Performance comparison of centralized versus distributed error recovery for reliable multicast

Martin S. Lacher, Jörg Nonnenmacher, Ernst W. Biersack

To appear in ToN, April 2000

Abstract—**We examine the impact of the loss recovery mechanism on the performance of a reliable multicast protocol. Approaches for loss recovery in reliable multicast can be divided into two major classes: centralized (source-based) recovery and distributed recovery. For both classes we consider the state of the art: For centralized recovery, an integrated transport layer scheme using parity multicast for error recovery (hybrid ARQ type 2) as well as timer-based feedback suppression. For distributed recovery, a scheme with local data multicast retransmission and feedback processing in a local neighborhood. We also evaluate the benefits of combining the two approaches into distributed error recovery with local retransmissions using a type 2 hybrid ARQ scheme. The schemes are evaluated for up to 10^6 receivers under different loss scenarios with respect to network bandwidth usage and completion time of a reliable transfer. We show that using distributed error recovery with type 2 hybrid ARQ gives best performance in terms of bandwidth and latency. For networks, where local retransmission is not possible, we show that a centralized protocol based on type 2 hybrid ARQ comes close to the performance of a protocol with local retransmissions.**

Keywords—**Reliable Multicast Protocol, Error Control, ARQ, FEC, Performance Evaluation.**

I. Introduction

DATA dissemination applications such as software updates, distribution of movies or newspapers require reliable data transfer from one sender to many receivers. The requirements for reliable multicast communications vary widely, depending on the application and network scenarios. A large number of protocols providing reliable multicast services for different applications have been presented and can be expected to co-exist in the future. The approaches differ, among others, by the various error control mechanisms used. Several taxonomies were presented to classify the different multicast protocols (see [1], [2], [3], [4], [5]). With respect to participation of group members in multicast *error recovery*, protocols can be classified as:

- **Centralized error recovery (CER)** allows retransmissions exclusively to be performed by the multicast source, referred to also as **source-based recovery**.
- **Distributed error recovery (DER)** allows retransmissions potentially to be performed by all multicast members. The burden of recovery is decentralized over the whole group.

Distributed error recovery can further be sub-classified (see Figure 1). If neighboring nodes in the multicast routing tree are organized as **DER groups**, within which retransmissions are performed locally, we refer to **grouped DER**. The absence of local groups is referred to as **ungrouped DER**, where retransmissions can be performed by *any node in the tree* to the *global* multicast group.

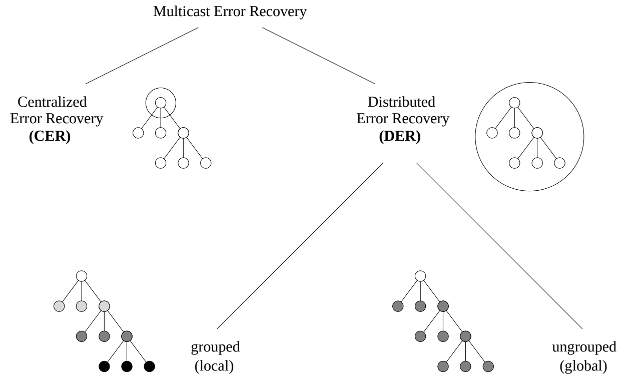
Multicast Error Recovery
Centralized Error Recovery (CER)
Distributed Error Recovery (DER)
grouped (local)
ungrouped (global)

Fig. 1. Classification of multicast error recovery techniques

Existing protocols and classifications can be mapped to our classification scheme in agreement with what their authors classified them as. Further, there are no conflicts with other existing classifications ([3], [4]). RMTP [6] is based on a hierarchical structure with local groups, each with a designated receiver that performs retransmissions. RMTP is a grouped DER protocol. SRM [7] allows retransmissions potentially by all nodes and proposes extensions for local recovery. Hence, SRM is an ungrouped DER protocol in our classification. In the case of the extension, SRM is a grouped DER protocol. In NP [8] only the multicast source can perform retransmissions, so NP can be classified as CER. MESH [9] is a DER protocol that incorporates both local and global recovery. SHARQFEC [10] can be classified as grouped DER protocol.

Error recovery mechanisms either retransmit original data or transmit parity data for loss recovery. We refer to the latter as **hybrid ARQ**. In [8] two types of hybrid ARQ are introduced: *layered FEC*, where parity transmission is performed in an extra layer below the transport layer, and *integrated FEC*, where parity transmission is integrated into the transport layer. We refer to layered FEC as **hybrid ARQ type 1** and to *integrated FEC* as **hybrid ARQ type 2**.

It is shown in [8] that transmission of parity has excellent scaling properties for large receiver groups. Parity transmission leads to a significant reduction of the number of total transmissions compared to retransmission of original data.

At the sender, h *parity* packets are coded, for example with a Reed Solomon code [11], from a group of k *original* data packets forming a **transmission group (TG)**.The reception of *any* k out of those $k + h$ packets is sufficient to reconstruct the k originals. A parity packet can repair the loss of *any* original packet. When multicast to several receivers, a single parity packet can repair the losses of several distinct original packets at different

Martin S. Lacher is with the Computer Science Department at Technische Universität München, 80290 München, Germany (e-mail: lacher@in.tum.de)

Jörg Nonnenmacher is with the Networking Research Department at Bell Labs, Murray Hill, NJ 07974, USA (e-mail: nonnen@research.bell-labs.com)

Ernst W. Biersack is with the Corporate Communications Department at Institut Eurecom, 06904 Sophia Antipolis, France (e-mail: erbi@eurecom.fr)

receivers.

Several comparisons between generic protocols of the DER class and the CER class exist. In [3], it is shown that DER protocols are superior to CER protocols concerning throughput, when both protocols use original packet retransmission. In [12] a grouped DER and a modified ungrouped DER protocol are compared and better performance is obtained for the grouped DER protocol. In [10] the SHARQFEC protocol (grouped DER) is compared to a CER protocol (both featuring parity transmission recovery). It is shown that for a fixed network topology with 113 receivers DER has superior bandwidth performance compared to CER. Latency issues as well as the influence of network and transmission mode parameters on all results are not considered in [10]. In [13] CER and DER protocols with optimizations have been compared regarding buffer size requirements and bandwidth. The results presented there underline our results. However, latency is not considered.

CER protocols are attractive since they are easier to deploy than DER protocols and require less functionality from the receivers and the network (no multicast retransmission capability). The findings about hybrid ARQ type 2 [8] in the context of multicast make us reconsider CER protocols. In the following we will compare a CER protocol based on hybrid ARQ type 2 to a grouped DER protocol with respect to bandwidth consumption and completion time for a reliable transfer. We also investigate how parity transmission for error recovery improves the performance of a grouped DER protocol.

The paper is organized as follows: Section II presents our network model for the comparison. Section III describes the protocols. Section IV gives the bandwidth consumption analysis. Section V compares the respective bandwidth performance results of the protocols considering various loss scenarios. Section VI gives the latency behavior analysis for the protocols. Section VII compares the protocols' latency performance results with respect to various loss scenarios. Section VIII presents a summary of the results and our conclusions.

II. Model

We are looking at 1:R communication. The multicast routing tree is created by some multicast routing algorithm. We consider loss due to buffer overflows in network nodes of the tree. In our simplified tree structure, one logical link represents several physical hops. We assign loss properties to each of the logical links in the tree model and refer to this as link loss. The spatial loss correlation among receivers that leads to several receivers losing the same packet, is given by the topology of the tree model shown in Figure 2. The first tree level consists of one logical link, the **source link** (1 physical hop), connecting the multicast source to a backbone router. Loss on the source link is experienced by all receivers (**shared loss**). At the second tree level, we have G **backbone links** (w_b physical hops each) leading to G DER nodes.[1] The DER nodes are connected by Z **receiver links** (1 physical hop each) to the receivers that are located at the leaves of the tree. Therefore the tree connects $R = G \cdot Z$ receivers to the source. The tree is similar to the one in [12], which is based on loss measurements for Internet multicast [14]. It was shown there that loss occurs mainly on the source link and on the receiver links while backbone loss is negligible. We can model such a loss pattern by assigning no loss to backbone links. The tree and the loss models we propose can also accommodate findings from [15], which will be explained along with the results in Section V and Section VII.

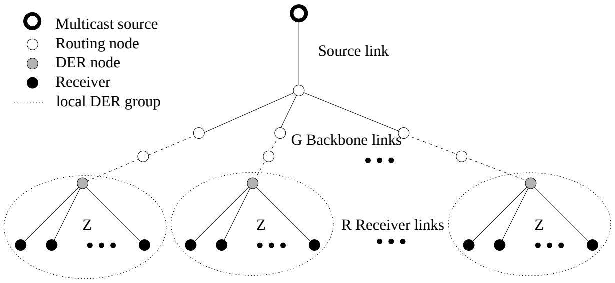

Fig. 2. Tree model.

Figure 2 shows the tree model for DER, where Z receivers connected to the same backbone link belong to one local **DER group**. Each DER group constitutes a separate multicast group and the **DER node** at the end of a backbone link can perform retransmissions to the local DER group. The CER topology is the same, with the single difference that only one multicast group exists that connects all receivers to the source. Local groups do not exist and DER nodes are just internal nodes that only perform routing of multicast packets. To show the influence of loss patterns, we will examine different loss scenarios:

- **homogeneous independent loss** with packet loss probability p only on the receiver links.
- **heterogeneous independent loss** only on the receiver links. We examine two sub-scenarios: **intra-group heterogeneous loss**, where in each of the DER groups a fraction f_h of the Z receivers experiences high loss with probability p_h, the rest low loss with probability p. With **inter-group heterogeneous loss**, DER groups consist exclusively either of high loss receivers or of low loss receivers. There is a fraction f_h of high loss groups among the G DER groups. Due to the higher average loss that receivers see, the heterogeneous scenario cannot be compared quantitatively to the other scenarios. However, the scenario is sufficient for the derivation of qualitative results for the influence of heterogeneous loss.
- **shared source link loss** with a homogeneous loss probability p' on the source link and all receiver links. The loss probability p' is calculated such that the loss probability that each receiver sees is p. This allows for the analysis of the influence of shared loss compared to homogeneous independent loss.
- **burst loss** influence is briefly discussed according to results from [8].

