SLIDING-WINDOW PACKETIZATION FOR FORWARD ERROR CORRECTION BASED MULTIPLE DESCRIPTION TRANSCODING

*Tong Gan and Kai-Kuang Ma**
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore
* Email: ekkma@ntu.edu.sg

ABSTRACT

Forward error correction based multiple description (MD-FEC) transcoding, which performs *unequal loss protection* (ULP) for embedded bitstream, allows robust video transmission over packet erasure channels. However, most of the existing works focus on rate-distortion optimization of individual encoding unit, e.g., a *group of pictures* (GOP), and did not examine the problem of rate allocation among different units. Such transcoding strategy would lead to noticeable video quality variations when the video signal is highly nonstationary, and/or when large transmission rate fluctuations occur. In this paper, a novel window-based packetization scheme is proposed for reducing such quality variation through GOP interleaving. Performance evaluations are conducted by using 3D-SPIHT embedded video encoder.

1. INTRODUCTION

With the explosion of the Internet, video streaming across packet erasure networks has received much attention recently. In the current deployment of the Internet, routers do not differentiate the importance of each packet, and would randomly discard packets when encounter congestion [1]. Thus, in video transmission, it is essential to make the bitstream not sensitive to the position of packet loss. To achieve this goal, *forward error correction based multiple description* (MD-FEC) transcoding scheme was proposed [1]. The end-to-end system architecture based on MD-FEC is depicted in Figure 1. On the sender side, video is encoded using 3D-SPIHT [2] and pre-stored. Upon request, the embedded bitstream is first converted into an MD packet stream using the MD-FEC transcoder, and then delivered to the receiver across the Internet [1]. The *congestion control* (CC) module is responsible for performing end-to-end TCP-friendly rate control. More specifically, the CC module would periodically adjusts the transmission rate over the discrete-time periods, called *epochs* [1]. At the beginning of each epoch, according to the feedback information, the CC module would determine the number of packets to be transmitted based on the rate control algorithm employed, such as *linear increase/multiplicative decrease* (LIMD) [3], or TCP-friendly rate control (TFRC) [4].

In [1], one *group of pictures* (GOP) is transcoded and then packetized within one epoch. Although such packetization scheme is able to minimize the distortion for individual GOP, there are two problems encountered. First, if conventional LIMD is employed, which throttles the transmission rate by half upon packet loss [3], large transmission rate fluctuations would be incurred and lead to noticeable video quality variations at the receiver. Secondly, even if such rate fluctuation is eliminated by using a modified version of LIMD as proposed in [3], or by employing TFRC [4], "smooth" video quality still can not be maintained due to the inherent nonstationarity of video signal, such as varying motion intensity and scene changes. To address this issue, we propose a sliding-window packetization scheme. The main idea is to interleave adjacent GOPs into one epoch. In this way, available bandwidth is adaptively allocated among the GOPs to reduce video quality variations.

The rest of this paper is organized as follows. Section 2 gives a brief review of existing works in the literature. In Section 3, we present our proposed sliding window scheme. In Section 4, the proposed scheme is compared with that of [1] under both LIMD and TFRC rate control frameworks. Section 5 provides the conclusions and future works.

2. REVIEW OF EXISTING WORKS

MD-FEC transcoding, which performs *unequal loss protection* (ULP) for embedded bitstream, has been extensively studied in the literature [5]-[9]. In this section, a brief review of the existing transcoding and packetization schemes is given. Our notation closely follows that in [1].

In epoch k, suppose N packets with L symbols each will be sent. Without loss of generality, let us assume that each symbol takes one byte. The major steps of MD-FEC transcoding is summarized as follows. First, the embedded bitstream of GOP k is marked with $N+1$ positions $\{R_{k,0}, R_{k,1}, \ldots, R_{k,N}\}$ subject to

$$0 = R_{k,0} \leq R_{k,1} \leq \ldots \leq R_{k,N},$$

so that the bitstream is partitioned into N (possibly unequal-length) *segments*, $\{S_1, S_2, ..., S_N\}$, as shown in Fig. 2.

