ADJUSTING FORWARD ERROR CORRECTION FOR TCP-FRIENDLY STREAMING MPEG

Huahui Wu, Mark Claypool, Robert Kinicki

Computer Science Department
Worcester Polytechnic Institute
Worcester, MA 01609, USA
{flashine,claypool,rek}@cs.wpi.edu

ABSTRACT

The growing requirement of TCP-Friendly bandwidth use by streaming video plus the proven advantages of Forward Error Correction (FEC) to combat packet loss drives need to optimize the amount of FEC in a TCP-Friendly video stream. In this paper, we derive an analytical model for predicting the playable frame rate in a TCP-Friendly MPEG stream with FEC. Our model characterizes the Group Of Pictures (GOP) and FEC that are part of the MPEG video transmission. Assuming a network estimate for the packet loss probability, our model incorporates TCP-Friendly data rate constraints to calculate a total playable frame rate. For a given packet loss probability, we use our model to search the variable space to find the MPEG and FEC configuration that yields the optimal playable frame rate. Analysis over a range of network conditions indicates that adjusting FEC can provide a significant performance improvement, while adjusting a well-chosen GOP will contribute little improvement. Further analysis shows that a poor choice for a GOP can result in a large degradation of the playable frame rate. Overall, by introducing moderate amounts of FEC overhead, frame rates can be improved 10 to 50 times under network conditions of moderate to high loss rates.

1. INTRODUCTION

The growing number of users with high bandwidth Internet connections and the increasing power of desktop computers have fueled the use of the Internet to carry potentially high-quality video.

Unlike typical Internet traffic, streaming video is sensitive to delay and jitter, but can tolerate some data loss. Additionally, streaming video transmissions tend to yield better quality video when the underlying protocol provides a steady data rate rather than a bursty data rate. For these reasons, streaming video applications often use UDP as a transport protocol rather than TCP.

Unfortunately, UDP does not reduce its data rate when an Internet router drops packets to indicate congestion. Using UDP as the underlying transport protocol for streaming video flows may mean that UDP will both introduce traffic to cause congestion and not respond to relieve the saturation of a bottleneck due to the congestion. As network researchers have argued [4], response to congestion is critical to the health of the Internet, to the extent that some Internet researchers have pushed that all Internet flows should be required to be *TCP-Friendly*. Recent research has proposed rate-based TCP-Friendly protocols for streaming media [14, 5, 15] as alternatives to UDP.

In an effort to improve MPEG [9] video quality in the presence of network congestion while reducing the data rate, [3] and [17] vary the size and makeup of the Group of Pictures (GOP) to reduce network bandwidth during congestion. However, these studies address neither the best GOP pattern to use with MPEG nor the issue of TCP-Friendly bandwidth use. Thus, the question remains whether there exists an optimal GOP pattern for a streaming MPEG flow given TCP-Friendly constraints.

Besides the problem of avoiding congestion, video streams must react to the problem of packet loss. While streaming video applications can tolerate some data loss, excessive data loss can produce unacceptable media quality. Moreover, the inter-frame dependencies needed to achieve high-compression rates in video exacerbate the degradation in quality when the primary frames are lost. While retransmissions are an appropriate response to packet loss for most applications and even many streaming applications, large network timeout periods from retransmissions cause problems for applications such as streaming videos that have short end-to-end delay requirements.

By adding redundant data [2, 11, 13, 16], Forward Error Correction (FEC) can be used to repair the damage to the video stream due to packet loss. Used properly, FEC can reduce or eliminate packet loss and partially or fully insulate video applications from degraded quality [7]. However, since FEC adds overhead to the video stream, if a video flow is to operate within TCP-Friendly throughput constraints, adding FEC will reduce the playable rate of the transmitted video content. Thus, given a estimated network loss rate and the constraint of a TCP-Friendly data rate, the choice of FEC can be cast as a constraint optimization problem where the goal is to choose the combination of FEC packets that optimizes the quality of the video stream.

This paper derives an analytical model for streaming video with FEC in a TCP-Friendly environment. Given an estimate of the loss rate that a streaming video flow experiences, our model uses GOP patterns and FEC weighted by frame type to compute the total playable frame rate. We design experiments to search the space of possible GOP and FEC combinations to find the maximum playable frame rate given a TCP-Friendly bandwidth constraint. Over a range of network and video conditions, our results show: 1) adjusting FEC to different packet loss levels provides significant benefit to the playable frame rate and video quality; 2) adjusting GOP over a well-chosen GOP pattern has only moderate benefit; 3) using a poorly-chosen GOP can greatly reduce playable frame rates.

