RATE SHAPING FOR VIDEO WITH FRAME DEPENDENCY[*]

Trista Pei-chun Chen and Tsuhan Chen[+]
Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.
Email: {peichun, tsuhan}@andrew.cmu.edu

ABSTRACT

Streaming of precoded video, which is both source- and channel-coded, over packet-loss networks faces challenges of the time-varying packet loss rate and fluctuating bandwidth. Rate shaping has been proposed to reduce the bit rate of a precoded video bitstream to adapt to the real-time bandwidth variation. In our earlier work, rate shaping was extended to consider not only the bandwidth but also the packet loss rate variations. In practice, the reconstructed result of the previous frame will affect the following frames if the video is predictive coded, and/or the error concealment method performed at the receiver utilizes temporal information. However, none prior work considers rate shaping for pre source- and channel- coded video with such frame dependency. In order to incorporate frame dependency into the rate shaping process, we propose to send the location (and mean) of the corrupted macroblock back to the sender, and use such feedback information to determine the distortion measure in the rate-distortion optimized rate shaping. Experiments have shown improved performance with the aids of the feedback information.

1. INTRODUCTION

Due to the rapid growth of wireless communications, video over wireless networks has gained a lot of attention. Challenges as to cope with the time-varying error rate and fluctuating bandwidth bring out the need of error resilient video transport.

Joint source-channel coding techniques [1][2] are often applied to achieve error resilient video transport with online coding. However, joint source-channel coding techniques are not suitable for streaming *precoded* video. The precoded video is both source- and channel- coded prior to transmission. The network conditions are not known at the time of coding. “Rate shaping”, which was called dynamic rate shaping (DRS) in [3][4], was proposed to “shape”, that is, to reduce, the bit rate of the single-layered pre source-coded (pre-compressed) video, to meet the real time bandwidth requirement.

To protect the video from transmission errors in the wireless networks, source-coded video bitstream is often protected by forward error correction (FEC) codes [5]. Redundant information, known as parity bits, is added to the original source-coded bits. Conventional DRS did not consider shaping for the parity bits in addition to the source-coding bits. In our earlier work, we extended rate shaping for transporting the precoded video that is both pre source- and channel- coded, which we refer to as “baseline rate shaping (BRS)” [6]. By means of discrete rate-distortion (R-D) combination, BRS drops part of the precoded video to achieve the best video quality. The part being dropped can consist of bits from the scalable coding or the parity bits from the FEC coding.

Later, our work on “error concealment aware rate shaping (ECARS)” [7] has also considered the receiver-side error concealment (EC) in the rate shaping process. Knowing the EC method used at the receiver, the *gain* of successfully transmitting one part of the video as opposed to losing it and reconstructing it using EC, can be evaluated. The better the EC can reconstruct the part of the video, the smaller the gain of this part of the video is. We then use this gain in the R-D optimization for ECARS.

Since the video is often predictive coded with standards as H.263 [8] and MPEG-4 [9], and many EC methods used at the receiver utilize temporal information, i.e., information from the previous frames, we need a rate shaping mechanism that considers the frame dependency. By sending back to the sender some auxiliary information about the previous reconstructed frame, we can incorporate frame dependency into the rate shaping process of the current frame. We proposed to use feedback information to determine the distortion measure in the R-D optimized rate shaping.

The paper is organized as follows. In Section 2, we introduce the rate shaping system. R-D optimization problem is formulated and two-stage R-D optimization is illustrated. In Section 3, we propose to modify the distortion measure in the R-D optimized rate shaping to accommodate the effect of frame dependency with the feedback information from the receiver. In Section 4, experiment results are shown. Concluding remarks are given in Section 5.

2. RATE SHAPING SYSTEM

2.1. Wireless Video Transport

In this section, we start from describing the system of wireless video transport, including precoding, streaming with rate shaping for video with frame dependency, and decoding (Figure 1 to Figure 3). We then introduce the R-D optimized rate shaping.

