

IPTV multicast with peer-assisted lossy error control

Zhi Li^a, Xiaoqing Zhu^b, Ali C. Begen^b and Bernd Girod^a

^aDept. of Electrical Engineering, Stanford University, Stanford, CA 94305, USA;

^bCisco Systems Inc., 170 West Tasman Drive, San Jose, CA 94134, USA

ABSTRACT

Emerging IPTV technology uses source-specific IP multicast to deliver television programs to end-users. To provide reliable IPTV services over the error-prone DSL access networks, a combination of multicast forward error correction (FEC) and unicast retransmissions is employed to mitigate the impulse noises in DSL links. In existing systems, the retransmission function is provided by the Retransmission Servers sitting at the edge of the core network. In this work, we propose an alternative distributed solution where the burden of packet loss repair is partially shifted to the peer IP set-top boxes. Through Peer-Assisted Repair (PAR) protocol, we demonstrate how the packet repairs can be delivered in a timely, reliable and decentralized manner using the combination of server-peer coordination and redundancy of repairs. We also show that this distributed protocol can be seamlessly integrated with an application-layer source-aware error protection mechanism called forward and retransmitted Systematic Lossy Error Protection (SLEP/SLEPr). Simulations show that this joint PAR-SLEP/SLEPr framework not only effectively mitigates the bottleneck experienced by the Retransmission Servers, thus greatly enhancing the scalability of the system, but also efficiently improves the resistance to the impulse noise.

Keywords: IPTV, error-resilient video, impulse noise, peer-to-peer, reliable multicast, scalability

1. INTRODUCTION

Internet Protocol TeleVision (IPTV) services aim to multicast to the end-users QoS-guaranteed television programs over a carefully-engineered infrastructure. Digital subscriber lines (DSL) are often chosen as the low-cost last-mile access technology, over which IPTV shares the link with other services, such as Web browsing and Voice over IP (VoIP). DSL links, however, are susceptible to various types of interference, among which the impulse noise is the most common.

Riding on reliable multicast protocols, current IPTV deployments integrate source-specific multicast (SSM) with an error-control mechanism, which combines multicast forward error correction (FEC) and unicast retransmissions, to mitigate the impact of impulse noise on delivered video quality.³ When experiencing packet losses, the IP set-top boxes (STBs) first attempt to recover using received FEC packets; upon failure they initiate retransmission requests to the Retransmission Servers, located at the edge of the network. In this framework, each Retransmission Server can only support a limited number of downstream STBs lest they are overwhelmed by retransmission requests. Furthermore, the recoverable bursty loss duration is limited by the bitrate allocated for retransmissions.

In this work, we first propose a paradigm shift from the current error-control mechanism for IPTV multicast. The main idea is to leverage many STBs in the loss-repair process, so as to lessen the burden on the Retransmission Servers. In the *network/transport layer*, we propose the *Peer-Assisted Repair (PAR)* protocol, which coordinates the server and the peers to deliver repair packets in a timely, reliable and decentralized way. We then investigate the performance of PAR in combination with an *application-layer* error-correction technique called *forward and retransmitted Systematic Lossy Error Protection (SLEP/SLEPr)* and show that this further mitigates the server burden and increases the recoverable bursty loss duration.

This paper is structured as follows. Section 2 describes the various packet loss-repair protocols, including the currently deployed server-assisted repair, and the proposed PAR protocol. Section 3 describes how to combine the PAR protocol with the video coding scheme SLEP/SLEPr. Simulation results are presented in Section 4.

Send correspondence to Z.L.: E-mail: leeoz@stanford.edu, Telephone: +1 650 868 8155. Preliminary results were published in Packet Video Workshop (PV) '09 and ACM Multimedia (MM) '09.^{1,2}

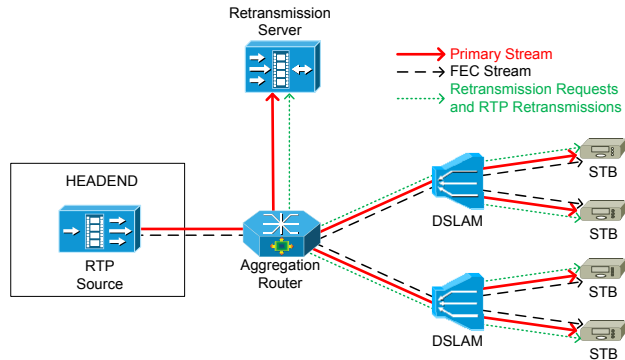


Figure 1. IPTV access network architecture with hybrid error control. The source video stream and the FEC stream are generated at the headend, and delivered to IP STBs via an SSM session. The Retransmission Server caches the source packets and responds to STBs’ requests by retransmitting the source packets. Abbreviations: STB – set-top box, DSLAM – DSL Access Multiplexer.

2. PACKET LOSS-REPAIR PROTOCOLS

In this section, we describe the protocols used in the network/transport layer to achieve reliable multicast of IPTV streams over error-prone DSL access networks. We use $\{\cdot\}$ to denote a set and its cardinality (or size) is denoted by $|\{\cdot\}|$. A network node is denoted by P ; specifically, P_{RS} denotes the Retransmission Server, and P_n , $1 \leq n \leq N$ denote the STBs. S_i denotes a source packet and C_i denotes a generated parity packet. We start by introducing the system architecture of DSL access network in Section 2.1. Section 2.2 describes the currently deployed hybrid error-control mechanism through dedicated Retransmission Servers. Section 2.3 presents the proposed peer-assisted packet loss-repair protocol.

