Adjusting Forward Error Correction with Quality Scaling for Streaming MPEG

Huahui Wu, Mark Claypool, and Robert Kinicki
Computer Science Department
Worcester Polytechnic Institute
100 Institute Rd, Worcester, MA 01609, USA
{flashine|claypool|rek}@cs.wpi.edu

ABSTRACT

Packet loss can severely impact streaming video quality. Repair techniques protect streaming video from packet loss but at the price of a reduced effective transmission rate when streaming a flow in a capacity constrained situation. This paper proposes an algorithm that optimizes the choice of Forward Error Correction (FEC) to repair packet loss for streaming MPEG videos under a capacity constraint with quality scaling. An analytic model is developed to estimate the video quality of streaming MPEG given a quality scaling level and a specific FEC strength. Given network conditions in terms of packet loss rate, the model searches the total variable space to find the combination of FEC and scaling that yields the optimal quality under the capacity constraint. Analysis over a range of network conditions indicates that adjusting FEC with quality scaling provides significant performance improvement.

Categories and Subject Descriptors: C.2.m Computer-Communication Networks: Miscellaneous

General Terms: Performance, Design

Keywords: Streaming MPEG, Quality Scaling, Forward Error Correction

1. INTRODUCTION

The growth in the power and connectivity of the Internet has sparked an even larger growth in streaming media. Although many compression techniques have been introduced to reduce the streaming bitrate [11], video streaming over the Internet is often still limited by the network capacity. For example, one capacity constraint comes from the Internet Service Provider's (ISP) negotiated rate that restricts capacity on the home user's last mile. Another capacity constraint comes from the growing consensus that all Internet traffic must be TCP-Friendly. A flow is *TCP-Friendly* if its data rate does not exceed the maximum data rate of a conformant TCP connection under equivalent network conditions. There are proposed approaches to detect and restrict the capacity of non-TCP friendly flows [9]. Streaming media applications that ignore these constraints suffer data loss in the form of dropped packets.

While multimedia applications can tolerate some data loss, excessive packet loss yields unacceptably low quality. Since video encoding relies upon frame dependencies to achieve high compression rates, the random dropping of packets by routers can seriously degrade video quality. For example, as little as 3% MPEG packet loss can cause 30% of the frames to be undecodable.

To avoid the latency and variance in latency caused by retransmission of lost packets, streaming media flows often use Forward Error Correction (FEC) to reconstruct lost streaming media packets. However, FEC adds redundant repair data to the original video stream. Current approaches use either apriori, static FEC choices or FEC that adapts to perceived packet loss on the network. However none of the published adaptive FEC schemes account for the additional FEC overhead against a capacity constraint. With a capacity constraint, adding FEC reduces the effective transmission rate of the original video content [10].

To preserve real-time playout, multimedia servers use media scaling to reduce the streaming data rate to match the capacity constraint. With quality scaling, the multimedia server adjusts the quantization level before transmission to reduce the streaming bitrate. When used in conjunction with FEC, a multimedia application can increase the quantization level even higher to save capacity for the FEC overhead. Hence, selecting the appropriate amount of FEC and the corresponding quantization level can be cast as a constrained optimization problem that attempts to optimize the video streaming quality.

Previously, we derived a model of playable frame rate for streaming MPEG [21] that did not account for media scaling. This paper extends the analytic model to characterize quality-scaled MPEG video performance with FEC in the presence of packet loss. For a given loss rate and MPEG characteristics, the new model utilizes a specified quality scaling level and number of FEC packets for each MPEG frame type to estimate video distortion. Based on the model, the optimization algorithm exhaustively searches all possible combinations of FEC and scaling level to determine the configuration that yields the best capacity-constrained video quality. Experimental results show that adjusting FEC with quality scaling (QAFEC) provides significant benefits to video quality.

NOSSDAV'05, June 13–14, 2005, Stevenson, Washington, USA.

