quality measure used is the peak signal-to-noise-ratio (PSNR), defined as PSNR$= 10\log(255^2/\mathrm{MSE})$ dB.

E. Packetization and Error Concealment

We consider a system where each group of blocks (GOB)[3] is coded as one source packet, and every packet is independently decoded. For error concealment, we consider a simple but efficient error concealment scheme similar to the one in [16]. Specifically, we use a temporal replacement error concealment strategy. The concealment strategy is spatially causal, i.e., the decoder will only use the information from previously received packets in concealing a lost packet. When a packet is lost, the concealment motion vector for a MB in the lost packet is the median of the three motion vectors of its top-left, top, and top-right

[1]Because ACK/NAK packets are typically very small, they can be transmitted with negligible probability of loss through retransmissions or other forms of protection. For small packets, these techniques will only lead to a small amount of additional overhead. From the application's point of view, the RTT can be treated as constant as long as its variation is limited to one frame's time, since our optimization is frame based. This is addressed in detail in Section IV.

[2]It is important to clarify that in order to accommodate other traffic and share the network resources fairly, a TCP-friendly congestion control is usually used to constrain the source bit rate. In this case, as long as the video traffic generated by our application follows the assigned transmission rate, the effect of our traffic on the overall network congestion can usually be ignored.

[3]Following the H.263 standard, we use a GOB to denote one row of macroblocks (MBs).

MBs. If the previous packet is also lost, then the concealment motion vector is zero, i.e., the MB in the same spatial location in the previously reconstructed frame is used to conceal the current loss. In this case, the expected distortion can be written as

$$E[D_k] = (1-\rho_k)E[D_{R,k}] + \rho_k(1-\rho_{k-1})E[D_{C,k}] + \rho_k\rho_{k-1}E[D_{Z,k}] \quad (2)$$

where $E[D_{C,k}]$ and $E[D_{Z,k}]$ are the expected distortions after concealment when the previous packet is either received correctly or lost, respectively. Note that, for this packetization and error concealment scheme, the choice of prediction mode and the loss probability for source packet $k-1$ affects the distortion of the kth source packet.

III. Joint Source-Channel Coding

In this section, we discuss approaches for jointly optimizing error resilient source coding and channel coding. First, we discuss approaches that sequentially optimize these resources by first allocating bits between source and channel coding, and then optimizing the source coding given the bit budget. Next, we present our IJSCC framework which optimizes both the bit allocation and source coding in a single step.

A. Sequential JSCC

Most of the JSCC work to date has focused on the bit allocation between source and channel coding, such as in [13]–[15], [18]–[20]. Source coding is performed based on the given bit budget, after the bit allocation between source and channel is completed. The optimization of source coding can be achieved in the form of mode selection by taking into account the residual errors after channel coding, such as in [1]–[3] and [16].

Let $\mathcal{Q}$ be the set of source coding parameters, which include the prediction mode and quantization step size. The FEC parameter set is defined as $\mathcal{R} = \{(N_1, M), \ldots, (N_q, M)\}$, where q is the number of available code options. Let $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ denote the vector of source coding parameters and channel coding parameters for one frame, respectively. Let the superscript (n) denote the nth frame, and the subscripts s and c stand for source and channel coding, respectively. Then, this sequential two-step JSCC can be formally presented as

$$\min_{\{\boldsymbol{\nu}\in\mathcal{R}\}} E\left[D^{(n)}(\boldsymbol{\nu})\right]$$
$$\text{s.t.} \quad T^{(n)}(\boldsymbol{\nu}) = \frac{B_s^{(n)}(\boldsymbol{\mu}(\boldsymbol{\nu}))}{R_T} + \frac{B_c^{(n)}(\boldsymbol{\nu})}{R_T} \le T_0^{(n)} \quad (3)$$

and

$$\min_{\{\boldsymbol{\mu}\in\mathcal{Q}\}} E\left[D^{(n)}(\boldsymbol{\mu})\right]$$
$$\text{s.t.} \quad T_s^{(n)}(\boldsymbol{\mu}) = \frac{B_s^{(n)}(\boldsymbol{\mu})}{R_T} \le T_{s,0}^{(n)} \quad (4)$$

where $E[D]$ is the expected distortion; R_T is the transmission rate; B_s and B_c are the source bits and channel bits, respectively; T is the associated transmission delay; and T_0 and $T_{s,0}$ are the transmission delay constraint for the whole frame (including both source and channel bits) and the source bits, respectively. In (3), the constraint is on the total transmission delay for the nth frame $T^{(n)}$; in (4), the constraint is on the source transmission delay[4] $T_s^{(n)}$. Several channel coding techniques have been considered for solving (3). For work utilizing preencoded video, such as [15] and [18], source coding is fixed. Thus, the objective in these studies is to minimize the channel induced distortion, and the second step (4) is not necessary. For work on coding the source on the fly, one way to characterize the distortion in (3) is to use a source R-D model, as in [19] and [20]. For example, a universal R-D model is used in [20]. In [19], the distortion is expressed as the sum of source and channel distortion, both of which are model based. By assuming uncorrelated source and channel distortion, the first step of the minimization in [19] aims at minimizing the channel distortion, while the second-step minimizes the source distortion. There has also been considerable work in the area of JSCC for scalable video coders, such as [14], [21], and [23]. The inherent prioritization of information in a scalable video bitstream makes the implementation of JSCC more straightforward. For block-based motion compensated video coding, JSCC is more challenging because the relative importance of packets is not explicitly available.

The above studies, however, do not fully consider the interaction between source coding and channel coding. More specifically, they do not optimally account for how error resilient source coding affects the bit allocation between source and channel. The goal of JSCC is to optimally add redundant bits in the source (error resilient source coding) and the channel (channel coding) to achieve the best tradeoff between error robustness and compression efficiency. The optimal way to achieve this requires joint consideration of error resilient source coding and channel coding. In addition, we would like to emphasize that one of the main objectives in our work is not to rely on approximations but instead to accurately compute the expected distortion. Because the effect of source coding and channel coding on the end-to-end distortion is intertwined, accurate calculations of the expected distortion cannot separate these two components. In other words, although the tradeoffs between source and channel coding are considered in our approach, to accurately calculate the expected distortion, the effects of source and channel coding must be considered jointly (i.e., in an integrated fashion).

B. IJSCC

Next, we present our IJSCC framework for jointly optimizing error resilient source coding and channel coding. That is, instead of separating the overall expected distortion into source distortion and channel distortion, as in (3) and (4), we consider the interaction between these components.

A related framework was presented in [16] for jointly considering error resilient source coding and channel coding. In that study, the distortion measurement was model based, where the concealment distortion for each block is calculated by weighing the distortions of the surrounding blocks in the previous frame(s) that overlap with the motion-compensated block (with the weights proportional to the overlap area).

[4]Note that both of these constraints can also be interpreted as specifying bit budgets of $T_0^{(n)}R_T$ and $T_{s,0}^{(n)}R_T$.

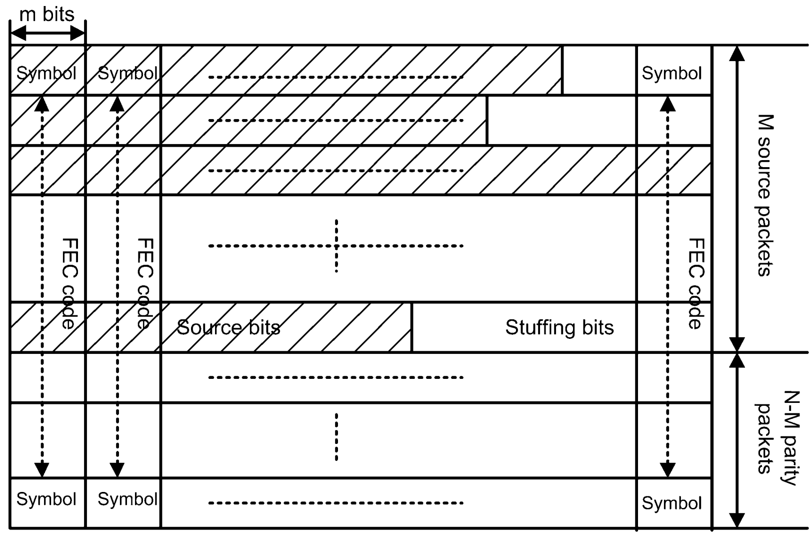


Fig. 2. Packetization scheme: one row corresponds to one source packet.