2.1 System Architecture

We focus on the IPTV access network, as shown in Figure 1. The IPTV stream is sourced from the streamers in a headend and transmitted over the aggregation router(s) and a DSL access multiplexer (DSLAM). It is then delivered to the IP STBs over a DSL link (*i.e.*, telephone wire). To overcome the impulse noise and other types of noise within the DSL links, dedicated Retransmission Servers are deployed to provide repair services to the STBs. To minimize the burden on the network, Retransmission Servers are pushed to edge of the network, usually co-located with the aggregation routers.

The IPTV primary stream is carried in a MPEG2 Transport Stream (MPEG2-TS)⁴ and encapsulated in Real-time Transport Protocol (RTP)⁵ packets. Each IP STB may join one of many multicast groups, each of which carries one video program channel. Each group shares a primary stream, which are delivered to the STBs over an SSM session.

2.2 Server-Assisted Repair (SAR)

The Retransmission Server-assisted hybrid error-control mechanism is shown in Figure 1, and its abstraction model is illustrated in Figure 2(a). The primary video stream and FEC stream are multicast from the RTP source to the IP STBs. The primary stream is cached by the Retransmission Server P_{RS} . When interference or noise on the DSL links corrupts the stream, the IP STB first attempts to recover the corrupted source packets from the FEC packets. If this fails, it sends a request to P_{RS} , which then retransmits the requested packets over a unicast session. As packet retransmissions are solely handled by the Retransmission Server, we name this approach *Server-Assisted Repair (SAR)*.

2.3 Peer-Assisted Repair (PAR)

In PAR, the repair function is partially shifted to the peer IP STBs. The Retransmission Server only serves as the last resort when the peer-assisted repair is not available. The PAR protocol is designed with *low latency* in mind, that is, unlike TCP or Scalable Reliable Multicast (SRM),⁶ the system cannot afford to let the peer

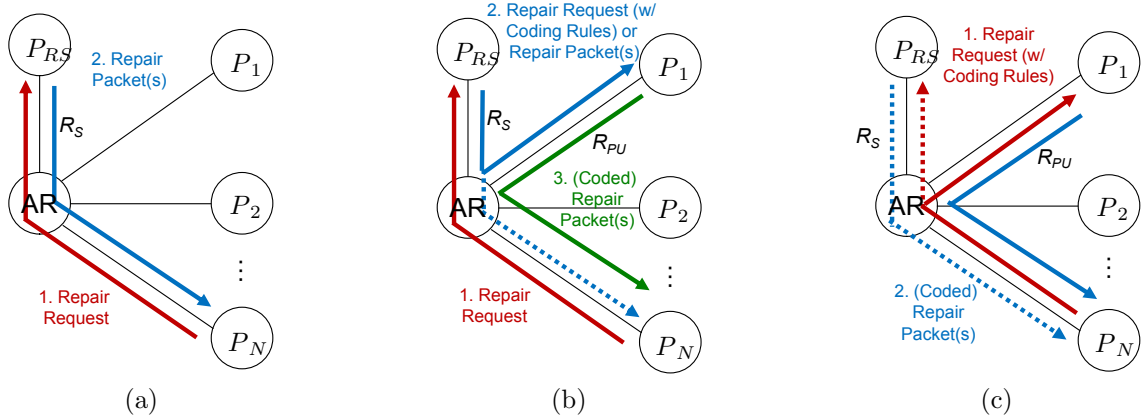


Figure 2. Packet loss-repair protocols illustrated in simplified network topology. (a) Server-Assisted Repair (SAR), (b) Peer-Assisted Repair with Centralized Tracking (PAR-CT) and (c) Peer-Assisted Repair with Distributed Tracking (PAR-DT). Only the retransmissions are depicted. Abbreviation: AR – aggregation router.

continue sending requests until the repair is received; instead, the system must ensure that the repair information is received *in one retransmission with high probability*. In addition, the communication between the peers does not rely on exchanging state information (*e.g.*, the current multicast session the peer is in) between the peers. Instead, the Retransmission Server promptly pushes the state information to the peers.

One challenging issue faced by PAR (but not by SAR) is the peer departure (or peer churn) that could lead to repair failures. For example, when a peer changes the program (*i.e.*, leaves a multicast session), it is deemed available for packet repair until the state update is pushed through the Retransmission Server and the other peers. Besides peer departure, other uncertainties could also lead to repair failures. For example, due to imperfect synchronization, the requested packet may fall outside a peer’s cache window. In this work, we propose to address all these problems with a unified solution – *redundancy of repairs*. In Section 2.3.1, we discuss how to introduce redundancy in the repair packets sent from different peers. In Section 2.3.2 and 2.3.3 respectively, we discuss two variations of the PAR protocol – PAR with Centralized Tracking (PAR-CT) and PAR with Distributed Tracking (PAR-DT). Figures 2(b) and (c) illustrate these two variations.

2.3.1 Redundancy of Repairs

Uncoded Case When a peer P_n experiences an error burst \mathcal{B} with size $B \triangleq |\mathcal{B}|$, that causes loss of packets $\{S_i\}_{i \in \mathcal{B}}$ which cannot be corrected by FEC, the simplest solution is to request redundant repairs for each individual lost packet S_i , thereby increasing the probability that P_n receives *at least one* retransmitted packet for each S_i . Exactly L copies are requested for each S_i from P_n ’s neighboring peers. When not enough neighboring peers are available, retransmission can always be performed by the Retransmission Server P_{RS} , where redundant repairs are not needed since the Retransmission Server is considered reliable. Note that the redundancy degree L should not be too large, otherwise the redundant packets will cause congestion in the downlink of P_n . The uncoded case can be considered as a special coded case of a simple $(L, 1)$ repetition code.

