FORWARD AND RETRANSMITTED SYSTEMATIC LOSSY ERROR PROTECTION FOR IPTV VIDEO MULTICAST

Zhi Li[1], *Xiaoqing Zhu*[2], *Ali C. Begen*[2] *and Bernd Girod*[1]

[1]Department of Electrical Engineering, Stanford University, Stanford, CA 94305
[2]Cisco Systems, Inc. 170 West Tasman Drive, San Jose, CA 95134
{leeoz, bgirod}@stanford.edu, {xiaoqzhu, abegen}@cisco.com

ABSTRACT

Emerging IPTV deployments combine Forward Error Correction (FEC) and packet retransmissions to resist the impulse noise of the Digital Subscriber Line (DSL) links. In this work, we re-engineer such a solution to improve its robustness against impulse noise while keeping it backward-compatible with the current network infrastructure. We propose forward and retransmitted Systematic Lossy Error Protection (SLEP/SLEPr), which effectively provides error resiliency at the expense of some slight loss of video quality. We demonstrate its effectiveness through a set of experiments. We further present an analytical model for SLEP/SLEPr and show that the experimental results can be explained by the analysis.

Index Terms— IPTV, error-resilient video, impulse noise, Systematic Lossy Error Protection, error control

1. INTRODUCTION

Advances in video and networking technologies have made the delivery of television over telephone lines a reality. Internet Protocol TeleVision (IPTV) service is delivered over a carefully engineered network infrastructure and thus capable of meeting stringent quality-of-service (QoS) measures. IPTV is currently being deployed by many telcos throughout the world.

The deployment of IPTV over DSL has its own challenges. One of these challenges is the potentially devastating impact of impulse noise on the subscriber line. Possible sources of impulse noise include AC power switches, motors and lightning strikes. Depending on the duration, impulse noise can be put into three categories, namely, Repetitive Electrical Impulse Noise (REIN) with a duration less than 1 ms, Prolonged Electrical Impulse Noise (PEIN) with a duration between 1 ms and 10 ms, and Single Isolated Impulse Noise (SHINE) with a duration larger than 10 ms.

In this work, we address the impulse noise problem of IPTV from a video coding perspective. Our goal is to reduce video quality fluctuations due to damaged IP packets in a backward-compatible way; that is, the proposed solution makes a modification to an existing IPTV system only in the application layer, with as little impact as possible on the underlying network infrastructure. The proposed solution extends the Systematic Lossy Error Protection (SLEP) [1] scheme to a hybrid forward and retransmission scenario. Its effectiveness does not rely on the accurate statistical modeling of the impulse noise [2], thus is potentially robust against the mismatch between the noise model and reality.

This paper is presented as follows. In Section 2, we give an introduction to a hybrid error-control method that has been developed for recovering the packet losses in real time in IPTV systems. Section 3 introduces SLEP and hybrid ARQ – two related concepts to our work. Section 4 describes the proposed SLEP/SLEPr solution. Section 5 presents an analytical model of SLEP/SLEPr. In Section 6, we present and discuss the experimental and analytical results.

2. HYBRID ERROR CONTROL FOR IPTV

This section provides an overview of a hybrid error-control method [3] that has been deployed for IPTV services running over DSL. Its major system components are illustrated in Fig. 1.

In this system, the primary video stream, which is carried in an MPEG2 Transport Stream (MPEG2-TS) [4], is encapsulated in Real-time Transport Protocol (RTP) [5] packets and sent from the source to the IP set-top boxes (STBs). Also sent from the source (not necessarily the same source as the primary video) and encapsulated in RTP are the generated FEC packets. Attached to the edge router (or access aggregation router) is one or more repair servers that temporarily cache the primary video stream. The source video packets as well as the corresponding FEC packets are delivered to the IP STBs over a single-source multicast (SSM) session. The IP STBs first attempt to recover the missing source packets by using the FEC packets. If this fails, they request a retransmission for the missing packet(s) from the repair server. The repair server then retransmits the requested packets over a unicast session.

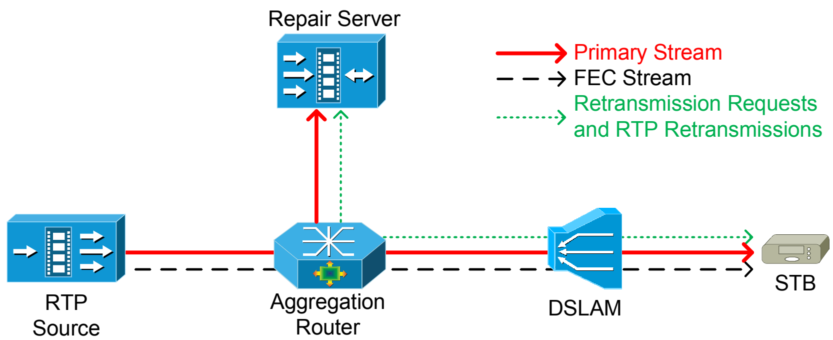


Fig. 1. A hybrid error-control method for IPTV systems running over DSL.

For the packet-level FEC, many types of FEC codes may be used. One of the most widely used FEC codes is the parity FEC code [3]. In this code, a block of source packets are arranged in a matrix of D rows and L columns. Then, a bit-by-bit XOR operation is applied across each row and column to generate L interleaved and D non-interleaved parity packets. This erasure channel FEC code has a correction capability, which is upper bounded by $(N - K)$ [6].