Here, we recursively calculate packet distortion based on (1), which takes into account both source distortion and channel distortion, as well as error propagation due to channel errors.[5] Our objective is to minimize the total expected distortion for the nth frame, given a transmission delay constraint, i.e.,

$$\min_{\{\boldsymbol{\mu}\in\mathcal{Q},\boldsymbol{\nu}\in\mathcal{R}\}} E\left[D^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})\right]$$
$$\text{s.t.}\quad T^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu}) = \frac{B^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})}{R_T} \le T_0^{(n)} \tag{5}$$

where $B^{(n)}$ represents the total bits used for both source coding and channel coding, R_T is the transmission rate, and $T_0^{(n)}$ is the transmission delay constraint for this frame.

In calculating the expected distortion for each source packet, the loss probability for the source packet ρ needs to be determined. The relationship between the source packet loss probability and transport packet loss probability depends on the specific transport packetization scheme and channel coding chosen [32]. We will discuss next the particular case of hybrid FEC and selective retransmission.

IV. Hybrid FEC and Selective Retransmission

In this section, we consider hybrid FEC and application-layer selective retransmission to perform optimal error control in the IJSCC framework.

A. Packetization Scheme

For packet-based video transmission, FEC is usually done across packets, as discussed in Section II-C. Fig. 2(a) illustrates the packetization we use for each frame, where one row corresponds to one GOB, which is directly packetized into one transport packet. Since the source packet sizes (shown by the shaded area in Fig. 2) are usually different, the maximum packet size of a block (a group of packets protected by one RS code) is determined first, and then all packets are padded with stuffing bits in the tail part to make their sizes equal. The stuffing bits are removed after the parity codes are generated and so are not transmitted [16], [30]. The resulting parity packets are all of the same size (the maximum packet size). Each source packet in Fig. 2 is protected by an $\mathrm{RS}(N,M)$ code, where M is the number of GOBs and N the number of transport packets.

In this scheme, a source packet is regarded as lost after error recovery at the receiver only when the corresponding transport packet is lost and the block containing the lost transport packet cannot be recovered (i.e., more than $N-1-M$ other packets are also lost). Therefore, the probability of source packet loss ρ after error recovery is defined as

$$\begin{aligned}\rho &= \epsilon\left(1-\sum_{i=0}^{N-1-M}\binom{N-1}{i}\epsilon^i(1-\epsilon)^{N-1-i}\right)\\ &= \sum_{i=N-M+1}^{N}\frac{i}{N}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}\end{aligned} \tag{6}$$

where ϵ is the probability of transport packet loss. Note that, in this scheme, all source packets in a given frame have the same probability of loss ρ [16].

B. Problem Formulation

Assume that there are up to A frames in the sender's buffer that are eligible for retransmission.[6] Let $\sigma_k^{(n)} \in \{0,1\}$ denote the retransmission parameter for the kth source packet in frame n, where 0 denotes no retransmission and 1 denotes retransmission. Let $\boldsymbol{\sigma^{(n)}} = \{\sigma_1^{(n)},\ldots,\sigma_M^{(n)}\}$ denote the retransmission parameter vector for frame n, and $\boldsymbol{\sigma} = \{\boldsymbol{\sigma}^{(n-A)},\ldots,\boldsymbol{\sigma}^{(n-1)}\}$ the vector for the A frames. For video transmission applications, usually a higher-level rate controller is used to constrain the bits, or equivalently the transmission delay for each frame. For simplicity, let $T_0^{(n)}$ be the transmission delay for the nth frame obtained from the rate controller. Following the structure of the

[5]The effect of error propagation can be fully captured based on the acknowledgment information after 1 RTT's delay.

[6]The maximum number of frames that are eligible for retransmission can usually be obtained based on the occupation level of the encoder buffer and the maximum end-to-end delay $T_{\max}$.

IJSCC framework in (5), we consider the following problem formulation:

$$\begin{aligned}\min_{\{\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\sigma}\}} \sum_{i=0}^{A} E\left[D^{(n-i)}\right] &= \sum_{i=1}^{A} E\left[D^{(n-i)}\left(\boldsymbol{\sigma}^{(\boldsymbol{n-i})}\right)\right] + \sum_{k=1}^{M} E\left[D_k^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})\right] \\ \text{s.t.} \quad \sum_{i=1}^{A}\sum_{k=1}^{M} \sigma_k^{(n-i)} T_k^{(n-i)} &+ \sum_{k=1}^{M} T_k^{(n)} \le T_0^{(n)}. \end{aligned} \tag{7}$$

Gains might be obtained by grouping the retransmitted packets and the packets in the current frame together to perform FEC. However, this introduces additional delay for the retransmitted packets and considerably complicates the solution of the problem. In this work, we only consider FEC for the current frame.

The above formulation is for an optimization scheme with a sliding window of size $A+1$ frames. The optimization window shifts at the frame level instead of at the packet level, since the latter usually leads to much higher computation complexity. In addition, the packets in one frame typically have the same deadline for playback. In this formulation, when processing each frame, we assume that all the raw data for the frame is available in a buffer, and the optimization (retransmission policy for the first A frames based on feedback, and source coding and FEC for the current frame) is performed on the $A+1$ frames in the window. After optimization is done, the retransmitted packets and the transport packets in the current frame (including source packets and parity packets) are transmitted over the network. After the transmission of these packets, the window shifts forward by one frame, and the optimization is solved again based on the updated feedback.

When each frame is encoded, the probability of packet loss for all the past A frames is updated based on the received feedback. For example, if one packet is known to be received, its probability of loss becomes 0; if one is lost, its loss probability becomes 1 if no further retransmission for this packet has been initiated. Based on the updated probabilities of packet loss, the expected distortion of all packets in the encoder buffer is recursively re-calculated as in (1). In using this model, the error propagation due to packet loss (after 1 RTT) can be fully captured and consequently the effect of previously lost packets on the future frames is taken into account. Since each time we do not consider re-encoding the past A frames, the complexity in updating the expected distortion is not significant.

Additional gain may be obtained by considering the future frames when the current frame is encoded. For example, by doing so, the effect of the parameter decisions in the optimization window on future frames can be taken into account. This will generally result in better performance due to the motion-compensation dependencies of video frames. However, this leads to a very complicated and usually intractable problem. In addition, for a real-time applications, future frames may not be available when the current frame is encoded.

C. Calculation of Probability of Packet Loss

We discuss next how to calculate the probability of packet loss ρ_k in order to find the expected distortion in (1). For a packet in the current frame, the probability of packet loss can be defined as $\rho_k^{(n)} = \rho_{k,\mathrm{FEC}}^{(n)} \rho_{k,\mathrm{RET}}^{(n)}$, where $\rho_{k,\mathrm{FEC}}^{(n)}$ and $\rho_{k,\mathrm{RET}}^{(n)}$ denote, respectively, the probability of packet loss due to FEC and retransmission. $\rho_{k,\mathrm{FEC}}^{(n)}$ is defined in (6). The probability of loss in future retransmissions can only be estimated since the acknowledgment information and retransmission decisions (note that lost packets are selectively retransmitted) are not available in the encoding of the current frame. In this work, we give an approximate formula to estimate it, i.e., $\rho_{k,\mathrm{RET}}^{(n)} = \epsilon^{\tilde{m}}$, where $\tilde{m}$ denotes the estimate of the total number of retransmissions for the kth packet, m. Note that m is not a constant and is dependent on how $\rho_{k,\mathrm{RET}}^{(n)}$ itself is calculated and the future video content. In addition, the effect of packet recovery due to other packets' retransmissions should also be taken into account when calculating $\rho_{k,\mathrm{RET}}^{(n)}$. However, it is almost impossible to accurately estimate this factor due to the use of block code. In our estimation formula, although this factor is not explicitly indicated, it has been taken into account by the estimate of $\tilde{m}$. In this work, we use an estimate of $\tilde{m}$ developed from simulations. Fig. 3 shows the performance of the hybrid FEC/retransmission system versus m for the QCIF format (176 $\times$ 144) Foreman sequence and Akiyo sequence (here m is fixed for the entire sequence). In both tests, the number of frames that are eligible for retransmissions is $A = 4$ and the frame rate is 15 fps. The transmission rate is 480 and 360 kbps, respectively. Based on these results, we use $\tilde{m}$, calculated by $\tilde{m} = A/(1+\mathrm{RTT})^2$, where RTT is the RTT in the units of one frame's duration T_F. This appears to provide good performance and is used subsequently. Note that the maximum number of available retransmission opportunities is $\lfloor A/(1+\mathrm{RTT}) \rfloor$. In addition, from Fig. 3, we can see that the system performance is not very sensitive to the choice of $\tilde{m}$, i.e., the PSNR variation is less than 0.3 dB.