Coded Case A more efficient solution is to apply erasure codes (*e.g.*, Reed-Solomon codes) across peers to recover a burst of packet losses, as illustrated in Figure 3. Specifically, for a burst of size B , a (BL, B) code shall be applied. Notice that unlike the uncoded case, the redundancy degree L now can be any fractional number ≥ 1 . In total BL retransmission requests are generated and sent to the neighboring peers. Each retransmission request also includes a coding rule CR_l , in the following form:

$$CR_l = \tilde{g}(l) \triangleq [g(l, 1), g(l, 2), \dots, g(l, B)], \quad l = 1, \dots, BL \quad (1)$$

where $\tilde{g}(l)$ corresponds to a row in the Reed-Solomon code generator matrix $G \in \text{GF}(2)^{BL \times B}$, and has support $\{S_i\}_{i \in \mathcal{B}}$. The co-program peer P_l (*i.e.*, peer that is watching the same program as P_n), after receiving the request

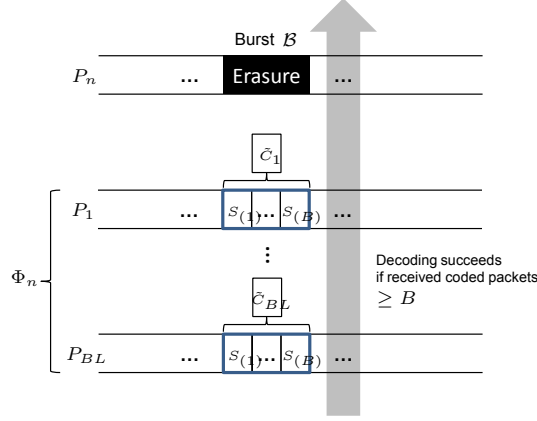


Figure 3. Generation and retransmission of coded redundant repair packets from peers in the co-program set Φ_n to the requesting peer P_n . Note that in this illustration we assume $|\Phi_n| = BL$.

with the coding rule CR_l , generates the coded packet \tilde{C}_l according to

$$\tilde{C}_l = \sum_{i=1}^B g(l, i) S_{(i)} \quad (2)$$

where $\{S_{(i)}\}_{i=1}^B \triangleq \{S_i\}_{i \in \mathcal{B}}^*$ and retransmits the coded packet \tilde{C}_l to the destination peer P_n . P_n performs decoding after successfully receiving B coded packets. Denote by $\tilde{l} \in \{1, \dots, B\}$ the index of the received packets out of BL coded packets. The decoding procedure is to solve $[S_{(1)} \dots S_{(i)} \dots S_{(B)}]^T$ from

$$\begin{bmatrix} \tilde{g}(1) \\ \dots \\ \tilde{g}(\tilde{l}) \\ \dots \\ \tilde{g}(B) \end{bmatrix} \begin{bmatrix} S_{(1)} \\ \dots \\ S_{(i)} \\ \dots \\ S_{(B)} \end{bmatrix} = \begin{bmatrix} \tilde{C}_1 \\ \dots \\ \tilde{C}_{\tilde{l}} \\ \dots \\ \tilde{C}_B \end{bmatrix}. \quad (3)$$

2.3.2 PAR-CT

In PAR-CT, a peer's status is continuously tracked by the Retransmission Server. When a peer P_n joins a multicast session M_k , $k \in 1, \dots, K$ [we denote this by $M_k = M(P_n)$, where $M(\cdot)$ is the operator that maps a peer to its current multicast session], the Retransmission Server P_{RS} is promptly informed of this event via a special message.⁷ During the session, P_n periodically sends messages to P_{RS} to confirm that it continues to receive the multicast. If no such message is received for some time, P_{RS} no longer considers P_n as a receiver of that multicast session. When P_n leaves the multicast session, it promptly informs P_{RS} of this event. P_{RS} also maintains a table that keeps track of the peer-program pairs $\{(P_n, M(P_n))\}_{n=1}^N$. Thus, it knows any peer P_n 's co-program set $\Phi_n \triangleq \{P_j : j \neq n, M(P_j) = M(P_n)\}$, *i.e.*, the set of peers currently watching the same channel as P_n .

When P_n experiences an error burst \mathcal{B} that causes loss of packets $\{S_i\}_{i \in \mathcal{B}}$ which cannot be corrected by FEC, it sends a repair request $REQ(\{S_i\}_{i \in \mathcal{B}})$ to P_{RS} . P_{RS} determines L – the number of redundant repairs per packet needed to maintain a high probability of recovery. The total number of repair packets is then BL . P_{RS} then spreads the BL requests as evenly as possible to P_n 's co-program set Φ_n . If coding needs to be applied across the repair packets, P_{RS} also specifies the coding rules CR_l , $l = 1, \dots, BL$ in each request packet (See Section 2.3.1). If not enough peers are found, *i.e.*, $|\Phi_n| < L$, P_{RS} replies to P_n with the requested packets $\{S_i\}_{i \in \mathcal{B}}$ itself. Notice

⁷Note that we use the subscript (i) for the indexing of packets within \mathcal{B} , which is different from the index i used for the set of all packets.

that by forwarding the request instead of sending the repair packets, Retransmission Server bitrate can be saved. Upon receiving the retransmission request, each requested peer P_l looks for the requested packets in its cache. If found, it responds to P_n with the requested packets. If necessary, coding is applied to generate the coded repair packets.