Our loss scenarios take into account the measurements of Internet multicast loss done by Handley [15]. It was found that a fraction of the receivers experiences relatively high loss (heterogeneous independent loss scenario). Two main overlaying loss patterns were found: a receiver correlated high loss, caused by a small number of shared lossy links (shared source link loss scenario) and an independent loss noise pattern (homogeneous independent loss scenario). In [15] it was also shown that, ex-

[1] By assigning a variable number of hops to the backbone links, we can later examine the influence of larger backbone parts on the performance.

cept for periodic high loss patterns, due to assumed routing instabilities in parts of the network, burst loss patterns are largely insignificant.

We assume a constant total Round Trip Time (RTT) from the source to any receiver. Further we assume a constant and equal propagation delay for one physical hop, independent of the position in the tree. Considering the variable number w_b of hops on the backbone links, we get for each physical hop a delay of $d = \frac{RTT}{2 \cdot (2+w_b)}$.

III. Protocols

Both, for CER and DER, we examine generic protocols in each class with characteristics that have been shown to give, up to this day, the highest performance for this class. We define one protocol featuring hybrid ARQ type 2 and one featuring plain ARQ within both, the CER and the DER class.

All protocols feature the following characteristics:

- Receiver-based loss detection with negative acknowledgments (NACK), realized by ordered transmission with gap detection.
- (Re-)transmissions of original or parity packets are multicast globally (CER), or within the respective DER group.
- One Application Data Unit (ADU), consisting of N packets, is partitioned into **transmission groups (TGs)** of k packets each for transmission. The size N of an ADU can become very large in our protocols. k is announced to the receivers in header information in data packets.
- Accumulated feedback (NACK) is sent by the receivers for each TG. Feedback and poll for feedback packets are assumed to consume neither bandwidth nor transmission time and to be transmitted without loss.
- Transmissions and retransmissions of different TGs can be interleaved in time. Interleaving improves the protocol throughput, since the source can use the time waiting for feedback to transmit new packets.

The CER protocols feature a timer-based feedback suppression algorithm like the one introduced in [7] and improved in [16]. The DER nodes perform hierarchical feedback filtering in the multicast tree.

For hybrid ARQ type 2, parity packets are coded on demand. We used the encoder introduced in [17] and [18] and found that coding delays are negligible (for the measurement result see Section VII). The coder uses Reed-Solomon codes and the coding algorithm has a complexity of $O(k^2)$. With the recently introduced Tornado codes [19], the coding complexity is reduced to $O(k)$. This renders coding delay irrelevant.

A. Protocol C (CER with hybrid ARQ type 2)

Protocol C is a CER protocol featuring hybrid ARQ type 2. The source multicasts the data packets to all receivers. Accumulated feedback for a TG is sent by a receiver on reception of a poll for feedback. Due to the ability of a parity packet to recover any packet of a TG lost by an arbitrary receiver, only the maximum number of lost packets at any receiver is required as feedback. If a receiver needs to send feedback at all, is decided by the feedback suppression algorithm with exponential timers [16] (explained in more detail below). The transmission of a TG of k packets is done the following way:

The multicast source (For each TG):
1. Sends the k original packets of the TG; a poll for feedback is piggybacked with the last transmitted packet of the TG to indicate the end of the TG.
2. If the feedback from the receivers indicates that less than k packets are received by any receiver, a_{max} new parity packets are generated and transmitted. a_{max} is the maximum number of additional parity packets needed at any receiver to reconstruct the k original packets. Again, a poll for feedback is piggybacked.
3. Step 2 is repeated until no feedback about missing packets is received anymore within a certain timeout interval.

The receiver (For each TG):
1. Original and parity packets of a TG are buffered.
2. If k or more packets of a TG, be it parity or data packets, have been received, the k original packets are decoded and sent to a higher layer.
3. If less than k packets have been received and a poll for feedback for the TG is received, the receiver calculates the number of additional parity packets required. If the feedback suppression algorithm decides that the receiver sends feedback, the receiver will globally multicast its NACK indicating the number of missing packets.
4. Step 3 is repeated until at least k packets of the respective TG have been received.

The feedback suppression algorithm presented in [16] decides whether or not a receiver has to send feedback in the following way: When a receiver joins the multicast group, the one-way delay OTT between sender and receiver is estimated. Upon reception of a request for feedback, each receiver schedules a timer in an interval that depends on the estimated OTT between sender and receiver. Feedback is sent in two cases: if the timer expires before reception of a feedback message of another receiver, or a received feedback message from another receiver indicates a lower number of lost packets than at the current receiver. The feedback message stating the number of additionally required packets is multicast to the entire group. Additional recovery latency is introduced through timer-based feedback suppression (see Section VI). The parameters of the feedback suppression mechanism [16] are chosen such that the expected number of feedback messages arriving at the source is in the worst case equal to the number of receivers in a DER group in our tree model (see Section VI).

B. Protocol C_{noFEC} (CER with ARQ)

We define a generic CER protocol with the same characteristics as protocol C, but instead of parity retransmission, C_{noFEC} uses data retransmission. C_{noFEC} is introduced to show that CER protocols profit more from hybrid ARQ type 2 than do DER protocols. We will not provide any further analysis for this protocol, since it can easily be derived from the analysis of C and D1.

C. Protocol D1 (grouped DER with ARQ)

We define D1 as grouped DER protocol with *data retransmission (ARQ)*. The source is a group leader for all the internal DER nodes in the tree model (Figure 2). The DER nodes in turn are group leaders for the receivers at the leaves. Protocol D1 works in a store-and-forward manner. The first transmission is done by the source to all nodes in the multicast tree. Thereafter, error recovery only for the DER nodes, which constitute an extra multicast group, is performed. A DER node does not perform error recovery for its local DER group, which constitutes another multicast group, until it has received all packets of a TG from the source. Error recovery for the receivers is performed in parallel for the different DER groups. Feedback messages and retransmissions are restricted to the respective groups, such that the maximum number of feedback messages to be processed by any group leader is equal to the number of group members. We assume a hierarchical feedback filtering mechanism in the upper tree levels to reduce the number of feedback messages, while introducing negligible delay. The transmission of a TG of k packets is done the following way:

The multicast source (For each TG):

1. Sends the k original packets of the TG and a poll for feedback to all nodes in the tree (global multicast).
2. On the reception of feedback (NACKs) from the DER nodes, the corresponding packets are retransmitted from the source to the DER nodes (scope restricted multicast), again with a poll for feedback.
3. Step 2 is repeated until no missing packets are indicated anymore by the DER nodes.

The DER node/ the receiver (For each TG):

1. Original packets of a TG are buffered.
2. On the detection of a loss and reception of a poll for feedback, a NACK is sent (unicast to the source or the DER node respectively), indicating the sequence numbers of the missing data packets.
3. Step 2 is repeated until the TG is fully received. In the case of a DER node, loss recovery for the receivers is now performed.

D. Protocol D2 (grouped DER with hybrid ARQ type 2)

D2 is identical to D1 except for the fact that parities are used for error recovery, the same way as for protocol C. In most cases, parity packets received by the DER nodes from the source are sufficient for error recovery at the receivers as well. If additional parity packets have to be coded, the additional delay is negligible and will thus not be considered. In any case, a DER has to wait until the full TG is received, just as with protocol 1.

IV. Bandwidth Analysis

Table I summarizes the variables and notation that will be used for the bandwidth analysis.

We define the bandwidth B as the bandwidth consumed by a multicast packet[2] per link, averaged over all links in the multicast tree [20]. The bandwidth of a multicast packet in a multicast group i is the product of the number M_i of transmissions per packet (original and retransmissions) and the number H_i of links traversed. Given $H = R + w_b \cdot G + 1$ links in total, where w_b is the number of physical hops in a backbone link as defined in Section II, the **average bandwidth of a multicast packet per link** is:

$$E[B] = \frac{1}{H} \sum_i E[M_i] \cdot H_i \tag{1}$$

TABLE I

Identifiers for Bandwidth and Latency analysis

Id	**Meaning**
B	Average bandwidth consumed by a multicast packet per link
f_h	Fraction of high loss receivers among all receivers
$F_X(x)$	$= P(X \leq x)$ (Cumulative Probability distribution of the random variable X)
G	Number of Backbone links
H	total number of links in the multicast tree
k	Number of packets in a TG
L	number of additional packets (NOAP) required by all receivers
L_r	NOAP required by a random receiver
L_{rh}	NOAP required by a random high loss receiver
M M_C $M_{D1,G}$ $M_{D1,Z}$ $M_{D2,G}$ $M_{D2,Z}$	Number of transmissions per packet (NOTPP) NOTPP for protocol C NOTPP to all DER nodes for D1 NOTPP to Z receivers from a DER node for D1 NOTPP to all DER nodes for D2 NOTPP to Z receivers from a DER node for D2
N	Number of packets in one ADU
p	Packet loss probability
p_h	Packet loss probability for high loss receivers
p'	Packet loss probability in the shared loss case
R	Number of receivers in the multicast group
w_b	Number of physical hops in a backbone link
Z	Number of group members in a DER group

As a base for comparison of performance of the CER and DER protocols in different loss scenarios we consider the **relative bandwidth** requirements $E[B_{D_1}]/E[B_{C_{noFEC}}]$ and $E[B_{D_2}]/E[B_C]$.

