Adaptive Bitstream Switching of Pre-encoded PFGS Video

Osama Lotfallah
Dept. of Computer Science and Engineering, Arizona State University
Tempe, AZ 85287, USA
Oslatif@asu.edu

Martin Reisslein
Dept. of Electrical Engineering, Arizona State University
Tempe, AZ 85287, USA
Reisslein@asu.edu

Sethuraman Panchanathan
Dept. of Computer Science and Engineering, Arizona State University
Tempe, AZ 85287, USA
Panch@asu.edu

ABSTRACT
With Progressive Fine Granularity Scalability (PFGS) video coding, one given encoding (with a prescribed bit rate) can flexibly be transmitted at any lower bit rate. However, the transmitted video is only efficiently encoded when the transmission bit rate is in the vicinity of the encoding bit rate; for transmission bit rates far from the encoding bit rate up on the order of 4 dB in video quality are lost. In this paper we develop and evaluate a suite of policies for accounting for this coding efficiency issue, which has been largely overlooked in previous PFGS streaming studies, in uni- and multicast streaming. Our adaptive policies select the PFGS encoding rate from a small number of pre-encoded versions and drop packets so as to maximize the reconstructed video qualities. Our policies consider both the visual video content, expressed using the motion activity level of MPEG-7 descriptor, as well as the channel variability. We find that an optimal non-adaptive streaming policy overcomes the 4 dB inefficiency and on top of this efficiency gain, our adaptive unicast streaming policy achieves 0.8 dB improvement over the optimal non-adaptive streaming. We also find that our content-dependent packet drop policies enforce fairness among multiple streams in terms of reconstructed video qualities and that our multicasting policy improves the average reconstructed video qualities at a group of receivers by up to 2 dB.

Categories and Subject Descriptors
C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks-Internet; H.4.3 [**Information Systems**]: Communications Applications

General Terms: Algorithms, Performance, Design, Human Factors, Verification

Keywords: PFGS, motion activity level, content-dependent coding, content-dependent packet drop, quality-dependent multicast

1. INTRODUCTION
A key challenge of video streaming in peer-to-peer (P2P) and overlay networks is to regulate the video transmission rate so that the video streams fairly share the available bandwidth, e.g., in a TCP friendly manner [16,22]. One basic approach is to encode a given video at many different rates, i.e., into different versions, and then to transmit the version with the highest rate that still fits into the available bandwidth. This approach of transmitting different encoded versions of the same video, also known as simulcast or bit stream switching, while simple, has a number of significant drawbacks. These include the need to encode an impractically large number of versions (on the order of several tens of versions are required to adapt a Mbps video at the granularity of 100 kbps) and no flexibility to scale down the bit rate/video quality of a stream during network transport, unless typically complex and computationally demanding transcoding is performed at intermediate network nodes. Bitstream switching can also be applied over versions of different coding schemes, but an increase in the computational complexity of video decoding is inevitable [18].

Scalable (layered) video coding strives to overcome these drawbacks by encoding a video into a base layer and several enhancement layers [6,8,13]. The base layer, which is required to provide a basic video quality, is typically transmitted with higher protection (often achieved with unequal error protection channel coding), and the enhancement layers, which improve the video quality, are transmitted with lower protection [12,15,26]. A key limitation of layered video coding is that the video bit rate can only be adapted at the granularity of complete enhancement layers, whereby the number of layers is typically limited to a small number (at most 4-5 in practical encoders) resulting in rather coarse rate adaptation by adding and dropping layers. When a part of an enhancement layer has to be dropped to adapt to the available bandwidth during network transport, then the entire enhancement layer is lost for the decoder.

Fine granularity scalable (FGS) encoding overcomes this shortcoming by encoding the video into one base layer and one enhancement layer, whereby the enhancement layer bit rate can be very finely adapted [13]. In FGS coding this flexibility in bit rate adaptation comes at the expense of a relatively low compression efficiency due to lack of motion compensation in the enhancement layer. Progressive FGS (PFGS) coding overcomes this disadvantage and provides generally good compression efficiency as well as high flexibility in adapting the enhancement layer bit rate [6,8].

The adaptation capabilities of PFGS coding make it possible to encode the enhancement layer of the video with one (high, say 2 Mbps) bit rate and then to transmit the enhancement layer at any bit rate that is lower than the encoding bit rate. This flexibility, however, comes even with the PFGS codec with some compression inefficiencies. In particular, *for transmission at a low* *bit rate (say around 1 Mbps) the video could be coded (with the PFGS codec) much more efficiently by employing an enhancement layer encoding bit rate in the vicinity of the transmission bit rate. We quantify this efficiency gain achieved by selecting the encoding bit rate reasonably close to the actual transmission bit rate in Section 3.1 and demonstrating that it can reach on the order of 4 dB for individual video shots.* In summary, PFGS encoding is only optimized for the rate that is used during the encoding process [6,8]. Essentially all existing studies on PFGS video streaming have ignored the important aspect of the encoding bit rate selection [3,4,12,25,26] and focus primarily on modeling the rate-distortion characteristics of a given encoding or the optimal selection of the transmission bit rate based on the rate-distortion model of one encoding.

Our main contribution in this paper is to develop and evaluate mechanisms for optimally selecting the enhancement layer encoding bit rate when streaming PFGS videos. To the best of our knowledge this important problem has to date only been observed in [24] but no solution has been proposed, as detailed in Section 1.1. Our main approach is to pre-encode a given video with PFGS into a small number of versions with different enhancement layer coding bit rates. In a sense, our approach combines simulcast with PFGS coding. In contrast to simulcast which requires many versions to adapt the video bit rate to the available transmission rate, our approach requires only a small number of versions so that the transmission bit rates are reasonably close to the encoding bit rates (about 4 versions are sufficient to extract the full gain, see Section 3.3). The storage overhead of these encoding versions is negligible due to minimal hardware cost of current storage devices. Our approach achieves the adaptation to the available transmission bit rate by first selecting the version with the lowest encoding bit rate that is still above the available transmission bit rate and then relying on the PFGS enhancement layer adaptation for fine-tuning the transmission rate.

Key distinctions of our work are that we (*i*) exploit the video content characteristics for efficient PFGS streaming, (*ii*) develop and evaluate suite of schemes that can be used to augment existing unicast streaming, link rate regulation, and multicast schemes and is therefore of particular relevance for application layer streaming over P2P overlay networks, and content distributed networks, and (*iii*) that our schemes do not require any modification to existing PFGS decoders; the minor added complexity is only in the video server, which makes our proposed approach easy to deploy.

The paper is organized as follows. In Section 2, an overview of PFGS video coding scheme is presented. Section 3 discusses the unicast streaming of PFGS streams and presents an adaptive scheme to improve the reconstructed qualities. Section 4 explains multiple packet dropping policies for unicast schemes and its implication on the reconstructed qualities for each receiver. In Section 5, quality-adaptive multicasting scheme is presented. Finally, the discussion is presented in Section 6 followed by references.