2.3.3 PAR-DT

Similar to PAR-CT, in PAR-DT a peer's status is always promptly updated. However, now each peer P_n maintains its own tracking table that contains the co-program set Φ_n . Whenever there is a status change, P_{RS} is informed first, which, in turn, sends a message to update the tracking table of P_n if needed.

When P_n experiences the loss of packets $\{S_i\}_{i \in \mathcal{B}}$, it first determines the repair redundancy L needed. P_n then spreads the BL requests as evenly as possible to the co-program set Φ_n . It also specifies the coding rules CR_l , $l = 1, \dots, BL$ if coding shall be applied. If P_n does not find a sufficient number of peers receiving the same multicast, *i.e.*, $|\Phi_n| < L$, it sends the repair request $REQ(\{S_i\}_{i \in \mathcal{B}})$ to P_{RS} . Either P_{RS} or the peers in Φ_n respond with the repair packets, similar to PAR-CT.

PAR-CT and PAR-DT differ from each other in (i) where the tracking table(s) are maintained, and (ii) who takes the initiative to decide where to look for repairs. Compared to PAR-CT, PAR-DT demands less server bitrate and reduces queuing delay at the server, but it is more difficult for PAR-DT to achieve tracking table synchronization.

3. CROSS-LAYER DESIGN WITH SLEP/SLEPr

SLEP is an application-layer forward error protection scheme for robust transmission of video over packet erasure channels.⁸ The scheme is *systematic* in the sense that the protection stream is separable from the source stream, and *lossy* in that robustness is achieved at the expense of some slight loss of video quality. 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 has been proposed.⁸ In our earlier work, we also extended the scheme to a hybrid of forward/retransmission repairs, termed as SLEP/SLEPr.¹

In this section, we describe how to combine PAR with the source-aware error protection scheme SLEP/SLEPr to effectively deal with *impulse noise* and *peer departures* within a unified framework. SLEP/SLEPr, when used alone, provides a backward-compatible solution to the current IPTV infrastructure. Compared to the source-oblivious hybrid error-control method described in Section 2.1, SLEP/SLEPr effectively increases the correctable error burst duration with some slight degradation of the received video quality. In Section 3.1, we first illustrate the SLEP/SLEPr procedure as an extension of SLEP in the hybrid error-control scenario. In Section 3.2, we show how to combine the application-layer SLEP/SLEPr with network/transport-layer PAR.

3.1 SLEP/SLEPr Procedure

Consider SLEP is applied to a hybrid error-control scenario. First, divide the source stream into blocks of \bar{K} packets, and denote by \mathcal{D} one of the blocks. Generate for each block $\bar{N} - \bar{K}$ parity packets $\{\hat{C}_{(i)}\}_{i=1}^{\bar{N}-\bar{K}}$. Among the parity packets, let $\{\hat{C}_i^{FWD}\}$ be the set of parity packets used in forward error protection, with size $\bar{M}_{FWD} \triangleq |\{\hat{C}_i^{FWD}\}|$; let $\{\hat{C}_i^{RET}\}$ be the set of parity packets used in retransmission, with size $\bar{M}_{RET} \triangleq |\{\hat{C}_i^{RET}\}|$, satisfying $\bar{N} - \bar{K} = \bar{M}_{FWD} + \bar{M}_{RET}$. In practice, the forward parity packets are generated at the source encoding stage; depending on the repair schemes, the retransmitted parity packet may be either generated at the source encoding stage, or generated at the Retransmission Server (as in SAR) or the peers (as in PAR) upon request. Figure 4 illustrates the procedure of forward and retransmission of SLEP packets.

- *Forward transmission.* At the sender, the source packets $\{S_i\}$ and the forward parity packets $\{\hat{C}_i^{FWD}\}$ are transmitted to the receiver over the error-prone erasure channel. Recall that \mathcal{B} denotes an erasure.

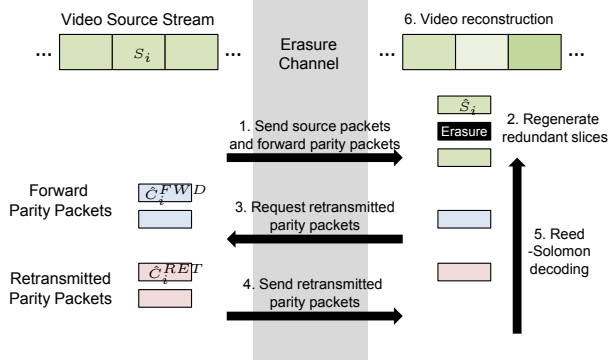


Figure 4. Packet-level illustration of forward and retransmission of SLEP packets.