A. Protocol C

For the CER protocol C, we have only one multicast group and all transmissions are multicast over all links. Thus we get:

$$E[B] = E[M_C] \tag{2}$$

where $E[M_C]$ is the average number of transmissions per packet required for reliable delivery to all receivers. In the following, a general formula for $E[M_C]$ is derived and then evaluated for the different types of loss. Let L_r describe the **number of additional packet transmissions required by a random receiver** to receive a complete TG, using parity transmission. And let L describe the **number of additional packet transmissions required by all receivers**, to receive the complete TG. We generalize the distributions of L and L_r and the expectations of L and M_C given in [8]. For a fraction f_h of high loss receivers with loss probability p_h and the rest low loss receivers with loss probability p we get:

$$F_{L_r}(l,p) = \sum_{i=0}^{l} \binom{k+i-1}{k-1} p^i (1-p)^k \qquad ,l \geq 0 \tag{3}$$

[2] We assume feedback and poll for feedback packets to consume no bandwidth.

$$\begin{aligned} F_L(l,f_h) &= F_{L_r}(l,p_h)^{R\cdot f_h}\cdot F_{L_r}(l,p)^{R\cdot(1-f_h)} &(4)\\ E[L] &= \sum_{l=0}^{\infty}(1-F_L(l)) &(5)\\ E[M_C] &= 1+E[L]/k &(6)\end{aligned}$$

For homogeneous independent loss we can now simply evaluate equation (4) as $F_L(l,0)$, for both of the heterogeneous cases we evaluate equation (4) as $F_L(l,f_h)$. Afterwards we can calculate the respective $E[M_C]$ with equations (5) and (6).

For shared source link loss in our model multicast tree, no analytical formula for $E[M_C]$ could be derived. Therefore we estimate the value of $E[M_C]$ by simulation.[3] The loss with probability p seen by a receiver is kept constant. Thus, p is equally split to a loss probability p' on the source link and the receiver link:

$$p' = 1-\sqrt{1-p} \qquad (7)$$

B. Protocol D1

The reliable transmission of a packet from the multicast source to the G DER nodes is done via $w_b\cdot G+1$ links and requires $M_{D1,G}$ transmissions per packet. From each DER node, $M_{D1,Z}$ transmissions over Z links are needed to reliably transmit a packet to the receivers of the local group. The bandwidth cost for D1 is given by:

$$E[B]=\frac{1}{H}\left(E[M_{D1,G}]\cdot(1+w_b\cdot G)+E[M_{D1,Z}]\cdot R\right) \quad (8)$$

With the recursive calculation method from [21] we derive a general formula for the **number of transmissions to Z receivers** among which there is a fraction f_h of high loss receivers:

$$F_{M_{D1,Z}}(m,f_h)=(1-p_h^m)^{Z\cdot f_h}\cdot(1-p^m)^{Z\cdot(1-f_h)} \qquad (9)$$

For homogeneous independent loss, we evaluate equation (9) as $F_{M_{D1,Z}}(m,0)$ and use it to calculate $E[M_{D1,Z}]$ as in equation (5). Since the upper links are lossless we get $E[M_{D1,G}]=1$ and can now calculate $E[B]$ as in equation (8).

For heterogeneous independent loss, we first consider each recovery group to consist of a fraction f_h of receivers with high loss p_h and the rest of the receivers experiencing low loss p (intra-group heterogeneous loss). Again, we have $E[M_{D1,G}]=1$ and $E[B]$ can be calculated with equations (9) and (8).

We now also consider the case that a fraction f_h of the groups consists of high loss receivers exclusively and the rest of the groups consists of low loss receivers exclusively (inter-group heterogeneous loss). With $E[M_{D1,G}]=1$ and equation (8) we directly derive:

$$\begin{aligned} E[B] &= 1+\frac{R}{H}\cdot f_h\cdot(E[M_{D1,Z_h}]-1)\\ &\quad +\frac{R}{H}\cdot(1-f_h)\cdot(E[M_{D1,Z}]-1) \qquad (10)\end{aligned}$$

$E[M_{D1,Z}]$ is the expected number of transmissions in a low loss group, $E[M_{D1,Z_h}]$ the same in a high loss group. Both can be calculated by evaluating equation (9) as $F_{M_{D1,Z}}(m,0)$ (low loss) and $F_{M_{D1,Z}}(m,1)$ (high loss) respectively. Both the expectations are calculated as in equation (5).

For shared source link loss, the loss probability p' in equation (7) is the same for the source link and the receiver links. Since the number of transmissions for G DER nodes behind the single lossy source link is the same as for only one DER node behind the lossy source link, we get:

$$F_{M_{D1,G}}(m)=(1-p'^m) \qquad (11)$$

We calculate $E[M_{D1,G}]$ with equations (11), (5) and (8). We get $E[M_{D1,Z}]$ by evaluating equation (9) as $F_{M_{D1,Z}}(m,0)$ while replacing p by p'. We get the result with equations (5) and (8).

C. Protocol D2

For the DER protocol D2, the bandwidth $E[B]$ can be derived from equation (8) by substituting $M_{D2,G}$ for $M_{D1,G}$ and $M_{D2,Z}$ for $M_{D1,Z}$. We can calculate $M_{D2,G}$ and $M_{D2,Z}$ for the case of parity transmission analogously to the calculations for protocol C (see equations (3) to (6)). For the independent loss scenarios we get directly from equation (4):

$$F_L(l,f_h)=F_{L_r}(l,p_h)^{Z\cdot f_h}\cdot F_{L_r}(l,p)^{Z\cdot(1-f_h)} \qquad (12)$$

For all independent loss scenarios is $M_{D2,G}=1$. For homogeneous independent loss we evaluate equation (12) as $F_L(l,0)$, calculate $E[M_{D2,Z}]$ with equations (5) and (6) and the resulting $E[B]$ with equation (8).

For intra-group heterogeneous loss we evaluate equation (12) as $F_L(l,f_h)$ and calculate $E[B]$ with equations (5), (6) and (8).

For inter-group heterogeneous loss we can derive $E[B]$ from equation (10) by substituting $M_{D2,Z}$ for $M_{D1,Z}$ and M_{D2,Z_h} for M_{D1,Z_h}. We calculate $E[M_{D2,Z_h}]$ from equations (6) and (5) by evaluating equation (4) as $F_L(l,0)$ (low loss) and $F_L(l,1)$ (high loss) respectively.

For shared source link loss we can calculate $E[B]$ as in equation (8). We evaluate equation (12) as $F_L(l,0)$ with $Z=1$ and $p=p'$ for the shared link and as $F_L(l,0)$ with $p=p'$ and Z as a variable for the lower tree level. We can then calculate both $E[M_{D2,G}]$ and $E[M_{D2,Z}]$ with equations (5) and (6).

V. Bandwidth Comparison

In the following, the bandwidth requirements of the defined protocols are evaluated. We will see how *parity* transmission can diminish the performance gap between CER and DER. The influence of the parameters k (TG size), p (loss probability), Z (DER group size), as well as the scalability of the protocols with the number of receivers R, is explored in the homogeneous independent loss scenario (loss only on the last hop from the DER nodes to the receivers). The three protocols are then compared regarding their scalability with the number of receivers for the different loss scenarios, using the measure of relative bandwidth requirement. Thereafter the influence of bursty loss patterns is examined. Unless stated otherwise, a packet loss probability of $p=0.01$ [4] used and $R=10^6$ receivers are in the global multi-

[3] All simulations were done with MATLAB. The simulation produces sample values for M_C. $E[M_C]$ is estimated as the average of the sample values for M_C.

[4] We use $p=0.01$ in contrast to MBone measurements in [15]. There, it was shown that a large portion of receivers experiences a median loss rate of $5\% \leq p \leq 10\%$. However, as can be seen in Figure 6, the loss rate does not substantially influence the relative performance of the protocols.

cast group.

A. Homogeneous independent loss

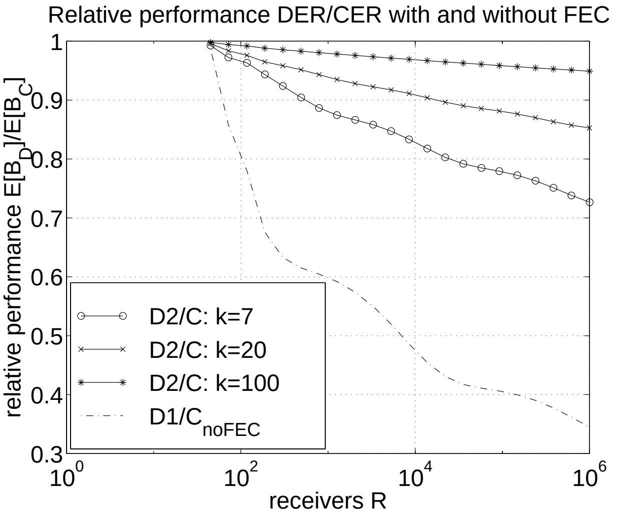

Fig. 3. Relative bandwidth $E[B]_{DER}/E[B]_{CER}$ for independent homogeneous loss with and without parity retransmission, $p = 0.01$, $Z = 30$, $w_b = 1$.

The rationale for our comparison is to demonstrate that CER profits more from parity retransmission than does DER. In [3], grouped DER with plain ARQ was shown to outperform CER with plain ARQ in terms of throughput. The hierarchical structure of grouped DER was identified as the dominating performance factor. The paper argued that any technique employed in a CER protocol could also be employed with a DER protocol and would yield the same relative performance. We see that this is *not the case for hybrid ARQ type 2*. Figure 3 shows the bandwidth requirements of the DER protocols $D1$ and $D2$ relative to the bandwidth requirements of the CER protocols C and C_{noFEC}. It can be seen that the relative performance of CER to DER is improved through hybrid ARQ type 2. This is due to the fact that protocol C serves a larger number of receivers with one parity retransmission than protocol $D2$; each parity packet can repair different losses at different receivers, an effect that is not exploited to the same extent in the DER case, where retransmissions are limited to a local group. Since CER with parity transmission has been shown to outperform CER with data retransmission ([8]) and DER/ARQ has been shown to outperform CER/ARQ in [3], we will not consider protocol C_{noFEC} anymore for our comparison.

