Overlay and Peer-to-Peer Multicast with Network-Embedded FEC

Hayder Radha and Mingquan Wu

Michigan State University, Department of Electrical and Computer Engineering

East Lansing MI, 48824 (radha, Wu)@egr.msu.edu

ABSTRACT

Under traditional IP multicast, application-level FEC can only be implemented on an end-to-end basis between the sender and the clients. Emerging overlay and peer-to-peer (*p2p*) networks open the door for new paradigms of *network FEC*. The deployment of FEC within these emerging networks has received very little attention (if any). In this paper, we analyze and optimize the impact of Network-Embedded FEC (NEF) in overlay and *p2p* multimedia multicast networks. Under NEF, we place FEC codecs in *selected* intermediate nodes of a multicast tree. The NEF codecs detect and recover lost packets within FEC blocks at earlier stages before these blocks arrive at deeper intermediate nodes or at the final leaf nodes. This approach significantly reduces the probability of receiving undecodable FEC blocks. In essence, the proposed NEF codecs work as signal regenerators in a communication system and can reconstruct most of the lost data packets without requiring retransmission. We develop an optimization algorithm for the placement of NEF codecs within random multicast trees. Our theoretical analysis and simulation results show that a relatively small number of NEF codecs placed in (sub-)optimally selected intermediate nodes of a network can improve the throughput and overall reliability dramatically.

1. INTRODUCTION

Overlay and peer-to-peer (*p2p*) networks (e.g., [1]-[6]) are becoming increasingly popular for the distribution of shared content over the Internet. Most of the studies conducted for these networks have focused on multicast tree building. Further, these studies assume that reliable transport and congestion control are performed by the underlying end-to-end transport protocol such as TCP. However, this assumption may not be appropriate for realtime multicast applications. More importantly, the deployment of FEC *within* these networks for realtime multimedia applications has received very little attention (if any).

In this paper, we analyze and optimize the impact of Network-Embedded FEC (NEF) in overlay and *p2p* multimedia networks. The proposed NEF approach exploits these emerging networks by placing FEC codecs at *selected* internal nodes within random multicast trees. We develop a recursively optimum (globally sub-optimum) scheme for the placement of a given (small) number of NEF codecs within any randomly-generated multicast network of known (yet random) link loss rates. In essence, the proposed NEF codecs work as signal regenerators in a communication system, and hence, they can reconstruct the vast majority (and sometimes all) of the lost data packets without requiring retransmission. Figure 1 shows an example of NEF.

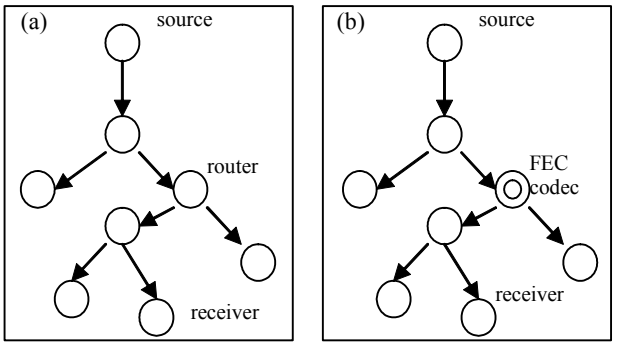


Figure 1 (a) In IP multicast, routers do not perform FEC (b) A NEF codec in a multicast tree can recover lost data and parity packets and send these lost packets downstream.

In the two forms of networks considered here, "overlay" and "*p2p*" (e.g., [1]-[6]), multicast functions such as membership management and data replication are promoted to the application layer. Here, to distinguish it from a *p2p* network, an *overlay* network is equivalent to a *proxy-based* network[1] [4]. In a *p2p* multicast network, each node in the multicast tree can also be a multicast client (receiver). In a (proxy-based) overlay network, only the leaf nodes are clients. Within both networks, and at each intermediate node, data packets reach the application layer, and then get replicated and forwarded. Hence, in both cases (proxy-based or *p2p*), packet-loss recovery as an application level service can be placed in the intermediate nodes of the network.

We show through extensive analysis and simulations that a small number of NEF codecs can significantly improve the overall throughput and the probability of decodable FEC blocks over a given multicast tree. These NEF-based networks can be designed with the desired level of reliability for the delivery of realtime multimedia (e.g., video and audio). The remainder of the paper is organized as follows. Section 2 presents an analytical model for Network-Embedded FEC routes within a multicast network. Section 3 describes and analyzes an optimization NEF codec placement algorithm. Simulation results are presented in Section 4. A brief summary is presented in Section 5.