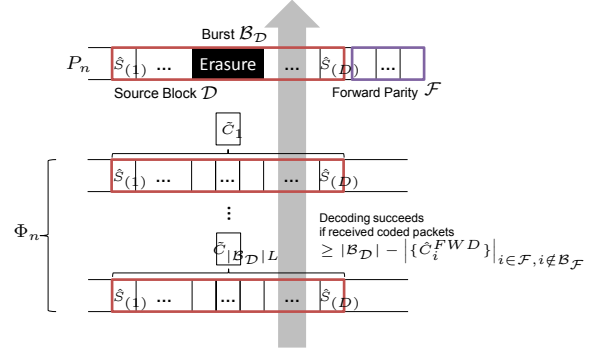


Figure 5. Generation and retransmission of coded redundant repair packets from peers in the co-program set Φ_n to the requesting peer P_n – joint PAR-SLEP/SLEPr error protection case. Note that in this illustration we assume $|\Phi_n| = |\mathcal{B}_D|L$ and $\mathcal{B}_F = \emptyset$.

- *Forward error correction.* At the receiver side, use the received source packets $\{S_i\}_{i \in \mathcal{D}, i \notin \mathcal{B}}$ to regenerate the redundant packets $\{\hat{S}_i\}_{i \in \mathcal{D}, i \notin \mathcal{B}}$. Within coding block \mathcal{D} , if the total number of redundant packets regenerated and the parity packets received is larger than \bar{K} , *i.e.*, $|\{\hat{S}_i\}_{i \in \mathcal{D}, i \notin \mathcal{B}}| + |\{\hat{C}_i^{FWD}\}_{i \in \mathcal{D}, i \notin \mathcal{B}}| \geq \bar{K}$, apply Reed-Solomon decoding to recover the corrupted packets. Otherwise, the receiver sends a retransmission request for additional parity packets from $\{\hat{C}_i^{RET}\}$.
- *Retransmission.* Upon receipt of the additional parity packets, apply Reed-Solomon decoding if enough packets are received, *i.e.*, $|\{\hat{S}_i\}_{i \in \mathcal{D}, i \notin \mathcal{B}}| + |\{\hat{C}_i^{FWD}\}_{i \in \mathcal{D}, i \notin \mathcal{B}}| + |\{\hat{C}_i^{RET}\}_{i \in \mathcal{D}, i \notin \mathcal{B}}| \geq \bar{K}$. Then decode the redundant slices and splice back to the motion-compensated primary video stream. If not enough packets are received, then the corresponding source packets are considered missing in the video decoding process, in which case error concealment could be applied to mitigate the distortion.

3.2 Joint PAR-SLEP/SLEPr Error Protection

We have seen that SLEP/SLEPr operates in the application layer whereas PAR operates in the network/transport layer. Notice that Reed-Solomon codes are used in both layers, but for different purposes – for resisting the impulse noise in the application layer and for mitigating peer departures in the transport layer. This introduces inefficiency. If we can unify SLEP/SLEPr and PAR into one single framework, then Reed-Solomon coding is performed only once and the redundancy can be utilized more efficiently. This can be achieved by modifying the PAR protocol and make it aware of the encoding in SLEP/SLEPr. Specifically, we modify the coded case described in Section 2.3.1 as follows.

The repair packet generation and retransmission procedure is illustrated in Figure 5. Either P_{RS} (in PAR-CT) or P_n (in PAR-DT) must be aware of the coding blocks used in SLEP/SLEPr. Let \mathcal{D} be one source coding block with size $D \triangleq |\mathcal{D}|$ and \mathcal{F} be the forward parity block corresponding to \mathcal{D} . Let \mathcal{B}_D be the error burst within \mathcal{D} and \mathcal{B}_F be the error burst within \mathcal{F} . Either P_{RS} or P_n computes the redundancy L needed and specifies the coding rules

$$CR_l = \tilde{g}(l) \triangleq [g(l, 1), g(l, 2), \dots, g(l, D)], \quad l = 1, \dots, |\mathcal{B}_D|L \quad (4)$$

Parameter	Default Value	Range	Parameter	Default Value	Range
Number of peers N	100	10~1000	Packet loss rate	0.4%	0.2%~0.8%
Number of programs K	200		Req. pkt size B_{CTL}	64 Bytes	
Zipf distr. shape s	1		Source pkt size B_{SRC}	1375 Bytes	
Source stream bitrate R	5 Mb/sec		Repair redundancy L	1	1 ~3
Video duration	10 sec		Stream compression ratio β	1	0.5 ~ 1
FEC delay D_{FWD}	50 ms		Peer departure rate γ_{DEPT}	0	0 ~0.25
Retrans. delay D_{RET}	0.5 sec	0.1 ~0.5 sec	Peer uplink bitrate m_{PU}	1	0.1 ~3
Peer-router delay	20 ms		Peer downlink bitrate m_{PD}	1	0.1 ~3
Router-retrans. server delay	0 ms		Number of peer neighbors n_{NB}	10	
FEC budget α_{FWD}	0%	0 ~5%	Source innovation std σ_W	5	
Retrans. budget α_{RET}	10%	5 ~10%	Source corr. coef. ρ	0.75	
Mean burst length T_{BURST}	8 ms	2~16 ms	Quantization step size δ	1.2	
Mean time btw burst T_{MTBB}	2 sec		Quantization ratio m	1	1 ~ 4

Table 1. Simulation parameters.