Service providers usually offer IPTV services in a bundle with other services, such as voice over IP (VoIP) and Internet access (also known as triple-play services). In a typical solution, different services share the DSL bandwidth as sketched in Fig. 2. Traffic is put into different priority classes, with VoIP and IPTV traffic having the highest priority whereas the best-effort Web traffic the lowest. Furthermore, different types of IPTV traffic also have different priorities – the primary video stream always has the highest. This is followed by the FEC traffic and the retransmission traffic. Typically, the primary video is encoded with constant bitrate. As a result, FEC also occupies a constant bitrate. On the other hand, the retransmission stream exhibits bursty behaviors, *stealing* bandwidth from the best-effort services as needed, up to a certain limit. This limit is imposed to maintain the interactivity of Web browsing.

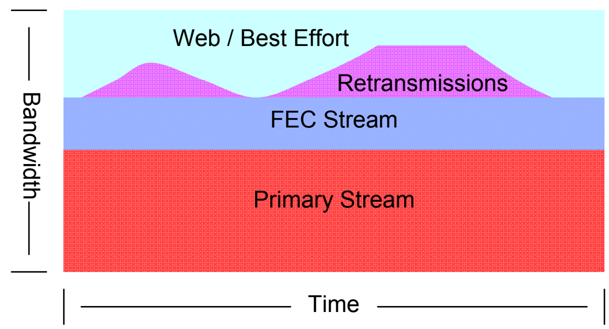


Fig. 2. Bandwidth sharing between IPTV and Internet access services.

3. RELATED WORK

3.1. Hybrid ARQ

Hybrid Automatic Retransmission reQuest (ARQ) is a variation of the standard ARQ where FEC is combined with retransmissions for error correction. Two types of Hybrid ARQ schemes have been used. In Type I, the source message and the parity are forward transmitted. At the receiver, error correction decoding is attempted. If the attempt fails, retransmission of the source data is requested. The error-control method described in Section 2 essentially uses Type I Hybrid ARQ. In Type II, the receiver sends a request asking for more parity information instead. When received, the parity and the forward transmitted parity are both used in a stronger error correction decoding. Type II Hybrid ARQ is also known as incremental redundancy (IR) or on-demand FEC. Rate-adaptive codes such as shortened Reed-Solomon (RS) codes or Fountain Codes are required in Type II Hybrid ARQ.

3.2. SLEP

SLEP is a content-level forward error protection scheme for robust transmission of video over packet erasure channels [1]. The scheme is *systematic* in the sense that the protection stream is separable from the source stream, and *lossy* in the sense that robustness is achieved at the expense of some slight loss of video quality. The concept originates from the distributed source coding principle and the practice of digital enhancement of an analog channel [7]. SLEP is able to achieve a graceful degradation performance even under very noisy channel conditions because it can avoid the *cliff effect* that is usually experienced in conventional FEC. A practical implementation of SLEP using H.264/AVC redundant slices is described in [1].

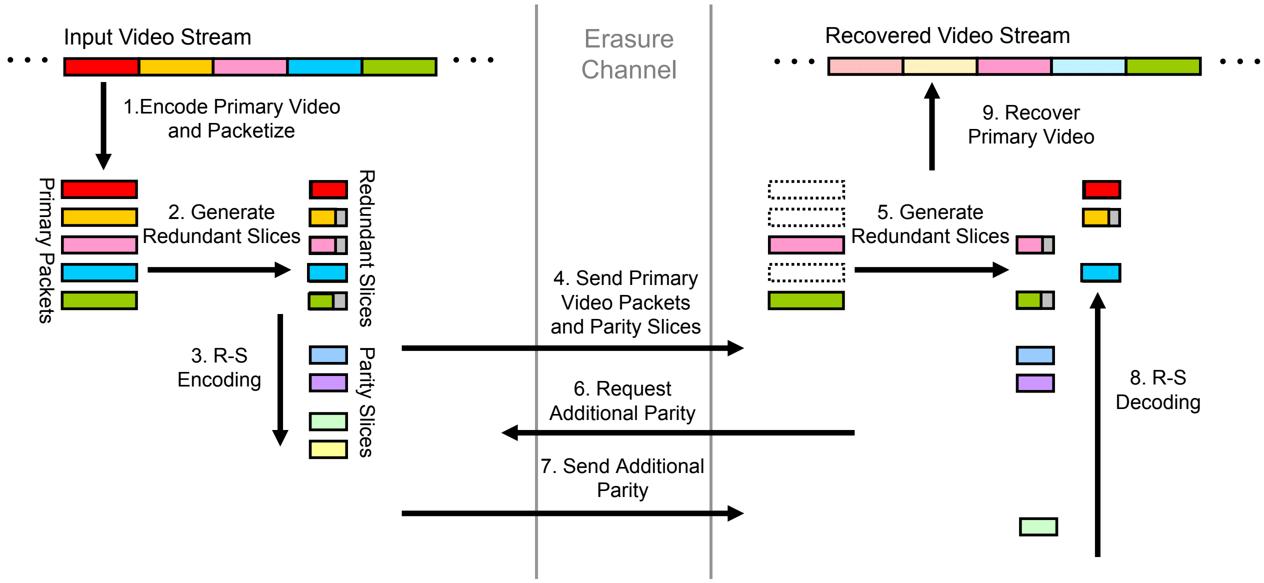


Fig. 3. Packet-level illustration of the proposed SLEP/SLEPr scheme.

4. PROPOSED SOLUTION – SLEP/SLEPR

Recall that our design goal is to alleviate the video quality degradation due to the damaged packets in a way that is backward-compatible with the existing error-control framework. Our solution addresses the impulse noise issue from a video coding perspective by extending the SLEP scheme to a hybrid forward and retransmission scenario. Since at the expense of some slight loss of quality, SLEP applies stronger FEC, we expect that it will be effective against bursty errors. We reserve the term *SLEP* to specifically refer to the forward transmitted SLEP and dub the retransmitted SLEP as *SLEPr*. The procedure of using the forward and retransmitted SLEP to combat bursty packet losses is described as follows. Refer to Fig. 3 for a packet-level illustration.