In the remainder of the paper, Section 2 provides background knowledge; Section 3 introduces our analytic model; Section 4 discusses our methodology and experimental settings; Section 5 analyzes the results; and Section 6 summarizes the paper.

2. BACKGROUND

2.1. TCP-Friendly

A flow is considered to be TCP-Friendly if if its data rate does not exceed the maximum data rate of a conformant TCP connection in the same network circumstances. Padhye et al [12] analytically derived the following equation for TCP throughput:

$$T = \frac{S}{t_{RTT}\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1+32p^2)} \qquad (1)$$

We assume that streaming flows are TCP-Friendly, and use T as an upper bound of sending rate.

2.2. Forward Error Correction

Media independent FEC [6] takes K original packets, adds $(N-K)$ redundant packets, and sends the N packets out. If any K or more packets are received, then all the original packets can be completely reconstructed. Thus the probability $q(N,K,p)$ that a K-packet video frame is successfully received with $N-K$ redundant

$$q(N,K,p) = \sum_{i=K}^{N}\left[\left(\begin{array}{c} N \\ i \end{array}\right)(1-p)^i * p^{N-i}\right] \qquad (2)$$

2.3. MPEG

MPEG video typically continually repeats the pattern of I, P, B frames (known as a Group of Pictures or GOP) in an individual video stream. Because of the dependencies of the I, P, B frames, the loss of one P frame renders some of other P and B frames useless, and the loss of one I frame results in the loss of the whole GOP.

Let N_P represent the number of P frames in a GOP, N_B represent the number of B frames in a GOP, and N_{BP} represent the number of B frames in between an I and a P frame or two P frames[1]. Obviously, $N_B = (1+N_P) \times N_{BP}$. Using these terms to characterize GOP patterns, a specific GOP pattern can be identified uniquely by G(N_P,N_{BP}). For example, GOP(2,2) signifies the pattern 'IBBPBBPBB'.

In the rest of this paper, we will use subscripts to identify individual frames within a GOP. The single I frame of a GOP is referred to as I_0, while P frames are referred to as P_i, where $1 \leq i \leq N_P$, and B frames are referred to as B_{ij}, where $0 \leq i \leq N_P$ and $0 \leq j < N_{BP}$. For example, P_3 is the third P frame, and B_{01} is the second B frame in the first interval of I and P frames.

3. ANALYTICAL MODEL

This section provides the details for the analytic model we develop to investigate means to improve the playable frame rate of TCP-Friendly streaming video flows in the presence of network packet loss. First, we identify system parameters related to TCP-Friendly MPEG flows (see Section 3.1). Next, working from frame size parameters and representations of the amount of FEC to add per frame type, we create a system of equations to characterize the probability of a successful transmission for each MPEG frame type (see Section 3.2). By adding information about the Group Of Picture (GOP) structure and frame dependencies, we derive formulas for flow transmission rate and playable frame rate in Section 3.3.

3.1. Parameters and Variables

The following six parameters are treated as fixed attributes of a streaming MPEG flow for our analytical model in the rest of this paper.

t_{RTT}: the Round-Trip Time
t_{RTO}: the TCP retransmition timeout
S: the packet size (in bytes)
S_I,S_P,S_B: the size of the I,P,B frames (in packets)

The remaining six attributes of an MPEG video flow over a lossy network are treated as variables for the remainder of our discussion:

p: the packet loss percentage

[1]We assume B frames are distributed evenly in the intervals between I and P frames.

N_P: the number of P frames in the GOP

N_{BP}: the number of B frames in an interval of I and P frames

S_{IF},S_{PF},S_{BF}: the number of FEC packets added to each I,P,B frame respectively.

p is determined by the network, whereas N_P and N_{BP} can be adjusted to provide an adjusted GOP, and S_{IF}, S_{PF} and S_{BF} can be adjusted to provide an adjusted FEC.

The strategy in our model is to assume that the streaming flow's network protocol is able to provide an estimate of the current network loss probability. Using this estimate, the model can be used to adjust the other five variables so as to optimize the total playable frame rate and explore the effect of today's typically non-adjusted MPEG video.