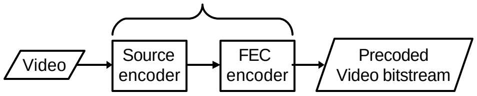


Figure 1. System diagram of the precoding process: MB prioritization followed by source encoding and FEC encoding

---

[*] Work supported in part by Industrial Technology Research Institute.
[+] Corresponding author: Prof. Tsuhan Chen

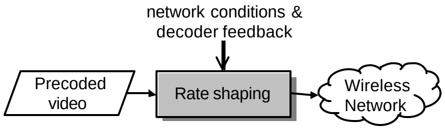


Figure 2. Transport of the precoded video with rate shaping for video with frame dependency

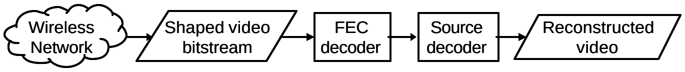
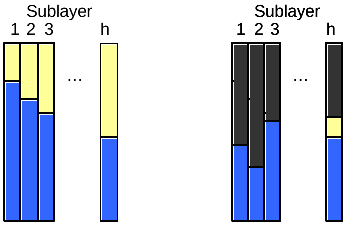


Figure 3. System diagram of the decoding process: FEC decoding followed by source decoding

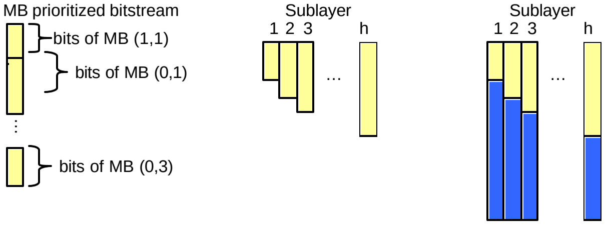
2.2. R-D Optimized Rate Shaping

Given the precoded video, which is both source- and channel-coded, rate shaping will perform bandwidth adaptation in a R-D optimized manner. Suppose the precoded video is given with several *sublayers*. Each sub layer consists of symbols[1] from the source coding, which is shown as the upper portion of each stripe in Figure 4 (a), and symbols from the channel coding, which is shown as the lower portion of each stripe in Figure 4 (a). The construction of the sublayers with the knowledge of the EC method used at the receiver is shown in Figure 5. Interested readers can refer to [7] for more information.



(a) (b)

Figure 4. (a) Precoded video in sublayers and (b) rate shaping decision on which symbols to send



(a) (b) (c)

Figure 5. Precoded video: (a) MB prioritized bitstream, (b) MB prioritized bitstream in sublayers, and (c) FEC coded MB prioritized bitstream

The problem formulation is as follows. The total gain is increased (or the total distortion is decreased) as more sublayers are correctly decoded. With Sublayer 1 correctly decoded, the total gain is increased by G_1 (accumulated gain is G_1); with Sublayer 2 correctly decoded, the total gain is increased further by G_2 (accumulated gain is $G_1 + G_2$); and so on. Note that G_i of Sublayer i is calculated given the EC method used at the receiver. For example, G_i can be the square of the residue between the true sublayer and its error-concealed version without sending this sublayer. G_i is different with different EC method. Note that we should reconsider about the value of G_i knowing there is frame dependency. Detailed explanation will be given in the next section. The expected accumulated gain is then:

$$G = \sum_{i=1}^{h} G_i v_i \qquad (1)$$

where v_i is the recovery rate of Sublayer i. With Reed-Solomon codes used in this paper, Sublayer i is recoverable (or successfully decodable) if the number of erasures is no more than $r_i - k_i$. k_i is the message (symbols from the source coding) size in Sublayer i and r_i is the number of symbols selected to send in Sublayer i. Thus, the recovery rate v_i is the summation of the probabilities that no loss occur, one erasure occurs, and so on until $r_i - k_i$ erasures occur.