2. ANALYSIS OF NETWORK-EMBEDDED FEC ROUTS

Previous studies analyzed the packet-loss model for FEC-enhanced multicast trees (e.g., [7][8]). These studies are based on the IP multicast model, in which intermediate nodes do not participate in FEC. Here, we study the packet-loss model of a

[1] Please note that both *p2p* and *proxy-based* networks are forms of *overlay* networks [4]. In this paper, we use the term *overlay networks* to refer to proxy-based networks.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

multicast tree when FEC codecs are placed in the intermediate nodes of a tree. In our analysis, we use the following notations:

$RS(n,k)$	Reed Solomon code with *k* data packets and *n-k* parity packets.
T ; $\mid T \mid$	A multicast tree with a root node *r* & a total number of nodes $\mid T \mid$.
T^c ; T_l^c	T^c is a sub-tree rooted at some node $c \in T$ but does not include the node c; T_l^c is the set of leaf nodes of T^c.
$P_v(i)$	Probability that node $v \in T$ receives exactly *i* packets.
$P_{v\mid v-1}(i,j)$	Probability that node v receives i packets given that its parent $v-1$ sends j packets.
p	the packet loss probability between the link from $v-1$ to v

Similar to previous studies, we assume a binomial distribution for the packet losses. For node v, if its parent $v-1$ sends j packets, the probability that it receives i packets is:

$$P_{v|v-1}(i,j) = \binom{j}{i}(1-p)^i p^{j-i} \quad (1)$$

When computing the probability $P_v(i)$ that a node v receives exactly *i* packets, we need to consider two cases; first, we consider the case when the parent node $v-1$ has no codec; second, we consider the case when the parent node $v-1$ has a NEF codec. If node v's parent does not have a codec, the probability that node v receives i packets is:

$$P_v(i) = \sum_{j=i}^{n} P_{v-1}(j) P_{v|v-1}(i,j) \quad (2)$$

Note that $P_{v\mid v-1}(i,j) = 0, \ \forall\, j < i$. In other words, node v can receive i packets only when its parent $v-1$ sends at least i packets. For the root node (*r*) of the tree, we define

$$P_r(i) = \begin{cases} 0 & 0 \le i \le n-1 \\ 1 & i = n \end{cases} \quad (3)$$

Equation (2) is a recursive function, and hence with the initial condition from (3), we can calculate the probability $P_v(i)$, for any node v in the multicast tree, that it receives exactly i packets. When a node has a codec for a $R(n,k)$ block, and if that node receives less than k packets and cannot decode the FEC block, it will just forward the received packets as usual; if it receives k or more packets, the node can decode the block and reconstruct the original data. It can also reproduce the lost parity packets. In fact, a codec can produce more or less than $n-k$ parity packets if desired; however, in this paper, we assume that the NEF codecs reconstruct the original data and reproduce the lost parity packets using the same $R(n,k)$ code. These packets are then multicasted downstream. (The design of NEF codecs with an adaptive FEC erasure codes is a problem that we are currently pursuing, and it is beyond the scope of this paper.)

A node that has a NEF codec and which receives $k \le j \le n$ packets will send *n* packets. If v is the immediate child of a codec, the probability that it receives i packets becomes

$$P_v'(i) = \begin{cases} \sum_{j=k}^{n} P_{v-1}(j) P_{v|v-1}(i,n) & k \le i \le n \\ \sum_{j=k}^{n} P_{v-1}(j) P_{v|v-1}(i,n) + \sum_{j=i}^{k-1} P_{v-1}(j) P_{v|v-1}(i,j) & 0 \le i \le k \end{cases} \quad (4)$$

Once a node c is assigned a NEF codec, the probability $P_v(i)$ for all $v \in T^c$ will change and need to be recomputed. We use (4) to calculate $P_v(i)$ for the immediate children of the codec. For nodes that are not immediate children of a codec, the calculation of $P_v(i)$ is the same as equation (2).