- The input video stream is encoded into the primary stream using a standard H.264/AVC encoder and packetized, one primary slice per packet. From the primary stream, the redundant slices are generated by requantizing and encoding the predicted residues. The requantizer used is coarser than the quantizer used in generating the primary slice such that the redundant slice could have smaller size than the primary. It is possible to have redundant slices of different sizes. In one solution, they can be zero-padded and aligned exactly. Another solution is to use different quantizers to fine-tune the coefficients to make the redundant slices the same size.

- (N, K) RS encoding is applied across the K redundant slices to generate $(N-K)$ parity slices. The parameters N and K are chosen to meet the rate and delay constraints. After that, the redundant slices are discarded and only the parity slices are kept. Each parity slice is put into one packet, along with some additional information. Among the parity packets generated, some are used for forward transmission whereas the others for retransmission.

- The primary and forward parity packets are sent to the receiver over the error-prone erasure channel.

- At the receiver side, the received primary packets are used to regenerate the redundant slices. With the regenerated redundant slices and the received parity slices, RS decoding is attempted. If decoding succeeds, the missing redundant slices can be recovered. If it fails, the receiver sends a retransmission request for additional parity packets.

- Upon reception of the additional parity packets, RS decoding is attempted again using the regenerated redundant slices and all the currently available parity slices. If it succeeds, the missing redundant slices can be recovered. The redundant slices are then decoded and spliced back to the motion-compensated primary video stream. This is followed by the standard H.264/AVC decoding process.

Note that the combination of SLEP and SLEPr is one of many possible ways of hybrid forward and retransmission error protection. Other possible ways could be SLEP with conventional retransmission (RET), conventional FEC with SLEPr, etc. Depending on match or mismatch of the

	Forward Quality	Retransmission Quality	Hybrid ARQ Type
FEC/RET	Fine	Fine	I or II
FEC/SLEPr	Fine	Coarse	I
SLEP/RET	Coarse	Fine	I
SLEP/SLEPr	Coarse	Coarse	I or II

Table 1. Possible combinations of forward and retransmitted error correction information.

forward and retransmission stream quality, different hybrid ARQ types could be applied. Table 1 lists all the possible combinations.

5. ANALYTICAL MODEL

In this section, we extend the analysis in [8] and develop an analytical model of SLEP/SLEPr. This model can help us understand the performance of SLEP/SLEPr as well as its design tradeoffs.

5.1. Source and Channel Models

Consider the source as a stationary first-order Gaussian Markov process $\{X_n\}$, which can be modeled as $X_n = \rho X_{n-1} + W_n$, for $|\rho| < 1$, where ρ is the correlation coefficient between X_n and X_{n-1}, and $W_n \sim$ i.i.d. $\mathcal{N}(0, \sigma_W^2)$ is the innovation process. We choose this process to model the temporal correlation between the video frames.

For the impulse-noise channel, we assume that an impulse of burst length T_{bst} occurs in every interval of T_{mtbb} seconds, where the subscript mtbb stands for the mean time between bursts. The duration of the impulse follows an exponential distribution with a mean between 2 ms to 16 ms.

5.2. Primary Stream Encoding

The block diagram of the encoder is illustrated in Fig. 4. First, a linear predictor $X_n = \rho \hat{X}_{n-1} + W_n$ is used. At high rates, $\hat{X}_n \approx X_n$. The predicted residue W_n is then quantized using a uniform quantizer $q_1(\cdot)$ with step size $\Delta_1 = \Delta$, producing the quantization index Q_{1n}. The minimum mean-square error (MMSE) reconstruction is $\hat{W}_{1n} = E[W_n|Q_{1n}]$, resulting in reconstruction error $E_{1n}^W = W_n - \hat{W}_{1n}$ with a uniform distribution $U[-\Delta/2, \Delta/2]$. The quantized index Q_{1n} is further entropy-coded and we assume that the coder achieves the entropy, *i.e.*,

$$R_1 = H(Q_{1n}) = h(W_n) - \log_2 \Delta = \frac{1}{2} \log_2 \left(2\pi e \frac{\sigma_W^2}{\Delta^2} \right). \tag{1}$$

The codeword generated by the entropy coder is transmitted through an erasure channel with an erasure probability of p.

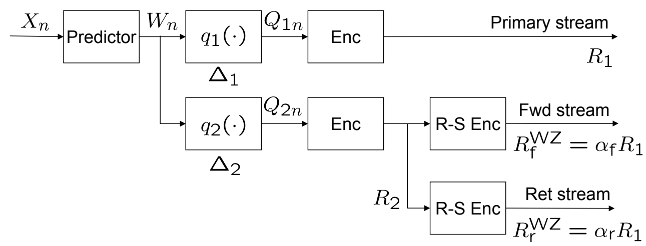


Fig. 4. Block diagram of the analytical model of the SLEP/SLEPr encoder.