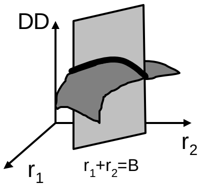


Figure 6. Intersection of the model-based hyper-surface and the bandwidth constraint, illustrated with $h = 2$

Stage 1 of the two-stage R-D optimization gives a near-optimal solution by intersecting the model-based hyper-surface with the bandwidth constraint. The solution can be refined by a hill-climbing based approach (Figure 7). We perturb the solution from Stage 1 to yield a larger accumulated gain under the bandwidth constraint. The process can be iterated until the solution reaches a stopping criterion such as the convergence.

```
While (stop == false)
    $z_i$ = $r_i$ for all i=1~h
    For (j=1; j<=h; j++)
        For (k=1; k<=h; k++)
            $z_k$ = $z_k$ + delta for k==j //Increase sublayer j
            $z_k$ = $z_k$ - delta/(h-1) for k!=j //Decrease others
        End-for
        Evaluate $G_j$ by equations (1) and (2)
    End-for
    Find the j* with the largest $G_{j^*}$.
    For (i=1; i<=h; i++)
        $r_i$ = $r_i$ + delta for i==j*
        $r_i$ = $r_i$ - delta/(h-1) for i!=j*
    End-for
    Calculate the stop criterion.
End-while
```

Figure 7. Pseudocodes of a hill-climbing algorithm

3. DISTORTION MEASURE FOR VIDEO WITH FRAME DEPENDENCY

---

[1] "Symbols" are used instead of "bits" since the FEC codes use a symbol as the encoding/decoding unit. In this paper, we use 14 bits to form one symbol. The selection of symbol size in bits depends on the user.

As we have discussed in the last section, the goal of rate shaping is to achieve the maximum accumulated gain. We should consider frame dependency in the process of R-D optimization, since the reconstructed result of the previous frame will affect the following frames if the video is predictive coded, and/or the error concealment method performed at the receiver utilizes temporal information. We propose to use feedback information from the receiver to carry information about the previous reconstructed frame for the use of the current frame in rate shaping.

In this paper, the feedback information for rate shaping can consist of (1) the location of the corrupted macroblock; or (2) the mean of the corrupted macroblock in addition to the location of the corrupted macroblock. In the later experiments, we will use “none” to represent no feedback is used, “loc” to represent feedback with the location of the corrupted macroblock is used, and “mean” to represent feedback with the location and mean of the corrupted macroblock is used.

For the case of “loc”, we will explain how to determine the gain G_i' with respect to three cases where frame dependency can occur. The three cases are denoted as (0,1), (1,0), and (1,1). An “1” in the first field represents INTER coding, and an “1” in the second field represents an EC method that utilizes temporal information. Likewise, we will explain how to determine the gain with respect to (0,1), (1,0), and (1,1) for the case of “mean”.

In general, the gain G_i' for each macroblock remains the same as G_i if the corresponding macroblock of the previous frame is successfully decoded. We then see how to determine the gain G_i' for each macroblock if the corresponding macroblock of the previous frame is corrupted.

3.1. Feedback with the location information “loc”

- (0,1): If a macroblock of the previous frame is corrupted, we want to increase the gain of the corresponding macroblock of the current frame. Since temporal EC will use information from the corrupted macroblock of the previous frame if the corresponding macroblock of the current frame cannot be decoded successfully, we want to make sure the macroblock of the current frame is sent with good protection. A natural way is to double the value of the gain as:

$$G_i' = 2G_i \qquad (2)$$

- (1,0) and (1,1): If a macroblock of the previous frame is corrupted, since INTER coding will use it for predictive coding, we want to decrease the gain of the corresponding macroblock of the current frame. Sending the residues of the macroblock of the current frame is useless if the prediction to this macroblock is already erroneous. A natural way is to set the gain to zero as:

$$G_i' = 0 \qquad (3)$$

3.2. Feedback with the location and mean information “mean”

Motivated by “how different” a corrupted macroblock of the previous frame is from a successfully decoded one, we determine the gain of the macroblock of the current frame. The further apart the corrupted macroblock is from its successfully decoded one, the more we should modify the gain of the macroblock of the current frame. The corrupted macroblock of the previous frame will affect either the prediction of the macroblock of the current frame, the EC reconstruction of the macroblock of the current frame, or both. We propose to send the mean of the prediction of the macroblock of the current frame p' and the mean of the EC reconstruction of the macroblock of the current frame c', both affected by the corrupted macroblock of the previous frame, back to the sender for calculation of G_i'.