In considering the possible retransmission of packets in the current frame, the expected additional transmission delay used for retransmission in the future should be taken into account; it is calculated by $E[\Delta T_k^{(n)}] = \sum_{k=1}^{M} \tilde{m} \rho_{k,\mathrm{FEC}}^{(n)} T_k^{(n)}$. The delay constraint in (7) is modified accordingly.

For a lost packet in the past frames, we let $\rho_k^{(n-i)} = \rho_{k,\mathrm{UPD}}^{(n-i)} \rho_{k,\mathrm{RET}}^{(n-i)}$ for $i = 1, \ldots, A$, where $\rho_{k,\mathrm{UPD}}^{(n-i)}$ is the updated probability of packet loss based on feedback and $\rho_{k,\mathrm{RET}}^{(n-i)}$ is the probability of packet loss due to retransmissions. Assume that one past frame is protected by an $\mathrm{RS}(N, M)$, and L packets are lost. Let $J = L + M - N$ and V be the number of retransmitted packets in that frame. Taking into account the RS codes, the calculation of $\rho_{k,\mathrm{RET}}^{(n-i)}$ is different for the lost packets that are either retransmitted or not. If $V < J$, we have

$$\rho_{k,\mathrm{RET}}^{(n-i)} = \epsilon^{\sigma_k^{(n-i)}}$$

if $V = J$, we have

$$\rho_{k,\mathrm{RET}}^{(n-i)} = \begin{cases} \epsilon, & \text{if } \sigma_k^{(n-i)} = 1 \\ 1 - (1-\epsilon)^J, & \text{if } \sigma_k^{(n-i)} = 0 \end{cases}$$

and if $V > J$ we have

$$\rho_{k,\mathrm{RET}}^{(n-i)} = \begin{cases} \sum_{j=V-J+1}^{V} \frac{j}{V} \binom{V}{j} \epsilon^j (1-\epsilon)^{V-j}, & \text{if } \sigma_k^{(n-i)} = 1 \\ \sum_{j=V-J+1}^{V} \binom{V}{j} \epsilon^j (1-\epsilon)^{V-j}, & \text{if } \sigma_k^{(n-i)} = 0 \end{cases}.$$

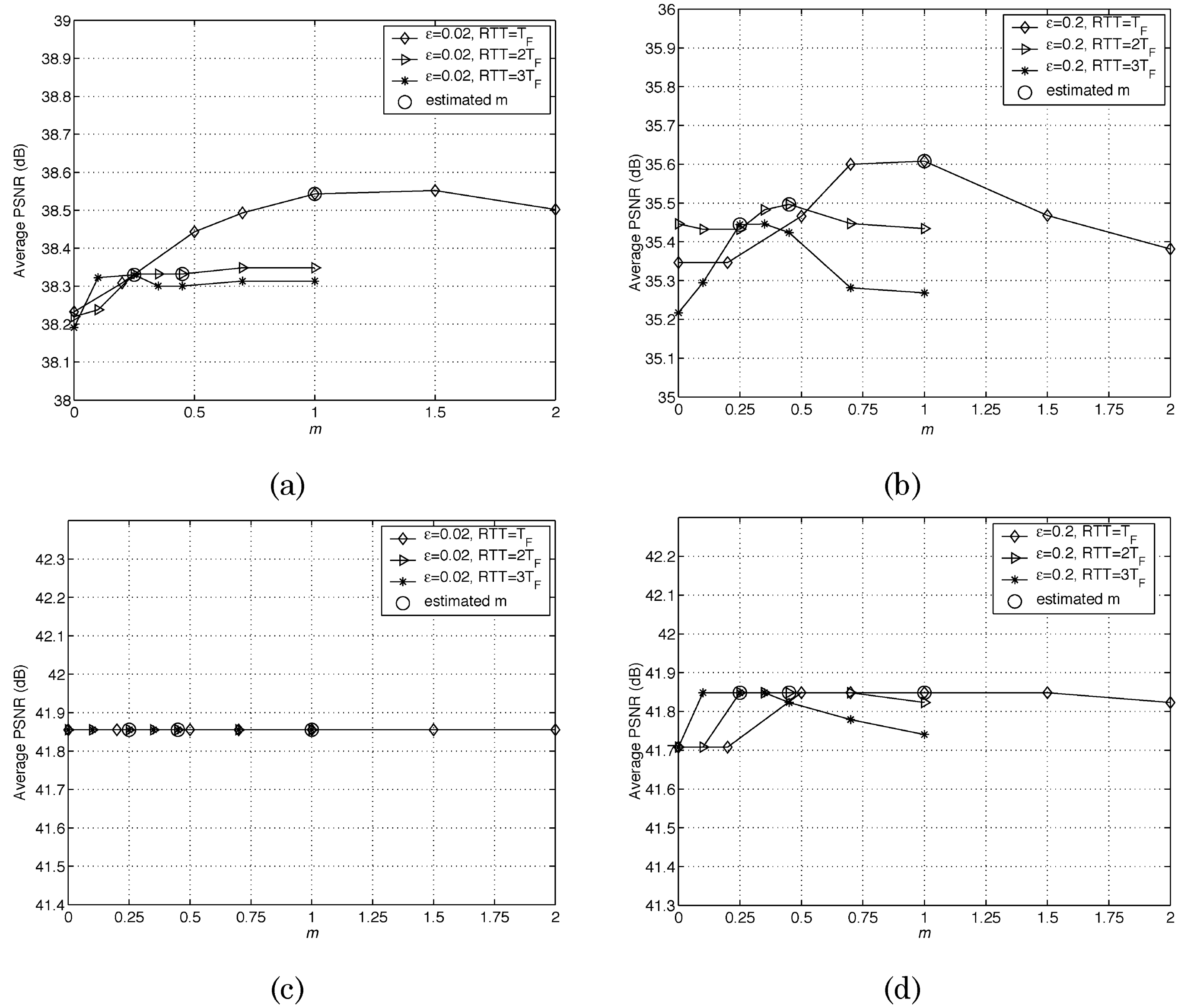


Fig. 3. Average PSNR versus m in the hybrid FEC/retransmission system. (a), (b) QCIF Foreman sequence at $F = 15$ fps, $R_T = 480$ kbps and $A = 4$. (c), (d) QCIF Akiyo sequence at $F = 15$ fps, $R_T = 360$ kbps and $A = 4$.

Note that, when the above formulas are derived, in order to maintain reasonable complexity, we make a conservative assumption regarding future retransmissions of packets for which we have received NAK. In particular, when calculating $\rho_{k,\mathrm{RET}}^{(n-i)}$, we only consider the possibility of retransmission at the current time instant. Without this assumption, the complexity required to estimate the probability of loss based on future retransmissions of previously lost packets increases significantly and is, therefore, less practical. In addition, our assumption results in a more conservative solution. In other words, by only considering retransmission at the current time, the sender often chooses to retransmit the lost packet earlier as opposed to waiting to retransmit it at a later time (although the algorithm does not eliminate this option). As discussed above, optimization performed within a localized time window (one frame time in our work) does not guarantee optimal performance for the sequence (i.e., the performance we could achieve if the entire sequence and channel conditions were known in advance). Based on simulation results, we have observed that our conservative approximation regarding future behavior can actually improve the average distortion for the sequence as compared to the less conservative approaches, which do not take into account the effect of current decisions on the future frames. One reason for this is that by being more conservative in the current frame, the algorithm tends to produce a more reliable reference for future frames.