In our tree model (Figure 2) we have a variable number w_b of physical hops for the backbone links. In Figure 4, we see that the bandwidth for protocol C is independent of w_b. This is because there is no loss on the additional backbone links and hence the number of transmissions is not increased. Retransmissions for protocols D1 and D2 are only performed in the DER groups over a constant number of links, while the total number of links in the tree increases. Hence, the bandwidth of protocols D1 and D2 decreases slightly (max. 10%). We also see that the influence of w_b on D1 and D2 diminishes with larger transmission group sizes k. The larger the transmission group size with parity transmission is, the smaller is the number of transmissions

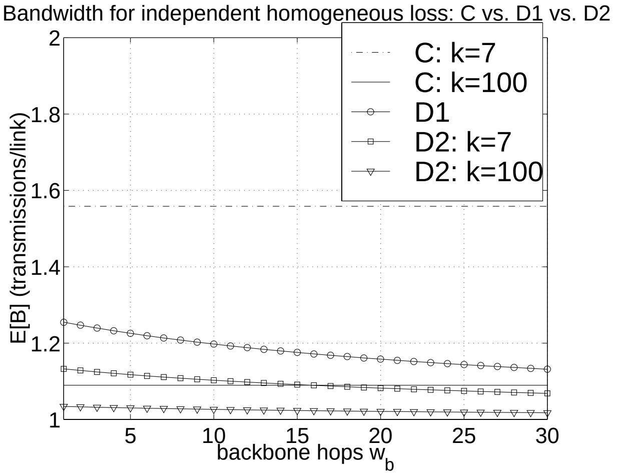

Fig. 4. Bandwidth dependent on number of backbone link hops for independent homogeneous loss: C vs. D1 vs. D2, $p = 0.01$, $R = 10^6$, $Z = 30$

per packet. A larger number of backbone hops only makes a difference for the retransmissions, which are performed locally. Thus, the smaller the number of retransmissions per packet, the smaller the influence of an increased number of backbone hops. We conclude that the number of backbone hops does not have a great influence on the relative performance of the protocols and consider the backbone links to consist of one hop from now on.

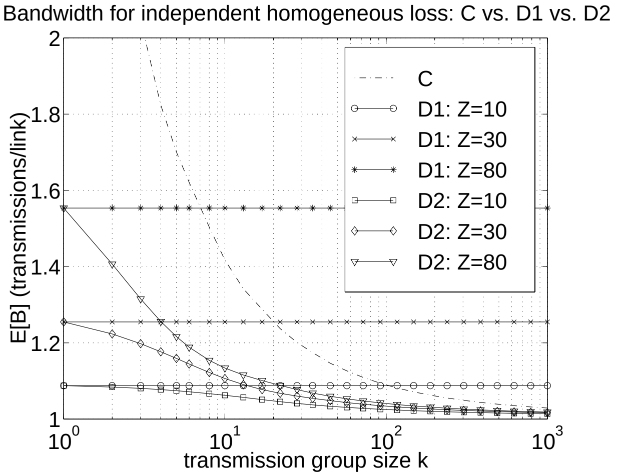

Fig. 5. Bandwidth dependent on TG size k for independent homogeneous loss: C vs. D1 vs. D2, $p = 0.01$, $R = 10^6$, $w_b = 1$.

It can be seen in Figure 5 for different DER group sizes $Z = \{10, 30, 80\}$ that the performance of the protocols C and D2 improves with increasing TG size k. This is due to the fact that a parity packet can repair the loss of *any* packet out of the TG, and that therefore a parity packet can repair the loss of *different packets* at *different receivers* – an effect that increases with the TG size k. Due to the small number of receivers in a DER group, D2 is not as susceptible to differences in k as protocol C (see also Figure 3). For large TG sizes $k \geq 100$ the performance of C comes close to the performance of D2 and is even better than the performance of D1. With the coder introduced in [18], coding complexity for parity packets is $O(k^2)$. The tradeoff between bandwidth and latency is explored in Section VI and Section VII.

Figure 5 shows that the performance of D1 and D2 improves with decreasing Z, since the exposure of retransmissions decreases with decreasing local group size Z. This does not contradict Figure 3: all schemes benefit from decreasing numbers of receivers/group sizes. But, comparing schemes with and without the use of parity transmission, schemes with larger group sizes benefit relatively more from parity transmission than schemes with smaller group sizes.

Protocol D2 performs better than D1 for all transmission group sizes. A result that we experienced for the entire parameter space. Thus, from now on, we will not consider protocol D1 anymore and exclusively compare protocols C and D2.

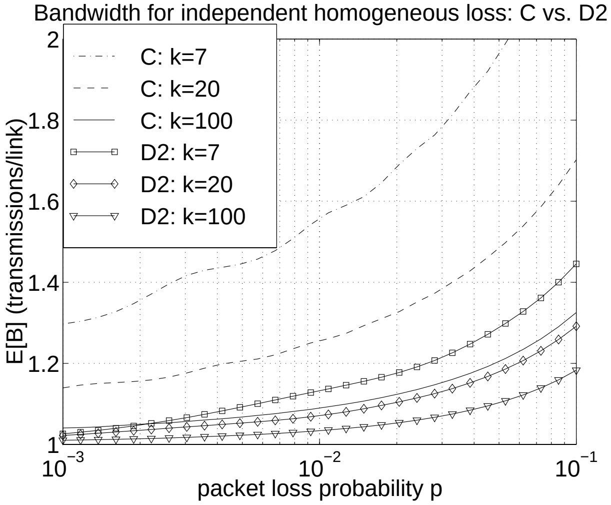

Fig. 6. Bandwidth dependent on packet loss probability p for independent homogeneous loss: C vs. D2, $R = 10^6$, $Z = 30$, $w_b = 1$.

We see in Figure 6 that protocol D2 with a DER group size of $Z = 30$ performs better than C for all loss probabilities p for TG sizes of $k \in \{7, 20, 100\}$. Moreover the bandwidth increase with increasing loss probability p is less for protocol D2 than for protocol C. Due to the scoped retransmissions for protocol D2, a high loss receiver merely increases the bandwidth requirements in one DER group. For C, however, a receiver with high loss dominates all receivers. In [15], measurements in current MBone showed, that loss probabilities for large portions of the receivers are $5\% \leq p \leq 10\%$. Figure 6 also demonstrates that the influence of the loss rate on the relative performance of the protocols is minor. We will use $p = 0.01$ from now on.

As can be seen in Figure 7, the bandwidth requirements of protocol D2 increase quasi-linearly with the local group size Z. Protocol D2 always performs better, or in the worst case, as well as protocol C, for the same transmission group size. If we choose a DER group size $Z = 1$, the performance of D2 is independent of the TG size k, since a retransmission for the single receiver always holds exactly the required packets, whereas for several receivers, unnecessary receptions are possible.

Figure 8 shows that both C and D2 scale very well in terms of bandwidth with large numbers of receivers R. For an increase in receivers of factor 10^6, the bandwidth of protocol C increases only about 50% and the bandwidth for protocol D2 stays constant. Since parity packets are multicast, the receiver with the maximum number of lost packets dominates the whole multicast group. For protocol C, the maximum loss increases with the group size. For protocol D2, the DER group sizes stay constant and so do the maximum loss and hence the bandwidth in each DER group. While protocol D2 performs better than protocol C for the whole range of numbers of receivers R, for large transmission group sizes $k = 100$, the performance difference is very small (see also Figure (5)).

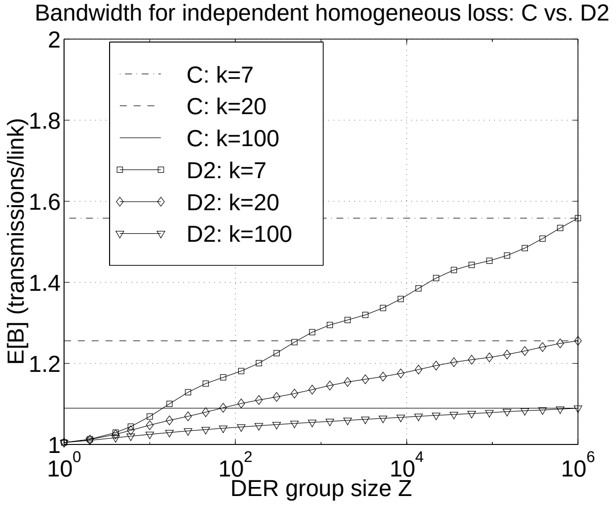

Fig. 7. Bandwidth dependent on DER group size Z for independent homogeneous loss: C vs. D2, $p = 0.01$, $R = 10^6$, $w_b = 1$

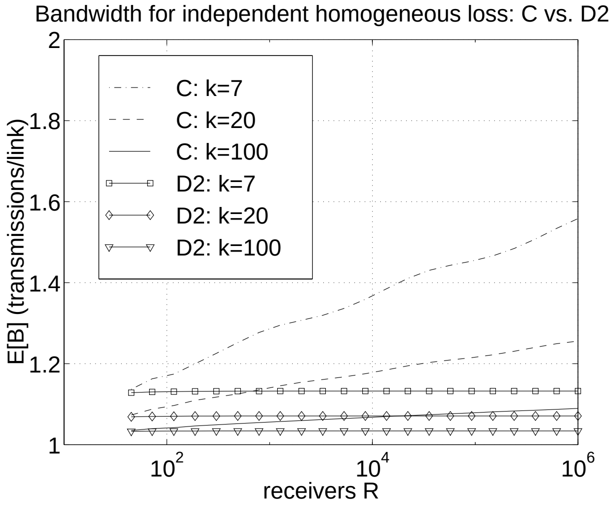

Fig. 8. Bandwidth dependent on number of receivers R for independent homogeneous loss: C vs. D2, $p = 0.01$, $Z = 30$, $w_b = 1$

B. *Other loss scenarios*

We are now going to compare the relative bandwidth performance of C and D2 for large numbers of receivers for the four different loss scenarios (Figure 9). For the homogeneous independent loss scenario, we have $p = 0.01$. For the two heterogeneous independent loss scenarios, for 90% low loss receivers the loss probability stays $p = 0.01$, whereas the 10% high loss receivers see a loss probability of $p_h = 0.25$. As we showed before (Figure 8) protocol C comes close to the performance of protocol D2 for *homogeneous independent loss*.