Here we use P_v^{dec} to represent the probability that node v can decode a $RS(n,k)$ block:

$$P_v^{dec} = P_v(i \ge k) = \sum_{i=k}^{n} P_v(i) \quad (5)$$

We define the average decodable probability of a tree T for *p2p* and proxy-based overlay networks, respectively, as:

$$P_{avg}^{dec} = \frac{\sum_{v \in T-r} P_v^{dec}}{\mid T \mid -1} \quad (6) \ ; \qquad P_{avg-leaf}^{dec} = \frac{\sum_{v \in T_l^r} P_v^{dec}}{\mid T_l^r \mid} \quad (7)$$

If we use $r_d(v)$ to represent the number of received data packets (not including the parity packets received) of a FEC block at node v, then, $E[r_d(v)] = \sum_{i=k}^{n} k P_v(i) + \frac{k}{n} \sum_{i=0}^{k-1} i P_v(i)$ (8)

Here we assume that for a $RS(n,k)$ block, if a node receives i packets, on average only $(k/n)i$ are data packets. For a *p2p* and overly networks, we define the data throughput as:

$$g = \frac{\sum_{v \in T-r} E[r_d(v)]}{(\mid T \mid -1)k} \quad (9) \ ; \qquad g_{leaf} = \frac{\sum_{v \in T_l^r} E[r_d(v)]}{\mid T_l^r \mid k} \quad (10)$$

3. OPTIMUM PLACEMENT OF NETWORK-EMBEDDED FEC CODECS

In this section, we develop a mechanism for placing NEF codecs within a given network topology. In a large topology, identifying the optimum locations for the NEF codecs is not a trivial task. One objective is to place codecs in the intermediate nodes of a topology to maximize the average throughput. Assuming that the loss rate for each link in the topology and the number of codecs to be placed are known beforehand, the problem is similar to (but different from) the well-known *P*-median problem [9][10]. A *P*-median problem is to find *P* locations in the network to place facilities in order to minimize the overall cost for servicing all of the nodes. Generally, in a *P*-median problem, the cost to serve a node is determined by the *weight* at the node and the distance between the node and its nearest available facility. The *P*-median cost has nothing to do with other facilities placed in the network. As we have seen in the previous section, in order to calculate the decodable probability and throughput, we need to know the locations of the codecs that have been placed on that path, not just the immediate codec that serves the node.

Because the throughput at a node in a NEF network is impacted by all the codecs placed along the path from that node to the source (root), the dynamic programming approaches that have been used in previous network-placement problems (e.g., [10]) cannot be used to solve the NEF codec placement problem. In the following, we use a greedy algorithm to place *m* codecs in the multicast tree.

The greedy algorithm finds the best location for the first codec, then the next best location for the second one, and so on. Once a node is selected, an FEC codec is added to regenerate any lost data or parity packets. Let $T^c \subset T$ be the sub tree rooted at node $c \in T$ not including c. If c is set as a "codec node", only those nodes $v \in T^c$ will benefit from this selection; meanwhile, the "codec node" c itself will not be affected. For nodes $v' \in T - T^c$, everything remains unchanged. Let $E[r_d(v)]$ and $E'[r_d(v)]$ denote the average received packets for node $v \in T^c$ before and after node c is set as a codec node, respectively. We need to find $c \in T$ that maximizes the following:

$$\max_{c \in T} \left[\sum_{v \in T^c} (E'[r_d(v)] - E[r_d(v)]) \right] \quad (15)$$

A similar optimization objective function can be expressed for proxy-based overlay networks, except here the summation takes place over the leaf nodes only. Under the proposed greedy algorithm, we use an exhaustive search to find the best place for the first codec, after we find the optimum $c \in T$ node, we place the codec at that node. We use the same method to place the next codec; this process continues until all of the *m* codecs are placed.

The proposed greedy algorithm does not guarantee a global optimum solution for the placement of the *m* FEC codecs. Nevertheless, its performance has been very close to the global optimum. Table 1 shows the performance (in terms of throughput) resulting from the placement of *m*=2 and 3 FEC codecs (within 100-node multicast trees) based on the greedy algorithm, and compares these numbers with the throughput of the actual optimum placement under three (average) packet-loss ratios (*p*) over the multicast trees' links. (More details on the simulations are presented in the next section.) It is clear from the table that the greedy algorithm provides an excellent set of (sub-)optimum solutions in all 6 cases covered in this example.