The remainder of the paper is organized as follows: Section 2 provides background for the work in this paper; Section 3 introduces the analytical quality model; Section 4 presents the optimization algorithm; Section 5 discusses experiments, including a preliminary user study, and analyzes the results; and Section 6 summarizes the paper and presents possible future work.

2. BACKGROUND

2.1 MPEG and Quality Scaling

There are four frame types defined in MPEG-1 [11]: I (intra-coded) frames, P (predictive-coded) frames, B (bi-directionally predictive-coded) frames and D (DC-coefficients only) frames. However, the seldom-used D frame is excluded from our model of MPEG. MPEG video typically repeats a pattern of I, P, and B frames (known as a Group of Pictures or GOP) for the duration of a video stream. Figure 1 shows the typical GOP used in this paper.

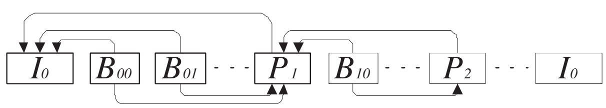


Figure 1: A typical MPEG Group Of Pictures

In MPEG, the DCT coefficient of the video signal is quantized by dividing by an integer (the *quantization value*), and rounding to the nearest integer. When using higher quantization values, each MPEG frame is encoded with lower precision, and transmitted with fewer bits. Thus, this quality scaling technique reduces the bitrate of the streaming video.

Proposed mechanisms for quality scaling include: adaptive quantization values [20], which adapts the encoding quantization value to network capacity; signal-to-noise ratio (SNR) scalability [16], which encodes a video clip into multiple layers and streams as many layers as possible; MPEG-4 fine granularity scalability (FGS) [7], which is a special case of SNR scalability that provides continuous scalability using partial enhancement; and scalable MPEG (SPEG) [6], which transcodes MPEG's DCT coefficients to a base level plus three enhanced levels and transmits different numbers of levels.

While this report focuses on using adaptive quantization values, the model developed (Section 3) is independent of the scaling technique and only requires the relationships between scaling level, encoding bitrate and video quality.

2.2 Forward Error Correction(FEC)

Reed-Solomon (R-S) code [15] is a media-independent FEC technique that can be applied at the packet level. As shown in Figure 2, an application level video frame is modeled as being transmitted in K packets where K varies with frame type, encoding method, and media content. R-S code adds $(N-K)$ redundant packets to the K original packets and sends the N packets over the network. Although some packets may be lost, e.g. packet 2 in Figure 2, the frame still can be completely reconstructed if any K or more packets are successfully received.

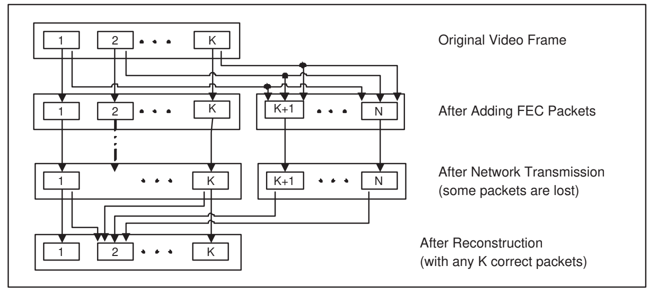


Figure 2: Reed-Solomon code

To analyze the effects of FEC on application layer frames, the sending of packets is modeled as a series of independent Bernoulli trials. Thus, the probability $q(N,K,p)$ that a K-packet video frame is successfully transmitted with $N-K$ redundant FEC packets along a network path with packet loss probability p is:

$$q(N,K,p) = \sum_{i=K}^{N} \left[\binom{N}{i} (1-p)^i * p^{N-i} \right] \quad (1)$$

3. MODEL

3.1 Parameters and Variables

System Layer	Parameters
Application	$S_I, S_P, S_B, N_P, N_B, R_F$
Presentation	$S_{IF}, S_{PF}, S_{BF}, l$
Network	p, t_{RTT}, s, T

Table 1: System Layers and Parameters

The system layers and parameters for our analytic model are indicated in Table 1, where the parameters are:

R_F: the maximum playable frame rate achieved when there is enough capacity and no loss (While typical frame rates are 25 fps (PAL) or 29.97 fps (NTSC), this paper uses the NTSC frame rate and rounds it up to $R_F = 30fps$).