1.1 Related Work

To the best of our knowledge the important problem of setting the enhancement layer encoding rate for streaming PFGS video has to date only been noted in [24] where a transcoding approach is developed that can lower the enhancement layer encoding rate of an encoding with a higher encoding rate to make the encoding more efficient for the transmission at lower bit rate. Our work is complementary to [24] in that we consider the problem of determining the optimal enhancement layer encoding bit rate. This optimal rate could be achieved by transcoding the enhancement layer stream from an encoding with a very high encoding bit rate, or by picking from a set of pre-encodings with different enhancement layer coding bit rate. We consider the latter approach throughout the paper.

The visual content, which typically varies drastically between shots of various actions and genres, especially for movie contents, has received relatively little attention in efforts to improve the application layer QoS of video streaming. The streaming scheme should adapt to the visual content variability as well as bit rate variability. Conventional video streaming techniques optimize user satisfaction by using a utility function that is based on the rates of the receivers [7,11,17]. Some other video streaming employs the frame type of the video stream to adapt the video streaming [9,27]. This frame type information is a low level visual content that is extracted from the syntax of the video stream. This frame type information is not well correlated with actual visual content descriptors that are expressed by MPEG-7 descriptors [10], which we consider in our study.

Video multicasting over the Internet has received significant attention in the past and bandwidth adaptation has been identified as an essential requirement for video multicasting [2,5,14,15]. In layered multicast schemes, each receiver decides which layers to receive, based on its own capabilities, see e.g., [19]. One of the key issues that layered multicast schemes have to overcome is the intersession fairness [14,15]. Hybrid Adaptation Layered Multicast (HALM) is a recent protocol that outperforms other layer multicasting protocols by allowing adaptation at the video server, in addition to adaptation at the receivers [16,17]. HALM is most suitable for video streams coded using FGS or PFGS schemes because of the simplicity of real-time rate adaptation. Initially, the receivers predict their own available bandwidth (using a mathematical model, e.g., [20]) and use this prediction to join the appropriate multicasting group. The video server then receives these predictions, in the form of RTP reports, and uses these predictions to assign transmission rates to each layer in a manner that maximizes the average bit rate fairness index. Our adaptive multicast scheme is an application layer multicast scheme that considers the reconstructed qualities in the rate adaptation and can be implemented over the existing HALM protocol

2. OVERVIEW OF PFGS CODING

The PFGS scheme improves bandwidth efficiency by providing motion compensation from reconstructed enhancement layer frames [6,8]. Fig. 1 illustrates a typical PFGS codec where the base layer stream is coded using an H.26L encoder. There are two predictions used at the enhancement layer. One is a low quality prediction, which is derived from the reconstructed base layer, while the other is a high quality prediction, which is derived from the reconstructed enhancement layer. Macroblocks can be coded with reference to high or low quality prediction [8]. In Fig. 1, two switches (s_1 and s_2) are used to select the reference for motion compensation and reconstruction. The residue after prediction is discrete cosine transformed (DCT), followed by bit plane coding, similar to the MPEG-4 FGS scheme [13]. Only the first $\alpha(t)$ bits (which represent the bit rate of the overall enhancement layer) are used to reconstruct the enhancement reference for the next frame. If part of the enhancement layer bitstream is lost due to channel bandwidth fluctuations, the decoder will reconstruct a degraded image, compared to the transmitted image. This will inevitably result in a drifting error at the decoder. In other words, a drifting error occurs if the enhancement layer transmission rate (r) is less than the encoding rate (α), which is referred to as $\alpha(t)$ in Fig. 1.

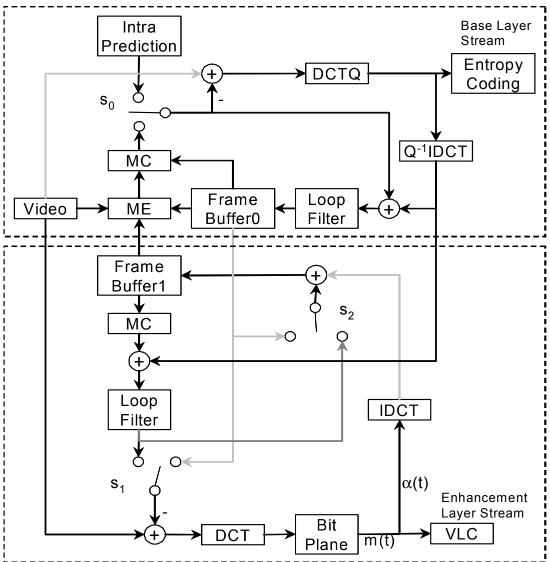


Figure 1: Encoder structure of H.26L-PFGS

To illustrate the performance of PFGS codec as a function of the enhancement layer transmission rate r, we present encoding results for the first 15 minutes of the *Star Wars I* movie. In our performance evaluation, we take the video content features into consideration by extracting the motion activity levels of the encoded video shots. Video shots can be considered as the minimal logical video sequence of the underlying movie. A video shot is defined as a sequence of frames captured by a single camera in a single continuous action in time and space. Automatic shot detection techniques have been extensively studied and simple shot detection algorithms are available [1]. This enables us to code the first frame in every shot as an intra frame. The shot detection techniques produced 200 video shots with a range of motion activity levels. The intensity of the motion activity in a video shot is represented by a scalar motion descriptor, which ranges from 1 for a low level of motion to 5 for a high level of motion, and correlates well with the human perception of the level of motion in the video shot [10]. For each video shot, 10 human subjects estimated the perceived motion activity levels, according to the guidelines presented in [21]. The motion activity level was then computed as the average of the 10 human estimates. QCIF (176×144) video format was used, with a frame rate of 30 fps, and I-frame coded every 25 frames. The video shots were coded using a quantization scale of 28 (which is selected to guarantee a base layer rate of about 100 Kbps).

Fig. 2 shows the average reconstructed qualities denoted as $Q(r,\alpha,\lambda)$, where r represents the enhancement layer transmission rate, α represents $\alpha(t)$ in Fig. 1, i.e., the enhancement layer encoding rate, and λ represents the motion activity level of the underlying video shot. In the case of $r = 0$, the reconstructed quality is obtained by decoding the base layer stream and this base layer quality can be of any PSNR value depending primarily on the color complexity of the video shot. Importantly, we observe that the relationship between $Q(r,\alpha,\lambda)$ and r can be approximated linearly in the range of $r \in [0,1800]$. The non-linearity in the range of $r \in [1800,2000]$ is due to the significance of the least bit planes in the reconstructed quality. This effect is more pronounced in shots of low motion activity levels, where large number of frame blocks is predicted with reference to the enhancement layer frame. Overall, the combination of the base layer quality and the close to linear increase in quality with the encoding rate gives rise to the ordering of the curves observed in Fig. 2 with activity level 5 lying between activity levels 1 and 2. Normalizing the quality, as done in Fig. 5, gives rise to the typical ordering of activity levels, see Sec. 4.1.