where $\tilde{g}(l)$ corresponds to a row in the Reed-Solomon code generator matrix $G \in \text{GF}(2)^{|\mathcal{B}_D|L \times D}$. Notice that each coding rule is now specified with the support $\{\hat{S}_i\}_{i \in \mathcal{D}}$, *i.e.*, all the redundant packets within the coding block, instead of only for the corrupted source packets $\{S_i\}_{i \in \mathcal{B}}$ as in Section 2.3.1. Each peer P_l , after receiving the request with the coding rule CR_l , first regenerates the redundant packets $\{\hat{S}_i\}_{i \in \mathcal{D}}$ from source packets $\{S_i\}_{i \in \mathcal{D}}$, then generates the coded packets \tilde{C}_l according to

$$\tilde{C}_l = \sum_{i=1}^D g(l, i) \hat{S}_i \quad (5)$$

where $\{\hat{S}_i\}_{i=1}^D \triangleq \{\hat{S}_i\}_{i \in \mathcal{D}}$ and retransmits the coded packet \tilde{C}_l to the destination peer P_n . After successfully receiving $|\mathcal{B}_D| - \left| \left\{ \hat{C}_i^{FWD} \right\}_{i \in \mathcal{F}, i \notin \mathcal{B}_F} \right|$ coded packets, together with the regenerated redundant packets and the received forward parity packets, P_n performs Reed-Solomon decoding.

4. SIMULATION RESULTS

4.1 Simulation Setup

We have implemented a proof-of-concept simulation program of the peer-assisted loss-repair system for the IPTV video multicast. PAR is implemented in the Network Simulator (ns-2); SLEP/SLEPr is implemented based on H.264/AVC reference software JM version 13. Table 1 summarizes the parameters used in the simulations.

To allow simulations of up to 1000 peers, we have made some simplifications. Only the retransmitted packets are simulated in ns-2. We make the assumption that packet losses are uncorrelated at different links, but correlated for one link.[†] The channel error model is assumed to be on-off exponential. By picking proper values for mean burst duration T_{BURST} and mean time between any two bursts T_{MTBB} , we simulate an environment with packet loss rate (PLR) in the order of $1e-3$.[‡]

To evaluate the received video quality, we use the SOCCER sequence, which has a resolution 704×576 . The sequence is encoded at 30 frames per second with an H.264/AVC JM (version 13.2) encoder. The frame structure is IPPP and the GoP size is 20. We also use the default motion-compensated error-concealment method in JM13.2. The quantization parameter (QP) for the SLEP/SLEPr redundant slice generation is chosen such that the stream compression ratio β equals 0.5.

[†]This assumption is based on the fact that the access network is single-hop and most of the interference occurs on the last-mile DSL links. There could be correlated losses as well (e.g., a lightning strike), but their occurrence is rare.

[‡]A realistic PLR on a DSL line is at the level of $1e-5$, but this is a rather unrealistic condition for simulations, since it would take too long to generate meaningful statistics. In this work, we choose a PLR of $1e-3$ in the simulations instead.

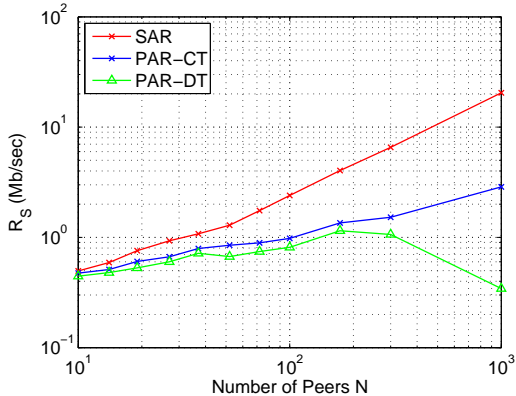


Figure 6. Retransmission Server bitrate R_S , for a post-repair PLR of $3e-4$, as a function of the number of supported peers N , for (1) SAR, (2) PAR-CT and (3) PAR-DT.

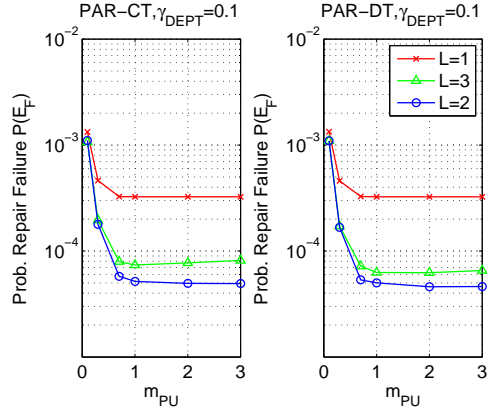


Figure 7. Probability of repair failure event $\mathbf{P}(E_F)$ as a function of the normalized peer uplink bitrate m_{PU} , for (a) PAR-CT and (b) PAR-DT with request redundancy $L = 1, 2, 3$, uncoded case. The results are measured at $\gamma_{DEPT} = 0.1$.

4.2 Retransmission Server Bitrate

We first investigate the bitrate needed at the Retransmission Server to support loss repairs in different schemes. For this experiment, we fixed the target PLR and find the bitrate needed to achieve this PLR. Figure 6 plots the Retransmission Server bitrate obtained in simulations. We can observe that the bitrate of SAR grows almost linearly with N . The bitrate of PAR-CT consists of two portions – control information (*i.e.*, request packets) and repair packets. For PAR-DT, only the repair packets count towards the bitrate. The repair packet portion of the bitrate exhibits an interesting behavior – as N increases, it grows until it reaches a maximum and then it drops. This could be explained as follows. As N grows, the number of multicast groups increases, contributing to the overall bitrate. The number of peers within each group also increases, thus the availability of peers that possess the requested repair packets increases, alleviating the demand on the Retransmission Server. The expected bitrate reaches its maximum when each multicast group has at least one peer.