3.2. Frame Transmission Probabilities

Given specific I, P, and B frame sizes, and the distribution of redundant FEC packets added to each frame type, Equation 2 can be used to compute the probabilities of successful transmission for each frame type as following:

$$\begin{aligned} q_I &= q(S_I + S_{IF}, S_I, p) \\ q_P &= q(S_P + S_{PF}, S_P, p) \\ q_B &= q(S_B + S_{BF}, S_B, p) \end{aligned} \tag{3}$$

3.3. Playable Frame Rate

Now the GOP rate (GOPs per second) can be expressed analytically under the TCP-Friendly constraint expressed in Equation 1. Subsequently, the frame dependency relationships for I, P, B frames are used to compute the playable rate for each frame type.

3.3.1. GOP Rate

Assuming Equation 1 provides an expression for T, the GOP rate G is computed as:

$$G = \frac{T/S}{(S_I + S_{IF}) + N_P(S_P + S_{PF}) + N_B(S_B + S_{BF})} \tag{4}$$

3.3.2. Playable Rate of I Frames

Since I frames are independently encoded, the playable rate of I frames R_I is simply the number of I frames transmitted successfully over the network. We have:

$$R_I = G \cdot q_I \tag{5}$$

3.3.3. Playable Rate of P Frames

The first P frame, P_1, can only be displayed when its preceding I frame and itself are successfully transmitted. Thus P_1's playable frame rate is $R_{P_1} = R_I \cdot q_P$. Since each subsequent P_i in the GOP depends upon the success of P_{i-1} and its own successful transmission, we have:

$$R_{P_i} = R_I \cdot {q_P}^i \tag{6}$$

and the playable P frame rate for all P frames:

$$R_P = \sum_{i=1}^{N_P} R_{P_i} = G \cdot q_I . \frac{q_P - q_P^{N_P+1}}{1 - q_P} \tag{7}$$

3.3.4. Playable Rate of B Frames

All B frames in the same interval between an I or P frame have the same dependency relationship and thus these B frames all have the same playable rate.

A B frame that precedes a P frame depends only on that P frame. It is not necessary to consider the I or P frames before this P frame since these dependency effects have already been accounted for in the success probability of the P frame. Thus:

$$R_{B_{ij}} = R_{P_{i+1}} \cdot q_B \;\; when \; 0 \leq i \leq N_P - 1 \tag{8}$$

When a B frame precedes an I frame, it depends on both the preceding P frame and the succeeding I frame. For these B frames:

$$R_{B_{ij}} = R_{P_i} \cdot q_B \cdot q_I \;\; when \; i = N_P \tag{9}$$

Finally, the playable B frame rate for all B frames is:

$$\begin{aligned} &R_B = N_{BP} \cdot \textstyle\sum_{i=0}^{N_P} R_{B_{i0}} \\ &= N_{BP} \cdot G \cdot q_I \cdot q_B \cdot (\tfrac{q_P - q_P^{N_P+1}}{1-q_P} + q_I \cdot q_P^{N_P}) \end{aligned} \tag{10}$$

3.3.5. Total Playable Frame Rate

The total playable frame rate is:

$$R = R_I + R_P + R_B \tag{11}$$

Using the above equations for R_I, R_P and R_B, the total playable frame rate is:

$$\begin{aligned} &R = G \cdot q_I + G \cdot q_I . \tfrac{q_P - q_P^{N_P+1}}{1-q_P} \\ &+ N_{BP} \cdot G \cdot q_I \cdot q_B \cdot (\tfrac{q_P - q_P^{N_P+1}}{1-q_P} + q_I \cdot q_P^{N_P}) \\ &= G \cdot q_I \cdot (1 + \tfrac{q_P - q_P^{N_P+1}}{1-q_P} + N_{BP} \cdot q_B \\ &\cdot (\tfrac{q_P - q_P^{N_P+1}}{1-q_P} + q_I \cdot q_P^{N_P})) \end{aligned} \tag{12}$$

4. EXPERIMENTS

In this section, we use the closed form formula for the total playable frame rate derived in Section 3 to explore the space of GOP and FEC given TCP-Friendly bandwidth constraints. We use experimental instances of the probabilistic model to exhaustively search the system variables space to find the maximum playable frame rate for a given packet loss percentage.