D. Solution Algorithm

By using a Lagrange multiplier $\lambda \geq 0$, (7) can be converted into an unconstrained problem as

$$\begin{aligned}
&\min_{\{\boldsymbol{\mu}\in\mathcal{Q},\boldsymbol{\nu}\in\mathcal{R},\boldsymbol{\sigma}\in\mathcal{P}\}} \sum_{i=0}^{A} J^{(n-i)} \\
&\quad= \sum_{i=1}^{A} E\left[D_k^{(n-i)}\left(\boldsymbol{\sigma}^{(n-i)}\right)\right] + \sum_{k=1}^{M} E\left[D_k^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})\right] \\
&\qquad+ \lambda\left\{\sum_{i=1}^{A}\sum_{k=1}^{M}\sigma_k^{(n-i)}T_k^{(n-i)} + \sum_{k=1}^{M}T_k^{(n)}\right\}. \qquad (8)
\end{aligned}$$

The convex hull solution of this relaxed problem can be found by choosing an appropriate λ to satisfy the transmission delay constraint. This can be done using standard techniques such as a bisection search [33]. We can write the problem as

$$\begin{aligned}\min_{\{\boldsymbol{\mu}\in\mathcal{Q},\boldsymbol{\nu}\in\mathcal{R},\boldsymbol{\sigma}\in\mathcal{P}\}} &\sum_{i=0}^{A} J^{(n-i)} \\ = \min_{\{\boldsymbol{\sigma}\in\mathcal{P}\}} &\sum_{i=1}^{A} J^{(n-i)}\left(\boldsymbol{\sigma}^{(n-i)}\right) \\ &+ \min_{\{\boldsymbol{\nu}\in\mathcal{R}\}}\left\{\min_{\{\boldsymbol{\mu}\in\mathcal{Q}\}}\sum_{k=1}^{M} J_k^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})\right\} \end{aligned} \tag{9}$$

where $J^{(n-i)} = E[D_k^{(n-i)}] + \lambda\sum_{k=1}^{M}\sigma_k^{(n-i)}T_k^{(n-i)}$ and $J_k^{(n)} = E[D_k^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})] + \lambda T_k^{(n)}$. Given a specific λ, there are three minimizations in (9). They correspond to the bit allocation for retransmission, bit allocation for FEC, and the optimal mode selection for the current frame based on the remaining delay. The first and second steps can be solved by using exhaustive search, and the optimal mode selection can be found using a dynamic programming (DP) approach. The DP can be viewed as a shortest path problem in a trellis, where each stage corresponds to the mode selection for a given packet [31], [33]. Note that by using the error concealment strategy described in Section II-E, the distortion $E[D_k]$ depends on the encoding modes and probability of source packet loss selected for the previous source packet, as shown in (2). Thus, the Lagrangian $\sum_{k=1}^{M} J_k^{(n)}(\boldsymbol{\mu},\boldsymbol{\nu})$ in (9) is not separable. In this case, the time complexity is $O(|2^L \times |\mathcal{R}| \times M \times |\mathcal{Q}|^2)$, where L is the number of lost packets in the optimization window and $|\cdot|$ denotes the cardinality of the set inside. If the error concealment strategy does not introduce dependency across source packets, the time complexity would be $O(|2^L \times |\mathcal{R}| \times M \times |\mathcal{Q}|)$ [33].

V. Experimental Results

In the simulations, we use an H.263+ codec (Annex K supports slice structure, which is used for source layer packetization) [34] to perform source coding. The source coding parameters are the prediction mode (Intra/Inter/Skip) and the quantizer used for each video packet. Rate control is not implemented in the video streaming system. Thus, every frame has the same transmission delay constraint of one frame's duration, i.e., $T_0^{(n)} = T_F$. We assume that after 1 RTT, channel feedback is available to the encoder in the form of which packets are received or lost.

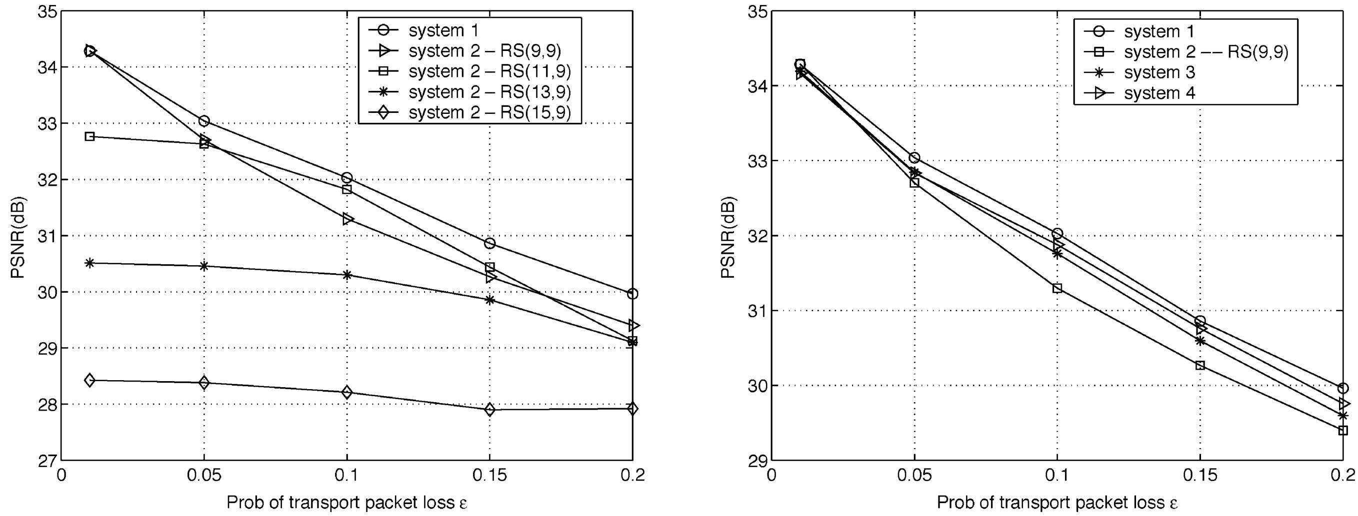
A. IJSCC

Our first experiment is to compare the performance of the IJSCC approach to a sequential JSCC approach. In this experiment, we will illustrate the advantages of this approach with the absence of ARQ, i.e., without taking into account retransmissions. Specifically, we consider applications that require a short end-to-end delay $T_{\max}$, and the RTT is set to equal to two frames. Under that situation, the feedback delay is long enough to preclude retransmissions, so only the source coding parameters $\boldsymbol{\mu}$ and FEC parameters $\boldsymbol{\nu}$ need to be specified. In all simulations of this subsection, we consider the QCIF format Foreman sequence.

In this experiment, four systems are compared: 1) system 1, which uses the proposed framework to jointly consider error resilient source coding and channel coding; 2) system 2, which performs error resilient source coding, but with fixed rate channel coding; 3) system 3, which performs only channel coding, but no error resilient source coding (i.e., source coding is not adapted to the modified channel characteristics after error recovery); and 4) system 4, which performs sequential JSCC. All four systems are optimized in the following manner. System 2 performs optimal error resilient source coding to adapt to the channel errors (with fixed rate channel coding). System 3 selects the optimal channel coding rate to perform FEC and does optimal source coding (without considering residue packet loss after channel coding) at the given bit budget. In the sequential JSCC, channel coding and error resilient source coding are performed sequentially, i.e., bit allocation between source and channel is performed with no awareness of error resilient source coding as in (3), and error resilient source coding is performed thereafter given the bit budget as in (4).

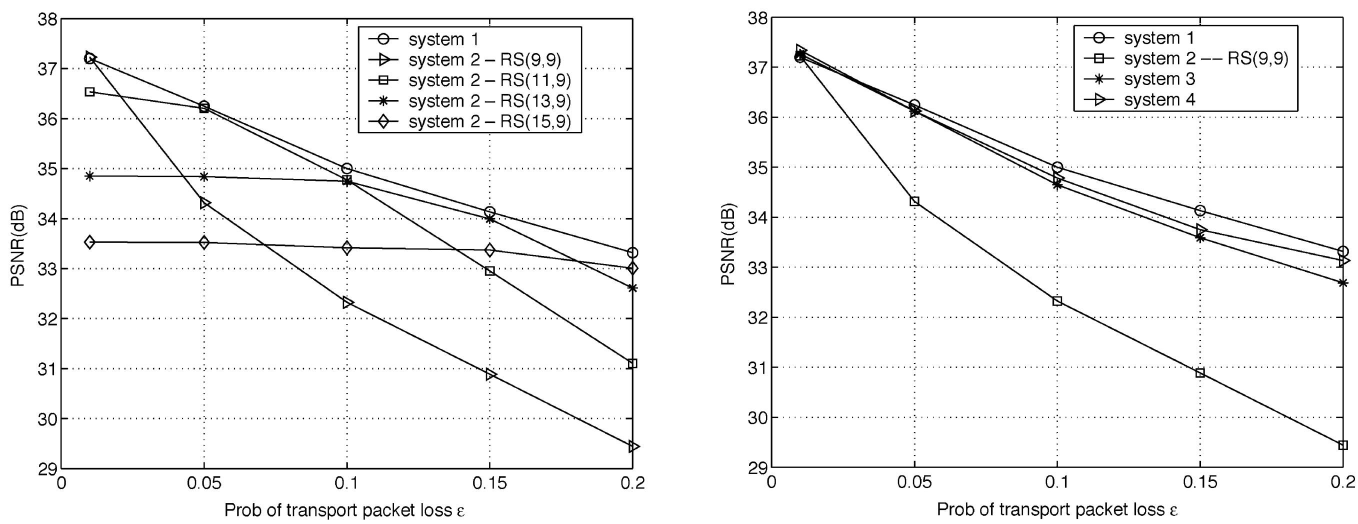
We illustrate the performance of the four systems in Fig. 4 at $R_T = 480$ kbps and the frame rate $F = 30$ fps. Here, we plot the average PSNR against different packet loss rates. All four systems have the same transmission delay constraints and transmission rate. It can be seen in Fig. 4(a) that system 1 outperforms system 2 with different preselected channel coding rates. In addition, system 1 outperforms the optimized system 2 (the upper bound of system 2 with different predefined channel rates) at different channel coding rates by 0.1 to 0.4 dB. This is due to the flexibility of system 1 from varying the channel coding rate in response to the video content.