5.3. Wyner-Ziv Error Protection Stream Encoding

Along with the primary stream of codeword, we also transmit a Wyner-Ziv (W-Z) error protection stream, which is generated as follows. The predicted residue W_n is quantized using a coarser uniform quantizer $q_2(\cdot)$ with step size Δ_2, which is larger than Δ_1. For convenience, assume an embedded quantizer, letting $\Delta_2/\Delta_1 = m \in \mathbb{Z}^+$. The generated quantization index is Q_{2n}. Similarly the MMSE reconstruction is $\hat{W}_{2n} = E[W_n|Q_{2n}]$. The reconstruction error $E_{2n}^W = W_n - \hat{W}_{2n}$ follows a uniform distribution $U[-m\Delta/2, m\Delta/2]$. Q_{2n} after entropy coding has a rate of

$$R_2 = H(Q_{2n}) = h(W_n) - \log_2(m\Delta) = R_1 - \log_2 m. \tag{2}$$

The codeword after entropy coding is then encoded by a systematic (N, K) RS code, with correcting capability $(N - K)$ symbols per block. Let $M = N - K = M_\text{f} + M_\text{r}$, where M_f is the number of symbols used in forward error protection and M_r in retransmission. After RS encoding, the systematic part is discarded. The resulting W-Z forward error protection stream has a rate of

$$R_\text{f}^{\text{WZ}} = \frac{M_\text{f} R_2}{K}, \tag{3}$$

and the W-Z retransmission stream has a rate of

$$R_\text{r}^{\text{WZ}} = \frac{M_\text{r} R_2}{K}. \tag{4}$$

For Type II Hybrid ARQ, M_r is the number of additional parity symbols transmitted. If Type I is used instead, M_r is the number of retransmitted symbols of the primary stream. Assume that the probability of the forward-retransmission scheme recovering the erasure is $\Pr(\text{recovery} \mid \text{erasure}) = \delta$.

5.4. Recovery Probability δ

Since in RS coding, each packet is treated as a symbol, in this section we use the terms symbol and packet interchangeably.

We first relate the erasure probability p to the burst length of the impulse noise T_{bst}. We assume T_{bst} is a deterministic value first and later generalized to the case that it is an exponentially distributed random variable. The data are arranged in packets of duration T_{pkt}. This leads to the average number of lost packets of

$$L = \frac{T_{\mathrm{bst}} + T_{\mathrm{pkt}}}{T_{\mathrm{pkt}}}. \tag{5}$$

The resultant packet erasure probability is

$$p = \frac{T_{\mathrm{bst}} + T_{\mathrm{pkt}}}{T_{\mathrm{mtbb}}}. \tag{6}$$

Consider the forward and retransmission error protection streams have rate $R_{\mathrm{f}}^{\mathrm{WZ}}$ and $R_{\mathrm{r}}^{\mathrm{WZ}}$, respectively. $R_{\mathrm{f}}^{\mathrm{WZ}}$ is subject to the forward transmission budget α_{f}, which is expressed as the percentage of the primary stream rate, by

$$R_{\mathrm{f}}^{\mathrm{WZ}} = R_1 \alpha_{\mathrm{f}}. \tag{7}$$

Relating (2), (3) and (7), the number of symbols for forward error protection is

$$M_{\mathrm{f}} = \left(\alpha_{\mathrm{f}} \frac{R_1}{R_1 - \log_2 m} \right) K. \tag{8}$$

Note that the length-K RS block coding will result in a forward transmission delay T_{f}, expressed as

$$T_{\mathrm{f}} = K T_{\mathrm{pkt}}. \tag{9}$$

Similar to $R_{\mathrm{f}}^{\mathrm{WZ}}$, $R_{\mathrm{r}}^{\mathrm{WZ}}$ is subject to the retransmission budget α_{r} which is expressed as percentage of the primary stream rate, by

$$R_{\mathrm{r}}^{\mathrm{WZ}} = R_1 \alpha_{\mathrm{r}}. \tag{10}$$

The number of retransmitted error correction symbols in a retransmission delay T_{r} is expressed as:

$$M_{\mathrm{r}} = \frac{\alpha_{\mathrm{r}} T_{\mathrm{r}} R_1}{T_{\mathrm{pkt}}(R_1 - \log_2 m)}. \tag{11}$$

Note that we have assumed that T_{f} is much larger than T_{bst} and the chance that the burst falls into two different coding blocks can be ignored. This is a worst-case assumption since if a burst falls into two blocks, it can be easily corrected. The assumption greatly simplifies our analysis. Further, the usual practice of interleaving can be ignored under this assumption.

Suppose an end-to-end play-out delay T_{e2e} is allowed. The total delay consists of the forward transmission delay T_{f}, one single-trip forward propagation delay, retransmission delay T_{r}, and one round-trip propagation delay. Denote the total propagation delay (*i.e.*, the sum of one single-trip and one round-trip time) as T_{p}. The total delay must satisfy:

$$T_{\mathrm{f}} + T_{\mathrm{r}} + T_{\mathrm{p}} \leq T_{\mathrm{e2e}}. \tag{12}$$

The number of retransmitted packets can be computed according to (4), (10), (11) and (12).