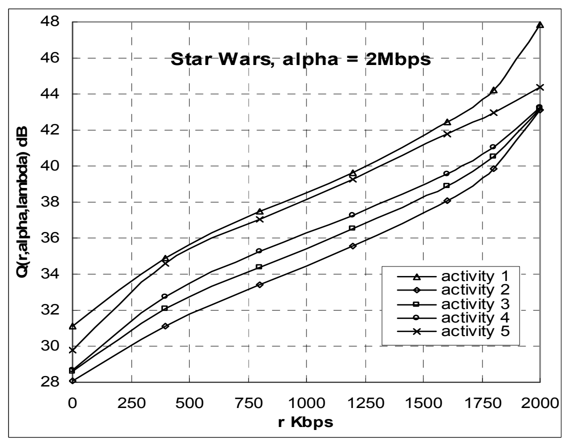


Figure 2: Reconstructed video quality Q as a function of enhancement layer transmission rate r for different motion activity levels λ, enhancement layer encoding rate α = 2 Mbps, fixed

3. CONTENT-DEPENDENT ENCODING RATE SELECTION FOR UNICAST PFGS STREAMING

In this section we present our novel adaptive point-to-point streaming scheme for pre-encoded PFGS videos. The basic idea of our scheme is to optimally select the encoding rate α of the PFGS enhancement layer from among a set of pre-encoded encoding rates α_i according to the motion activity level of the underlying video shot.

3.1 Foundation: Impact of Motion Activity Level on Selecting Best Encoding Rate

To illustrate the basis of our adaptive streaming scheme, we present the tradeoffs in selecting the enhancement layer encoding rate α for the video shots in the first 15 minutes of *Star Wars I* movie. We have coded the shots using different encoding rates α_i, i=1,..,N. We compare the reconstructed qualities of PFGS encodings with different α_i, by plotting quality difference curves of two PFGS encodings ($Q(r,\alpha_i,\lambda) - Q(r,\alpha_{i+1},\lambda)$) as shown in Fig. 3. If the video transmission rate is slightly higher than α_i =1 Mbps (see, e.g., transmission rate 1200 Kbps in Fig. 3(a)), the reconstructed qualities for various motion activity levels are higher for the PFGS encoding with encoding rate α_i =1 Mbps, as indicated by the positive quality difference. In other words, for such a case (i.e., the available transmission rate is 1200 Kbps) the video server is better off transmitting the video stream with a lower rate (i.e., 1 Mbps) than the channel can offer (1.2 Mbps) in order to achieve higher reconstructed quality. If the video transmission rate is much higher than 1 Mbps, significant quality degradation is incurred using the PFGS encoding with α_i =1 Mbps, as indicated by the negative quality difference. That is, with the higher transmission rate, a higher reconstructed quality is achieved with α_{i+1} = 2 Mbps encoding. Importantly, we observe from Fig. 3(a) that for video shots of low motion (activity 1) coding inefficiencies up to 4 dB are incurred by transmitting an encoding with α = 2 Mbps at rate of 1200 Kbps over streaming an encoding with α = 1 Mbps, underscoring the importance of appropriate encoding rate selection as a function of the visual content.

In Fig. 3, we define the critical point as the rate at which adapting the video streaming from the encoding rate α_i to the encoding rate α_{i+1} is beneficial in terms of the reconstructed qualities. The motion activity level of the underlying video shot plays an important role in determining these critical points. The critical point for high motion activity levels is located at a lower rate compared to the critical points of lower motion activity levels. This is because the reconstructed qualities of shots of high motion activities are dependent on the availability of larger number of bit-planes, which can be provided by PFGS streams with encoding rate $\alpha_{i+1}>\alpha_i$. Comparing Fig. 3(a) and Fig. 3(b), we observe that as the difference between α_i and α_{i+1} becomes smaller, the critical points move closer to α_{i+1}.

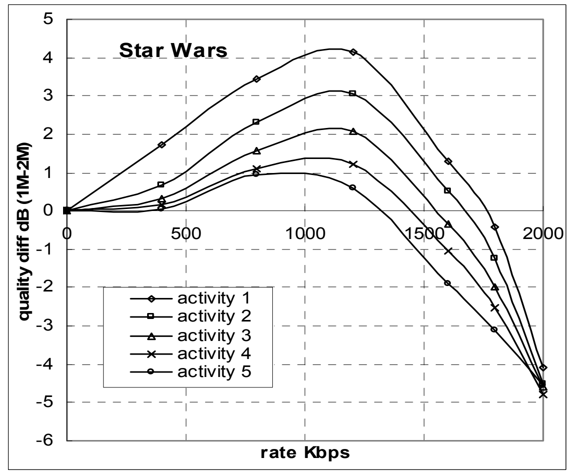
3.2 Adaptive Transmission Scheme

The relationships illustrated in the above figures suggest the following adaptive video transmission scheme that can be implemented at the video server. The video server can store *N* PFGS encodings of a given video with different enhancement layer encoding rates α_i, where i=1,..,N and $\alpha_i<\alpha_{i+1}$. The TCP-friendly transmission rate is computed using an equation based modeled as [20]:

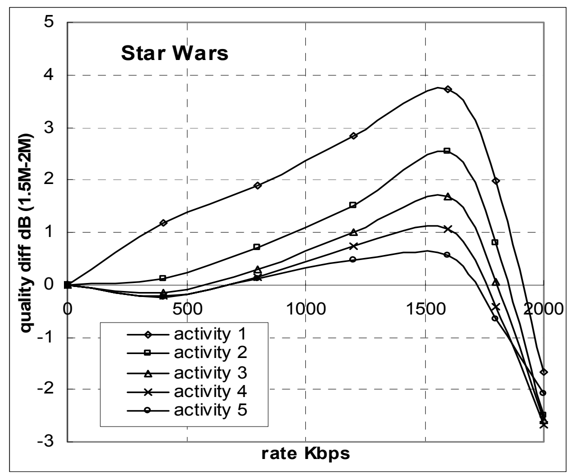
$$Bandwidth = 1.22 \times \frac{MTU}{RTT\sqrt{Loss}} \qquad (1)$$

where *MTU* denotes the packet size, *RTT* denotes the round-trip-time, and *Loss* denotes the steady state drop rate. In each RTT, the video server computes the transmission rate using eq. (1) and determines the PFGS video stream with optimal enhancement layer encoding rate α^*, such that the reconstructed quality is maximized. In order to find α^*, sample points of $Q(r,\alpha_i,\lambda)$ (shown in Fig. 2) are stored in the video server. A linear approximation between these sample points is used to predict the reconstructed quality $Q(r,\alpha_i,\lambda)$. For each video shot, the motion activity level (λ) is stored in the video server. Hence, the PFGS video stream that maximizes the $Q(r,\alpha_i,\lambda)$ is selected for the current video transmission.

$$i^* = \arg\max_i \{Q(r,\alpha_i,\lambda)\}, \alpha^* = \alpha_{i^*} \qquad (2)$$



(a) α_i = 1 Mbps, α_{i+1} = 2 Mbps



(b) α_i = 1.5 Mbps, α_{i+1} = 2 Mbps

Figure 3: Quality difference ($Q(r,\alpha_i,\lambda) - Q(r,\alpha_{i+1},\lambda)$) **as a function of enhancement layer transmission rate *r* for PFGS encodings with different encoding rates α_i and different motion activity levels**