Secondly, S_j (bounded by $R_{k,j-1}$ and $R_{k,j}$) is equally divided into j *sub-segments*, denoted as $\{P_{j,1}, P_{j,2}, ..., P_{j,j}\}$, and protected with an (N, j) Reed-Solomon (RS) code with maximal distance, which can correct *any* $N-j$ erasures out of N packets [1][9]. Finally, the contribution from each segment of GOP k is distributed into N packets as illustrated in Fig. 3. Through this way, it can be ensured that when *any* j out of N packets are received, the client can reconstruct the bitstream of GOP k to $R_{k,j}$. In [5]-[9], various algorithms are proposed to minimize the end-to-end expected distortion ε_k of GOP k, where

$$\varepsilon_k = \sum_{j=0}^{N} q_j D_k \left(R_{k,j}\right), \tag{1}$$

subject to the bit rate constraint

$$\sum_{j=1}^{N} \frac{R_{k,j} - R_{k,j-1}}{j} \leq L.$$

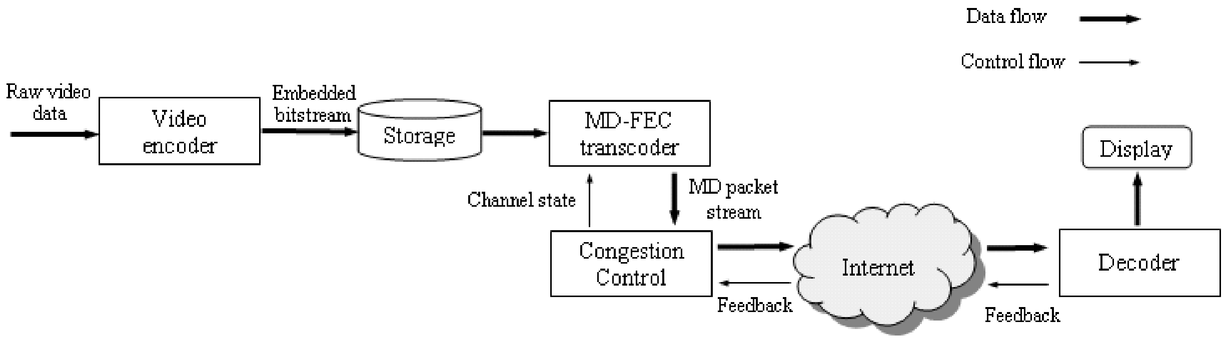

Fig. 1. Block diagram of the end-to-end video streaming system based on MD-FEC transcoding [1].

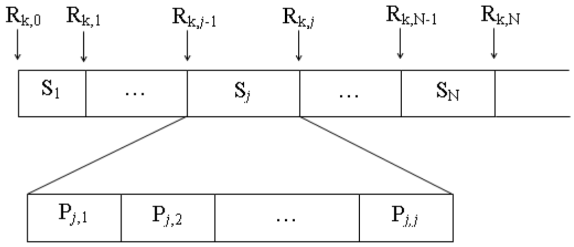

Fig. 2. Rate partition of the embedded bitstream of GOP k [1].

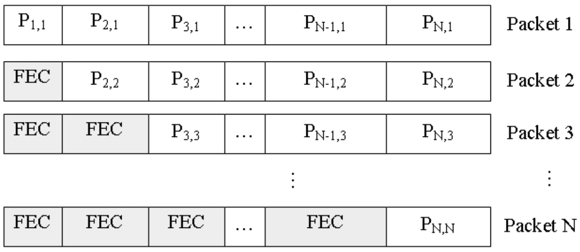

Fig. 3. Packetization of GOP k [1].

In (1), q_j for $0 \leq j \leq N$ denotes the probability that any j out of N packets would be successfully delivered, and $D_k(r)$ represents the rate-distortion function of GOP k. Among these algorithms, the recently proposed local search algorithm [9] can obtain near-optimal result with low time complexity of $O(NL)$, versus the $O\left(N^2 L^2\right)$ optimal algorithm in [8]; thus it would be employed in our proposed packetization scheme in the following section.

3. SLIDING WINDOW PACKETIZATION SCHEME

In order to reduce video quality variations caused by transmission rate fluctuation and/or nonstationarity of video signal, we interleave several GOPs into one epoch, so that the available bandwidth can be shared among these GOPs.

To this end, a *sliding-window* is introduced, which spans W consecutive epochs, as illustrated in Figure 4. Accordingly, instead of transmitting GOP i within epoch i, W GOPs from the ith to the $(i+W-1)$th are sent out. In other words, GOP i is transmitted from epoch $i-W+1$ to epoch i, and then decoded and displayed during epoch $i+1$. For example, with $W=3$, the window for GOP 3 spans from epoch 1 to epoch 3, while for GOP 4, the window slides forward by one epoch. Note that when $W=1$, this scheme boils down to that in [1].