4.3 Peer Uplink Bitrate

Compared to SAR, PAR-CT and PAR-DT require additional uplink bitrate from each peer. We use the normalized peer uplink bitrate m_{PU} as the measure, defined as the ratio of peer uplink bitrate and the bitrate needed for retransmissions for a single peer. Figure 7 plots the repair failure probability versus the peer uplink bitrate m_{PU} . We observe that no matter how much redundancy is introduced, it is sufficient to have the peer uplink bandwidth matched to the downlink bandwidth reserved for repair packets. The reason is that in DSL link, packet losses tend to be bursty but sparse. Given that the losses are uncorrelated for different links, the chance that a peer is burdened by simultaneous repair requests from more than one peer is small.

4.4 Peer Downlink Bitrate

Similarly, we use the normalized peer downlink bitrate m_{PD} as the measure, defined as the ratio of peer downlink bitrate and the bitrate needed for retransmissions for a single peer. Figure 8 plots the repair failure probability versus the peer downlink bitrate in the uncoded case. From the plot, we observe that if we keep the downlink bitrate low (*e.g.*, $m_{PD} = 1$), repair redundancy does not necessarily reduce the repair failure probability. In other words, the peer downlink bandwidth needs to increase in accordance with repair redundancy, so as to accommodate more packet arrivals and avoid congestion. Therefore, using uncoded repairs with high redundancy is not an effective repair strategy. We will show in Section 4.5 that coded repairs with moderate redundancy (*e.g.*, $L = 3/2$) is more effective in reducing the repair failure probability, even if we keep $m_{PD} = 1$.

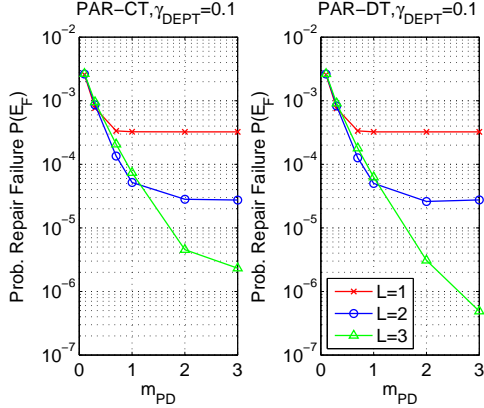


Figure 8. Probability of repair failure event $\mathbf{P}(E_F)$ as a function of the normalized peer downlink bitrate m_{PD} , for (a) PAR-CT and (b) PAR-DT with redundancy $L = 1, 2, 3$, uncoded case. The results are measured at $\gamma_{DEPT} = 0.1$.

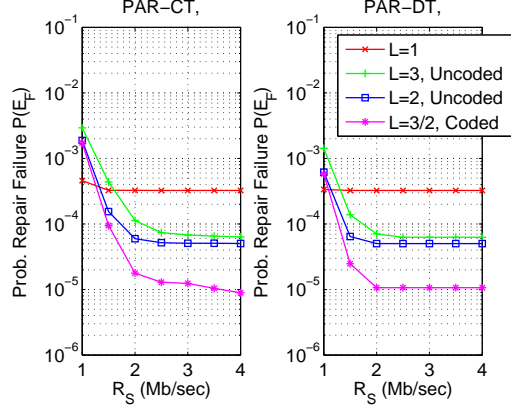


Figure 9. Probability of repair failure event $\mathbf{P}(E_F)$ as a function of Retransmission Server bitrate R_S for (1) PAR-CT and (2) PAR-DT at different redundancy and uncoded/coded cases. The results are measured at $\gamma_{DEPT} = 0.1$.

4.5 Redundancy of Repairs

Figure 9 shows the repair failure probability as a function of the Retransmission Server bitrate R_S , for various degrees of redundancy. Increasing the uncoded redundancy of repair packets is an effective tool to combat the peer departures, but is at the expense of potentially jamming of the communication links. This is evidenced by the ineffectiveness of increasing the redundancy from 2 to 3. This confirms the observation we made in Section 4.4. If we instead use a moderate redundancy (*e.g.*, $L = 3/2$) but apply Reed-Solomon codes across the redundant packets, as shown in Figure 9, we can avoid the peer downlink congestion while we further reducing the PLR in the presence of peer departures. Applying coding across packets is a more effective solution for mitigating the impact of peer departures.

4.6 Joint PAR-SLEP/SLEPr Error Protection

In this section, we evaluate the performance of combining PAR and SLEP/SLEPr, termed as the jointly coded scheme. Since SLEP/SLEPr would cause some slight video quality degradation even if the corrupted packet is successfully repaired, for fairness, we evaluate different systems based on the received video quality.

Figure 10 shows the empirical CDF of frame PSNR (fPSNR) for PAR-CT and PAR-DT in the uncoded, separately coded and jointly coded cases at $T_{BURST} = 8$ ms. As expected, the gain of applying coding across packets over the uncoded case can be reflected on the CDFs of reconstructed video quality. The Jointly coded case achieves even better performance than the separately coded case as it combines all the redundancy in network/transport/application layer in correcting the packet erasures.

We also evaluate the repair failure probability for the three schemes in Figure 11, in an scenario with stringent server rate. We compare the video quality for three schemes with different combinations of redundancy L and stream compression ratio β : (i) $\beta = 0.5$, $L = 2$, (ii) $\beta = 1$, $L = 1$ and (iii) $\beta = 1$, $L = 2$. It can be seen that both (ii) and (iii) incurs worse video quality. The case of (ii) fails because it does not provide redundancy against peer departures, whereas the case of (iii) fails due to server link congestion. In comparison, the case of (i) demonstrates robustness performance as it avoids congestion by transmitting packets of reduced sizes.