3.3 Performance Evaluation

Two simulation experiments have been conducted with average TCP throughputs (AVR_TCP) of 1 Mbps and 1.5 Mbps, where the packet loss ratios are generated using a Markov model with two states [23]. RTT is fixed to 500 ms. Simulations using real Internet traces or using the ns-2 simulator can be a topic for future research. The video server stores *N* PFGS different encodings with the enhancement layer encoding rates α_i set as follows:

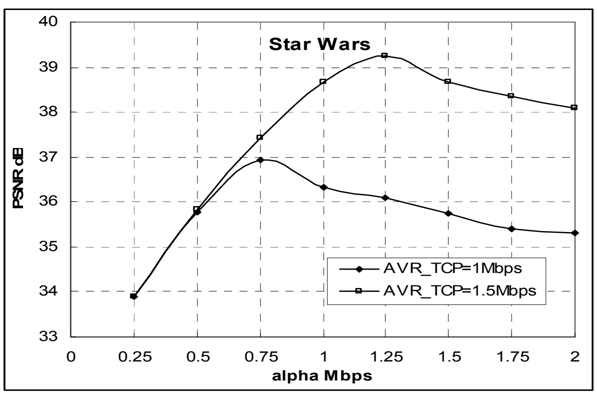
α_N=2 Mbps

$\alpha_i=\alpha_{i+1}$-0.25

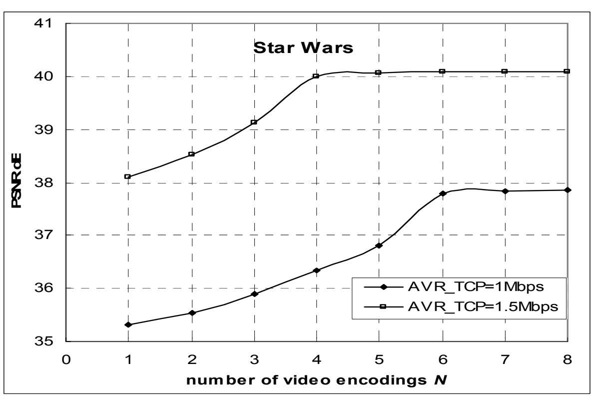
i=1,..,N-1

where the maximum value of *N* is 8. For instance if *N* equals 3, the video server contains three PFGS video streams with α_1, α_2 and α_3.

The performance evaluation of the adaptive video transmission scheme is computed as the average of reconstructed qualities over the first 15 minutes of *Star Wars I* movie. The performance of the proposed adaptive transmission scheme is shown in Fig. 4(b). As the number N of stored encodings of a video increases, the average reconstructed quality increases. The performance of the adaptive transmission reaches a peak value at a relatively small number of 4-6 stored encodings, where storing more encodings (with lower α_i) brings only negligible performance improvement. In addition, Fig. 4(a) shows the performance of the non-adaptive video transmission scheme. We observe that by encoding the video with enhancement layer coding rate α = 2 Mbps and then streaming this encoding with an average transmission rate of 1 Mbps, coding inefficiencies of up to 1.8 dB are incurred over selecting the optimal encoding rate of α = 0.75 Mbps. Generally, the optimal α^* for the non-adaptive transmission scheme depends on the average throughput of the transmission channel, which is impossible to predict for time-varying channels such as wireless and Internet. Comparing the peak values in Fig. 4(a) and Fig. 4(b), we observe that the proposed adaptive scheme achieves about 0.8 dB improvement over the best non-adaptive scheme. We thus conclude that for time-varying channels, it is beneficial to have multiple PFGS encodings with various α_i. The proposed adaptive transmission scheme is suitable for any time-varying channels and can be added to existing video servers with negligible overhead.



(a) Reconstructed quality Q with non-adaptive transmission as a function of enhancement layer encoding rate α



(b) Reconstructed quality Q with proposed adaptive transmission as a function of number of encodings N (with different enhancement layer encoding rates α_i)

Figure 4: Performance of adaptive scheme: Approximately 0.8 dB quality improvement with adaptive scheme over best-performing non-adaptive scheme.

4. CONTENT-DEPENDENT PACKET DROP POLICIES FOR PFGS UNICAST STREAMING

In this section, we propose and evaluate content-dependent packet drop policies for pre-encoded PFGS unicast streams that share a common networking resource, e.g., shared bottleneck link. The streaming context considered in this section is that we suppose that video streams with a prescribed enhancement layer encoding rate are streamed, with a prescribed transmission rate into the network, and encounter a bottleneck that requires the dropping of some packets downstream.

4.1 Foundation: Content-Dependent Quality Degradation Due to Packet Drops

The basic underlying observation for our packet drop policies is that dropping video packets carrying video shots of various motion activities result in different visual impacts. In order to illustrate this effect, the results presented in Fig. 2 are transformed into rate regulation and corresponding visual quality degradation in Fig. 5. The rate regulation represents the reduction in the enhancement layer transmission rate due to the packet drops at the bottleneck (equivalently, the rate regulation represents the packet loss ratio of the enhancement layer) normalized by the enhancement layer encoding rate α, i.e., rate regulation of value of 1 represents the complete dropping of the enhancement layer. The quality degradation represents the ratio of the reduction in the reconstructed quality (due to the rate reduction) to the quality of the enhancement layer with encoding rate α. In other words, we define quality degradation = (original quality with enhancement layer coded with rate α - reconstructed quality after rate regulation)/ original quality with enhancement layer coded with rate α, *i.e., the* quality degradation at rate r is given as $(Q(\alpha,\alpha,\lambda)-Q(r,\alpha,\lambda))/Q(\alpha,\alpha,\lambda)$, whereby the amount of rate regulation is $(\alpha - r)/\alpha$. Note that the maximum quality degradation (achieved for a rate regulation value of 1) corresponds to the reception of only the base layer stream. Fig. 5 shows the quality degradation as a function of the amount of rate regulation for shots of different motion activity levels. The rate regulation of shots of high motion activity levels reduces the reconstructed quality relatively less compared to the rate regulations of shots of lower motion activity levels. This difference in quality degradation is due to the difference in motion estimation loops that are used for shots of different motion activity levels. In a shot with higher motion activity, a larger fraction of the enhancement layer blocks is motion estimated with reference to the base layer frame. Hence, the quality degradation due to enhancement layer rate regulation is relatively smaller for shots of high motion activity levels. Conversely, in shots with a lower level of motion, a larger fraction of the enhancement layer blocks are motion estimated with reference to the preceding enhancement layer frame, resulting in more severe quality degradation due to rate regulation.

4.2 Proposed Packet Drop Policies

The basic idea of our packet drop policies is that when multiple video streams share a bottleneck communication link, it is likely that the link buffer contains packets carrying shots of diverse motion activity levels. As we have observed in Fig. 5, the visual quality degradation depends on the motion activity level. In the case of rate regulation (packet dropping to overcome congestion or to achieve TCP-friendly throughputs), the number of dropped packets from each video sequence can be specified using different policies, that exploit the content dependencies illustrated in Fig. 5.