S_I, S_P, S_B: the size of I, P or B frames respectively, in fixed size packets.

N_P, N_B: the number of P or B frames in one GOP, respectively.

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

l the quality scaling level, assuming I, P and B frames have the same level.[1]

s: the packet size (in bytes).

p: the packet loss probability.

t_{RTT}: the round-trip time (in milliseconds).

T: the capacity constraint, limited by the ISP or by a TCP-Friendly rate [13].

3.2 Distortion from Quality Scaling

When a video is streamed over an unreliable network under a capacity constraint, its perceptual quality is degraded

[1]While using different quality levels for different frame types is possible, capturing the quality dependencies between different frames is difficult. This remains as future work.

by two factors: quality scaling and frame loss. Scaling distortion, caused by a high quantization value, appears visually as coarse granularity in every frame. Frame loss, due to network packet loss, yields jerkiness in the video playout.

This paper uses the Video Quality Model (VQM) [14], an objective video quality measurement, to approximate the distortion due to quality scaling. Section 3.3 uses playable frame rate to estimate the distortion from frame loss. Section 3.4 presents a new measurement, namely, distorted playable frame rate that combines these two factors.

VQM takes an original video and a distorted video as input and returns a distortion value D between 0 (no distortion) and 1 (maximum distortion). Previous research [3] implies that perceptual video distortion varies exponentially with the quantization value. Employing VQM to measure D in videos encoded with varying quantization levels, our preliminary studies show that D can be approximated as an exponential function of the quantization value l:

$$D = \hat{D} \cdot l^{\lambda_D} \tag{2}$$

where l is the quantization level, $\hat{D}$ is the VQM distortion when $l = 1$, and λ_D is the exponential coefficient. Table 3 in Section 5.2 provides an example that shows how accurately this function fits real video data.

3.3 Playable Frame Rate

3.3.1 Frame Size

Frame sizes change with the quality scaling level. Previous researches [3, 19] demonstrate that MPEG streaming bitrate can be approximated by an exponential function of the quality scaling level l. Our preliminary experiments suggest frame size can be estimated by exponential functions of quantization value l given as:

$$\begin{cases} S_I = \hat{S_I} \cdot l^{\lambda_I} \\ S_P = \hat{S_P} \cdot l^{\lambda_P} \\ S_B = \hat{S_B} \cdot l^{\lambda_B} \end{cases} \tag{3}$$

where l is the quantization level, $\hat{S_*}$ is the frame size when $l = 1$, and λ_* is the exponential coefficient. Note, all the results S_* are rounded up to the nearest integer $\lceil S_* \rceil$ since video frames must be sent over the network in an integer number of packets. Table 3 in Section 5.2 shows how accurately these functions fit real video frame sizes.

3.3.2 Successful Frame Transmission Probability

Given I, P, and B frame sizes, and the distribution of redundant FEC packets added to each frame type, Equation 1 provides the probability of successful transmission for each frame type, knowing the amount of redundancy added by media-independent FEC:

$$\begin{cases} q_I = q(S_I + S_{IF}, S_I, p) \\ q_P = q(S_P + S_{PF}, S_P, p) \\ q_B = q(S_B + S_{BF}, S_B, p) \end{cases} \tag{4}$$

3.3.3 Playable Frame Rate

Our previous work [21] derived a model to estimate total playable frame rate for streaming MPEG:

$$\begin{aligned} R = G \cdot q_I \cdot (1 + \tfrac{q_P - q_P^{N_P+1}}{1-q_P} &+ N_{BP} \cdot q_B \\ &\cdot (\tfrac{q_P - q_P^{N_P+1}}{1-q_P} + q_I \cdot q_P^{N_P})) \end{aligned} \tag{5}$$

where G is the constant GOP rate, N_P is the number of P frames, N_{BP} is the number of B frames between two reference frames (I or P frame), and q_* is the successful transmission probability of I, P or B frames determined from Equation 4.