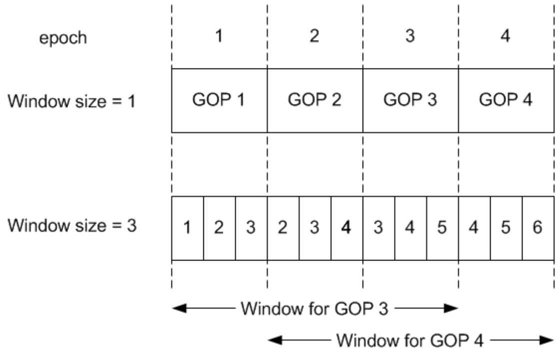

Fig. 4. Data interleaving with different window sizes. Content of each GOP is spread into W consecutive epochs.

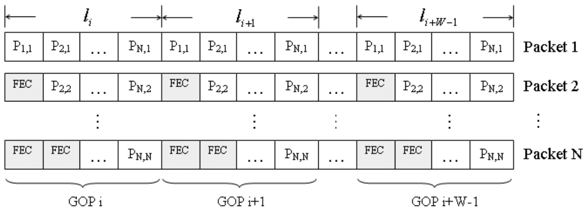

Fig. 5. Sliding-window packetization scheme. W GOPs are interleaved in one epoch.

Suppose within each packet, the number of bytes assigned to GOP k for $i \leq k \leq i+W-1$ is l_k. The proposed packetization scheme is illustrated in Figure 5. For individual GOP, similar to that in Figure 3, content from each segment is evenly distributed into each packet. After that, data from all the GOPs are concatenated to form the entire N packets. Obviously, this packetization strategy preserves the merits of multiple description in [1]. Under such packetization framework, there are two levels of rate allo-

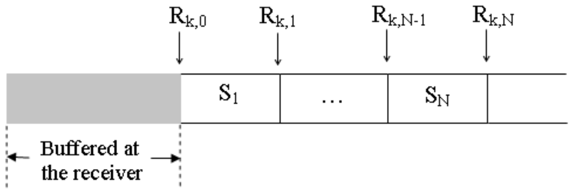

Fig. 6. Rate partition of GOP k. $R_{k,0}$ is shifted according to the data already buffered.

cation that need to be considered, i.e., intra-GOP allocation and inter-GOP allocation.

The main purpose of intra-GOP rate allocation is to minimize the distortion of individual GOP (i.e., ε_k) with a rate budget of $N \cdot l_k$ bytes for source and channel codes. Such problem has been well studied in [5]-[9]. Besides, it should be noted that in the sliding-window framework, since the content of GOP k is pre-fetched, the position of the first partition, $R_{k,0}$, should be shifted according to the amount of data already buffered at the receiver, as shown in Figure 6.

On the other hand, the objective of inter-GOP rate allocation is to minimize quality variations among different GOPs. To this end, one feasible criteria is to minimize the maximum distortion of the GOPs within the current epoch. Let $\varepsilon_k^*(l_k)$ denote the minimum expected distortion given that l_k bytes of each packet is allocated to GOP k. Also, let $\varepsilon_{\max}(l_i, l_{i+1}..., l_{i+W-1}) = \max_{i \le k \le i+W-1} \{\varepsilon_k^*(l_k)\}$ represent the maximum expected distortion given rate allocation $\{l_k\}\,(i \le k \le i+W-1)$. The optimal inter-GOP rate allocation is equivalent to minimize

$$\varepsilon_{\max}(l_i, l_{i+1}..., l_{i+W-1}) \tag{2}$$

subject to

$$\sum_{k=i}^{i+W-1} l_k = L.$$

Although the optimal solution can be obtained by exhaustively searching all the $\binom{W-1}{L+1}$ possibilities, the time complexity would be prohibitively high for real-time applications. Hence, a simple heuristic algorithm is proposed in this paper as follows.

Step 1: /* Initialization */
  $l_k = 0$ for $i \le k \le i+W-1$;
Step 2: Find k' such that
  $\varepsilon_{k'}^* = \max_{i \le k \le i+W-1} \{\varepsilon_k^*\}$;
Step 3: If $\sum_{k=i}^{i+W-1} l_k < L$
    $l_{k'} = l_{k'} + 1$,
    intra-GOP allocation for GOP k'
    go to Step 2;
  Else
    stop.

The algorithm starts from assigning 0 bytes to each GOP, and then iteratively increases rate allocation for certain GOPs until all $N \cdot L$ bytes are consumed. In each iteration, the algorithm first locates GOP k' with the maximum expected distortion based on the current allocation $\{l_i, l_{i+1}..., l_{i+W-1}\}$. Then one more byte in each packet (i.e., N bytes) is assigned to GOP k', and the intra-GOP allocation is updated with the local search algorithm in [9]. The time complexity of our algorithm is $O(NL^2)$, which is much lower than that of the exhaustive search.