Specifically, we introduce four different rate regulation policies: -

- Policy 1: random dropping of video packets from the shared PFGS video streams to meet the rate constraint (denoted as R Kbps) of the link resources.
- Policy 2: equal packet loss ratios (or equal rate regulation ratios) from the shared PFGS video streams to meet the rate constraint on the link resources.
- Policy 3: the packet loss ratios are determined according to underlying shot activity level. The objective is to achieve equal quality degradations for the PFGS video streams sharing the bottleneck resource. This can be implemented by simple algorithm that has access to the sample points of $Q(r,\alpha,\lambda)$ (shown in Fig. 2), and linearly approximating between these sample points.
- Policy 4: the bit budget is distributed among shared PFGS video streams such that the average reconstructed qualities are maximized. A greedy computational method can achieve this goal. The method evaluates every possible rate combinations using the approximation of $Q(r,\alpha,\lambda)$. The global maximum point is located and used to determining the rate (denoted as ρ_i) for each PFGS video stream.

Existing buffer management schemes employ either policy 1 or policy 2, which do not need access to the sample points of $Q(r,\alpha,\lambda)$. On the other hand, policy 3 and policy 4 require access to these points, which can easily be exchanged during connection setup. In the following subsection a performance evaluation of these policies is conducted

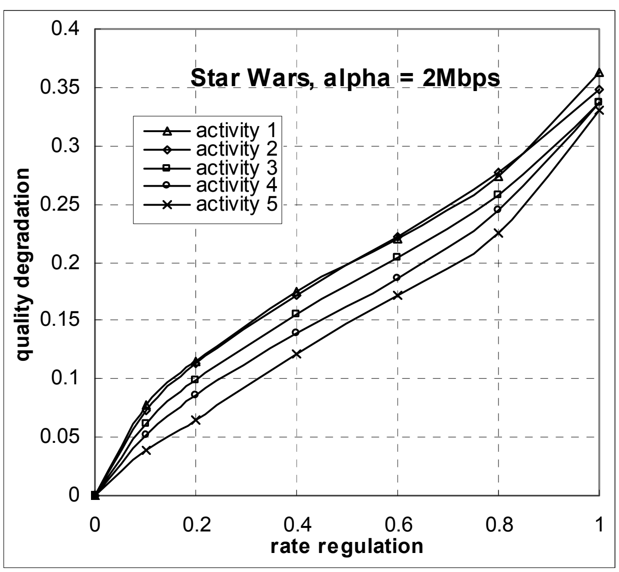


Figure 5: Quality degradation (reduction in reconstructed quality normalized by the quality of enhancement layer with encoding rate α) as a function of rate regulation (fraction of dropped enhancement layer packets) for different motion activity levels

4.3 Performance Evaluation

We have evaluated the four outlined dropping policies through simulation experiments with average TCP throughputs of 2 Mbps, 3 Mbps, 4 Mbps and 5 Mbps. For simplicity, four PFGS video streams share the communication resources. The four video streams are derived from the video shots from *Star Wars I* as follows. An encoding with enhancement layer encoding rate of α_i= 2 Mbps forms the basis for the streams. Four streams with enhancement layer transmission rates (at the server) of 1.75 Mbps, 1.5 Mbps, 1.25 Mbps, and 1 Mbps are streamed (denoted as r_j) from this α_i= 2 Mbps encoding. We adopt this streaming with the different transmission rates from the same encoding and do not employ the adaptive enhancement layer encoding rate selection of Section 3 to study in isolation the effects of the dropping policies; the adaptive rate selection of Section 3 and the adaptive dropping examined here could be combined for improved efficiency. In order to create randomness similar to user demands for the video streams, the start time of transmitting the video streams differs by 1 minute. In other words, the PFGS video stream with 1.75 Mbps starts at the initial simulation time, the PFGS video stream with 1.5 Mbps starts transmission at when the time equals 1 minute, .. etc. The rate regulation policy takes place every RTT (= 500 ms). The following results represent the average over the 15 minutes of the *Star Wars I* movie.

The performance of each rate regulation policy is evaluated using the average reconstructed qualities of the four streams and the quality fairness indices for the individual streams. The average reconstructed quality ($\overline{Q}(R)$) can be expressed as:

$$\overline{Q}(R) = \frac{1}{4}\sum_{j=0}^{3} Q(\rho_j, \alpha_j, \lambda_j) \quad (3)$$

$$R = \sum_{j=0}^{3} \rho_j$$

$0 \le \rho_j \le r_j$

where R represents the rate constraint on the communication link (which is also denoted as TCP_AVR), and ρ_j represents the enhancement layer rate for each individual PFGS video stream after applying the rate regulation policy.

The quality fairness index for video stream *j,* which is based on the reconstructed quality ratios, is calculated using: -

$$Fairness\_Index(j) = \frac{Q(\rho_j, \alpha_j, \lambda_j)}{Q(r_j, \alpha_j, \lambda_j)} \quad (4)$$

where $Q(r_j,\alpha_j,\lambda_j)$ represents the maximum reconstructed quality for video stream j, which can be obtained with maximum enhancement layer transmission rate of r_j Kbps. This fairness index considers the QoS from the application layer respective. It is therefore better suited for video streaming than the existing fairness indices that evaluate the bandwidth fairness, which does not correlate linearly with the application layer QoS.

Table 1 shows the average reconstructed qualities (PSNR_AVR) for these dropping policies, while Fig. 6 shows the corresponding individual fairness indices. The first three policies achieve very close average reconstructed qualities. This observation can be attributed to the fact that these drop policies are simulated over a bottleneck link of the same rate constraint R, and to the relatively little diversity in shot activity levels in the link buffer. It is expected that increasing the number of video streams sharing the bottleneck link can introduce sufficient diversity in shot activity levels, where rate regulation policies that consider shot activity levels achieve better average reconstructed qualities. Policy 4 only achieves 0.15 dB improvement, which is a negligible gain compared to the computational complexity of maximizing the average reconstructed qualities.

Table 1: The average reconstructed qualities for various drop policies and various TCP_AVR rates

PSNR_AVR	2Mbps	3Mbps	4Mbps	5Mbps
Policy 1	32.68	34.19	35.64	37.06
Policy2	32.89	34.25	35.64	37.04
Policy 3	32.76	34.21	35.65	36.97
Policy 4	32.94	34.34	35.79	37.15

