ABSTRACT

Video communication with quality of service (QoS) is an important and challenging task. To have QoS provision or guarantee at application level in the current best-effort Internet, congestion control and error control are two effective approaches. In this paper, we propose a new network-adaptive rate control scheme for scalable video communication in conjunction with Unequal Loss Protection (ULP). The proposed scheme not only adapts the bit rate to the dynamical network condition, but also has the error resilient capability. In other words, our proposed approach is capable of simultaneously controlling congestion and error occurred across the Internet. More specifically, first, we dynamically estimate the available network bandwidth on the fly. Secondly, congestion control is performed to smoothly adjust sending rate for continuous media. Thirdly, network packet loss rate is taken into account while performing rate adaptation at the bitstream level by minimizing end-to-end distortion. Lastly, error protection at packet level is added using ULP while performing rate control. Simulation results demonstrate effectiveness of our approach.

I. INTRODUCTION

With the advent of Internet revolution and availability of growing network resources, video communication is becoming increasingly popular among Internet multimedia applications. Since today’s Internet only offers best-effort service, video over the Internet with QoS guarantee faces many challenges due to network bandwidth fluctuation, packet loss, and delay. To achieve QoS guarantee or provision, two aspects of research have been conducted to date. One is to study how networks (e.g., routers and switches) support QoS services, which are referred to as integrated services and differentiated services. The other one is to study solely from application (end-system) without considering network at all. This includes network available bandwidth estimation, media rate adaptation to the available network bandwidth, and error resilient or/and error concealment in the codec. In the latter approach, Network monitoring deals with dynamic network bandwidth estimation either through feedback or probing. While media adaptation controls the media bit rate according to the network condition. Error resilience or/and concealments refer to the ability to track, localize and recover transmission errors.

In this work, we aim at application level QoS control. Congestion control and error control are the two most popular approaches at application level dealing with packet loss and delay in the Internet. Congestion control typically refers to rate control in the source and transport. It aims to reduce packet loss and delay. While error control is to avoid the packet loss across the Internet. It consists of forward error correction (FEC), retransmission such as ARQ, error resilience, and error concealment in the source and destination. Previous studies on application-level QoS considered congestion control and error control separately [1-4]. In this paper, we address rate control and error control simultaneously.

Scalable video coding is of great interest recently because of its scalability in bitstream and therein friendliness to network [5]. It is very suited for networking video applications such as video streaming. Scalable source coder encodes input video into many layers. Base layer, the most important layer that has to be successfully transmitted and protected, carries the basic information. Other layers called enhancement layer that can be dropped or transmitted based on the available network bandwidth. Due to the heterogeneities of networks and users as well as bandwidth fluctuation, studying scalable video transport over Internet is very important. This paper addresses simultaneous congestion control and error control for scalable video delivery over IP networks.

II. NETWORK-ADAPTATIVE RATE CONTROL WITH ULP

2.1. Available Network Bandwidth Estimation

To cope with packet dropping and bandwidth fluctuation in the current Internet, we use our proposed multimedia streaming TCP-friendly protocol (MSTFP) [4] for available bandwidth estimation. This protocol attempts to minimize the number of future packets that are likely to be dropped and smooth the sending rate. The underlying idea of the MSTFP is that the network status can be optimally tracked by iteratively combining forward estimation of network condition with information feedback control.

We use this 2-state Markov model (Gilbert model) to model packet loss in the Internet. The Markov model is a renewal model, i.e., the event of a loss frees the memory of the loss process and then is re-started. Meanwhile, the sender estimates the available bandwidth based on the observed packet loss rate, round trip time (RTT), and time-out (TO). The process of optimally tracking of network conditions using MSTFP consists of four stages: (1) estimating packet loss ratio; (2) estimating round trip time; (3) estimating available network bandwidth; (4) adjusting sending rate.

2.2. History Related Congestion Control

Usually TCP congestion control is based on Additive Increase/ Multiplicative Decrease (AIMD), which is independent of the packet lost ratio, bandwidth variation, and adjusting interval. We use the estimated network bandwidth to dynamically adjust the sending rate. Instead of using AIMD directly, we adjust the transmission rate related to packet loss ratio as follows.

- Increment in transmission rate: appropriate if the sender experiences loss when transmission rate is less than the available connection capacity.
- Slight reduction in transmission rate: appropriate if the sender experiences loss when transmission rate is at or around the available connection capacity.
• Aggressive reduction in transmission rate: appropriate if the sender experiences congestion loss when transmission rate is higher than the available connection capacity. Detailed formula can be found in [4]. The advantage of our scheme is that the sending rate can be increased or decreased very smoothly which is suited for the continuous media transmission. In other words, our scheme has less variation in the transmission rate and is less sensitive to packet loss.