4.1. Methodology

To study the effects of adjusted GOP and FEC on performance for a given loss percentage, p, we execute the following steps on the model to optimize the playable frame rate:

1. Given a specified p, determine the TCP-Friendly data rate T from Equation 1.

2. For each set of the other five system variables, compute the playable frame rate R using Equations 4 and 12. We characterize the variation of these five variables in terms of two searching dimensions: the GOP dimension and the FEC dimension. For different fixed combinations of these dimensions, we compute the rate R for each possible set of the remaining variables. Once the maximum rate is determined, we record the value of each of the five variable settings to produce this rate.

3. For all the combinations of searching dimensions, we compare the maximum rates to consider the benefits of adjustments in both the GOP and FEC dimensions. Later in the analysis, experiments with the model are run to consider the behavior of the optimal pattern of GOP and FEC in the face of different loss percentages.

4.2. System Settings

In our system, we use a RTT of 50ms and a packet size of 1KB, We use $t_{RTO} = 4t_{RTT}$ based on recommendations in [5]. The possible loss percentage p are 0.5%, 1%, 2%, 3%, ... 10%.

The MPEG frames' sizes were chosen based on the mean sizes of I, P, B frames measured in [10], and moved up to their ceiling to provide an integer number of packets. We assign S_I, S_P and S_B with 25, 8 and 3 packets respectively.

4.3. Searching Dimensions

The search choices for optimizing the playable frame rate are viewed as moving in two dimensions - one for FEC components and the other for GOP patterns. Table 1 lists the four possible combinations of the two searching dimensions we explore in Section 5.

Table 1. Searching dimension schemas

Schema	Description
A	Fixed
B	Adjusted GOP
C	Adjusted FEC
D	Adjusted FEC and GOP

The word 'Adjusted' implies the optimal playable rate over the search space dimension. Schema A is called 'Fixed' since it uses no FEC and a fixed GOP, which is a commonly used GOP [1, 8], GOP(4,2) ('IBBPBBPBBPBBPBB'). Schema B is called 'Adjusted GOP' since it has no FEC and adjusts N_P and N_{BP} to optimize the playable frame rate. Schema C is called 'Adjusted FEC' since it adjusts S_{IF}, S_{PF} and S_{BF} with a fixed GOP, GOP(4,2). Schema D adjusts all of N_P, N_{BP}, S_{IF},S_{PF} and S_{BF}.

A few reasonable constraints were imposed to reduce the time required to search the variable space. The maximum frame rate allowed is 30 frames per second, typical of full-motion video. The maximum size in frames of the GOP is 15, as recommended in [9], and the maximum number of B frames in the I-P frame intervals is three.

5. ANALYSIS

This section presents and analyzes some of the experimental results.[2] The goal is to investigate the performance benefits that can be derived from adjustments to GOP and FEC for different network packet loss rates.

5.1. Benefits from Adjusting GOP and Adjusting FEC

Figure 1 compares the playable frame rates for the four schemas for a range of packet loss percentages. Adjusting the GOP does not significantly improve the playable frame rate. When the packet loss percentage is higher than 6% the best possible GOP pattern yields an absolute playable frame rate below one frame per second (fps). On the other hand, the benefit from Adjusted FEC is very high. Adjusting FEC with increased packet loss manages to keep the playable frame rate above 4 fps. If the network can keep the packet loss percentage below 5%, adjusting FEC will provide a playable frame rate higher than 10 fps. The Adjusted GOP curves are very close to the FEC curves, suggesting that adjusting the GOP patterns (as in [3, 17]) yields insignificant improvement. This suggests that adjusting FEC is the better way to maximize the playable video frame rates in a lossy network.

[2] More analysis can be found in [18].