Treating I, P and B frame sizes as functions of the quality scaling level l given in Equation 3, the model can be extended to estimate the playable frame rate and subsequently used to estimate the quality distortion due to frame loss.

3.4 Distorted Playable Frame Rate

Quality scaling uses a higher quantization value to encode the video, causing intra-frame quality distortion. Frame loss lowers the playable frame rate and is referred to as inter-frame quality distortion.

Since the inter-frame and intra-frame distortion components are independent, it is assumed they contribute independently to the overall distortion. Hence, quality distortion can be represented by a function of these two factors. To stream the highest quality video possible, the media server needs to use the best quality scaling level and the media client needs to receive all the frames. Thus, these two factors are combined into a multiplicative function, referred to as the distorted frame rate, R_D,:

$$R_D = (1 - D) \cdot R \tag{6}$$

where D is the quality distortion from Equation 2 and R is the playable frame rate from Equation 5.

The motivation behind R_D is as follows. If a video is streamed with the best quantization value, its quality scaling distortion is 0 and video quality is determined only by the playable frame rate R. With any other quantization value, every frame carries less visual detail and its contribution to the video quality (measured by frame rate) is reduced by the quality distortion D. A preliminary user study (shown in Section 5.3) shows a correlation between user perceptual quality and distorted playable frame rate R_D. This suggests that R_D may be a reasonable representation of overall video quality.

4. OPTIMIZATION ALGORITHM

For given network conditions and MPEG video parameters, the total distorted playable frame rate R_D varies with quality scaling level and the amount of FEC for each frame type as a function $R_D(l, (S_{IF}, S_{PF}, S_{BF}))$ where streaming bitrate is limited by a capacity constraint, T. Thus, this model can optimize the distorted playable frame rate, R_D, using the following operation research equation:

$$\begin{cases} Maximize: \\ \quad R_D = (1 - D(l)) \cdot R(l, (S_{IF}, S_{PF}, S_{BF})) \\ Subject\ to: \\ \quad G \cdot ((S_I(l) + S_{IF}) + N_P \cdot (S_P(l) + S_{PF}) \\ \qquad + N_B \cdot (S_B(l) + S_{BF})) \leq T \end{cases} \tag{7}$$

Unfortunately, finding a closed form solution for the non-linear function R_D is difficult since there are many saddle points. However, given that the optimization problem is expressed in terms of integer variables over a restricted domain, an exhaustive search of the discrete space is feasible. With fixed input values for (p, RTT, s), (G, N_P, N_B) and functions of $(S_I(l)$,$S_P(l)$,$S_B(l))$, each set of values of l and

(S_{IF}, S_{PF}, S_{BF}) determine the distorted playable frame rate R_D using the following steps:

1. Approximate the video frame sizes (S_I,S_P and S_B) using l in Equation 3.
2. Estimate total video streaming bitrate using the video frame sizes and the FEC frame sizes. If the estimated bitrate is larger than the capacity constraint T, the set of values of l and (S_{IF}, S_{PF}, S_{BF}) are invalid and R_D is returned as 0.
3. Otherwise, use the video frame sizes and the FEC sizes to determine the successful transmission probabilities (q_I, q_P and q_B) from Equation 4.
4. Estimate playable frame rate R by inputting (q_I, q_P and q_B) into Equation 5.
5. Use l in Equation 2 to approximate D.
6. Employ R and D in Equation 6 to estimate the distorted playable frame rate, R_D.