Note that the gain of a macroblock G is originally defined as:

$$G_i = -\sum_m \left[(p_m + s_m - t_m)^2 - (c_m - t_m)^2 \right] \qquad (4)$$

where p_m is the value of the correct prediction of the macroblock of the current frame, c_m is the value of the correct EC reconstruction of the macroblock of the current frame, s_m is the value of what will be sent for the macroblock of the current frame, and t_m is the true value before quantization/compression. Inspired by (4) and the assumption that mean of a macroblock represents this macroblock well, we approximate the gain considering the frame dependency as:

$$G_i = -\left[(p + s - t)^2 - (c - t)^2 \right] \qquad (5)$$

where p, c, s, and t represent the means. As motioned, if the macroblock of the previous frame is corrupted, the prediction and concealment results will be affected to form p' and c'. With another assumption that $t \approx p + s$, we can express G_i' in terms of G_i as:

$$G_i' = \frac{-\left[(p' - p)^2 - (c' - p - s)^2 \right]}{(c - p - s)^2} G_i \qquad (6)$$

(6) can be applied to the calculation of the gain of (1,1).

- (0,1): When the macroblock is INTRA coded, there is no prediction associated with this macroblock. Therefore, (6) can be simplified to:

$$G_i' = \frac{(c' - s)^2}{(c - s)^2} G_i \qquad (7)$$

- (1,0): When the EC method does not utilize the information from the previous frame, $c' = c$. Therefore, (6) can be simplified to:

$$G_i' = \left(1 - \frac{(p' - p)^2}{(c - p - s)^2} \right) G_i \qquad (8)$$

After determining the gain G_i', we continue with the R-D optimized rate shaping described in the last section.

4. EXPERIMENT

In the experiment, we will show results of the proposed rate shaping methods “none”, “loc”, and “mean”, compared with the naïve rate shaping method “unequal error protection rate shaping (UPPRS)” described in Figure 8. UPPRS will drop from the bottom if the bandwidth is not enough. In that, UPP can be achieved since more parity symbols are sent for Sublayer i than Sublayer $i+1$.

Wireless networks are generally with time-varying packet loss rate and fluctuating bandwidth. The packet loss rate and bandwidth vary at each time interval. We simulate random bandwidth fluctuation and use the Gilbert-Elliot model [10][11] to simulate the bursty wireless bit errors. The test video sequences are “akiyo”, “foreman”, and “stefan” in common intermediate format (CIF) (Figure 9 (a)-(c)). We use H.263 [8] for video encoding. The frame rate is 30 frames/sec. All results in the following are shown for the luminance Y components only.

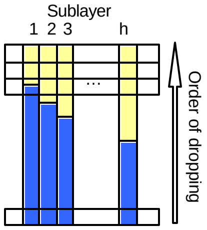


Figure 8. UPPRS

(a) (b) (c)

Figure 9. Test video sequences in CIF: (a) akiyo, (b) foreman, and (c) stefan

PSNR results are shown in Table 1 to Table 3 for frame dependency cases (0,1), (1,0), and (1,1), respectively. We can see the proposed rate shaping methods with R-D optimization—“none”, “loc”, and “mean”, perform much better than the naïve method UPPRS for about 10dB! The rate shaping methods considering frame dependency—“loc” and “mean”, in general outperform the rate shaping method with no feedback—“none”, by 0~1dB. We also see that rate shaping method “mean” outperforms the rest in most cases, while “loc” ranks the second in terms of overall performance.