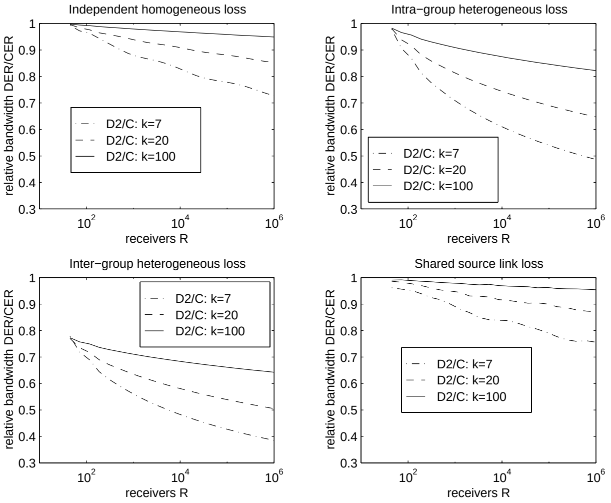

Fig. 9. Relative bandwidth dependent on number of receivers R for different loss scenarios: C vs. D2, $p = 0.01$, $Z = 30$, $w_b = 1$

Looking at *intra-group heterogeneous loss* compared to *homogeneous independent loss*, the performance of protocol D2 improves relative to C. This effect is more pronounced for smaller transmission group sizes. For protocol C, the receiver with the highest loss determines the number of parities transmitted to all receivers. For protocol D2, only one DER group is affected.

Of all loss scenarios, *inter-group heterogeneous loss* gives the highest performance advantage of protocol D2 relative to C. The property of D2 to restrict local retransmissions to the DER recovery group has even more impact than in the intra-group heterogeneous case. We note that DER protocols perform better with heterogeneous network loss characteristics. The assumption of homogeneous independent loss for the exploration of the parameter space benefits CER. However, the qualitative results for the comparison of C and D2 stay valid.

For *shared source link loss* the performance of protocol C improves minimally relative to protocol D2 compared to the homogeneous independent case. Shared loss for a group of R receivers to which parity transmissions are multicast, can be modeled by a smaller group of receivers ($R_{indep} < R, Z_{indep} < Z$) with independent loss ([8]). The reason for this is that only the maximum loss among all receivers determines the required bandwidth. Protocol C multicasts parity packets to R receivers, protocol D2 to Z receivers. Since $Z \ll R$, we also get $R - R_{indep} > Z - Z_{indep}$: the absolute (apparent) reduction in receivers through shared loss is higher for protocol C than for protocol D2. Protocol D2 also requires additional bandwidth on the now lossy source link and the backbone links. This effect is minimal however, since the group of G DER nodes shares all losses and thus behaves like one receiver to the source. We note that with respect to independent homogeneous loss, shared source link loss has negligible influence on the relative performance of C and D2.

C. Burst loss

We will briefly examine the influence of burst loss using the results from [8]. We look at bursty, spatially independent loss patterns as described in [8]. It was found in [8], that for hybrid ARQ type 2 protocols, the number of transmissions per packet is always higher for burst loss than for independent homogeneous loss. Moreover, the increase in transmissions with an increasing number of receivers is higher with burst loss. The difference between numbers of transmissions with burst loss and homogeneous loss is very small for small numbers of receivers and large (up to 60% increase for 10^4 receivers) for large numbers of receivers. The smaller the TG size k , the stronger the influence of burst loss. If the TG is large enough to span several burst loss periods, receivers essentially see homogeneous independent random loss.

Since our bandwidth measure is directly proportional to the number of transmissions per packet, we can directly derive conclusions from the results in [8]. For small TG sizes k, burst loss brings a great disadvantage for protocol C. The bandwidth for C increases up to 60% for 10^4 receivers. The bandwidth for D2 increases by a very small amount and stays independent of R. However, if TG sizes k are large enough, receivers see a random non-bursty loss pattern and protocol C comes very close to the performance of protocol D2.

VI. Latency Analysis

Table II summarizes the notation and random variables that are used in the latency analysis. Other variables can be found in Table I.

Our basic performance measure for latency is the **completion time** $E[D]$ for the reliable transfer of an ADU consisting of one TG with k packets. We define the completion time as the time that is required to fully and successfully transmit the ADU from the sender to *all receivers*. Completion time is the duration, for which resources in the multicast tree will be occupied by the transfer of the ADU. We will define a normalized measure, the **average completion time per packet** expressed in multiples of RTT as to stay independent of the absolute RTT and the total amount of data transmitted. We assume a packet size $P = 2kByte$. In practice, a group of k packets resembles the amount of data that is typically transferred for a HTML-page in WWW.[5] Multicast latency analysis is a highly complex task. So far, no analytical model for latency analysis of multicast parity transmission has been presented and existing approaches solve only parts of the problem ([22]). We will obtain our results for the case of multicast parity transmission by simulation.

For our analysis, we do not follow the chronological order of transmission inherent to the protocol transmission mode. Instead we sum up each delay contribution separately over the whole transmission and add those to the total completion time. We call the process from the beginning of the packet transfer by the sender, until feedback about missing packets is received by the sender one **transmission round**. A reliable transmission requires several transmission rounds. In each round we have account for **transmission delay** (the time it takes to send the packets), the **feedback propagation delay** (the time it takes from the completion of packet emission until feedback is received by the sender), the **feedback suppression delay** (the delay for feedback suppression/processing) and the **coding delay** (both en-

[5] By examining the transmission of a small amount of data we implicitly assume that no interleaving of different TGs is necessary.

TABLE II

NOTATION AND RANDOM VARIABLES FOR LATENCY ANALYSIS

Id	**Meaning**
c_d	Coding constant
d	Propagation delay (PD) for each physical hop
$d_{i,j}$	PD between receiver i and j
d_t	Parameter for feedback suppression algorithm
D	Completion time (CT) for ADU transmission
D_{norm}	Normalized CT per packet in multiples of RTT
D_t	Transmission delay
D_f	Feedback delay
D_{fs}	Feedback suppression delay
D_{fp}	Feedback propagation delay
D_c	Coding delay
D_p	Propagation delay
K	Number of transmission rounds for reliable delivery (NOTR) to all receivers (TAR)
K_C	NOTR TAR for protocol C
$K_{D1,S}$	NOTR to all DER nodes for protocol D1
$K_{D1,I}$	NOTR TAR from all DER nodes for D1
$K_{D2,S}$	NOTR to all DER nodes for protocol D2
$K_{D1,I}$	NOTR TAR from all DER nodes for D2
L_l	Number of parity packets required for decoding of one TG
Λ	Constant end-to-end packet throughput
λ_0	Parameter for feedback suppression algorithm
$M_{D1,I}$	NOTPP TAR from all DER nodes for D1
$M_{D2,I}$	NOTPP TAR from all DER nodes for D2
OTT	One-way latency
P	Packet size
RTT	Round Trip Time

coding and decoding, if parity transmission is used). Those different contributions to the **total completion time** D are denoted by the following random variables:

- The accumulated packet *transmission delay* denoted by D_t: We assume a constant overall end-to-end packet throughput Λ, which is determined by congestion and flow control, queuing, or available bandwidth. Along with a constant packet throughput, we assume the packet size P to be constant. Further we assume the packet throughput to be equal and constant for all paths in the tree model. The latency incurred for the transmission of one packet is then given as $D_t = \frac{1}{\Lambda}$.
- The *feedback delay*, denoted by D_f, accounts for all delays related to feedback. For the DER nodes, we assume the nodes in the upper tree levels (including the DER nodes) to perform effective hierarchical feedback filtering to avoid feedback implosion. Accordingly, we consider the feedback processing delay at the source negligible and look only at the Z feedback messages that must be processed at DER nodes. DER is given a slight benefit through this. The protocol C uses a timer-based feedback suppression mechanism ([16]) to reduce the maximum number of feedback messages arriving at the source down to Z. Now, both for CER and DER, on the way from a receiver to the source, Z feedback messages must be processed (since the DER nodes process in parallel). We can thus neglect feedback processing in both cases. However we must account for the *feedback suppression delay* D_{fs} that is incurred by protocol C through the feedback suppression algorithm. For both CER and DER, the *feedback propagation delay* D_{fp}, which accounts for the additional RTTs that are incurred in each retransmission round, has to be considered.
- The *coding delay* D_c accounts for both encoding and decoding of parity packets. Parity packets are not precoded. Coding can not be performed in parallel to waiting for feedback or transmission of packets, since a complete group of k packets must be available before decoding can begin. We use the coding algorithm introduced in [17] and [18], which has a complexity of $O(k^2)$. Measurements on a SUN SPARC-20 with this coder showed that coding introduces only negligible delay even for large transmission groups. Recent developments showed that coding complexity can be reduced to $O(k)$ ([19]).
- the *propagation delay* for the first transmission of the original packets denoted by D_p.

Equations (13) and (14) show the completion time $E[D]$ and the normalized completion time $E[D_{norm}]$ for one single *short ADU* consisting of one transmission group of k packets.

$$\begin{aligned} E[D] &= E[D_t] + E[D_f] + E[D_c] + D_p & (13) \\ E[D_{norm}] &= \frac{E[D]}{k \cdot RTT} & (14) \end{aligned}$$

We call the intermittent transmission of original packets and retransmissions of different TGs **interleaving**. Interleaving is an efficient way to use waiting times due to network propagation or coding associated with one TG, for the transmission of another TG. For short ADUs, interleaving is not possible. For a *long ADU*, which consists of several transmission groups, **interleaving** is possible. Analysis was done both for short and long ADUs. Evaluating equation (13) for long ADUs results in a performance measure that is largely based on the number of transmissions per packet. Thus we will not consider long ADUs here, since the results would be qualitatively similar to the results from the bandwidth analysis (Section V). In the calculations for the DER protocols for short ADUs, we need to consider that transmission of packets can overlap in time in different DER groups (**horizontal parallelism**).

A. Protocol C

For the transmission delay of one short ADU of size k we get:

$$E[D_t] = E[M_C] \cdot \frac{k}{\Lambda} \qquad (15)$$

where M_C is the number of transmissions per packet in a group of k packets. $E[M_C]$ for homogeneous independent loss as well as for heterogeneous loss is calculated as shown in the bandwidth analysis from equations (3)- (6). $E[M_C]$ is estimated by simulation for shared source link loss.