If Type II Hybrid ARQ is used, the forward-retransmission scheme can correct up to $(M_{\mathrm{f}} + M_{\mathrm{r}})$ packets. That is, the recovery probability for given T_{bst} is

$$\delta \mid T_{\mathrm{bst}} = \begin{cases} 1, & L \leq M_{\mathrm{f}} + M_{\mathrm{r}} \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

If Type I hybrid ARQ is used instead, when the forward decoding fails (*i.e.*, $L > M_{\mathrm{f}}$) the retransmitted M_{r} packets are used to cover the lost packets. Then we have

$$\delta \mid T_{\mathrm{bst}} = \begin{cases} \min\left(1, \frac{M_{\mathrm{r}}}{L}\right), & L > M_{\mathrm{f}} \\ 1, & \text{otherwise.} \end{cases} \tag{14}$$

Now assume the error burst length follows a exponential distribution,*i.e.*, $T_{\mathrm{bst}} \sim f_{\exp}(t)$, The average δ can be computed as

$$\delta = \int (\delta \mid t) f_{\exp}(t) dt. \tag{15}$$

5.5. Source Reconstruction at the Decoder

In the event [with probability $p(1-\delta)$] that the codeword is erased but the W-Z error recovery fails, the predicted residue is reconstructed as $E[W_n] = 0$. To summarize, at the decoder, the reconstruction is

$$\hat{W}_n = \begin{cases} \hat{W}_{1n}, & \text{w.p.} \quad (1-p) \\ \hat{W}_{2n}, & \text{w.p.} \quad p\delta \\ 0, & \text{w.p.} \quad p(1-\delta). \end{cases} \tag{16}$$

The reconstruction error is $E_n^W = W_n - \hat{W}_n$. The distribution of E_n^W follows a mixture model of

$$(1-p)U[-\Delta/2, \Delta/2] + p\delta U[-m\Delta/2, m\Delta/2] + p(1-\delta)\mathcal{N}(0, \sigma_W^2), \tag{17}$$

which has a zero mean and a variance of

$$\sigma_{EW}^2 = (1 - p + p\delta m^2)\frac{\Delta^2}{12} + p(1-\delta)\sigma_W^2. \tag{18}$$

The last step of decoding is to reconstruct $\hat{X}_n$ from $\hat{W}_n$ using $\hat{X}_n = \rho \hat{X}_{n-1} + \hat{W}_n$. The reconstruction error $E_n^X = X_n - \hat{X}_n$ can be expressed as:

$$\begin{aligned} E_n^X &= (\rho X_{n-1} + W_n) - (\rho \hat{X}_{n-1} + \hat{W}_n) \\ &= \rho(X_{n-1} - \hat{X}_{n-1}) + (W_n - \hat{W}_n) \\ &= \rho E_{n-1}^X + E_n^W. \end{aligned} \tag{19}$$

It is easy to see that E_n^W is i.i.d. and that E_n^X is an regressive process. Expressing E_n^X in $\{E_n^W\}$ gives:

$$\begin{aligned} E_n^X &= \rho E_{n-1}^X + E_n^W \\ &= \rho^2 E_{n-2}^X + \rho E_{n-1}^W + E_n^W = \ldots \\ &= \sum_{m=0}^{\infty} \rho^m E_{n-m}^W. \end{aligned} \tag{20}$$

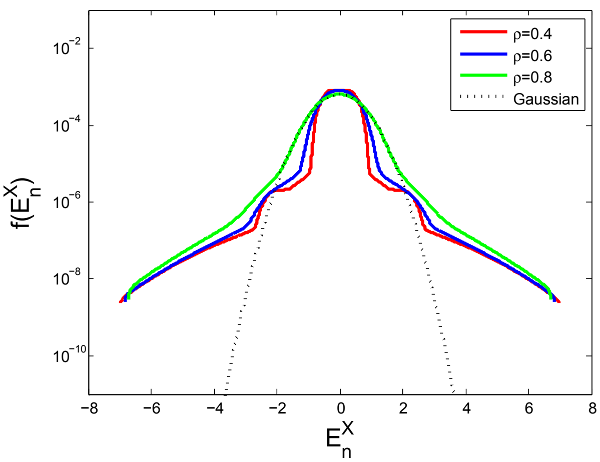


Fig. 5. Typical distributions of the reconstruction error E_n^X under different values of ρ.

From (20) we can see that E_n^X is the weighted sum of E_n^W's. We can find its distribution by convolving the distributions of $\rho^m E_{n-m}^W$'s. Fig. 5 shows some typical distributions of E_n^X under different values of ρ. We can see that, as ρ approaches 1, the distribution more closely resembles a Gaussian. However, E_n^X exhibits a distribution that has a longer tail than Gaussian. This tail captures the impulsive quality degradation due to the failure to recover the error, and is what we want to measure. In other words, we are interested in measuring the probability that the reconstruction SNR is below a certain threshold τ, or $\Pr(\mathrm{SNR} < \tau)$, where

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma_X^2}{\left(E_n^X\right)^2}. \tag{21}$$

Let $F_{EX}(x) = \Pr(E_n^X < x)$ be the cumulative distribution of E_n^X. We have:

$$\begin{aligned} \Pr(\mathrm{SNR} < \tau) &= \Pr\left(|E_n^X| > \sqrt{\frac{\sigma_W^2}{10^{\frac{\tau}{10}}(1-\rho^2)}}\right) \\ &= 2F_{EX}\left(-\sqrt{\frac{\sigma_W^2}{10^{\frac{\tau}{10}}(1-\rho^2)}}\right). \end{aligned} \tag{22}$$

6. RESULTS AND DISCUSSIONS

We first describe general experiment settings, then present the results for three sets of experiments. In the first experiment, we fix the forward transmission to be the conventional FEC with a rate budget $\alpha_{\mathrm{f}} = 5\%$ and compare SLEPr to the conventional retransmission (RET), both with a retransmission budget of $\alpha_{\mathrm{r}} = 2\%$. In the second experiment, we compare SLEP/SLEPr to other forward/retransmission combinations, with $\alpha_{\mathrm{f}} = 8\%$ and $\alpha_{\mathrm{r}} = 2\%$. In the third experiment, we fix the total repair budget $(\alpha_{\mathrm{f}} + \alpha_{\mathrm{r}})$ to 10% and vary the ratio between SLEP and SLEPr to observe the effects.