In Fig. 4(b), we can see that system 3 has higher average PSNR than system 2 without channel coding. Such a result is expected because FEC can change the channel characteristics to a greater extent [e.g., an RS(7, 5) code can change the packet loss probability from 10% to 1.1%] compared to error resilient source coding, which can only adapt to the channel characteristics to a limited degree. Also, as shown in Fig. 4(b), system 1 outperforms systems 3 and 4 by up to around 0.4 and 0.3 dB, respectively. The gain in system 1 compared to system 4 comes from the joint consideration of source coding and channel coding. The gain in system 4 in comparison to system 3 comes from the adaptation of source coding to the modified channel characteristics after error recovery (system 3 does not do error resilient source coding). Note that the gain of the IJSCC system (system 1) as compared to system 3 (without performing error resilient source coding) or system 4 (the sequential JSCC) may not be very significant. This is because in all systems, we perform the optimization by jointly considering several available error control components such as error concealment. Thus, the absence of one of the error control components, as in system 3, or the lack of joint consideration of source and channel coding, as in system 4, may not have a very significant effect due to mitigation from other error control components in the system. Another observation is that in practical situations where computation resources are constrained, application of the integrated system may not be necessary if the additional gain does not out-



(a) (b)

Fig. 4. Average PSNR versus transport packet loss probability. (a) System 1 versus System 2 with indicated channel rates. (b) System 1 versus System 2, 3, and 4 ($R_T = 480$ kbps, $F = 30$ fps).



(a) (b)

Fig. 5. Average PSNR versus transport packet loss probability. (a) System 1 versus System 2 with indicated channel rates. (b) System 1 versus System 2, 3, and 4 ($R_T = 480$ kbps, $F = 15$ fps).

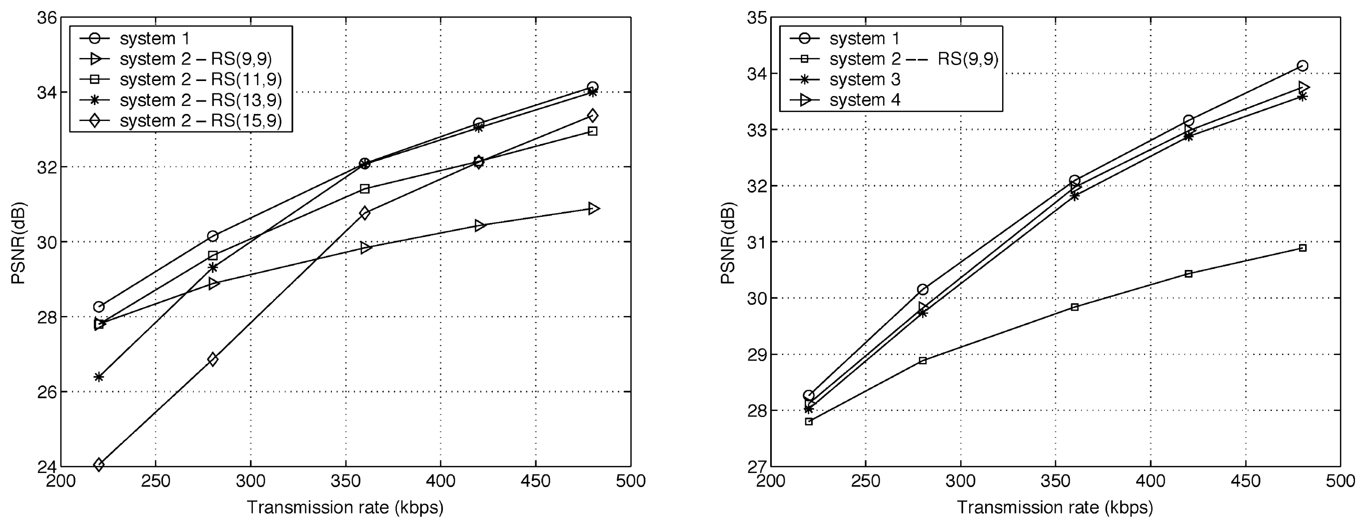
weigh the additional computational complexity. Nevertheless, the integrated system can still be useful in practical situations in that it provides an optimization benchmark against which the performances of other suboptimal systems can be evaluated.

Fig. 5 shows the same performance comparisons at a lower frame rate of $F = 15$ fps. Since the average bit budget per frame is given by R_T/F, this results in larger bit budget per frame. In this case, when the channel loss rate increases, the PSNR curve for system 2 without channel coding deviates from those with channel coding at a much higher rate compared to the situation in Fig. 4. The low bit budget in Fig. 4 restricts the ability to use channel coding, because a majority of the bits are needed for source coding. When the bit budget gets larger, the system becomes more flexible in allocating bits to the channel to improve the overall system performance. The resulting bit rate allocation between source and channel coding is illustrated in Table I.

The effect of bit budget is better illustrated in Fig. 6, where the PSNR is plotted against the transmission rate. It can be clearly seen that as the transmission rate increases (i.e., the bit budget per frame increases), the gap between the performance of system 2 (without channel coding) and the other systems (with channel coding) also increases. As shown in Fig. 6(a), system 1 outperforms system 2 with different preselected channel coding rates at various transmission rates. In addition, system 1 also outperforms systems 3 and 4 at various transmission rates, as shown in Fig. 6(b). The resulting bit rate allocation between source and channel coding is illustrated in Table II.

TABLE I
SOURCE BIT RATE (kbps) IN THE FOUR SYSTEMS BASED ON THE QCIF FOREMAN SEQUENCE. NOTE THAT SYSTEM 2 DENOTES THE OPTIMIZED SYSTEM 2 (THE UPPER BOUND OF SYSTEM 2 WITH DIFFERENT PRE-DEFINED CHANNEL RATES)

ϵ	F=30 fps, R_T=480 kbps				F=15 fps, R_T=480 kbps			
	System 1	System 2	System 3	System 4	System 1	System 2	System 3	System 4
0.01	480	480	475	470	474	480	425	425
0.05	434	480	436	434	369	393	368	368
0.1	410	393	404	406	311	393	311	311
0.15	400	393	387	387	284	332	279	281
0.2	392	480	378	378	259	288	257	255



(a) (b)

Fig. 6. Average PSNR versus transmission rate. (a) System 1 versus System 2 with indicated channel rates. (b) System 1 versus System 2, 3, and 4 ($\epsilon = 0.15$, $F = 15$ fps).