The feedback delay is given as:

$$E[D_f] = E[D_{fs}] + E[D_{fp}] \qquad (16)$$

where $E[D_{fs}]$ is the feedback suppression delay and $E[D_{fp}]$ is the feedback propagation delay. In protocol C, the timer-based feedback suppression algorithm presented in [16] is employed. If feedback suppression reduces the possibly R feedback messages from all receivers to a maximum number of Z feedback messages, the expected feedback suppression delay is given as derived in ([16]):

$$E[D_{fs}] = d_t \int_0^1 \left(1 - \frac{e^{\lambda_0 m} - 1}{e^{\lambda_0} - 1}\right)^{R_l} dm \qquad (17)$$

$$\lambda_0 = log(R) + 1 \qquad (18)$$

$$d_t = \overline{d} \cdot 1.2 \cdot \frac{\lambda_0}{log\,(Z)} \qquad (19)$$

where λ_0 is the optimal parameter of the exponential timer distribution, d_t the timer interval size, $\overline{d}$ is the average propagation delay between random receivers and $R_l = p \cdot R$ the expected number of receivers willing to send feedback. In our tree model we have homogeneous propagation delays $d_{i,j} = 2 \cdot (1 + w_b) \cdot d$ between a node i and all nodes j that are not in the same DER recovery group. Between node i and the nodes j that are in the same DER recovery group we have a propagation delay of $d_{i,j} = 2d$. We calculate $\overline{d}$ as $\overline{d} = \overline{d_{i,j}} \approx 2 \cdot (1 + w_b) \cdot d$, which leads to a very tight upper bound for the feedback suppression delay.

Since the number of potential feedback senders after the first retransmission round will be very small, we only consider the feedback suppression delay in the first round. We use the above calculation that applies for all loss scenarios. For the *heterogeneous independent loss* scenarios, the result will be a tight upper bound for the feedback suppression delay. The number of potential feedback senders R_l increases through high loss receivers. However, since we look at small fractions of high loss receivers, the increase is neglected.

After a receiver has sent feedback, a requested parity transmission can arrive at the receiver not before a full round trip time $RTT = 2 \cdot (2 + w_b) \cdot d$. Feedback is sent after each transmission round, for each transmission group. This way we get for the feedback propagation delay:

$$E[D_{fp}] = (E[K_C] - 1) \cdot (2 + w_b) \cdot 2d \qquad (20)$$

where $E[K_C]$ is the *number of transmission rounds* required for a group of packets for reliable delivery to all receivers. $E[K_C]$ is estimated by simulation for all loss scenarios as the average of the sample values obtained.

To calculate the parity packet coding time $E[D_c]$, we use measurements done with the coder presented in [17]. Since encoding and decoding time are approximately the same ([24]) we get from [17]:

$$E[D_c] = 2 \cdot k \cdot E[L_l] \cdot P \cdot c_d \qquad (21)$$

where k is the transmission group size, P is the transport layer packet size in kBytes and c_d is a machine dependent constant. $E[L_l] = p \cdot k$ is the expected number of parity packets to be used for decoding of the transmission group. Our completion time measure looks at the time elapsed until the last receiver is finished. The calculation of the coding delay in equation (21) considers a random receiver and thus gives a lower bound for the actual coding delay. Equation (21) is valid for all loss scenarios. For the *heterogeneous loss* scenarios, we consider the coding delay to be dominated by the high loss receivers, and set $p = p_h$, which is again a lower bound for the latency of the slowest receiver.

B. Protocol D1

For the transmission delay we account for two steps: the transmission from the source to the DER nodes and from the DER nodes to the receivers. Since the first transmission of original data is multicast to all nodes in the tree, we do not consider the additional transmission delay for the transmission of the original packets from the DER nodes to the receivers. Some packets may be lost on paths from the source to the DER nodes. Retransmission of those packets from the DER nodes to the receivers causes additional transmission delay. Since the loss probability p is very small, we neglect this transmission delay. For the total transmission delay for a TG of k packets for transmission to all receivers we get with (15):

$$E[D_t] = (E[M_{D1,G}] + E[M_{D1,I}] - 1) \cdot \frac{k}{\Lambda} \qquad (22)$$

$M_{D1,G}$ is the number of transmissions per packet required for reliable delivery to *all* of the DER nodes and $M_{D1,I}$ the number of transmissions per packet from the DER nodes to *all* of the Z receivers in *all* the groups. We get $E[M_{D1,G}]$ the same way as for the bandwidth analysis for the *independent* loss scenarios as $E[M_{D1,G}] = 1$, for *shared source link* loss with equations (11) and (5). We will estimate the value of $E[M_{D1,I}]$ by simulation for all loss scenarios.

The feedback delay for protocol D1 consists exclusively of feedback propagation delay, since protocol D1 does not perform feedback suppression. We calculate the maximum feedback propagation delay among all receivers in all groups as:

$$E[D_{fp}] = (E[K_{D1,S}] - 1) \cdot (1 + w_b) \cdot 2d + (E[K_{D1,I}] - 1) \cdot 2d \qquad (23)$$

where $E[K_{D1,S}]$ is the maximum number of transmission rounds required for delivery to all DER nodes and $E[K_{D1,I}]$ is the maximum number of transmission rounds required for delivery from the DER nodes to all receivers in all DER groups. For all independent loss scenarios we get $E[K_{D1,S}] = 1$. For *shared source link loss* we get the distribution of $K_{D1,S}$ for a TG of k packets with equation (11) as:

$$F_{K_{D1,S}}(m) = (1 - p'^m)^k \qquad (24)$$

$E[K_{D1,S}]$ can then be calculated with equation (5). $E[K_{D1,I}]$ will be estimated by simulation for all loss scenarios.

C. Protocol D2

The transmission delay for protocol D2 can be calculated as in equation (22) by replacing the respective variables for D1 with variables for D2. For all independent loss scenarios we get $E[M_{D2,G}] = 1$. For *shared source link loss*, $E[M_{D2,S}]$ can be calculated using the distribution from equation (4) with $R = 1$ together with equations (5) and (6). For all loss scenarios $E[M_{D2,I}]$ will be estimated by simulation.

The feedback delay for protocol D2 consists only of feedback propagation delay, which can be calculated the same way as in equation (23), again by replacing the respective variables. $E[K_{D2,S}]$ is the maximum number of transmission rounds required for delivery to all DER nodes and $E[K_{D2,I}]$ is the maximum number of transmission rounds required for delivery to all receivers in all DER groups, both using parity transmission. $E[K_{D2,S}]$ and $E[K_{D2,I}]$ will both be estimated by simulation for all loss scenarios.

The coding delay for protocol D2 is calculated the same way as for protocol C with equation (21). We assume that the parity packets coded by the source are sufficient for transmission from the DER nodes to the receivers and no additional parity packets have to be coded by the DER nodes.

VII. Latency Comparison

In the following, the completion time measure $E[D_{norm}]$ is compared for the protocols C, D1 and D2. The influence of the parameters p, Z, c_d and R on the protocols is evaluated for the *homogeneous independent loss* scenario. The scalability with the number of receivers of protocols C and D2 is compared for the different loss scenarios using the measure of relative performance. Unless stated otherwise, a constant packet size of $P = 2kB$ will be assumed. Through measurements with the FEC coder introduced in [17] on a SUN SPARC-20 workstation we got $c_d = 120 * 10^{-6}s$. The constant packet throughput is set to $\Lambda = 25/s$. [6] We set $RTT = 0.1s = 6d$ and the packet loss probability, that a receiver sees, to $p = 0.01$. The DER group size is $Z = 30$ and the number of receivers is $R = 10^4$.

A. Homogeneous Independent loss

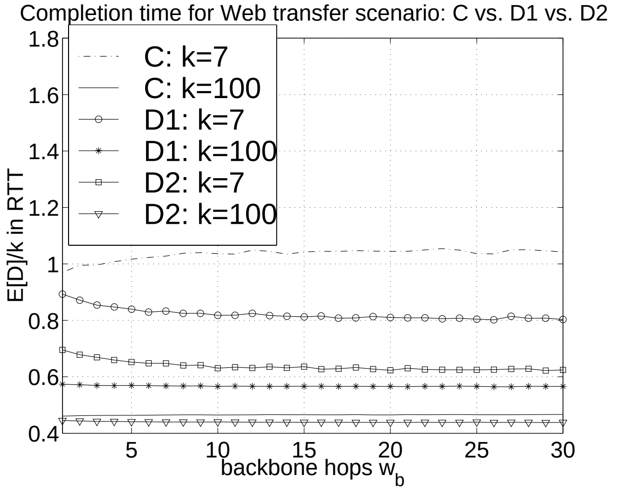

Fig. 10. Completion time $E[D_{norm}]$ dependent on number of backbone hops for homogeneous independent loss: C vs. D1 vs. D2, $p = 0.01$, $Z = 30$, $R = 10^4$, $c_d = 120 * 10^{-6}s$.

In our tree model (Figure 2) we have a variable number w_b of physical hops for the backbone links. The absolute total RTT between source and receiver is constant, but the fraction of the delay that is incurred on the backbone links is larger when w_b increases. In Figure 10 we see that the completion time increases slightly with increasing number of backbone hops for protocol C. The transmission delay, the feedback propagation delay and the coding delay are not influenced by a larger number of backbone hops, since the additional hops are loss free and the total RTT stays constant. However, the feedback suppression delay increases with an increasing number of backbone hops, since it is proportional to the propagation delay in between receivers. The completion time for protocol D1 decreases by no more than 10% with $k = 7$ and an increasing number of backbone hops. The delay for D2 decreases slightly less. Of all delay contributions, only the feedback propagation delay decreases due to the reduced RTT within the DER groups required for local retransmission. Since more retransmission rounds must be performed in the case of original packet retransmission, a smaller RTT has a stronger effect on $E[D_{norm}]$ for D1 than for D2. We also see for all protocols that the larger the transmission group size k, the smaller the influence of an increasing number of backbone hops. This is due to the larger impact of the transmission delay for large transmission groups, such that the feedback delay contributions that depend on w_b become relatively unimportant. We also found that the influence of w_b does not increase with a number of receivers $R > 10^4$. However, the influence of w_b increases and decreases with the packet throughput, as delay contributions other than the transmission delay become more important. Since the influence of w_b on the quality of the comparison is not considerable, we will set $w_b := 1$ from now on.