With these steps for each set of values, the space of possible values for l and (S_{IF}, S_{PF}, S_{BF}) is exhaustively searched to determine the quality level and the amount of FEC packets for each frame type that yields the maximum distorted playable frame rate under the capacity constraint. Since the search can be done in real-time [2], the determination of optimal choices for adaptive FEC and quality scaling is feasible for streaming MPEG.

While the quality adjusted FEC (QAFEC) model assumes MPEG frame sizes are constant for the entire video, in practice MPEG frame sizes change from one GOP to the next. To assess the impact of changing frame sizes on the model, we used the QAFEC model to determined the adjusted FEC and scaling assuming an average fixed frame size. Then, a simulation using trace-based frame sizes [18] was run and actual playable frame rate at the receiver was measured. The frame rate predicted by the model was slightly higher than the actual frame rate achieved. However, the difference was negligible for loss rates around 1% and was less than 10% for higher loss rates. Details can be found in [22].

5. EXPERIMENTS

5.1 Methodology

Using the optimization algorithm, the distorted playable frame rates over a range of network and application settings are explored. For each set of network and application parameters, the playable frame rates are compared for MPEG streaming with quality adjusted FEC, MPEG streaming with two types of fixed FEC, and MPEG streaming without FEC:

1. Quality Adjusted FEC: Before transmission, the server employs the optimization algorithm based on the QAFEC model to determine the FEC and quality scaling levels that produce the maximum distorted playable frame rate R_D and uses these for the entire video transmission.
2. Large Fixed FEC: The server protects each frame with 15% FEC packets (moved up to the nearest integer). This FEC pattern provides strong protection to each frame and roughly represents the relative importance of the I, P and B frames [4, 8].
3. Small Fixed FEC: Each I frame receives 1 FEC packet. This simple FEC pattern protects the most important frame, the I frame. Protecting the I frame is a scheme used by other researchers [2, 17].
4. Non-FEC: No FEC is added to the video.

[2]It takes about 100 milliseconds to find the best FEC and scaling pattern using our approach on a Pentium-3 800 MHz PC. Optimizations of the code and a faster machine will allow searching to be done even faster.

The total bitrate used by the MPEG video and FEC is scaled to meet a TCP-Friendly capacity constraint [13] using quality scaling. While other published approaches dynamically adapt FEC to current packet loss conditions, the resultant total bitrates do not satisfy a capacity constraint. Thus, there is no simple way to compare QAFEC to other dynamic FEC approaches.

5.2 System Settings

Network Layer		Application Layer	
t_{RTT}	50 ms	N_P	4 frames per GOP
s	1 Kbyte	N_B	10 frames per GOP
p	0.01 to 0.04	R_F	30 frames per second

Table 2: System Parameter Settings

Table 2 presents the system parameter settings for the network and application layers. A commonly-used MPEG GOP pattern, '`IBBPBBPBBPBBPBB`', and a typical full motion frame rate R_F of 30 frames per second (fps) are used. The packet size s, round-trip time t_{RTT} and packet loss probability p are chosen based on the characteristics of many network connections [1, 5]. For all experiments, the parameters are fixed, except for packet loss probability p, which was varied from 0.01 to 0.04 in steps of 0.002.

Two picture sequences are used. The first video, *Paris*, from PictureTel, shows two people sitting at a table and talking while making high-motion gestures. It has 900 raw images and lasts for 30 seconds, providing a frame rate of 30 fps. The image size is 352x288 pixels (CIF). The Berkeley MPEG encoder *mpeg_encode* [12] is used to encode the images with different quantization values as one-minute long videos. From the output videos, the frame sizes are extracted with the Berkeley MPEG statistics tool *mpeg_stat* [12] and the quality distortion is extracted with VQM. Statistical analysis software *SPSS* [3] is used to fit the relation between quality distortion and l to a function as in Equation 2, and the relation between frame sizes and l in Equation 3. The equations then become:

$$\begin{cases} D = 0.025 \cdot l^{0.87} \\ S_I = 81.51 \cdot l^{-0.70} \\ S_P = 52.94 \cdot l^{-1.21} \\ S_B = 15.47 \cdot l^{-0.79} \end{cases} \quad (8)$$