B. Hybrid FEC and Selective Retransmission

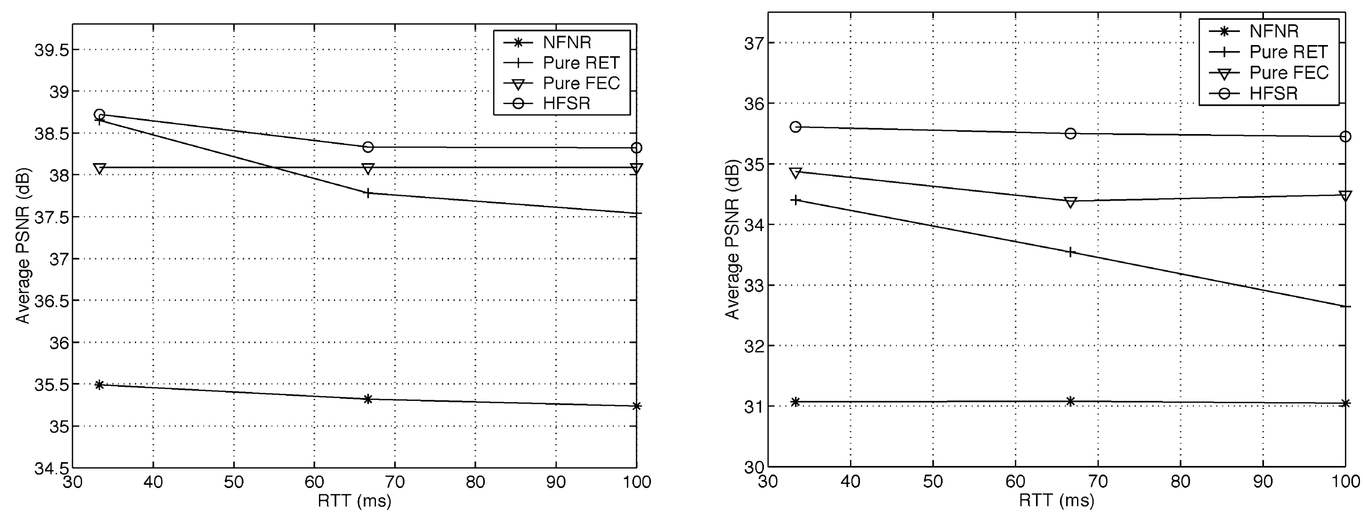
Next, we consider systems where retransmissions are feasible. Four schemes are compared: 1) neither FEC nor retransmission (NFNR); 2) pure retransmission; 3) pure FEC; and 4) hybrid FEC and selective retransmission (HFSR). All four systems are optimized using the IJSCC framework. Although the IJSCC framework in (7) is general, in our simulations, we restrict a packet's retransmission to the extent only when its NAK has been received. In all experiments of this subsection, we consider the QCIF Foreman sequence and set $A = 4$.

1) Sensitivity to RTT: Fig. 7 shows the performance of the four systems in terms of PSNR versus RTT, with different levels of channel loss rate. We set $R_T = 480$ kbps and $F = 15$ fps. As expected, the HFSR system offers the best overall performance. It can also be seen that the pure retransmission approach is much more sensitive to variations in RTT than in FEC. In addition, at low ϵ and low RTT, the pure retransmission approach outperforms the pure FEC system, as shown in Fig. 7(a). However, when the channel gets worse and the RTT becomes larger, the pure FEC system starts to outperform the pure retransmission system, as shown in Fig. 7(b). This means that retransmission is suitable for those applications where the RTT is short and channel loss rate is low, which confirms the observation in [5]. The disadvantage of retransmission when the RTT gets longer comes from two sources. 1) Given the same value of A, which is decided by the initial setup time $T_{\max}$, the number of retransmission opportunities becomes smaller. 2) The amount of errors accumulated due to error propagation from the motion compensation becomes larger, and consequently retransmission of lost packets becomes less efficient.

2) Sensitivity to Packet Loss Rate: In Fig. 8, we plot the performance of the four systems in terms of PSNR versus probability of transport packet loss for different values of RTT when $R_T = 480$ kbps and $F = 15$ fps. The RTT is set equal to T_F and $3T_F$ in Fig. 8(a) and (b), respectively. It can be seen that the HFSR system achieves the best overall performance of the four. The resulting PSNR in the pure retransmission system drops faster than the pure FEC system, which implies that retransmission is more sensitive to packet loss rate. In addition, the pure retransmission system only outperforms the pure FEC system at low ϵ. When the channel loss rate is high, FEC is more efficient since retransmission techniques require frequent retransmissions to recover from packet loss, which results in high bandwidth consumption and is also limited by the delay constraint. For example, when $\mathrm{RTT} = 3T_F$ and $A = 4$, each lost packet has only one opportunity for retransmission, which

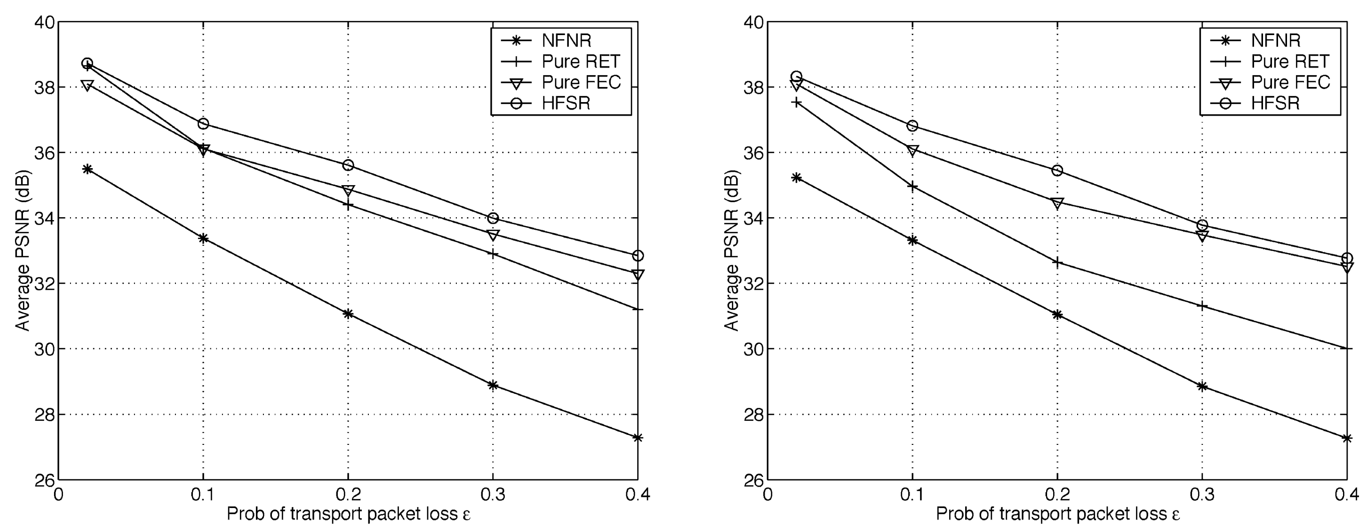
TABLE II
SOURCE BIT RATE (kbps) IN THE FOUR SYSTEMS BASED ON THE QCIF FOREMAN SEQUENCE. NOTE THAT SYSTEM 2 DENOTES THE OPTIMIZED SYSTEM 2 (THE UPPER BOUND OF SYSTEM 2 WITH DIFFERENT PRE-DEFINED CHANNEL RATES.)

R_T (kbps)	F=15 fps, ϵ=0.1				F=15 fps, ϵ=0.15			
	System 1	System 2	System 3	System 4	System 1	System 2	System 3	System 4
220	208	180	185	183	199	180	180	181
280	212	229	209	209	209	229	198	199
360	243	249	245	243	229	249	225	224
420	278	291	277	279	254	291	251	251
480	311	393	311	311	284	332	279	281



(a) (b)

Fig. 7. Average PSNR versus RTT, $R_T = 480$ kbps, $F = 15$ fps. (a) $\epsilon = 0.02$. (b) $\epsilon = 0.2$.