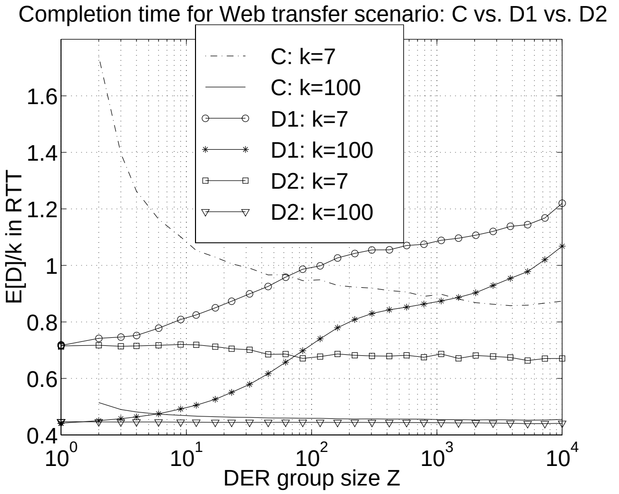

Fig. 11. Completion time $E[D_{norm}]$ dependent on DER group size Z for homogeneous independent loss: C vs. D1 vs. D2, $p = 0.01$, $R = 10^4$, $c_d = 120 * 10^{-6}s$, $w_b = 1$.

We can see in Figure 11 that protocol C performs better than protocol D1 in most of the range of Z: for $k = 7$, C performs better than D1 for values of $Z > 80$, for $k = 100$, C performs better than D1 for DER group sizes of $Z > 6$. For both protocols, larger transmission group sizes are an advantage. More thorough inspection of the results showed that the dominating delay contribution for protocol D1 is the transmission delay D_t. The transmission delay for protocol D1 increases with the group size because the receivers incur latency for unnecessary packet receptions. For protocol C, the feedback suppression delay increases for $Z \rightarrow 1$. The number of feedback messages for protocol C is set to be equal to Z, such that for small values of Z, the feedback suppression delay is very large. Protocol D2 per-

[6] A packet throughput of $\Lambda = 25/s$ has been reported by Bolot [25] for a loaded IP path between Sophia Antipolis, France (INRIA) and London (UCL).

forms better than protocol C for all values of Z. However, for a TG size $k = 100$, the difference between C and D2 is very small. The performance of protocol D2 is hardly influenced by Z. We look at the delay until the last receiver of all R receivers has received all packets. With parity transmission, the last receiver will not receive any unnecessary packets, no matter how large the DER group size is. In fact the transmission delay is the same for protocol C and D2. The feedback propagation delay, however, is lower for protocol D2. Protocol D2 performs better than protocol D1 over the whole range of Z.

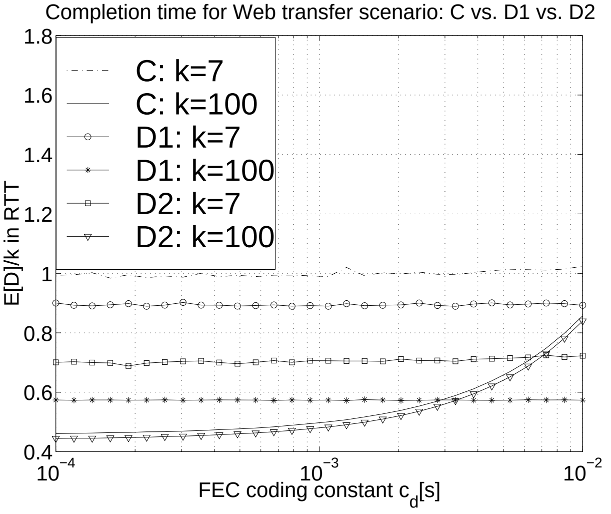

Fig. 12. Completion time $E[D_{norm}]$ dependent on FEC coding constant c_d for homogeneous independent loss: C vs. D1 vs. D2, $p = 0.01$, $R = 10^4$, $Z = 30$, $w_b = 1$.

To explore the influence of the coding constant c_d in Figure 12 we choose values for c_d of $10^{-4} \leq c_d \leq 10^{-2}$. The lower bound of $c_d = 10^{-4}$ was measured on a SPARC 20 workstation. For small values of c_d, the relative performance of the protocols as seen before is not changed. D1 performs better than C for small k and vice versa for large k. D2 performs better than both C and D1 for small c_d. Only for very large values of c_d and k, the coding delay for C and D2 becomes so dominant that they perform worse than D1. For the default value of $c_d = 120 * 10^{-6} s$, we saw that protocol D2 performs better than D1 in the rest of the parameter space. Thus, from now on we will leave out protocol D1 from our comparison.

It can be seen in Figure 13 that protocol D2 performs better than protocol C over the whole range of packet loss probabilities p. The reason is the smaller feedback propagation delay of the distributed scheme. The completion time increases with p for both protocols, but less steeply for protocol C. The reason is a decrease of the feedback suppression delay for protocol C with increasing p. The performance difference between C and D2 for large k is very small. This is because we look at the last receiver, such that the benefit of parallel transmission in the DER groups is partly lost.

It can be seen in Figure 14 that both protocol C and D2 scale very well with the number of receivers. Protocol D2 performs better than C over the whole range of R. For large k, C comes close to the performance of D2. D2 has a smaller feedback propagation delay and is through the constant DER group size hardly influenced by the increasing number of receivers. For protocol C, both the transmission delay and the feedback propagation delay increase with an increasing number of receivers. However, the feedback suppression delay decreases and the increasing feedback propagation delay is masked by the transmission delay for large TG sizes k.

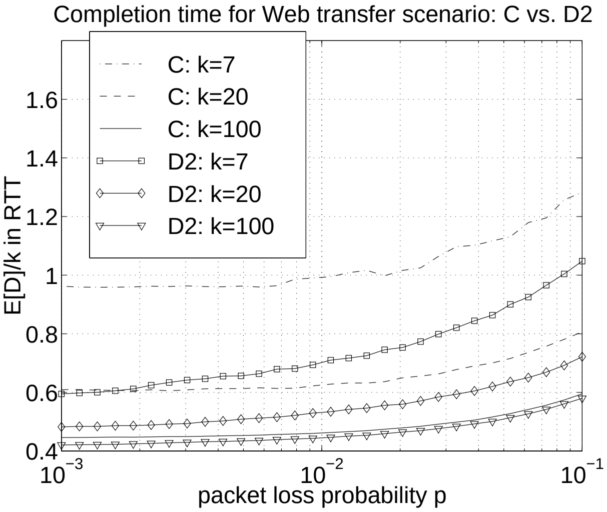

Fig. 13. Completion time $E[D_{norm}]$ dependent on packet loss probability p for homogeneous independent loss: C vs. D2, $R = 10^4$, $Z = 30$, $c_d = 120 * 10^{-6} s$, $w_b = 1$.

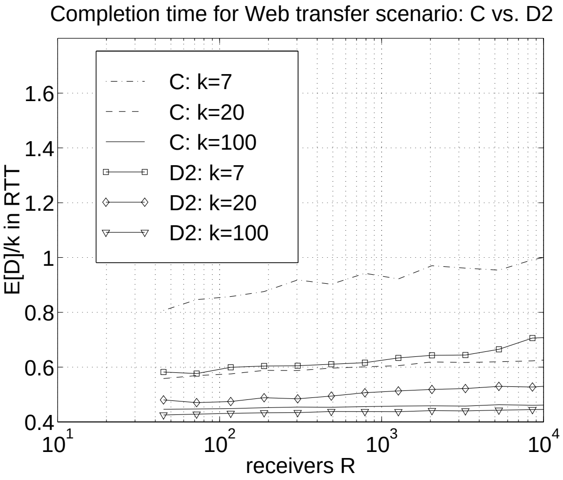

Fig. 14. Completion time $E[D_{norm}]$ dependent on number of receivers R for homogeneous independent loss: C vs. D2, $p = 0.01$, $Z = 30$, $c_d = 120 * 10^{-6} s$, $w_b = 1$.

B. *Other loss scenarios*

In Figure 15 the scalability of protocols C and D2 in the case of a large numbers of receivers is compared for the four different loss scenarios. Heterogeneous loss is modeled by 90% low loss receivers with the packet loss probability $p = 0.01$ and 10% high loss receivers with a packet loss probability of $p_h = 0.25$. With *Intra-group heterogeneous loss* there is a very slight improvement of the performance of protocol C relative to D2 compared to homogeneous loss. The feedback suppression delay for protocol C decreases even more than with homogeneous loss. This is due to the larger number of potential feed-

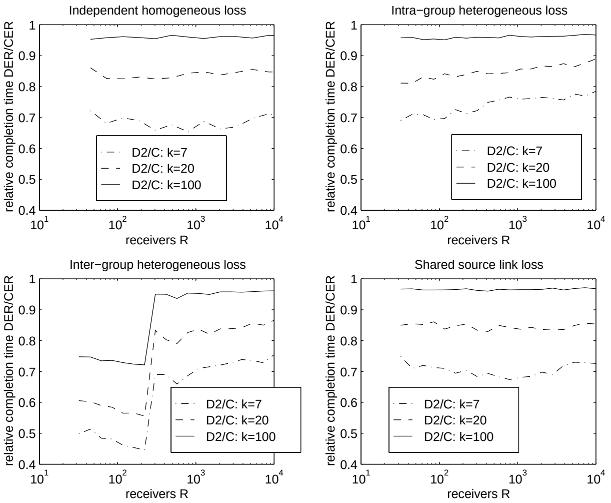

Fig. 15. Relative completion time dependent on R for different loss scenarios: C vs. D2, $p = 0.01$, $Z = 30$, $c_d = 120 * 10^{-6}s$, $w_b = 1$.

back senders. For large transmission groups, the transmission delay is too dominating to make the influence of the decreasing feedback suppression delay visible. The effect of *inter-group heterogeneous loss* is much the same as with intra-group heterogeneous loss. The discontinuity in the curve appears, when the first high loss group is introduced among the G DER groups.[7] Finally, *shared source link loss* does not make any noticeable difference to homogeneous independent loss. For calculation of the number of transmission, shared loss can be modeled through a smaller number of receivers R_{indep} experiencing independent loss (see Section IV). This improves the performance of both protocols, with a slight advantage for C. Shared loss does not have a great influence on the other delay contributions, such that the overall effect of shared loss is negligible.