Table 1 Average Throughput:

Comparison between Optimal and Greedy algorithm

Num of codecs	*p*=3%		*p*=4%		*p*=5%	
	opt	greedy	opt	greedy	opt	greedy
2	98.5%	98.4%	93.9%	91.9%	87.8%	87.8%
3	99.1%	99.1%	95.8%	95.4%	90.4%	90.4%

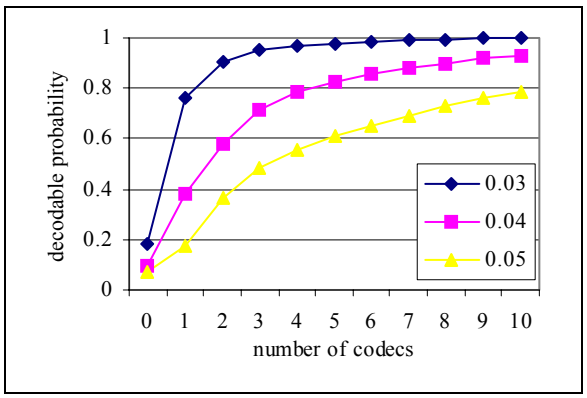
4. ANALYSIS AND SIMULATION RESULTS

We applied the performance analysis presented above to a general tree structure. We use the popular Georgia Tech gt-itm [11] network topology generator to produce a set of ten 100-node transit-stub graphs. For each graph, we use Dijkstra's Shortest Path First (SPF) algorithm to produce a tree rooted at a randomly selected node. We used the greedy algorithm described in the previous subsection to place the NEF codecs in the multicast tree. The number of codecs was increased from 0 to 10. After each codec is placed, we calculate the improvement on average decodable probability and throughput. In addition to applying the above performance analysis on the ten 100-node trees, we used the *network simulator2* (ns2) [12] with some modifications for the support of the proposed NEF codecs in intermediate nodes. We modified the simulator to allow packets to reach the UDP and application layers. We have implemented a FEC UDP agent and a FEC application in the simulator. The analysis and simulation results were virtually identical. Below, and due to space limitations, we only present the analysis results.

As mentioned above, in a *p2p* overlay multicast network, nodes in the multicast tree are also end users, which often are placed at the edge of the Internet. Each hop in the overlay network often consists of several underlying physical hops. This implies that the loss rate of each hop could be higher than the loss rate of a backbone link in an IP multicast model. Here, we show results when the loss rate per-link is set to 3%, 4%, and 5%. These loss rates are in accordance with previous studies [13]. We studied the performance improvement under each of these loss rates for a variety of RS codes. Here, we present the results for $RS(255, 223)$, which is a popular FEC code that has hardware and software implementations. (The channel coding rate[2] for $RS(255, 223)$ is 87.5%.)

[2] This code rate may be high for some of the loss rates that are evaluated in this paper. However, it is important to note that the main conclusions of our study are valid regardless of the specific RS codes used. In particular, the proposed NEF framework can be used in one of two ways. Under one approach, a *given* RS code is already being used (on an end-to-end basis) prior to adding any NEF codecs. In this case, NEF can significantly improve the overall throughput as shown extensively by our analysis and simulations in this paper. Under another approach, a reliable communication infrastructure is already in place. This reliable infrastructure would be normally based on using very conservative (low)

(a)



(b)

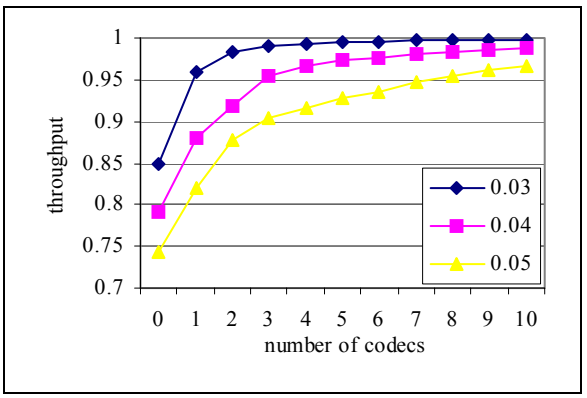


Figure 2 (a) Average decodable probability over all nodes. (b) Average data packets throughput over all nodes