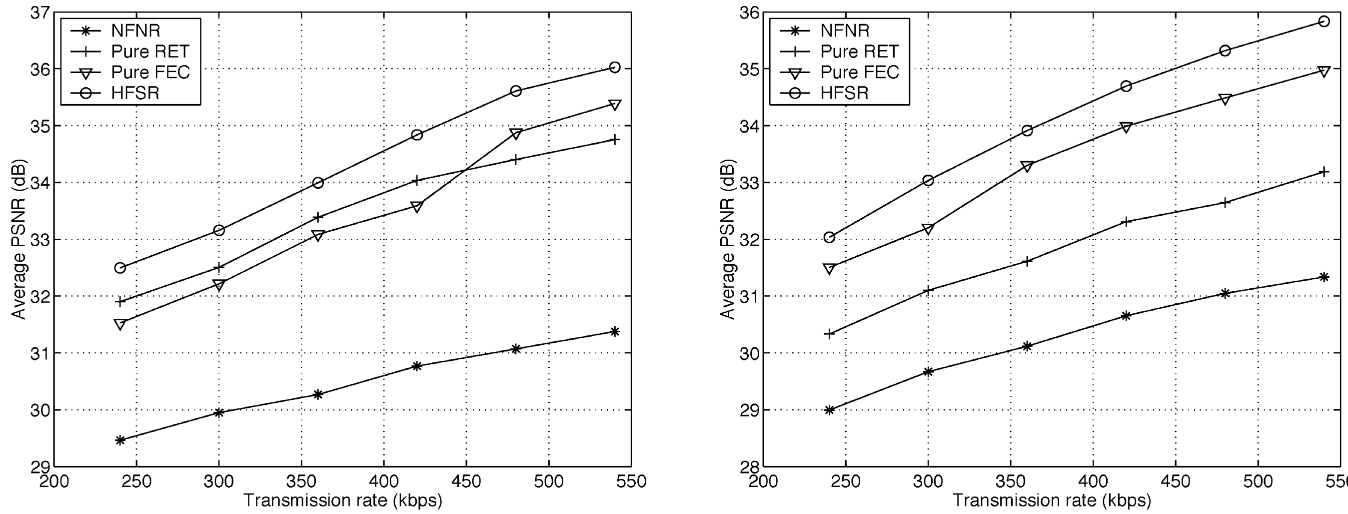


(a) (b)

Fig. 8. Average PSNR versus probability of transport packet loss ϵ, $R_T = 480$ kbps, $F = 15$ fps. (a) $\mathrm{RTT} = T_F$. (b) $\mathrm{RTT} = 3T_F$.

is not enough to recover many losses when $\epsilon = 0.3$. However, when the channel loss rate is small and the RTT is small, retransmission becomes more efficient, since FEC typically requires a fixed amount of bandwidth overhead. Consequently, the pure retransmission system performs close to the HFSR system, as shown in Fig. 8(a).

3) Sensitivity to Transmission Rate: Fig. 9 shows the performance of the four systems in terms of PSNR versus channel



(a) (b)

Fig. 9. Average PSNR versus channel transmission rate R_T, $\epsilon = 0.2$, $F = 15$ fps. (a) $\mathrm{RTT} = T_F$. (b) $\mathrm{RTT} = 3T_F$.

transmission rate when $\epsilon = 0.2$ and $F = 15$ fps. The RTT is set equal to T_F and $3T_F$ in Fig. 9(a) and (b), respectively. As shown in Fig. 9(a), when $\mathrm{RTT} = T_F$, the pure retransmission system outperforms the pure FEC system by up to 0.4 dB when the transmission rate is less than 450 kbps. When the transmission rate is greater than 450 kbps, the pure FEC system starts to outperform the pure retransmission system by up to 0.5 dB. When the RTT becomes longer, as shown in Fig. 9(b), although the pure FEC system always outperforms the pure retransmission system, the difference between the two systems increases from 1.2 to 1.8 dB when the transmission rate increases from 240 to 540 kbps, which means that FEC is more sensitive to variations in the transmission rate. These observations imply that FEC is more efficient than retransmission when the transmission rate becomes greater (resulting in a higher bit budget per frame). In addition, it can be seen that the HFSR system achieves the best overall performance of the four.

4) Time-Varying Channel: In Fig. 10, we show how the proposed HFSR system responds to network fluctuations. The top figure shows the average bit allocation between source coding, FEC, and ARQ. The network fluctuations including variations of channel transmission rate and packet loss probability are illustrated on the bottom graph. The variations of RTT are also indicated on the top graph. In the simulations, the sender does optimization based on the currently estimated CSI. The average bit allocation is obtained based on twenty different channel realizations with the same channel characteristics, as shown in the figure. It can be seen that the system intelligently allocates bits to source coding, FEC, and ARQ, in response to the changing network conditions. For example, during the time from frame 61 to 75, when the RTT is short, transmission rate is low and the packet loss probability is also low, most of the bits are allocated to source coding and the remaining bits are used for ARQ. When the transmission rate increases, more bits are allocated to FEC after the 75th frame, because of the higher flexibility of the proposed system in bit allocation. However, when the packet loss probability increases after the 150th frame, more bits are needed

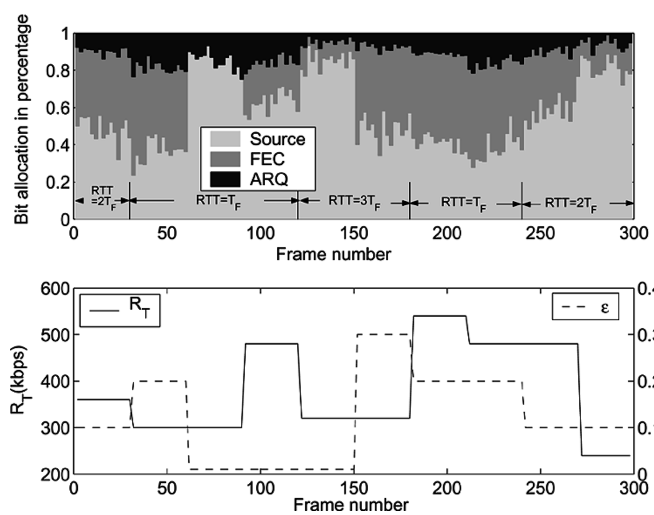


Fig. 10. Averagebit allocation of the HFSR system over a time-varying channel, $F = 15$ fps.

to combat channel errors and, therefore, the amount of bits allocated to source coding must decrease. The observations from Fig. 10 further confirm what we have seen in Figs. 7–9. Thus, the proposed system performs very well in response to the network fluctuations.

Although we only showed simulation results based on the QCIF Foreman sequence, extensive experiments have been carried out and similar results were obtained using other test sequences such as Akiyo, Container, and Carphone.

In summary, retransmission is suitable for short network RTT, low probability of packet loss, and low transmission rate, while FEC is more suitable otherwise. In general, our proposed hybrid FEC and selective retransmission scheme is able to identify the best combination of the two.

VI. Conclusion

This paper addresses the problem of optimal application-layer error control for real-time packetized video transmission. Specifically, we consider error resilient source coding at the encoder, FEC and retransmission at the application layer, and error concealment at the receiver. The optimization is carried out in the proposed IJSCC framework, which jointly considers these error control components to achieve the best video delivery performance. Based on the IJSCC framework, we have compared the performance of different application-layer error control scenarios such as pure FEC, pure ARQ, and hybrid FEC/selective retransmission. Simulation results show that the proposed hybrid FEC/retransmission system achieves better performance than pure FEC and pure retransmission systems.

REFERENCES

[1] R. O. Hinds, T. N. Pappas, and J. S. Lim, "Joint block-based video source-channel coding for packet-switched networks," *Proc. SPIE*, vol. 3309, pp. 124–133, Jan. 1998.

[2] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966–976, Jun. 2000.

[3] D. Wu, Y. T. Hou, B. Li, W. Zhu, Y.-Q. Zhang, and H. J. Chao, "An end-to-end approach for optimal mode selection in Internet video communication: theory and application," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 977–995, Jun. 2000.

[4] G. Côté, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 952–965, Jun. 2000.

[5] F. Hartanto and H. R. Sirisena, "Hybrid error control mechanism for video transmission in the wireless IP networks," in *Proc. IEEE 10th Workshop on Local and Metropolitan Area Networks*, Sydney, Australia, Nov. 1999, pp. 126–132.

[6] P. A. Chou, A. E. Mohr, A. Wang, and S. Mehrotra, "Error control for receiver-driven layered multicast of audio and video," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 108–122, Mar. 2001.

[7] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the Internet: challenges and approaches," *Proc. IEEE*, vol. 88, pp. 1855–1877, Dec. 2000.

[8] Y. Wang, G. Wen, S. Wenger, and A. K. Katsaggelos, "Review of error resilience techniques for video communications," *IEEE Signal Process. Mag.*, vol. 17, no. 7, pp. 61–82, Jul. 2000.

[9] J. Rosenberg and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction," Tech. Rep., Internet Engineering Task Force, Request for Comments (Proposed Standard) 2733, Dec. 1999.

[10] G. J. Wang, Q. Zhang, W. W. Zhu, and Y.-Q. Zhang, "Channel-adaptive error control for scalable video over wireless channel," presented at the IEEE MoMuc Conf., Oct. 2000.

[11] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, to be published.