Table 3 shows how these analytical functions fit the Paris data with some representative quantization values. In the table, l is the quantization value, D', S'_I, S'_P, S'_B are estimated by the analytical functions, and D, S_I, S_P, S_B are

[3]http://www.spss.com

the real values from the video analysis. Overall, the functions fit the data well.

l	D'	D	S'_I	S_I	S'_P	S_P	S'_B	S_B
5	0.10	0.09	26.4	26.5	7.5	7.8	4.3	4.5
8	0.15	0.15	19.0	19.5	4.2	4.6	2.9	2.8
12	0.21	0.22	14.3	14.5	2.6	2.8	2.1	2.0
18	0.31	0.33	10.7	10.7	1.6	1.6	1.5	1.4
24	0.39	0.38	8.8	8.7	1.1	1.1	1.2	1.2
31	0.49	0.48	7.3	7.2	0.8	0.7	1.0	1.1

Table 3: Estimated Value by Equation 8 versus Real Video Data

The second video sequence is *Tennis*, which comes with the Berkeley tools. *Tennis*, a short clip of two men playing ping-pong, has 150 raw images and lasts for 5 seconds, providing a frame rate of 30 fps. The size of each frame is 352x240 pixels. Again, the Berkeley MPEG tools, VQM and SPSS are used to approximate the quality distortion and frame sizes to functions of quality scaling level:

$$\begin{cases} D = 0.041 \cdot l^{0.69} \\ S_I = 74.55 \cdot l^{-0.86} \\ S_P = 96.22 \cdot l^{-1.31} \\ S_B = 33.27 \cdot l^{-1.01} \end{cases} \tag{9}$$

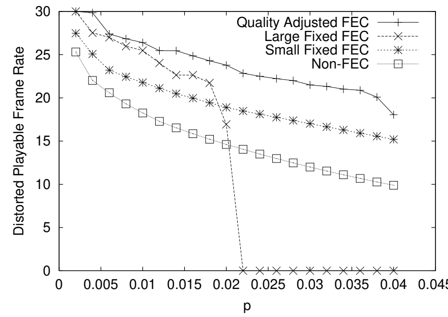
Due to space constraints, the functional fit for *Tennis* is not shown, but the approximation errors are less than 5%.

Note, analyzing videos and fitting the results to these exponential functions is a time-intensive operation. It may be possible to analyze a large variety of video types and find parameters for the exponential functions, based either on the frames sizes, frame rates and video content, that are generally effective. This analysis is left as future work.

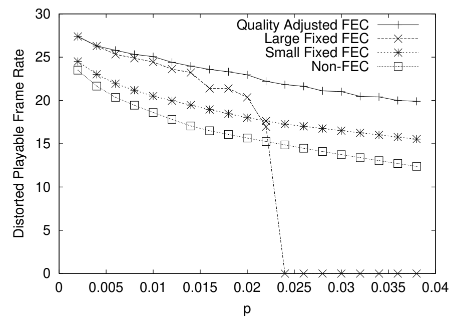
5.3 Analysis of Results

The *Paris* and *Tennis* videos are used to approximate their quality scaling distortion D and frame sizes to functions as Equation 2 and Equation 3 and instantiate them to Equation 8 and Equation 9. These functions and the optimization algorithm are used to search the FEC and quality scaling values to produce the maximum distorted playable frame rate for the four approaches in Section 5.1.