The average FEC block decodable probability and data throughput for each tree were evaluated. The results are the averages over all of the ten random trees that were analyzed. Figure 2(a) shows the average decodable probability (over all nodes in a *p2p* tree) when the loss rates are set to 3%, 4% and 5%. (Similar results where obtained for proxy-based overlay networks.) When no codecs are added, the FEC block decodable probabilities are very low for all three loss rates. For example, if the link loss ratio is 3%, the average decodable probability is just 18.6%. As the codec number increases, we see a dramatic increase in the decodable probability. It can be observed that a relatively small number of codecs can increase the decodable probability significantly. For a 3% per-link loss rate, the first codec increase the decodable probability from 18.6% to 76%; the first 3 codecs increase the decodable probability to above 95%. When the number of codecs increases to 10, the decodable probability reaches 99.9%; this implies that we can use NEF to achieve a very high level of reliability while using a very high (i.e., efficient) channel-coding rate. The results for the throughput are shown in Figure 2(b). For an average *p2p* node, when no codecs are added, the throughput is about 85% with per-link loss rate of 3%. The first codec raises the throughput to over 95%. With only 3 NEF codecs, the throughput increases to 99%. For a typical video application, reducing the effective packet losses from 15% (85% throughput) to less than 1% (higher than 99% throughput) will naturally have dramatic improvements in the decoded video quality, both in terms of PSNR and visual perception. Under high losses, traditional end-to-end FEC could resort to a significantly lower FEC coding rate (to lower the packet losses and achieve high reliability). However, this reduces the effective source video rate significantly. In this case, NEF could be used to maintain the high reliability performance while increasing the FEC rate significantly (i.e., increasing the effective source video bitrate). Either way, NEF provides salient and dramatic improvements in the delivery of realtime video over *p2p* and overlay networks.

FEC rates (i.e., much lower than the effective end-to-end channel capacity). In this case, NEF can be used to significantly improve the efficiency of the RS codes by increasing its rate while maintaining the same level of reliability provided by the original infrastructure. In this paper, we focused on the first scenario to illustrate the benefits of the proposed NEF-based framework.

5. SUMMARY

In this paper, we introduced and analyzed Network Embedded FEC (NEF) over *p2p* and overlay multicast multimedia networks. Under this framework, FEC codecs are placed in the intermediate nodes of a multicast tree. We implemented a simple sub-optimum codec placement algorithm on a general tree structure. Analysis and simulation results show that a small number of codecs placed in the intermediate nodes of an overlay network can significantly improve the overall throughput and decodable probability. Based on a very large number of network simulations that we conducted, NEF is very effective for small and large network topologies.

REFERENCES

[1] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, "Scribe: A large scale and decentralized application-level multicast infrastructure," in IEEE JSAC, vol. 20, no. 8, October 2002

[2] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in Proc of ACM SIGCOMM, August 2001.

[3] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," in Proc of Middleware, Nov. 2001.

[4] Y.-H. Chu, S.G. Rao, and H. Zhang, "A Case for End System Multicast," in Proc. ACM Sigmetrics, June 2000.

[5] Dimitrios Pendarakis, Sherlia Shi, Dinesh Verma, and Marcel Waldvogel, "ALMI: An Application Level Multicast Infrastructure," *Proc. of the 3rd USNIX Symposium on Internet Technologies and Systems,* March 2001.

[6] M. Castro, *et al*, "An Evaluation of Scalable Application-level Multicast Built Using Peer-to-peer overlays", Infocom 2003, San Francisco, CA, April, 2003.

[7] J. Nonnenmacher, E. Biersack, and D. Towsley. "Parity-Based Loss Recovery for Reliable Multicast Transmission." *IEEE Trans. On Networking*, pages 349-361, Aug. 1998.

[8] J. Nonnenmacher, L. Martin, J. Matthias, E. Biersack, G. Carle, "How bad is Reliable Multicast without Local Recovery?," Proc. I *IEEE INFOCOM* '98, volume 3, pp. 972-9, March 1998.

[9] Mark S. Daskin, Network and Discrete Location: Models, Algorithms, and Application, 1995, John Wiley & Sons, Inc.

[10] B. Li, M. J. Golin, G. F. italiano, X. Deng, and K. Sohraby, "On the optimal placement of web proxies in the internet," in *IEEE INFOCOM'99*, Mar. 1999, pp.1282-1290.

[11] E. W. Zegura, "*GT-ITM: Georgia tech internetwork topology models (software),*" http://www.cc.gatech.edu /fac/Ellen.Zegura/gt-itm/gt-itm.tar.gz, 1996.

[12] http://www.isi.edu/nsnam/ns/.

[13] M. Yajnik, J. Kurose, D. Towsley, "Packet Loss Correlation in the Mbone Multicast Network," IEEE Global Internet Conference, London, November 1996.