6.1. Experiment Settings

6.1.1. Evaluation Criteria

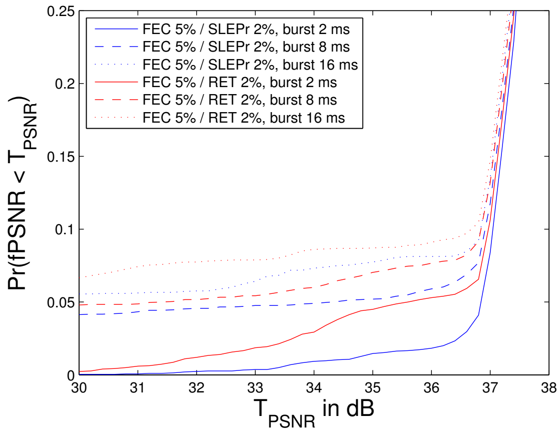
For fairness, we ensure that the channel condition, playout delay and bandwidth are the same for the proposed and benchmarked schemes. We are mainly interested in measuring the visual quality of video delivered over the error-prone erasure channel. We propose to use the empirical cumulative distribution (CDF) of video frame PSNR (fPSNR) as a visual quality measure. The reason is that the distortion generated by the impulse noise tends to be sparse but impulsive. Therefore, both the average video quality and its fluctuation matter in measuring the quality of experience. The conventional measure, such as the average PSNR, is only capable of capturing the average video quality, but not the fluctuation. On the other hand, both types of information is manifested in the distribution of fPSNR. Another related measure we will be using is the probability that the fPSNR falls below certain threshold (the probability of corrupted frames), which can be obtained by evaluating the cumulative distribution at a certain fPSNR threshold.

6.1.2. Video

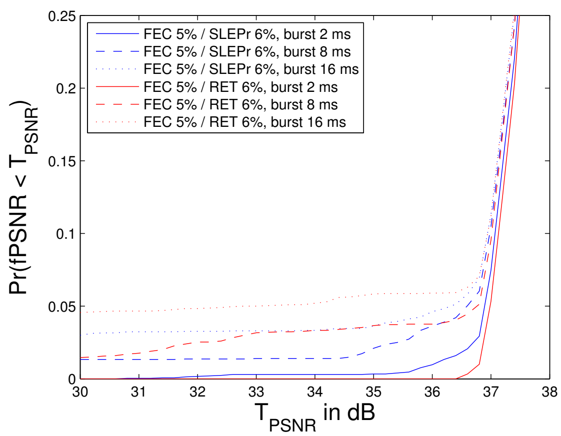
We performed the experiments with the 4CIF SOCCER sequence, which has a resolution of 704×576. The sequence was encoded at 30 frames per second with an H.264/AVC JM (version 13.2) encoder. The frame structure was IPPP and the GOP size was 20. We also used the default motion-compensated error-concealment method built in JM 13.2. We integrated the SLEP block into the encoder and used SLEP-50 (Refer to [1] for more details on SLEP settings).

6.1.3. Channel

A practical issue for simulating the impulse noise is that the realistic mean time between artifacts (MTBA) is typically in the order of thousands of seconds. To avoid inefficiency in the simulations, we use the following pragmatic solution. We truncate the video sequence into segments of 2 seconds and generate one error burst in each segment. As introduced in Section 5.1, we model the error-burst duration as exponentially distributed. We measure the performance under the mean durations between 1 ms and 16 ms. For channel coding, we use an encoding block size of 24. The retransmission deadline is 200 ms, and the round-trip time (RTT) is assumed to be 40 ms.

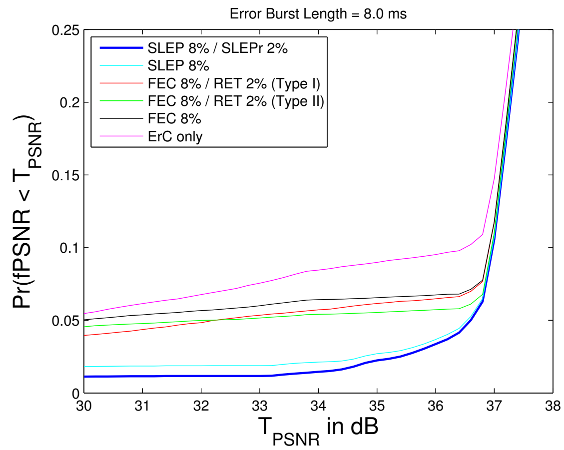


(a)

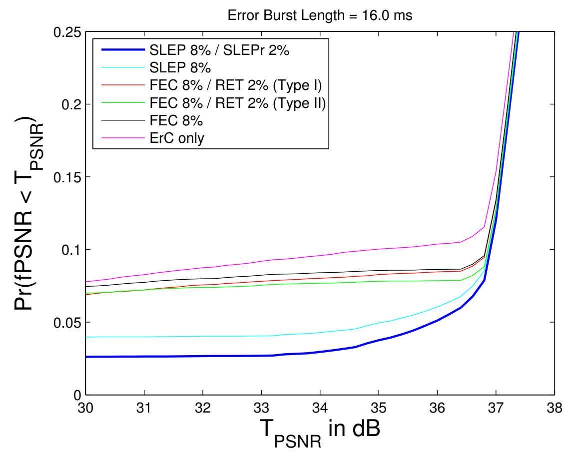


(b)

Fig. 6. CDF of fPSNR for FEC/SLEPr and FEC/RET, (a) $\alpha_{\mathrm{r}} = 2\%$, (b) $\alpha_{\mathrm{r}} = 6\%$.