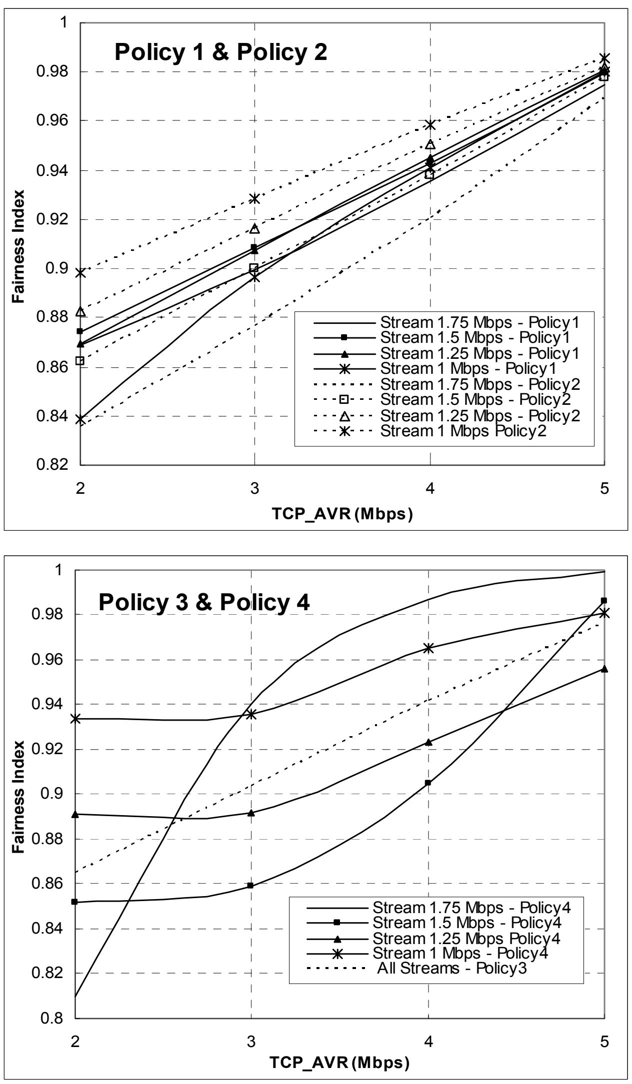


Figure 6: Fairness index of various dropping policies

The maximum reconstructed quality is relatively small due to the minimal diversity in motion activity levels of shared video streams and to the small difference in quality degradation between shots of various motion activity levels. Policy 1 and Policy 4 are random in nature, so that the quality fairness index for each PFGS video stream depends on the rate constraints of the link expressed by the TCP_AVR. On the other hand, policy 2 rewards low bit rate videos with higher fairness indices compared to higher bit rate videos. In other words, the quality degradation for low bit rate video is reduced, while the quality degradation for high bit rate video is increased. Policy 3 achieves equality in the fairness index among all PFGS video streams. In other words, the reconstructed qualities of each video stream are degraded fairly (i.e., the same quality degradation ratios) with the employment of policy 3 for rate regulation. This fair quality impact is not guaranteed with other rate regulation policies.

5. ENCODING RATE SELECTION FOR MULTICAST PFGS STREAM

In this section, we propose a scheme for optimally selecting the enhancement layer encoding rate so as to maximize the average reconstructed visual qualities at a group of multicast receivers. Our optimization scheme takes the rate constraints (number of subscribed multicast channels) of the individual receivers into consideration in the optimization. The proposed application layer scheme can be combined with any of the previously reported multicast methods, e.g., HALM [16,17], to improve the average reconstructed qualities of pre-encoded PFGS streams.

5.1 Foundation: Normalized Rate Regulation and Normalized Quality Degradation

As presented earlier in Section 3, video servers can store multiple versions of a video sequence, each based on a different encoding rate (α_i, where $1 \le i \le N$). During the entire video transmission, the effective Internet bandwidth varies dynamically. Multicasting protocols (such as HALM) dynamically adjust the enhancement layer transmission rates for each layer of the multicast tree. The challenging issue is to take into account the reconstructed visual qualities while adapting the encoding rate (α_i) so that the average reconstructed qualities are improved. In this work, we do not fully consider the motion activity levels expressed as λ in previous sections, due to the complexity of incorporating λ to existing multicast streaming. The motion activity levels are only considered by using four video sequences with such a level of motion activities. These video sequences are *news* (very low motion activity), *foreman* (low to moderate motion activity), *coastguard* (moderate motion activity), and *stefan* (high motion activity). We denote the quality of the base layer (which is constant) as $Q(0,\alpha_i)$, the number of multicast layers as M, and the aggregate bit rate of the first m multicast layers as c_m. In addition, p_m represents the probability that the expected bandwidth (R_j) of a receiver is between c_m and c_{m+1}. In other words, $p_m = P\{c_m \le r_j < c_{m+1}\}$ and $p_M = P\{c_M \le r_j\}$. If the aggregate bit rate is below the encoding rate, i.e., $\alpha_i \le c_m$, then the quality of multicast layer m is obtained using $Q(c_m,\alpha_i)$, whereas multicast layers with $c_m \ge \alpha_i$ achieve the same quality obtained using $Q(\alpha_i,\alpha_i)$. Formally, we let k denote the highest indexed multicast layer with an aggregate rate less than or equal the encoding rate, i.e., $k = \max\{m : c_m \le \alpha_i\}$

Therefore, the average reconstructed quality ($\overline{Q}(\alpha_i)$) for an encoding rate α_i is expressed as:

$$\overline{Q}(\alpha_i) = p_0 Q(0,\alpha_i) + \sum_{m=1}^{k} p_m Q(c_m,\alpha_i) + Q(\alpha_i,\alpha_i)\sum_{m=k+1}^{M} p_m \quad (5)$$

Hence, the encoding rate that results in the maximum quality can be expressed as the following optimization problem:

$$i^* = \arg\max_{1\le i\le N}\{\overline{Q}(\alpha_i)\}, \alpha^* = \alpha_{i^*} \quad (6)$$

Subject to: $\sum_{m=0}^{M} p_m = 1$

In order to solve this optimization problem, an approximation of Q(c,α_i) is required. Instead of storing sample points of Q(c,α_i) for every possible value of c and every possible α_i, we propose to normalize both the rate and quality of such a function using the following equations:

$$normalized\_rate = \frac{c}{\alpha_i} \quad (7)$$

$$normalized\_psnr = \frac{Q(c,\alpha_i) - Q(0,\alpha_i)}{Q(\alpha_i,\alpha_i) - Q(0,\alpha_i)} \quad (8)$$

Fig. 7 shows samples points of Q(c,α_i) using four α_i for *news*, *foreman*, *coastguard* and *stefan* video sequences after applying these normalizations. For each video sequence, the plotted values can be approximated using a polynomial function. The proposed normalization suggests that the values of all Q(c,α_i) can be reduced into a single polynomial for every normalized rate. This implies that for each video sequence (and hence each visual content) any reduction in the enhancement layer rate by a specific ratio results in a similar quality degradation, independent of the original encoding rate (α_i).

5.2 Proposed Quality-Adaptive Encoding Rate Selection

The basic idea of the proposed encoding rate selection is to periodically evaluate the multicast layer rates c_m, with an existing multicasting scheme, such as HALM, and then to find the encoding rate α^* that maximizes the average reconstructed qualities at the receivers. More specifically, each receiver estimates its expected bandwidth (r_j), which guarantees TCP-friendly behavior of the video multicasting scheme, with regard to other Internet traffic. The video server then receives these bandwidth estimates every control period, and specifies the multicast layer rates (c_m) using a dynamic programming technique [17]. These multicast layer rates are specified based on the distribution of various receiver rate estimates in terms of probabilities (p_m). A receiver joins the multicasting group with a rate that best matches its own. For each video sequence, the video server has several pre-encoded encodings at different encoding rates (α_i). Eq. (7) is applied to determine the normalized rate for each multicast layer and encoding rate. By approximating the curves of Fig. 7 (using polynomial coefficients), the normalized quality degradation can be obtained. The actual reconstructed quality for each multicast layer is determined by storing the sample points of $Q(\alpha_i,\alpha_i)$. For each available encoding rate, the average reconstructed qualities are calculated using eq. (5). By this method, the encoding with encoding rate (α^*) that produces the maximum average reconstructed quality at the group of multicast receivers is used for transmission. The proposed method can be deployed over existing video servers, since it is an application layer multicasting scheme.