[12] F. Zhai, Y. Eisenberg, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Rate-distortion optimized hybrid error control for real-time packetized video transmission," in *Proc. IEEE Int. Conf. Communications*, vol. 3, Paris, France, Jun. 2004, pp. 1318–1322.

[13] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: an approach to combined source-channel coding," *IEEE Trans. Inf. Theory*, vol. IT-38, no. 4, pp. 827–838, Jul. 1987.

[14] G. Davis and J. Danskin, "Joint source and channel coding for Internet image transmission," presented at the SPIE Conf. Wavelet Applications of Digital Image Processing XIX, Denver, CO, Aug. 1996.

[15] T. Stockhammer and C. Buchner, "Progressive texture video streaming for lossy packet networks," presented at the Int. Packet Video Workshop, Kyongju, Korea, Apr. 2001.

[16] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 357–372, Mar. 2001.

[17] Q. Zhang, W. Zhu, and Y.-Q. Zhang, "Network-adaptive scalable video streaming over 3G wireless network," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Thessaloniki, Greece, Oct. 2001, pp. 579–582.

[18] R. Zhang, S. L. Regunathan, and K. Rose, "End-to-end distortion estimation for RD-based robust delivery of pre-compressed video," presented at the 35th Asilomar Conf. Signals, Systems, Computers, Pacific Grove, CA, Oct. 2001.

[19] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 511–523, Jun. 2002.

[20] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, "Joint source-channel coding for motion-compensated DCT-based SNR scalable video," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1043–1052, Sep. 2002.

[21] J. Kim, R. M. Mersereau, and Y. Altunbasak, "Error-resilient image and video transmission over the Internet using unequal error protection," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 121–131, Feb. 2003.

[22] S. Appadwedula, D. L. Jones, K. Ramchandran, and L. Qian, "Joint source channel matching for wireless image transmission," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Chicago, IL, Oct. 1998, pp. 137–141.

[23] S. Zhao, Z. Xiong, and X. Wang, "Joint error control and power allocation for video transmission over CDMA networks with multiuser detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 425–437, Jun. 2002.

[24] G. Cheung and A. Zakhor, "Bit allocation for joint source/channel coding of scalable video," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 340–356, Mar. 2000.

[25] S. Falahati, A. Svensson, N. C. Ericsson, and A. Ahlén, "Hybrid type-II ARQ/AMS and scheduling using channel prediction for downlink packet transmission on fading channels," presented at the Nordic Radio Symp., 2001.

[26] I. Akyildiz, Y. Altunbasak, F. Fekri, and R. Sivakumar, "AdaptNet: an adaptive protocol suite for the next-generation wireless Internet," *IEEE Commun. Mag.*, pp. 128–136, Mar. 2004.

[27] J. Chakareski and P. A. Chou, "Application layer error correction coding for rate-distortion optimized streaming to wireless clients," presented at the Int. Conf. Acoustics, Speech, and Signal Processing, Orlando, FL, May 2002.

[28] M. Yajnik, S. Moon, and J. Jurose *et al.*, "Measurement and modeling of the temporal dependence in packet loss," Tech. Rep. 98-78, 1998.

[29] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1737–1744, Nov. 1996.

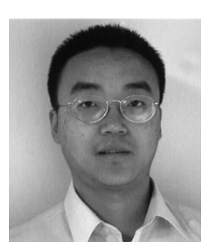
[30] X. Yang, C. Zhu, Z. Li, G. Feng, S. Wu, and N. Ling, "Unequal error protection for motion compensated video streaming over the Internet," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Rochester, NY, Sep. 2002, pp. 717–720.

[31] F. Zhai, C. E. Luna, Y. Eisenberg, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Joint source coding and packet classification for real-time video transmission over differentiated services networks," *IEEE Trans. Multimedia*, vol. 7, no. 4, pp. 716–726, Sep 2005.

[32] F. Zhai, Y. Eisenberg, C. E. Luna, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Packetization schemes for forward error correction in Internet video streaming," presented at the 41st Allerton Conf. Communication, Control and Computing, Oct. 2003.

[33] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion Based Video Compression: Optimal Video Frame Compression and Object Boundary Encoding*. Norwell, CA: Kluwer, 1997.

[34] "Video Coding for Low Bitrate Communication," ITU Telecom. Standardization Sector of ITU, Draft ITU-T Recommendation H.263 Version 2, Sep. 1997.



Fan Zhai (S'99–M'04) received the B.S. and M.S. degrees in electrical engineering from Nanjing University, Nanjing, China, in 1996 and 1998, respectively, and the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, IL, in 2004.

He is currently a System Engineer in the Digital Video Department, Texas Instruments, Dallas, TX. His primary research interests include image and video signal processing and compression, multimedia communications and networking, and multimedia analysis.

Yiftach Eisenberg (S'02–M'04) received the B.S. degree in electrical engineering from the University of Illinois, Urbana-Champaign, in 1999, and the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 2001 and 2004, respectively.

He is currently a Senior Engineer at BAE Systems, Advanced Systems and Technology, Merrimack, NH. In 2000, he was a Visiting Researcher with Motorola Labs, Schaumburg, IL, in the Multimedia Research Laboratory. His primary research interests include signal processing, wireless communications, networking, and video compression and analysis.

Thrasyvoulos N. Pappas (M'87–SM'95) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1979, 1982, and 1987, respectively.

From 1987 to 1999, he was a Member of the Technical Staff at Bell Laboratories, Murray Hill, NJ. In September 1999, he joined the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, as an Associate Professor. His research interests are in image and video compression, video transmission over packet-switched networks, perceptual models for image processing, model-based halftoning, image and video analysis, video processing for sensor networks, audiovisual signal processing, and DNA-based digital signal processing.

Dr. Pappas has served as Chair of the IEEE Image and Multidimensional Signal Processing Technical Committee, Associate Editor and Electronic Abstracts Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, Technical Program Co-Chair of ICIP'01 and IPSN'04, and, since 1997, he has been Co-Chair of the SPIE/IS&T Conference on Human Vision and Electronic Imaging. He was also Co-Chair of the 2005 IS&T/SPIE Symposium on Electronic Imaging: Science and Technology.

Randall Berry (S'93–M'00) received the B.S. degree in electrical engineering from the University of Missouri, Rolla, in 1993, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, in 1996 and 2000, respectively.

He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL. In 1998, he was on the technical staff at the MIT Lincoln Laboratory in the Advanced Networks Group. His primary research interests include wireless communication, data networks, and information theory.

Dr. Berry is the recipient of a 2003 National Science Foundation CAREER award.

Aggelos K. Katsaggelos (S'80–M'85–SM'92–F'98) received the Diploma degree in electrical and mechanical engineering from Aristotelian University of Thessaloniki, Thessaloniki, Greece, in 1979 and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1981 and 1985, respectively.

In 1985, he joined the Department of Electrical and Computer Engineering at Northwestern University, Evanston, IL, where he is currently a Professor, holding the Ameritech Chair of Information Technology. He is also the Director of the Motorola Center for Communications and a member of the Academic Affiliate Staff, Department of Medicine, at Evanston Hospital. He is the editor of *Digital Image Restoration* (New York: Springer-Verlag, 1991), coauthor of *Rate-Distortion Based Video Compression* (Norwell, MA: Kluwer, 1997), and co-editor of *Recovery Techniques for Image and Video Compression and Transmission* (Norwell, MA: Kluwer, 1998), and the co-inventor of eight international patents

Dr. Katsaggelos is a member of the Publication Board of the IEEE PROCEEDINGS, the IEEE Technical Committees on Visual Signal Processing and Communications, and Multimedia Signal Processing, the Editorial Board of Academic Press, Marcel Dekker: Signal Processing Series, *Applied Signal Processing*, and *Computer Journal*. He has served as Editor-in-Chief of the *IEEE Signal Processing Magazine* (1997–2002), member of the Publication Boards of the IEEE Signal Processing Society, the IEEE TAB Magazine Committee, Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1990–1992), Area Editor for the journal *Graphical Models and Image Processing* (1992–1995), member of the Steering Committees of the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1992–1997) and the IEEE TRANSACTIONS ON MEDICAL IMAGING (1990–1999), member of the IEEE Technical Committee on Image and Multi-Dimensional Signal Processing (1992–1998), and a member of the Board of Governors of the IEEE Signal Processing Society (1999–2001). He is the recipient of the IEEE Third Millennium Medal (2000), the IEEE Signal Processing Society Meritorious Service Award (2001), and an IEEE Signal Processing Society Best Paper Award (2001).