2.3. Network-Adaptive Rate Control with ULP

Once we know the available network bandwidth, we perform media rate adaptation. In the following we present a network-adaptive bit allocation scheme for multi-layer scalable video codec with ULP. We use PFGS (progressive fine granularity scalable) video codec [6] as an example although our approach can be applied to any scalable codec such as FGS (fine granularity scalable) [5]. In our scheme, we combine PFGS codec with network-adaptive ULP across packets. We strongly protect the base layer of PFGS against packet loss so as to be decodable even if no enhancement layer are available by employing ULP based on Reed-Solomon FEC code.

The difficulty encountered in joint bit allocation between source and Internet channel is how to add FEC so that the decoder can still recover the lost frames correctly. Obviously, it can be observed that under a given channel rate, the additional FEC packets reduces the available rate for source coding, thus resulting in a trade-off between source coding and FEC. In this work, we address how to optimally allocate bits between source and FEC based on the R-D function, so that the decoder can still successfully recover the lost packets. Specifically, in our scheme the optimal bit allocation is dynamically adjusted according to varying video characteristics and network conditions. We formulate this problem as follows. Let \( R(t) \) denote the network bandwidth available for transmission at time \( t \). Let \( R_s(t) \) and \( R_{FEC}(t) \) denote PFGS source rate and rate of FEC packets, respectively. Further, let \( D_s(t) \) and \( D_{FEC}(t) \) represent PFGS source distortion and distortion of FEC packets, respectively. Then this problem becomes to allocate the available bit rate at time \( t \) so that the optimal \( R_s(t) \) and \( R_{FEC}(t) \) are obtained by minimizing end-to-end distortion under the following constraint: \( R_s(t) + R_{FEC}(t) \leq R(t) \), i.e., minimize \( D = D_s(t) + D_{FEC}(t) \)

subject to \( R_s(t) + R_{FEC}(t) \leq R(t) \). (1)

The block diagram of our bit allocation scheme for the PFGS source and ULP is illustrated in Figure 1. PFGS source coder encodes input video into two layers: one is the base layer (BL) that carries the most important information; the other is the enhancement layer (EL) that carries less important information. The EL bit stream can be truncated anywhere. These layers are packetized and protected against packet loss according to their importance and network status using different FEC. The channel estimation module adaptively updates the network status as discussed before. On the receiver side, channel decoder reconstructs packets for each layer and display video after source decoding. To efficient deliver video over Internet, several error resilience mechanisms have been adopted in video coder, such as error localization, data partition, error concealment, and etc.

The idea of forward error correction across packets is to transmit additional packets that can be used in the receiver to reconstruct lost packets. Our FEC scheme uses Reed-Solomon (RS) codes across packets. RS codes are perfectly suitable for error protection against packet loss, because they are the only known non-trivial maximum distance separable codes, i.e., there are no other existing codes that can reconstruct erased symbols from a smaller fraction of received code symbols [7]. An RS\((n, k)\) code with length \( n \) and dimension \( k \) encodes \( k \) information symbols containing \( m \) bits per symbol into a codeword of \( n \) symbols. An example of RS codes across packets is given in Figure 2.

To evaluate the performance of an RS\((n, k)\) code, we need to know the probability that more than \( n-k \) packets are lost. We can compute this probability if we know the probability of which \( m \) packets are lost within \( n \) packets.

As mentioned above, we use the 2-state Markov model to estimate network status. Such model is determined by the distribution of error-free intervals (gap). Let gap length \( v \) be the event that after a lost packet, \( v \)-1 packets are received and then another packet is lost. The gap density function \( g(v) \) gives the probability of gap length \( v \). The gap distribution function \( G(v) \) is the probability of gap length greater than \( v \)-1. Let \( R(m, n) \) be the probability of \( m \)-1 packet losses within the next \( n \)-1 packets followed by a lost packet. Detailed derivation can be found in [7]. Then the probability of \( m \) lost packets within \( n \) packets is

\[ P(m, n) = \frac{n!}{m!(n-m)!} P_g(v) R(m, n-v+1) \quad \text{for } 1 \leq m \leq n, \quad (2) \]

where \( P_g \) is the average packet loss probability.

Using \( P(m, n) \), we can analytically calculate the residual loss probability experienced by a video decoder after RS decoding, which is the probability that more than \( n-k \) packets are lost within the \( n \) packets. It is denoted as \( \sum_{m=n-k+1}^{n} P(m, n) \), which can be used to design the overall system if how many losses are acceptable for a video decoder is known. In the multi-layer scalable video codec such as PFGS, the impact of the residual loss probabilities of different layers on the video quality is not equal. Each layer has its own distortion weight. Since some error resilience mechanisms have been used in PFGS, the distortion for packet loss may just affect the slice. To isolate the quality distortion, each packet in our delivery scheme is composed of several complete slices. The packetization of PFGS with ULP is depicted in Figure 3. The transmission order for the packets is marked as dash line.