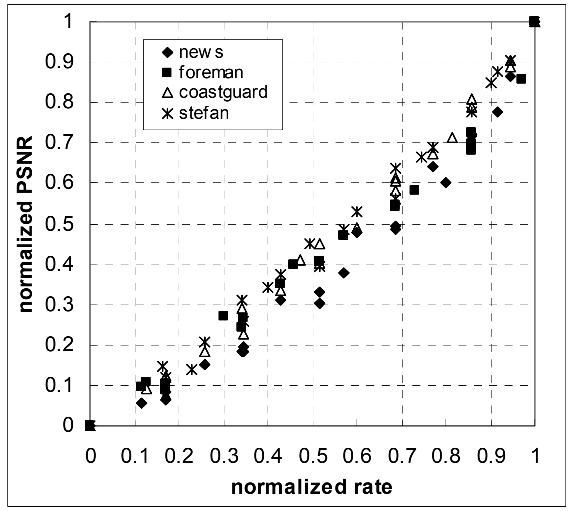


Figure 7: The normalized PFGS rate reduction curves

5.3 Performance Evaluation

Two separate simulation experiments were conducted, each targeting a particular rate distribution of the multicast layers (the total number of multicast layers is set to M = 4). CIF (352×288) video format was used, with a frame rate of 30 fps, and I-frame coded every 25 frames. The four video sequences (*news*, *foreman*, *coastguard*, and *stefan*) were coded using a quantization scale of 28. The first simulation distributes the receiver rates in a linear manner among the multicast layers, which guarantees graceful quality degradation in the event of network congestion. This rate distribution can be expressed as

$c_{m+1} = c_m + \Delta$, m=1,2 or 3   (9)

where Δ and c_1 are assigned values of 0.5 Mbps for the (low motion) *news* video sequence, and 1 Mbps for the other (moderate/high motion) video sequences. The second simulation distributes the receiver rates in an exponential manner among the multicast layers, which is consistent with congestion control protocols that employ multiplicative decrease in the event of congestion. The following equation represents this rate distribution.

$c_{m+1} = \beta \times c_m$, m=1,2 or 3   (10)

where c_1 = 193 Kbps for the *news* video sequence and 386 Kbps for the other video sequences. A value of 2 was selected for β to emulate the behavior of TCP-congestion control.

Fig. 8 shows the performance of the linear and the exponential rate distribution. To gain insight into the behavior of the proposed scheme for different receiver rate distributions, two separate sets of probabilities (p_m) are used for each simulation. Additionally, Table 2 and Table 3 show comparisons between the proposed

PFGS scheme (pfgs), the FGS scheme (fgs) and a baseline PFGS scheme with $\alpha_i = c_M$ (pfgs(max)), which corresponds to an existing multicast scheme that does not employ our proposed optimal encoding rate selection. The results reveal that the existing PFGS scheme with encoding rates set to c_M does not achieve the maximum quality, even if the last multicast layer (i.e., layer number M) serves the largest number of receivers (see entries of p=(0.1,0.1,0.2,0.2,0.4) in Table 3). It is obvious that the proposed PFGS scheme outperforms FGS coding. Sequences of low motion activities achieve the optimal quality at lower encoding rates, compared to sequences with high motion activities (see Fig. 8). For linear enhancement layer rates, and uniform probability distribution, the optimal coding rate (α*) is close to c_3. Hence, effectively three multicast layers are employed, where the receivers are distributed among these layers with probabilities 0.2, 0.2 and 0.4, respectively (with 20% of the receivers receiving only the base layer). We also observe, in the case of exponential multicast layer rates and a Gaussian-like probability distribution (see Fig. 8), that the optimal coding rate (α*) is close to c_2 for the *coastguard* sequence, while it is close to c_3 for the *news*, *foreman* and *stefan* sequences. Such correlation between the visual content and the optimal coding rate (α*) requires further investigation.

6. CONCLUSION

Adaptive video transmission schemes have proposed for streaming of pre-encoded PFGS videos for uni-cast, multiple uni-casts over bottleneck link, and multicast communication. The adaptive scheme for unicast communication selects an appropriate PFGS encoding from a number of stored PFGS coding versions of the video. The adaptation scheme is tuned for visual content variability and bandwidth fluctuations. The reconstructed quality of the adaptive scheme for unicast communication is improved by 0.8 dB compared to the best of non-adaptive schemes. We have conducted a performance analysis between four content-dependent packet drop policies for unicast streams that share a bottleneck link. Each packet drop policy utilizes the available bit budget by either randomly dropping video packets, equalizing the packet drop among shared video streams, equalizing the quality degradation among shared video streams, or optimizing the average reconstructed quality. The performance is evaluated using the average reconstructed quality and quality-based fairness index. The performance metrics show comparable results between these packet drop policies, which suggest the employment of packet drop policies that equalize the quality degradation due to its fairness of the reconstructed qualities. Multicast communication can also be improved (additional 2 dB) by storing multiple coding versions of a video and selecting the appropriate PFGS coding according to the condition of current multicasting tree and the underlying visual content. Motion related visual content shows a high correlation with reconstructed qualities for video streaming over heterogeneous networks such as P2P overlay networks deployed over the Internet and wireless networks. Future work can extend this frame work by investigating other visual content descriptors and run simulations using real Internet traces as well as other multicast streaming tools.

7. ACKNOWLEDGMENTS

We are grateful to Dr. Feng Wu from Microsoft China for providing the H.26L-PFGS codec. This work was supported in part by the National Science Foundation through Grant No. ANI-0136774. Any opinions, findings, and conclusions or recommendations expressed in this material are these of the authors and do not necessarily reflect the views of the National Science Foundation.

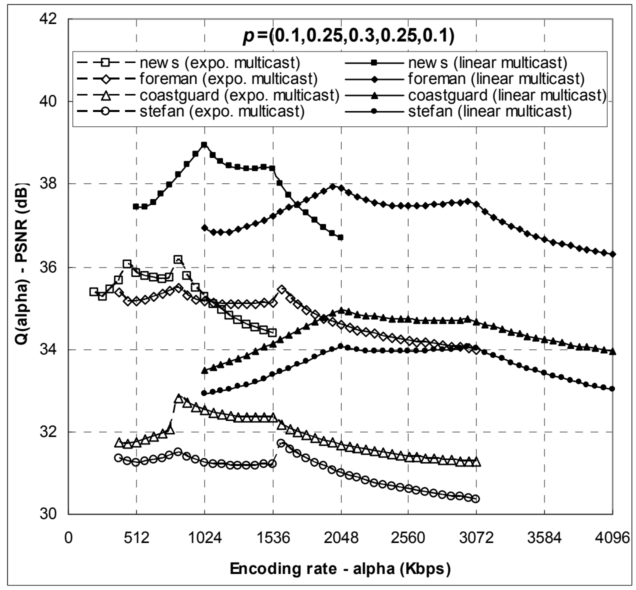


Figure 8: The impact of the encoding rate (alpha) selection on the average reconstructed qualities for receiver rate distributed with probabilities p_i and multicast layers distributed linearly or exponentially.