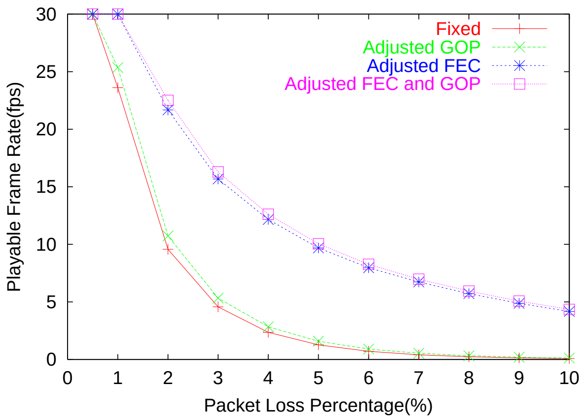

Fig. 1. Playable frame rates

5.2. GOP Behavior

5.2.1. Adjusting GOP

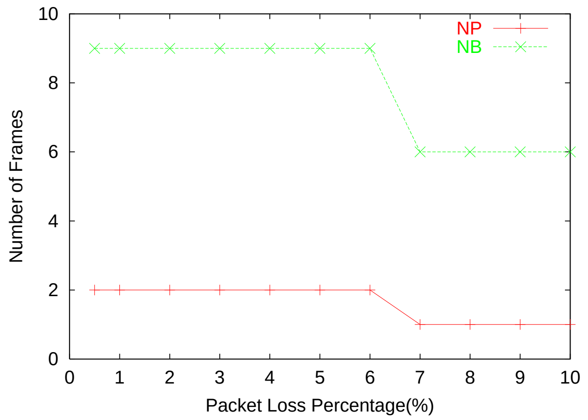

Fig. 2. Adjusted GOP

Figure 2 captures the limited movement in the choice of the optimal GOP pattern in schema B, the Adjusted GOP, as p increases. While N_P does decrease once the loss rate goes above 7%, the optimal N_{BP} is always three for all loss percentages.

In schema D, when both the FEC and GOP space are searched, GOP(2,3) ('IBBBPBBBPBBB') is the best choice regardless of the expected packet loss percentage. A change in loss results in only a change in the FEC pattern. Hence, it might be more accurate to label this schema as 'Adjusted FEC, Optimal GOP'.

5.2.2. Fixed GOP

In the prior discussion the initial Fixed GOP choice of GOP(4,2) was so well-chosen that there was no advantage in trying to adjust GOP to loss expectations. However, this not necessarily true for every choice of GOP.

To investigate the impact of different GOP choices, we fixed the packet loss percentage at 5% and used our model to compute the playable frame rate R, for possible fixed GOPs. The results indicate that there can be a significant performance difference caused by the GOP pattern. For examples, with no FEC, GOP(2,3) gets almost twice the playable frame rate of GOP(0,0) (a GOP with only I frames). With Adjusted FEC, the frame rate difference is even larger, with GOP(2,3) getting more than three times the frame rate of GOP(0,0). Thus, it is important to select a good GOP, especially when using FEC.

Knowing these trends, it makes sense to investigate the behavior of FEC by starting with an optimal GOP choice. Although GOP(2,3) is the optimal GOP, we use GOP(4,2) in next section because its performance is very close to that of GOP(2,3) and it is a very commonly used pattern on the Internet [1].

5.3. FEC Behavior

5.3.1. Adjusting FEC

Figure 3 depicts the breakdown of how redundant FEC packets should be adjusted for I, P, and B frames to yield the optimal playable frame rate as the packet loss percentage varies.

In optimally adjusted FEC, the I frames always receive more FEC packets than the P frames. Similarly, the P frames always receive more FEC than the B frames. This fits intuition, since this allocation represents both the relative importance and relative sizes of I, P, and B frames respectively.

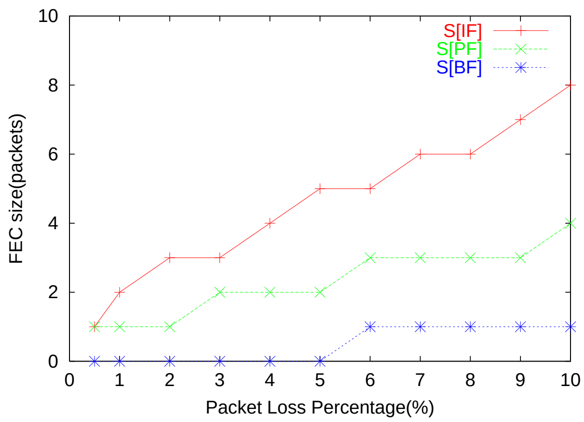

Fig. 3. Adjusted FEC pattern

5.3.2. *Fixed FEC vs. Adjusted FEC*

This section compares the Fixed FEC patterns and the Adjusted FEC. Three fixed FEC choices are taken from Figure 3 as Small, Medium and Large FEC, which are the adjusted FEC for 0.5% loss, 5% loss and 10% respectively. The playable frame rate of Fixed FEC and Adjusted FEC are compared. The result shows adjusted FEC for a given loss probability adds the minimum redundancy needed to achieve the optimum frame rate. The small Fixed FEC adds a low overhead to the MPEG stream and works well with a low packet loss rate, but at higher loss rates, the playable frame rate decreases dramatically since the small Fixed FEC cannot provide enough protection. In contrast, the large Fixed FEC gives adequate protection in the presence of high loss rates, but also introduces a large amount of unnecessary overhead when the loss rates are low.