Figure 3 graphs the distorted playable frame rates for the four FEC choices for the *Paris* and *Tennis* videos. The x-axes are the packet loss probabilities, and the y-axes are the distorted playable frame rates. From the data in these figures, quality adjusted FEC provides the best quality under all network and video conditions. The benefits of quality adjusted FEC over non-FEC are substantial, with quality adjusted FEC providing 5-10 more frames per second for all rates. The small fixed FEC approaches usually improves playable frame rates over non-FEC video, especially when loss rates are high. However, the small fixed FEC frame rates are still much lower than the frame rates with quality adjusted FEC. Large FEC achieves the playable frame rate provided by quality adjusted FEC for low loss rates because the TCP-Friendly rate is relatively high. With limited capacity (at high loss rates), the quality scaling level is high (> 16), and the large FEC overhead becomes significant. There is little reduction in the frame sizes as the scaling level increases (to a maximum of 31) and quality scaling is



a. Paris Video



b. Tennis Video

Figure 3: Distorted Playable Frame Rates

unable to conserve enough bitrate to overcome the effect of the large FEC overhead. The result is that none of the original video data is sent. These trends hold for both videos despite differences in the content between the two clips.

A preliminary user study was conducted to compare the impact of FEC choices on actual users and begin to ascertain the efficacy of the distorted playable frame rate measure, R_D, as a measure of perceived visual quality. Four versions of the *Paris* video clips were generated, simulating the Quality Adjusted FEC approach, the two fixed FEC strategies and the Non-FEC approach on a network with 0.02 packet loss and a TCP-Friendly constraint of 1.17 Mbps. No local concealment technique is used; if a frame was not playable, the preceding playable frame was repeated. Ten undergraduate students were asked to rate the quality from 0 (worst) to 10 (best) of the four videos using the original videos with no packet loss as a reference. The ratings were provided after viewing each clip twice in a different random order by each student.

Repair method	D	R	R_D	Q
Quality Adjusted FEC	0.17	28.55	23.78	6.89
Small Fixed FEC	0.20	23.58	18.90	4.44
Large Fixed FEC	0.44	30.00	16.93	3.89
Non-FEC	0.28	20.17	14.61	3.50

Table 4: Preliminary User Study

Table 4 displays the average user quality rating Q compared with the VQM distortion D, the playable frame rate R, and the distorted playable frame rate R_D for the four videos. The results indicate that the perceived qualities of the videos with FEC are significantly higher than the videos without FEC. Additionally, QAFEC videos appear noticeably better under all the conditions than videos with fixed FEC. Perhaps most importantly, the correlation between Q and R_D suggests that the distorted playable frame rate R_D appropriately represents perceived quality, accounting for both the temporal aspects of the video that influence perceived quality (R) and the quality aspects of the video that influence perceived quality (D).

It may be surprising that non-FEC has a higher VQM distortion D since non-FEC has a higher encoding bitrate without any FEC overhead. The reason is, if non-FEC chooses a better (lower) quantization level l under the capacity constraint, it will increase the frame sizes, reducing the successful transmission probability for each frame and getting a much lower playable frame rate. So, non-FEC selects a conservative quantization level to get better overall quality.

6. SUMMARY

This investigation studies adjusting FEC with quality scaling for streaming MPEG. An analytic model (QAFEC) is proposed that captures the quality distortion of streaming MPEG in the presence of quality scaling and frame loss. Using this model, an optimization algorithm determines the optimal adjustment of FEC and quality scaling under a capacity bound, accounting for both the network conditions and video parameters.

The analytic experiments show that QAFEC has significant advantages. Quality adjusted FEC always achieves higher quality than MPEG video without FEC and provides higher video quality than fixed FEC approaches when taken over a range of MPEG encoding and network conditions. A preliminary user study confirms the experiments in demonstrating that the perceived quality of the videos with quality adjusting FEC are significantly better than the videos with fixed FEC or without FEC. The user study also suggests that the proposed distorted playable frame rate, R_D, has the appropriate trend in capturing the distortion to video quality from quality scaling and frame loss.

Our ongoing work includes: implementing a streaming MPEG system that includes the model and the optimization algorithm; adding other quality scaling techniques, such as MPEG FGS to the model; and extending the model to compare and combine quality scaling and temporal scaling.