Table 2: Average reconstructed qualities with proposed PFGS encoding rate selection (pfgs), FGS streaming (fgs), and conventional PFGS (pfgs) for linear distribution of multicast layers

Sequence	p=(0.2,0.2,0.2,0.2,0.2)			p=(0.1,0.25,0.3,0.25,0.1)		
	pfgs	fgs	pfgs (max)	pfgs	fgs	pfgs (max)
News	38.2	35.5	37.4	38.9	35.6	37.1
Foreman	37.5	36.1	36.6	38	36.3	36.4
Coastguard	34.6	33.7	34.3	34.9	33.9	34.1
Stefan	33.8	32.4	33.5	34.2	32.5	33.3

Table 3: Average reconstructed qualities with proposed PFGS encoding rate selection (pfgs), FGS streaming (fgs), and conventional PFGS (pfgs) for exponential distribution of multicast layers

Sequence	p=(0.1,0.1,0.2,0.2,0.4)			p=(0.1,0.25,0.3,0.25,0.1)		
	pfgs	fgs	pfgs (max)	pfgs	fgs	pfgs (max)
News	38.5	35	37.8	36.2	33.7	34.4
Foreman	37.3	35.6	37	35.4	33.9	34
Coastguard	34.1	33.1	34.1	32.8	31.4	31.3
Stefan	33.5	31.7	33.4	31.7	29.9	30.4

8. REFERENCES

[1] J. S. Boreczky and L. A. Rowe. “Comparison of video shot boundary detection techniques,” SPIE Storage and Retrieval for Still Images and Video Databases IV 1996, vol. 2664, pp. 170 - 179.

[2] S. Cheung, M. Ammar, and X. Li, “On the use of destination set grouping to improve fairness in multicast video distribution,” Proc. IEEE INFOCOM 1996, pp. 553-560.

[3] P. de Cuetos, P. Seeling, M. Reisslein, and K.W. Ross, “Comparing the streaming of FGS-encoded video at different aggregation levels: frame, GoP, scene,” International Journal of Communication Systems 2005, vol. 18, no. 5, pp. 449-464.

[4] M. Dai, D. Loguinov, and H. Radha, “Rate-Distortion modeling of scalable video coders,” Proc. IEEE ICIP 2004, vol. 2, pp. 1093-1096

[5] W. Deng, Y.T. Hou, Wenwu Zhu, Ya-Qin Zhang and J.M. Peha, “Streaming video over the Internet: approaches and directions,” IEEE Trans. on Circuits and Systems for Video Technology (CSVT) 2001, vol. 11, no. 3, pp. 282 –300.

[6] Wu Feng, Li Shipeng and Ya-Qin Zhang, “A framework for efficient progressive fine granularity scalable video coding,” IEEE Trans. on CSVT 2001, vol. 11, no. 3, pp. 332 – 344

[7] S. Gorinsky and H. Vin “The utility of feedback in layered multicast congestion control” Proc. ACM NOSSDAV 2001, pp. 93 – 102.

[8] Y. He, F. Wu, S. Li, Y. Zhong and S. Yang, “H.26L-based fine granularity scalable video coding,” Proc. IEEE ISCAS 2002, vol. 4, pp. IV-548 - IV-551

[9] M.R. Ito and V. Bai, “A packet discard scheme for loss control in IP networks with mpeg video traffic,” The 8th International Conference on Communication Systems 2002, vol. 1, pp. 497 –503

[10] S. Jeannin and A. Divakaran “MPEG-7 visual motion descriptors,” IEEE Trans. on CSVT 2001, vol. 11 no. 6, pp. 720 –724.

[11] T. Jiang, E. Zegura, and M. Ammar, “Inter-receiver fair multicast communication over the Internet,” Proc. ACM NOSSDAV 1999, pp. 103-114.

[12] S.-R. Kang and D. Loguinov, “Impact of FEC overhead on scalable video streaming,” to appear Proc. ACM NOSSDAV 2005

[13] W. Li, “Overview of the fine granularity scalability in MPEG-4 video standard,” IEEE Trans. on CSVT 2001, vol. 11, no. 3, pp. 301-317.

[14] X. Li, M. Ammar, and S. Paul, “Video multicast over the Internet,” IEEE Network Magazine 2003, vol.13, no.2, pp. 46-60.

[15] J. Liu, Bo Li and Ya-Qin Zhang, “Adaptive video multicast over the Internet,” IEEE Multimedia 2003, vol. 10, no. 1, pp. 22 – 33.

[16] J. Liu, Bo Li and Ya-Qin Zhang “A hybrid adaptation protocol for TCP-friendly layered multicast and its optimal rate allocation,” Proc. IEEE INFOCOM 2002, vol. 3, pp. 1520 – 1529.

[17] J. Liu, Bo Li, and Ya-Qin Zhang, “An end-to-end adaptation protocol for layered video multicast using optimal rate allocation,” IEEE Trans in Multimedia 2004, vol. 6, no.1, pp. 87- 102

[18] O. Lotfallah and S. Panchanathan, “Adaptive scheme for internet video transmission,” Proc. of IEEE ISCAS 2003, vol. 2, pp. 872-875.

[19] S. McCanne, V. Jacobson, and M. Vetterli, “Receiver-Driven layered multicast,” Proc. ACM Sigcomm 1996, pp. 117-130.

[20] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP throughput: a simple model and its empirical validation,” Proc. ACM SIGCOMM 1998, vol. 28, no 4, pp 303-314.

[21] K.A. Peker and A. Divakaran, “Framework for measurement of the intensity of motion activity of video segments,” Journal of Visual Communications and Image Representation 2004, vol. 15, no. 3, pp. 265-284.

[22] N. Wakamiya, M. Miyabayashi, and M. Murata, “MPEG-4 video transfer with TCP-friendly rate control,” Springer-Verlang Berlin 2001, vol. 2216, pp. 29-42

[23] S. Wenger, “Error patterns for internet experiments,” in ITU Telecommunications Standardization Sector, Oct. 1999, Document Q15-I-16r1.

[24] J. Xu, Feng Wu, S. Li, “Transcoding for progressive fine granularity scalable vide coding” Proc. IEEE ISCAS 2004, vol. 3, pp III - 765-8

[25] F. Yang, Q. Zhang; W, Zhu, and Ya-Qin Zhang, “Bit allocation for scalable video streaming over mobile wireless Internet,” Proc. IEEE INFOCOM 2004, vol. 3, pp. 2142 – 2151

[26] Q. Zhang, G. Wang, W. Zhu, and Y.-Q. Zhang, “Robust scalable video streaming over Internet with network-adaptive congestion control and unequal loss protection,” Proc. Packet Video Workshop 2001, Kyongju, Korea.

[27] Z.-L. Zhang, S. Nelakuditi, R. Aggarwa, and R. P. Tsang, “Efficient server selective frame discard algorithms for stored video delivery over resource constrained networks,” Proc. IEEE INFOCOM 1999, pp. 472–479.