On the encoder side, distortion for each slice can be measured independently in advance. Let \( D_s(R_s) \) stand for the source perceptual distortion-rate function. Our problem becomes to find the optimal FEC scheme \((n, k)\) for different layers to minimize the end-to-end distortion \( D \):

Minimize \( D = D_s(R_s) \times P(0, R_s / S_p) + \sum_{i=1}^{n} (w_i \times \sum_{j=1}^{n} (D(i,j) \times \sum_{\{l=0 \ldots n-i-1\}} \sum_{\{y=0 \ldots n-l-1\}} P(l, n) \prod_{v=1}^{n-l} P(y, n))) \), (3)
where $D(i, j)$ represents the distortion that the $j^{th}$ packet at the $i^{th}$ layer is lost, $w_i$ is the distortion weight for the $i^{th}$ layer. Based on the decoder performance of PFGS, if the corresponding packet at any lower layers is lost, the packet of this layer is treated as lost no matter whether it is received or not.

![Figure 3. The packetization scheme for PFGS with ULP.](image)

**III. SIMULATION RESULTS**

This simulation is to demonstrate effectiveness of our proposed network-adaptive bit allocation with ULP scheme. In this simulation we tested: (1) our network-adaptive rate control with ULP scheme; (2) PFGS without knowledge of network bandwidth and packet loss ratio. In both cases, the first frame was intra-coded, and the remaining frames were inter-coded. The testing video sequence is Foreman that is coded in CIF at a temporal resolution of 15 fps. We conducted simulations under network condition from 320 kb/s to 480 kb/s. The corresponding packet loss ratio varies from 0.5% to 5%.

Figure 4 shows comparison results of PSNR for Foreman sequence using our proposed rate control with ULP scheme and the PFGS scheme.

Table 1 depicts comparison results of average PSNR for whole sequence and average protection ratio using our proposed network-adaptive rate control scheme with ULP and the PFGS scheme. Notice that the total available bandwidth is the same in both schemes.

Figure 5 shows two reconstructed frames of sequence Foreman using our rate control scheme with ULP and the PFGS scheme. In this figure, the upper left is the reconstructed 29th frame using our scheme, while the upper right shows the reconstructed 29th frame using the ULP scheme. The lower left shows the reconstructed 46th frame using our scheme and the lower right is the reconstructed frame using the PFGS scheme.

From Figures 4-5 and Table 1, it can be seen that for Foreman sequence, our proposed network-adaptive rate control approach with ULP obtains better results than the PFGS scheme under packet loss network both subjectively and objectively.

**IV. CONCLUSIONS AND DISCUSSIONS**

In this paper, we presented a new network-adaptive rate control method with unequal loss protection for scalable video communications. First, the available network bandwidth can be estimated via our proposed TCP-friendly protocol, the MSTFP. Secondly, history related congestion control is performed to smoothly adjust sending rate. Thirdly, our bit rate control in bit stream level takes packet loss rate into account so to minimizes the overall distortion for a given network condition. Lastly, FEC error control is added while performing rate control. The simulation demonstrated that our approach achieves significantly better results than the one without jointly considering congestion control and error control. One of the future works is to study how to incorporate ARQ into our scheme.

**V. ACKNOWLEDGEMENT**

Authors thank GuiJin Wang from Tsinghua University at Beijing for performing some of the simulation. Authors also would like to thank Drs. Shipeng Li and Feng Wu for providing the PFGS codec.

**VI. REFERENCES**


Figure 1. Block diagram of our network-adaptive bit allocation scheme for PFGS streaming with ULP.

Table 1. Comparison of average PSNR and overhead for Foreman

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Cases</th>
<th>Tested schemes</th>
<th>Average PSNR</th>
<th>Average protection ratio (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td>Case1</td>
<td>Our approach</td>
<td>30.16</td>
<td>8.25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PFGS</td>
<td>28.52</td>
<td>0</td>
</tr>
<tr>
<td>(320kb/s ~ 480kb/s)</td>
<td>Case2</td>
<td>Our approach</td>
<td>30.33</td>
<td>8.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PFGS</td>
<td>29.25</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 5. Comparisons of the reconstructed 29th frame (top) and 46th frame (bottom) of sequence Forman. The images on the left are reconstructed by our scheme, and those on the right are reconstructed by the PFGS scheme.