(a)



(b)

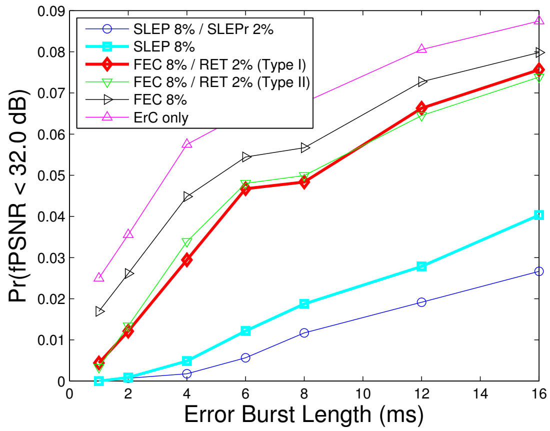
Fig. 7. CDF of fPSNR for SLEP/SLEPr and other schemes, (a) average burst length $=$ 8 ms, (b) average burst length $=$ 16 ms.

6.2. SLEPr vs. RET

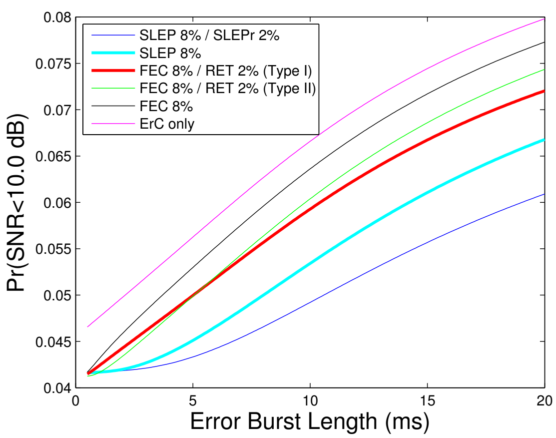
The main objective of this experiment is to measure how much gain SLEPr alone can achieve. Fig. 6 (a) shows the CDF of FEC/SLEPr and FEC/RET under 2% retransmission budget for the average burst lengths of 2, 8 and 16 ms. It is observed that in all three cases, FEC/SLEPr has a lower CDF curve than FEC/RET, implying that SLEPr incurs less visual distortion than RET. Fig. 6 (b) shows the case under 6% retransmission budget. For error-burst durations of 8 and 16 ms, the curves follow similar trends as the previous case. For error-burst duration of 2 ms, however, FEC/SLEPr has an inferior performance compared to FEC/RET. This is expected, since SLEP achieves robustness against errors at the expense of some slight quality degradation; on the other hand, when FEC/RET has enough repair budget to handle error bursts with a duration of 2 ms, it does not incur any video quality degradation.

6.3. SLEP/SLEPr vs. Others

We evaluate the combination of SLEP and SLEPr and compare it to FEC/RET. For completeness, we also plot the cases of error concealment only (ErC), FEC with 8% budget, and SLEP with 8% budget. In Fig. 7, we plot the CDF under (a) average burst length of 8 ms, and (b) average burst length of 16 ms. The observation is that SLEP/SLEPr generates lowest CDF among all schemes, implying that it has the lowest potential of creating impulsive quality degradations. In Fig. 8 (a), we fix the target fPSNR to be 32 dB and measure the probability of fPSNR being below the target (*i.e.*, measuring the probability of corrupted frames), against the average error burst length. It is expected that as the error burst length gets larger, it is more likely that the error burst cannot be corrected, hence, the probability of corrupted frames will go up. For all the measured burst lengths, SLEP/SLEPr has the best performance. This experimental result is consistent

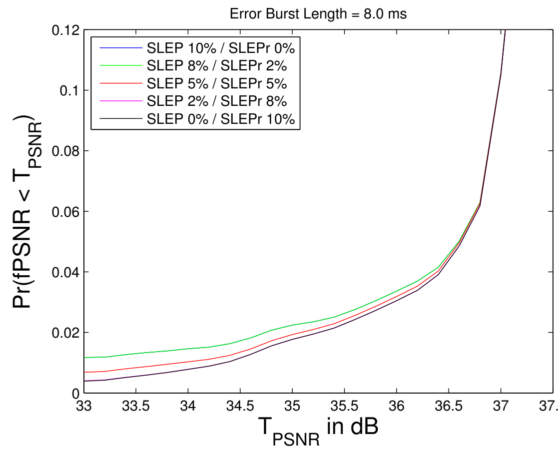


(a)

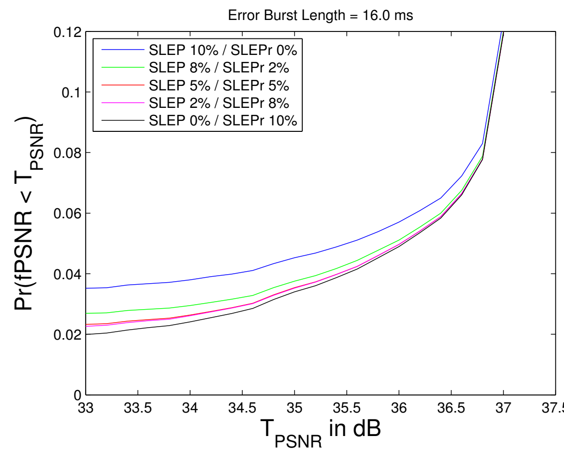


(b)

Fig. 8. Probability of corrupted frames for SLEP/SLEPr and other schemes, (a) experimental, (b) analytical.



(a)



(b)

Fig. 9. CDF of fPSNR for SLEP/SLEPr of various forward/retransmission ratios, (a) average burst length $=$ 8 ms, (b) average burst length $=$ 16 ms.

with the analytical result obtained using the model developed in Section 5, as plotted in Fig. 8 (b).