7. REFERENCES

[1] J. Chung, M. Claypool, and Y. Zhu. Measurement of the Congestion Responsiveness of RealPlayer Streaming Video Over UDP. In *Proceedings of the Packet Video Workshop (PV)*, Nantes, France, Apr. 2003.

[2] N. Feamster and H. Balakrishnan. Packet Loss Recovery for Streaming Video. In *Proceedings of 12th International Packet Video (PV) Workshop*, Pittsburgh, PA, USA, Apr. 2002.

[3] P. Frossard and O. Verscheure. Joint Source/FEC Rate Selection for Quality-Optimal MPEG-2 Video Delivery. *IEEE Transactions on Image Processing*, 10(12):1815–1825, Dec. 2001.

[4] F. Hartanto and H. R. Sirisena. Hybrid Error Control Mechanism for Video Transmission in the Wireless IP Networks. In *Proceedings of LANMAN*, Nov. 1999.

[5] S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, and D. Towsley. Inferring TCP Connection Characteristics Through Passive Measurements. In *Proceedings of IEEE Infocom*, Hongkong, China, Apr. 2004.

[6] C. Krasic, J. Walpole, and W.-C. Feng. Quality-adaptive Media Streaming by Priority Drop. In *Proceedings of NOSSDAV*, pages 112–121, Monterey, CA, USA, June 2003. ACM Press.

[7] W. Li. Overview of Fine Granularity Scalability in MPEG-4 Video Standard. *IEEE Transaction on Circuits and System for Video Technology*, 11(3), Mar. 2001.

[8] Y. Liu and M. Claypool. Using Redundancy to Repair Video Damaged by Network Data Loss. In *Proceedings of ACM MMCN*, Jan. 2000.

[9] R. Mahajan, S. Floyd, and D. Wetherall. Controlling High-Bandwidth Flows at the Congested Router. In *Proceedings of the 9th International Conference on Network Protocols (ICNP)*, Nov. 2001.

[10] 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.

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

[12] U. of California Berkeley. Berkeley MPEG-2 Encoder and Player. Internent site http://bmrc.berkeley.edu/frame/research/mpeg/.

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

[14] M. Pinson and S. Wolf. A New Standardized Method for Objectively Measuring Video Quality. *IEEE Transactions on Broadcasting*, 50(3):312–322, Sept. 2004.

[15] 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.

[16] R. Rejaie, M. Handley, and D. Estrin. Quality Adaptation for Congestion Controlled Video Playback overthe Internet. In *Proceedings of SIGCOMM*, pages 189–200, Cambridge, MA, Aug. 1999.

[17] I. Rhee. Error Control Techniques for Interactive Low-Bit Rate Video Transmission over the Internet. In *Proceedings of ACM SIGCOMM Conference*, Sept. 1998.

[18] O. Rose. Statistical Properties of MPEG Video Traffic and their Impact on Traffic Modeling in ATM Systems. In *Proceedings of 20th Conference of Local Computer Networks*, pages 397–406, Oct 1995.

[19] S. Sakazawa, Y. Takishima, M. Wada, and Y. Hatori. Coding Control Scheme for a Multi-Encoder System. In *Proceedings of 7th International Workshop on Packet Video (PV)*, Mar. 1996.

[20] W.-T. Tan and A. Zakhor. Real-Time Internet Video Using Error Resilient Scalable Compression and TCP-Friendly Transport Protocol. *IEEE Transactions on Multimedia*, 1(2):172–186, 1999.

[21] H. Wu, M. Claypool, and R. Kinicki. A Model for MPEG with Forward Error Correction and TCP-Friendly Bandwidth. In *Proceedings of NOSSDAV*, Monterey, CA, USA, June 2003.

[22] H. Wu, M. Claypool, and R. Kinicki. Adjusting Forward Error Correction with Temporal Scaling for TCP-Friendly Streaming MPEG. Technical Report WPI-CS-TR-03-10, CS Department, Worcester Polytechnic Institute, Apr. 2003.