6. SUMMARY

This paper presents an analytic model that permits investigation of the impact of adjusting Group of Pictures (GOP) and Forward Error Correction (FEC) on the playable frame rate of TCP-Friendly MPEG video streams. We use our model to search the system state space for a range of network conditions to optimize the playable frame rate for a given loss probability.

We find adjusting GOP and FEC provides the best playable frame rate. The benefit to playable frame rate from adjusting GOP is slight for typical GOPs, while the benefit to playable frame rate from adjusting FEC is large, especially for higher loss rates. However, the choice of a good GOP is still important. Adjusting FEC provides significant benefit to the playable frame rate with quite low overhead.

7. REFERENCES

[1] S. Acharya and B. Smith. An Experiment to Characterize Videos Stored on the Web. In *Proceedings of MMCN*, January 1998.

[2] J.-C. Bolot, S. Fosse-Parisis, and D. Towsley. Adaptive FEC-Based Error Control for Internet Telephony. In *Proceedings of IEEE INFOCOM*, Mar. 1999.

[3] J. Chung and M. Claypool. Better-Behaved, Better-Performing Multimedia Networking. In *Proceedings of SCS Euromedia*, May 2000.

[4] S. Floyd and K. Fall. Promoting the Use of End-to-End Congestion Control in the Internet. *IEEE/ACM Transactions on Networking*, Feb. 1999.

[5] S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-Based Congestion Control for Unicast Applications. In *Proceedings of ACM SIGCOMM Conference*, pages 45 – 58, 2000.

[6] I.S.Reed and G.Solomon. Polynomial Codes Over Certain Finite Fields. *Journal of the Society of Industrial and Applied Mathematics (SIAM)*, 8(2):300–304, June 1960.

[7] Y. Liu and M. Claypool. Using Redundancy to Repair Video Damaged by Network Data Loss. In *Proceedings of IS&T/SPIE/ACM Multimedia Computing and Networking (MMCN)*, Jan. 2000.

[8] K. Mayer-Patel, L. Le, and G. Carle. An MPEG Performance Model and Its Application To Adaptive Forward Error Correction. In *Proceedings of ACM Multimedia*, December 2002.

[9] J. Mitchell and W. Pennebaker. *MPEG Video: Compression Standard*. Chapman and Hall, 1996. ISBN 0412087715.

[10] M.Krunz, R.Sass, and H.Hughes. Statistical Characteristics and Multiplexing of MPEG Streams. In *Proceedings of IEEE Infocom*, pages 455–462, April 1995.

[11] C. Padhye, K. Christensen, and W. Moreno. A New Adaptive FEC Loss Control Algorithm for Voice Over IP Applications. In *Proceedings of IEEE International Performance, Computing and Communication Conference*, Feb. 2000.

[12] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Throughput: A Simple Model and Its Empirical Validation. In *Proceedings of ACM SIGCOMM*, 1998.

[13] K. Park and W. Wang. QoS-Sensitive Transport of Real-Time MPEG Video Using Adaptive Forward Error Correction. In *Proceedings of IEEE Multimedia Systems*, pages 426 – 432, June 1999.

[14] R. Rejaie, M. Handley, and D. Estrin. RAP: An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet. In *Proceedings of IEEE Infocom*, 1999.

[15] I. Rhee, V. Ozdemir, and Y. Yi. TEAR: TCP Emulation at Receivers – Flow Control for Multimedia Streaming. Technical report, Department of Computer Science, NCSU, 2000.

[16] T.Nguyen and A.Zakhor. Distributed Video Streaming with Forward Error Correction. In *12th International Packet Video Workshop*, April 2002.

[17] J. Walpole, R. Koster, S. Cen, C. Cowan, D. Maier, D. McNamee, C. Pu, D. Steere, and L. Yu. A Player for Adaptive MPEG Video Streaming Over The Internet. In *Proceedings of the SPIE Applied Imagery Pattern Recognition Workshop*, Oct. 1997.

[18] H. Wu, M. Claypool, and R. Kinicki. Adjusting Forward Error Correction for TCP-Friendly Streaming MPEG. Technical Report WPI-CS-TR-02-29, Worcester Polytechnic Institute, 2002.