C. *Burst loss*

We showed in the bandwidth Section (Section V) that burst loss increases the number of transmissions per packet and therefore the number of transmission rounds will also be increased. The performance advantage of protocol D2 over C compared to homogeneous loss will thus be even more striking when considering latency than it was when considering bandwidth. However, if transmission group sizes k are large enough to span several burst loss periods, the receivers essentially see a random non-bursty loss pattern. For large TG sizes k, protocol C then comes close to protocol D2. Our results on relative latency performance will stay qualitatively valid also for bursty loss patterns.

VIII. CONCLUSION

A. *Summary of results*

We compared the performance of one CER protocol (C), using parity transmission for error recovery, and two DER protocols, one using parity transmission (D2), and one using original data retransmission (D1) in terms of bandwidth and latency. Since D2 outperforms D1 in all cases, we further only considered D2.

We found that parity transmission for error recovery gives performance improvements to both schemes at almost no cost. Even coding schemes with complexity $O(k^2)$ contribute negligible delay, let alone recently discovered schemes with complexity $O(k)$ [19]. Parity transmission also guarantees excellent scalability for both DER and CER. For large transmission group sizes, the performance of CER comes close to the performance of D2. Considering the negligible coding delay, other issues in the comparison CER/DER now gain more importance (e.g. congestion control and network deployment).

Our results showed that DER clearly exhibits superior bandwidth performance if transmissions suffer from burst loss or very heterogeneous loss patterns among receivers. CER can partly catch up with DER, if larger transmission group sizes are used. In terms of latency, the diverse loss patterns have almost no influence on the relative performance of DER and CER: for large transmission group sizes, CER comes close to DER.

We have derived most of our results using a simple tree model with homogeneous, independent loss at the receivers. With results from varying loss models, and integration of MBone performance measurements, we showed that our tree model is valid.

B. *Why CER is desirable*

The excellent performance of DER does not come for free. To perform local retransmissions, there has to be either network support or support from the receivers. This means either that routers or users must devote part of their workstation processing time to performing retransmissions for other users. In the case of network support, protocol deployment and the choice of the DER nodes is a serious practical problem.

DER attempts to perform retransmissions as close as possible to the point of loss. In case of congestion, congestion control must be performed. However, congestion control for multicast with distributed error recovery seems to us a much more difficult problem than in case of CER.

For DER, the receivers must be organized in groups according to some metric. In order to adapt to changing network characteristics, dynamic grouping is recommended. A self-organizing technique with network support as a solution to this problem was proposed in [26]. The complexity of a dynamic grouping is at least $O(R)$.

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for their constructive comments. Eurecom's research is partially supported by its industrial partners: Ascom, Cegetel, France Telecom, Hitachi, IBM France, Motorola, Swisscom, Texas Instruments, and Thomson CSF.

REFERENCES

[1] S. Pingali, *Protocol and Real-Time Scheduling Issues for Multimedia Applications*, Ph.D. thesis, UMass, Sept. 1994.

[2] D. Towsley, J. Kurose, and S. Pingali, "A comparison of sender-initiated and receiver-initiated reliable multicast protocols," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 398–406, 1997.

[3] B. Levine and J. J. Garcia-Luna-Aceves, "A comparison of known classes of reliable multicast protocols,," in *Proc. Conference on Network Protocols (ICNP-96)*, Columbus, Ohio, Oct. 1996.

[7] In our simulation, we can only allow for natural numbers of high loss groups.

[4] K. Obrasczka, "Multicast transport mechanism: A survey and taxonomy," *IEEE Communications Magazine*, vol. 36, no. 1, pp. 94–102, Jan. 1998.

[5] C. Diot, W. Dabbous, and Crowcroft. J, "Multipoint communication: A survey of protocols, functions and mechanisms," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 277–290, April 1997.

[6] S. Paul, K. K. Sabnani, J. C. Lin, and S. Bhattacharyya, "Reliable Multicast Transport Protocol (RMTP)," *IEEE Journal on Selected Areas in Communications, special issue on Network Support for Multipoint Communication*, vol. 15, no. 3, pp. 407 – 421, April 1997.

[7] 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*, vol. 5, no. 6, pp. 784–803, Dec. 1997.

[8] J. Nonnenmacher, E. W. Biersack, and Don Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission," *IEEE/ACM Transactions on Networking*, vol. 6, no. 4, pp. 349–361, Aug. 1998.

[9] Matthew T. Lucas, Bert J. Dempsey, and Alfred C. Weaver, "MESH: Distributed error recovery for multimedia streams in wide-area multicast networks," in *International Conference on Communication, ICC'97*, Montreal, Canada, June 1997.

[10] Roger Kermode, "Scoped Hybrid Automatic Repeat reQuest with Forward Error Correction (SHARQFEC)," in *Proceedings of ACM SIGCOMM '98*, Vancouver, BC, Canada, October 1998.

[11] S. Lin and D. J. Costello, *Error Correcting Coding: Fundamentals and Applications*, Prentice Hall, Englewood Cliffs, NJ, 1983.

[12] Sneha K. Kasera, Jim Kurose, and Don Towsley, "A comparison of server-based and receiver-based local recovery approaches for scalable reliable multicast," in *Proceedings of IEEE INFOCOM '98*, San Francisco, CA, USA, March 1998.

[13] D. Rubenstein and D. Towsley, "Improving reliable multicast using active parity encoding services," in *INFOCOM 99*, March 1999.

[14] Maya Yajnik, Jim Kurose, and Don Towsley, "Packet loss correlation in the MBone multicast network," in *Proceedings of IEEE Global Internet*, London, UK, November 1996.

[15] M. Handley, "An examination of mbone performance," Tech. Rep. ISI/RR-97-450, USC/ISI, Jan. 1997.

[16] J. Nonnenmacher and E.W. Biersack, "Scalable Feedback for Large Groups," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 375–386, Jun. 1999.

[17] Luigi Rizzo, "Effective erasure codes for reliable computer communication protocols," *Computer Communication Review*, vol. 27, no. 2, pp. 24–36, April 1997.

[18] L. Rizzo, "On the feasibility of software FEC," Tech. Rep., Univ. di Pisa, Italy, Jan. 1997.

[19] Michael Luby, John W. Byers, Michael Mitzenmacher, and Ashutosh Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data," in *Proceedings of ACM SIGCOMM '98*, Vancouver, BC, Canada, October 1998.

[20] Sassan Pejhan, Mischa Schwartz, and Dimitris Anastassiou, "Error control using retransmission schemes in multicast transport protocols for real-time media," *IEEE/ACM Transactions on Networking*, vol. 4(3), pp. 413–427, June 1996.

[21] Pravin Bhagwat, Partho P. Mishra, and Satish K. Tripathi, "Effect of topology on performance of reliable multicast communication," in *Proceedings of IEEE INFOCOM'94*, Toronto, Ontario, Canada, June 1994, vol. 2, pp. 602–609.

[22] Dan Rubenstein, "Using packet-level FEC with real-time data delivery," available from the author, CS department, University of Massachusetts, Amherst, May 1997.

[23] Dan Rubenstein, Jim Kurose, and Don Towsley, "Real-time reliable multicast using proactive forward error correction," in *Proceedings of 8th International Workshop NOSSDAV*, 1998.

[24] Georg Carle and Ernst W. Biersack, "Survey of error recovery techniques for IP-based audio-visual multicast applications," *IEEE Network Magazine*, November/December 1997.

[25] J. C. Bolot, "Analysis and control of audio packet loss in the internet," in *5th Workshop on Network and Operating System Support for Digital Audio and Video*, T. D. C. Little and R. Gusella, Eds. April 1995, vol. 1018 of *LNCS*, pp. 163–174, Springer Verlag, Heidelberg, Germany.

[26] Christos Papadopoulos, Guru Parulkar, and George Varghese, "An error control scheme for large-scale multicast applications," in *Proceedings of IEEE INFOCOM '98*.

Martin S. Lacher (A '97 / ACM S 2000) received his M.Sc. degrees in Computer Science from the Technische Universität München, Munich, Germany as well as from the City College of the City University of New York, NY. He is a PhD student at Technische Universität München, where he does research on applications of Software Agents in Knowledge Management. His email address is: lacher@in.tum.de

Jörg Nonnenmacher received the M.Sc. degree in Computer Science from the University of Karlsruhe, Germany, in 1995, and the Ph.D. degree in 1998 at Institut Eurecom, Sophia Antipolis, France. In 1999 he joined Lucent Bell Labs. His email address is: nonnen@research.bell-labs.com

Ernst W. Biersack (M '88 / ACM '84) received his M.S. and Ph.D. degrees in Computer Science from the Technische Universität München, Munich, Germany. Since March 1992 he has been a Professor in Telecommunications at Institut Eurecom, in Sophia Antipolis, France. For his work on synchronization in video servers he received in 1996 (together with W. Geyer) the Outstanding Paper Award of the IEEE Conference on Multimedia Computing & Systems. For his work on reliable multicast he received (together with J. Nonnenmacher and D. Towsley) the 1999 W. R. Bennet Price of the IEEE for the best original paper published 1998 in the ACM/IEEE Transactions on Networking. Mr. Biersack is currently an associate editor of IEEE Network Magazine, ACM/IEEE Transactions on Networking, and of ACM Multimedia Systems. His email address is: erbi@eurecom.fr