5. CONCLUSIONS

We have discussed a cross-layer framework for achieving error resilience in IPTV multicast. The Peer-Assisted Repair (PAR) protocol draws context from conventional reliable multicast, but with the support of specific network infrastructure, it is designed with simultaneously meeting reliability, low latency and scalability in mind. PAR can be seamlessly integrated with the application-layer source-aware error protection scheme SLEP/SLEPr, which is an efficient solution to provide resistance to impulse noise at a slight sacrifice of video quality. The joint

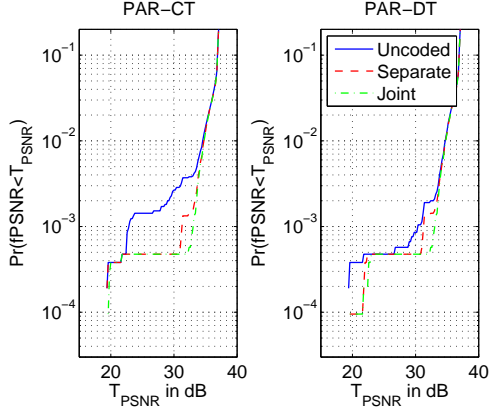


Figure 10. CDF of frame PSNR (fPSNR) over 100 peers and 100 frames at peer departure rate $\gamma_{DEPT} = 0.1$ for (a) PAR-CT and (b) PAR-DT for (i) uncoded case, (ii) SLEP/SLEPr separately coded case and (iii) SLEP/SLEPr jointly coded case. The FEC budget and retransmission budget is set at 5% and 5%, respectively. T_{BURST} is set to 8 ms.

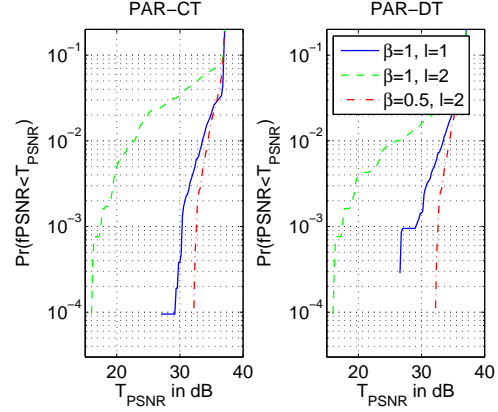


Figure 11. CDF of frame PSNR (fPSNR) over 100 peers and 100 frames at peer departure rate $\gamma_{DEPT} = 0.15$ for (a) PAR-CT and (b) PAR-DT for different combinations of redundancy L and stream compression ratio β : (i) $\beta = 0.5$, $L = 2$, (ii) $\beta = 1$, $L = 1$ and (iii) $\beta = 1$, $L = 2$. R_S is set at 1 Mb/sec, which is stringent enough to cause server link congestion for (iii). The repair bitrate is set to be 10% of the primary video stream bitrate.

PAR-SLEP/SLEPr framework is efficient in providing error protection to channel errors and resilient to peer departures.

One interesting question arisen in the system design is to determine the optimal resource allocation between forward error protection and retransmissions. Our previous results have indicated that from an end-to-end video delivery point of view, given the limited overall bandwidth, allocating the whole resources to retransmission always leads to better received video quality. However, in practice it is desirable to have some level of forward error protection for reducing Retransmission Server burden. Furthermore, the proposed PAR protocol is most effective when the downstream packet losses show an uncorrelated pattern in the spatial domain. For correlated losses (in the spatial domain) caused by events such as a lightning strike, *multicast repair* appears to be a more effective solution. How to determine the optimal switching point between forward error protection, unicast retransmissions and multicast retransmissions are interesting and important future research directions.

REFERENCES

1. Z. Li, X. Zhu, A. Begen, and B. Girod, "Forward and retransmitted Systematic Lossy Error Protection for IPTV video multicast," in *Proc. Packet Video Workshop 2009*, Seattle, WA, May 2009.
2. Z. Li, X. Zhu, A. Begen, and B. Girod, "Peer-assisted packet loss repair for IPTV video multicast," in *Proc. ACM Multimedia*, Beijing, China, Oct. 2009.
3. A. Begen, "Error control for IPTV over xDSL networks," in *Proc. IEEE Consumer Communications and Networking Conference, 2008. (CCNC 2008)*, Las Vegas, NV, Jan. 2008.
4. "ISO/IEC 13818-1, Information technology: Generic coding of moving pictures and associated audio information: Systems," Dec. 2000.
5. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RFC 3550: RTP: A transport protocol for real-time applications," Available online: <http://www.ietf.org/rfc/rfc3550.txt>.
6. S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing," *IEEE/ACM Transactions on Networking* **5**, pp. 784–803, Dec. 1997.
7. B. VerSteeg, A. Begen, T. VanCaenegem, and Z. Vax, "Internet draft: Unicast-based rapid acquisition of multicast RTP sessions," Available online: <http://tools.ietf.org/html/draft-ietf-avt-rapid-acquisition-for-rtp>.
8. S. Rane, P. Baccichet, and B. Girod, "Systematic lossy error protection of video signals," *IEEE Transactions on Circuits and Systems for Video Technology* **18**, pp. 1347 – 1360, Oct. 2008.