4. EXPERIMENTAL RESULTS

In this section, we present some preliminary experimental results for the proposed sliding-window packetization scheme. The test sequence is the 352×288 "Foreman" video. The first 288 frames are encoded using 3D-SPIHT [2] with 16 frames per GOP, and these GOPs are transmitted cyclically. Each epoch has a duration of 650 msec, as that in [1].

Simulations are carried out for both TFRC [4] and LIMD [3] rate-control schemes. In both cases, for simplicity, we assume the traffic flow in the network is in the steady state. First, with TFRC, the TCP-friendly transmission rate r is determined by $r = \frac{1.3 \cdot L}{RTT\sqrt{p}}$ [4], where RTT is the round-trip time, and p is the packet loss rate. Thus, the steady-state transmission rate would be constant with fixed RTT and p. Secondly, when LIMD is employed, the sender would transmit one more packet if there is no loss in the previous epoch; otherwise, the transmission rate is throttled by half [3][4], i.e.,

$$N = \begin{cases} N+1, & \text{if no packet loss;} \\ 0.5 \times N, & \text{else.} \end{cases}$$

In this simulation, in case of TFRC, we assume the transmission rate is 300 kbps with packet loss rate of 1%; while for LIMD, the steady-state transmission rate periodically ranges from 200 kbps to 400 kbps, and the packet loss rate is updated in the same way as that in [1].

Figure 7 and Figure 8 depict the reconstructed video quality under TFRC and LIMD respectively. As can be seen, in both cases, the playback video quality varies noticeably with the conventional packetization scheme (i.e., $W = 1$). Specifically, in Figure 7, the PSNR decreases dramatically for about 5 dB during $8 \le t \le 10$ seconds. Given that there is no change in transmission rate, such quality variation is solely due to the inherent characteristics of the video signal. In Figure 8, the PSNR variation becomes larger from $t = 20$ seconds to $t = 22$ seconds due to transmission rate fluctuation in LIMD. On the other hand, with the introduction of sliding-window, our proposed rate allocation scheme can effectively reduce quality variations. For LIMD, the overall quality becomes much smoother; while for TFRC with $W = 16$, the playback quality becomes nearly constant.

Table 1 and Table 2 further report the performance evaluation of the proposed scheme through the following objective criteria: the average PSNR, μ; the standard deviation of PSNR, σ; and the maximum PSNR difference among 4 adjacent GOPs, Δ, which reflects the quality variation in a short period of 2.6 seconds. Also, the upper bound of the buffer size, B (in bytes), required by the client for application-level buffering, and the start-up delay T_d for pre-fetching the first $W-1$ GOPs are also listed. To be more specific, B is given by $B = N_{\max} \cdot L \cdot W$, where $N_{\max}$ is the maximum number of packets that can be transmitted within one epoch, and T_d is $W-1$ epochs if the first $W-1$ GOPs are pre-fetched with the same packetization strategy as in Section 3. From Table 1 and Table 2, one can see that the buffer capacity and start-up delay are acceptable in streaming applications. Furthermore, by increasing W, since more future GOPs can be taken into account for rate allocation, quality variations in both long term (i.e., σ_Q) and short term (i.e., δ_Q) are reduced substantially. However, the

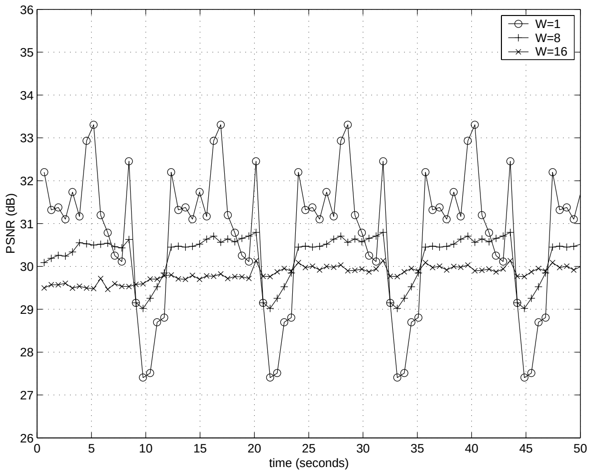

Fig. 7. Performance of sliding-window scheme with $W = 1$ and $W = 16$ when TRFC [4] is employed for rate control