Table 1. PSNR of sequences “akiyo”, “foreman”, and “stefan” using methods “UPPRS”, “none”, “loc”, and “mean” in (0,1)

PSNR (dB)	UPPRS	none	loc	mean
Akiyo	23.58	36.80	37.60	37.87
Foreman	20.32	27.18	28.14	28.25
Stefan	18.89	23.05	24.15	23.70

Table 2. PSNR of sequences “akiyo”, “foreman”, and “stefan” using methods “UPPRS”, “none”, “loc”, and “mean” in (1,0)

PSNR (dB)	UPPRS	none	loc	mean
Akiyo	29.29	37.18	37.18	37.18
Foreman	21.20	30.26	30.27	30.41
Stefan	19.01	23.13	23.85	23.47

Table 3. PSNR of sequences “akiyo”, “foreman”, and “stefan” using methods “UPPRS”, “none”, “loc”, and “mean” in (1,1)

PSNR (dB)	UPPRS	none	loc	mean
Akiyo	30.61	37.12	37.23	37.13
Foreman	21.66	30.73	30.77	30.88
Stefan	18.79	23.04	23.43	23.96

5. CONCLUSION

We proposed in this paper rate shaping methods for pre source- and channel- coded video considering the frame dependency. To incorporate frame dependency into the rate shaping process, the receiver sends the location (and mean) of the corrupted macroblock back to the sender. Such feedback is used in the sender-side rate shaping to determine the distortion measure in the R-D optimization. Experiments have shown improved performance with the aids of the feedback information. It was also shown in the experiments that rate shaping with feedback consisting of both the location and mean information slightly outperforms rate shaping with feedback consisting of only the location information. We will study in the future to find alternative feedback information to mean to improve the rate shaping performance.

REFERENCES

[1] G. Cheung and A. Zakhor, “Bit Allocation for Joint Source/Channel Coding of Scalable Video”, *IEEE Transactions on Image Processing*, 9(3), March 2000.

[2] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, “Joint Source-Channel Coding for Motion-Compensated DCT-based SNR Scalable Video”, *IEEE Transactions on Image Processing*, 11(9), September 2002.

[3] W. Zeng and B. Liu, “Rate Shaping by Block Dropping for Transmission of MPEG-precoded Video over Channels of Dynamic Bandwidth”, *ACM Multimedia 96*, Boston, MA, U.S.A, 1996.

[4] S. Jacobs and A. Eleftheriadis, “Streaming Video Using Dynamic Rate Shaping and TCP Congestion Control”, *Journal of Visual Communication and Image Representation*, 9(3), 211-222, 1998.

[5] S. Wicker, Error Control Systems for Digital Communication and Storage, Prentice-Hall, 1995.

[6] T. P.-C. Chen and T. Chen, “Adaptive Joint Source-Channel Coding using Rate Shaping”, *ICASSP 2002*, Orlando, FL, U.S.A., May 2002.

[7] T. P.-C. Chen and T. Chen, “Error Concealment Aware Rate Shaping for Wireless Video Transport”, *Packet Video 2003*, Nantes, France, April 2003.

[8] D. S. Turaga and T. Chen, “Fundamentals of Video Compression: H.263 as an Example”, in Compressed Video over Networks, edited by M.-T. Sun and A. R. Reibman, Marcel Dekker, Inc., 2001.

[9] Motion Pictures Experts Group, "Overview of the MPEG-4 Standard", ISO/IEC JTC1/SC29/WG11 N2459, 1998.

[10] H. S. Wang and N. Moayeri, “Finite-State Markov Channel-A Useful Model for Radio Communication Channels”, *IEEE Trans. On Vehicular Technology*, 44(1), February 1995.

[11] J.-P. Ebert and A. Willig, “A Gilbert-Elliot Bit Error Model and the Efficient Use in Packet Level Simulation”, TKN *Technical Reports Series of Technical University Berlin*, March 1999.