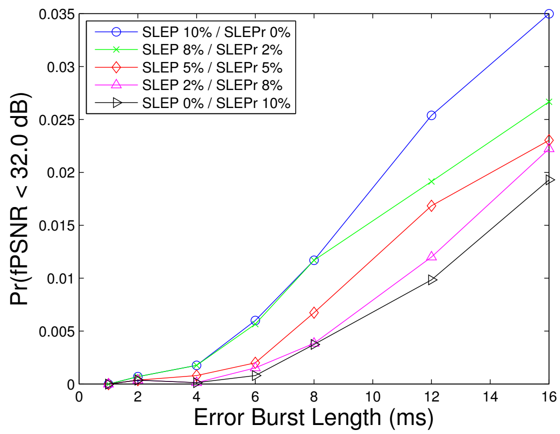
6.4. Varying the Ratio between SLEP and SLEPr

In this experiment, we are interested in understanding what could be the optimal resource allocation between SLEP and SLEPr for a given fixed loss-repair budget. We measure various SLEP/SLEPr resource allocation schemes, ranging from allocating all resources to SLEP to allocating all resources to SLEPr. Fig. 9 shows the CDF of these schemes under average burst lengths of 8 ms and 16 ms. Fig. 10 (a) shows the plot of the probability of corrupted frames versus average burst length. Fig. 10 (b) is the analytical results based on the model developed in Section 5. All the experimental and analytical results suggest that, for end-to-end transmission, allocating the loss-repair budget *entirely* to SLEPr maximizes the received video quality.

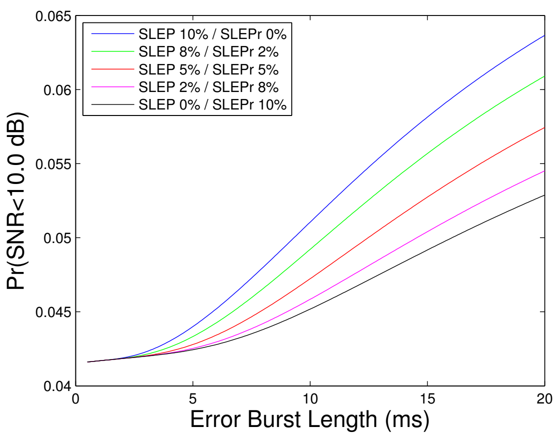
We provide an intuition for this observation. Forward error protection is a proactive mechanism that allocates resources to protect against errors regardless of whether errors actually happen or not. This mechanism is especially inefficient for bursty errors, which tend to happen in clusters, but with a rare occurrence. On the contrary, retransmission protects against errors only when errors actually happen. Therefore, given fixed loss repair budget, forward error protection evenly distributes the resource whereas retransmission can allocate resources to provide stronger protection for error bursts of longer length. From an information theory perspective, this is also true since retransmission is a capacity-achieving mechanism for erasure channels [9].

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have shown through analysis and experimental results that SLEP/SLEPr is an efficient solution



(a)



(b)

Fig. 10. Probability of corrupted frames for SLEP/SLEPr of various foward/retransmission ratios, (a) experimental, (b) analytical.

to provide resistance against impulse noise in IPTV systems. This proposed solution is backward-compatible with the current IPTV network infrastructure and therefore easy to deploy.

One interesting point raised in our analysis and experiments is that from an end-to-end video delivery perspective, allocating the loss-repair resources to SLEPr entirely maximizes the video quality. From a system perspective, however, it might create performance bottleneck at the server side in a multicast scenario, since the server may easily get overwhelmed by the retransmission requests from the multiple clients. One possible solution to this problem is to partially shift the repair function from the central server to the clients. This approach could not only help with the scalability issues, but could also improve the delivered video quality. Our future work is to study from a system level, the feasibility of such a distributed approach.

8. ACKNOWLEDGMENTS

The authors would like to thank Shantanu Rane and Pierpaolo Baccichet for sharing the SLEP source code, and Dave Oran for helpful discussions.

References

[1] S. Rane, P. Baccichet, and B. Girod, “Systematic lossy error protection of video signals,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 10, pp. 1347–1360, Oct. 2008.

[2] W. Henkel, T. Kessler, and H. Chung, “Coded 64-CAP ADSL in an impulse-noise environment – modeling of impulse noise and first simulation results,” *IEEE J. Select. Areas Commun.*, vol. 13, no. 9, pp. 1611–1621, Dec. 1995.

[3] A. C. Begen, “Error control for IPTV over xDSL networks,” in *IEEE Consumer Communications and Networking Conf. (CCNC)*, 2008.

[4] I. 13818-1, “Information technology: Generic coding of moving pictures and associated audio information: Systems,” Dec. 2000.

[5] RFC 3550, “RTP: A transport protocol for real-time applications.” [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt

[6] Internet Draft: draft-ietf-fecframe-1d2d-parity-scheme-00, “RTP payload format for non-interleaved and interleaved parity FEC.” [Online]. Available: http://www.tools.ietf.org/id/draft-ietf-fecframe-1d2d-parity-scheme

[7] S. Shamai, S. Verdu, and R. Zamir, “Systematic lossy source/channel coding,” *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 564–578, Mar. 1998.

[8] S. Rane, D.R.-Monedero, and B. Girod, “High-rate analysis of systematic lossy error protection of a predictively encoded source,” in *Data Compression Conf. (DCC)*, 2007.

[9] T. M. Cover and J. A. Thomas, *Elements of information theory*. Wiley, New York, 1991.