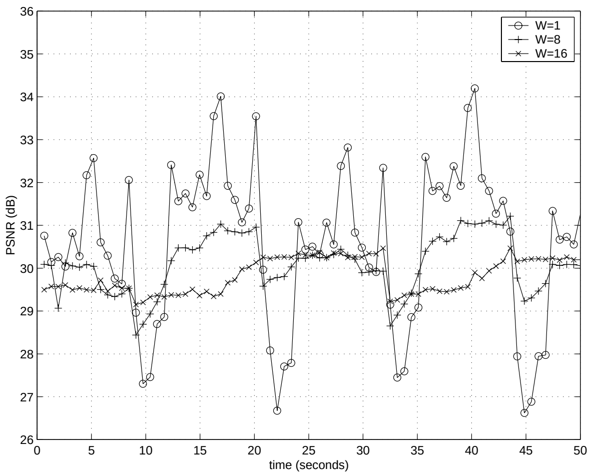

Fig. 8. Performance of sliding-window scheme with $W = 1$ and $W = 16$ when LIMD [3] is employed for rate control.

average PSNR is degraded by about 0.5 and 0.9 dB when W is increased from 1 to 8 and 16, respectively. This is because when one GOP is spread into W epochs and the intra-GOP rate allocation is carried out independently within each epoch, it may introduce more FEC redundancy. Such observation motivates our on-going work to choose adaptive window size for a better trade-off between higher average quality and less quality deviation.

5. CONCLUSION

In this paper, we proposed a sliding-window packetization scheme for transmitting MD-FEC transcoded video across packet erasure networks. In contrast to conventional packetization scheme [1] which transmits one GOP per epoch, the proposed packetization scheme allows the sender to adaptively allocate rate budget among adjacent GOPs. In this way, the quality variations due to both nonstationarity of video signal and transmission rate fluctuation can be effectively reduced.

Table 1. Performance under TFRC [4] rate control

W	$\mu\,(dB)$	$\sigma\,(dB)$	$\Delta\,(dB)$	$B\,(Kbytes)$	T_d (sec)
1	30.76	1.67	5.05	24	0.65
8	30.24	0.55	1.77	192	5.20
16	29.83	0.25	0.41	384	10.40

Table 2. Performance under LIMD [3] rate control

W	$\mu\,(dB)$	$\sigma\,(dB)$	$\Delta\,(dB)$	$B\,(Kbytes)$	T_d (sec)
1	30.62	1.82	6.86	32	0.65
8	30.09	0.64	1.98	256	5.20
16	29.83	0.40	1.24	512	10.40

There are still several related issues that need to be studied more thoroughly in the future. First, simulation would be carried out in a more realistic network environment to study the effect of unstable network state using the proposed scheme. Second, the pre-fetch strategy could be improved to reduce the start-up delay. Finally, sliding window with adaptive window size is under investigation.

6. REFERENCES

[1] R. Puri, K. Lee, K. Ramchandran and V. Bharghavan, "An integrated source transcoding and congestion control paradigm for video streaming in the Internet," IEEE Trans. on Multimedia, Vol. 3, pp. 18-32, Mar. 2001.

[2] B.-J. Kim, Z. Xiong, and W. A. Pearlman, "Low bit-rate scalable video coding with 3D set partitioning in hierarchical trees (3D SPIHT)," IEEE Trans. Circuits Syst. Video Technol., Vol. 10, pp. 1374-1387, Dec. 2000.

[3] T. Kim, S. Lu, and V. Bharghavan, "Improving congestion control through loss differentiation," Proc. International Conference on Computers and Communication Networks, Boston, MA, Oct. 1999.

[4] J. Mahdavi and S. Floyd, "TCP-friendly unicast rate based flow control," End2end-interest mailing list, January 1997.

[5] R. Puri and K. Ramchandran, "Multiple description source coding through forward error correction codes," Proc. Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, Oct. 1999.

[6] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal loss protection: graceful degradation of image quality over packet erasure channels through forward error correction," Proc. Data compression Conference, Snowbird, UT, Mar. 1999.

[7] A. E. Mohr, R. E. Ladner and E. A. Riskin, "Approximately optimal assignment for unequal loss protection," Proc. IEEE ICIP-2000, vol. 1, pp. 367-370, Vancouver, Sept. 2000.

[8] S. Dumitrescu, X. Wu, and Z. Wang, "Globally optimal uneven error-protected packetization of scalable code streams," Proc. DCC'02, Snowbird, Utah, Apr. 2002.

[9] V. Stankovic, R. Hamzaoui, and Z. Xiong, "Packet loss protection of embedded data with fast local search," Proc. IEEE ICIP-2002, Rochester, NY, Sept. 2002.