

Error Concealment Aware Rate Shaping for Wireless Video Transport¹

Trista Pei-chun Chen and Tsuhan Chen²

Abstract— Streaming of video, which is both source- and channel- coded, over wireless networks faces the challenge of time-varying packet loss rate and fluctuating bandwidth. Rate shaping (RS) 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. Rate-distortion optimized rate adaptation is performed on the precoded video that is a scalable coded bitstream protected by forward error correction codes. In this paper, we propose a rate shaping scheme that further takes into account the error concealment (EC) method used at the receiver. We refer to this scheme as EC aware RS (ECARS). When performing ECARS, first ECARS needs to know the benefit/*gain* of sending each part of the precoded video, as opposed to not sending it but reconstructing it by EC. Then given a certain packet loss probability, the expected gain can be derived and be included in the rate-distortion optimization problem formulation. Finally ECARS performs rate-distortion optimization to adapt the rate of the precoded video. A two-stage rate-distortion optimization approach is proposed to solve the ECARS rate-distortion optimization problem. In addition to ECARS, the precoding process can be EC aware to prioritize the precoded video based on the gain. We present an example EC aware precoding process by means of macroblock prioritization. Experiment results of ECARS together with EC aware precoding are shown to have excellent performance.

Index Terms—rate shaping, error concealment, rate-distortion optimization, wireless video

This paper is accepted for publication on EURASIP Signal Processing: Image Communication, Special Issue on Recent Advances in Wireless Video, 2003.

¹ Work supported in part by Industrial Technology Research Institute.

² The authors are with Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A. Corresponding author: Prof. Tsuhan Chen, Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A. Tel: (412) 268-7536 Fax (412) 268-3890, E-mail: tsuhan@cmu.edu.

I. 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 *pre-coded* video. The pre-coded 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]-[7], 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. In [3]-[5], it was proposed to drop the discrete cosine transform (DCT) coefficients beyond the “breakpoint” to reduce the bit rate of the pre-compressed video; while in [6][7], it was proposed to drop some blocks in a frame and reconstruct those that were dropped by interpolation at the receiver, to reduce the bit rate of the pre-compressed video.

To protect the video from transmission errors in the wireless networks, source-coded video bitstream is often protected by forward error correction (FEC) codes [8]. Redundant information, known as parity bits, is added to the original source-coded bits. Parity bits are included in the pre-coded video because FEC encoding at the time of transmission may not be feasible given the capability of the node that is transporting the video. On the other hand, this node should be able to perform rate shaping for both the source- and channel- coded bitstream since rate shaping has less complexity than full decoding. This node is able to perform full decoding if it wants to view the content of the video.

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 pre-coded video that is both pre

source- and channel- coded [9], which we refer to as “baseline rate shaping (BRS)”. The source coding in particular refers to scalable video coding as used by H.263 [10] and MPEG-4 [11]. 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.

In this paper, we propose a rate shaping scheme that further takes into account the error concealment (EC) method used at the receiver. We refer to this scheme as EC aware RS (ECARS). Related work that utilized EC information for rate shaping on pre source- coded bitstream only can be found in [6][7]. When performing ECARS, first ECARS needs to know the benefit/*gain* of sending each part of the precoded video, as opposed to not sending it but reconstructing it by EC. The gain of sending some part of the precoded video is large if the EC method used at the receiver cannot reconstruct this part very well. Such gain will be different if the EC method considered is different. Gain information can either be computed at the time of transmission or be embedded in the bitstream. Then given a certain packet loss probability, the expected gain can be derived and be included in the R-D optimization problem formulation. Finally ECARS performs R-D optimization to adapt the rate of the precoded video. A two-stage R-D optimization approach is proposed to solve the ECARS R-D optimization problem. Prior work on R-D optimization includes [12]-[15], where [12]-[14] solved the linear programming/integer programming problem by Lagrangian multiplier with pruning and iterative bisection, and [15] used a hill climbing based approach called sensitivity adaptation (SA) algorithm. The proposed two-stage R-D optimization aims for both efficiency and optimality by using the model-based hyper-surface and the hill climbing based refinement.

In addition to ECARS, the precoding process can be EC aware to prioritize the precoded video based on the gain. We present an example EC aware precoding process by means of macroblock (MB) prioritization. A MB in a frame is ranked according to its gain, which depends on how well this MB can

be reconstructed by the EC method used at the receiver. The gain of sending a MB is large if the EC method used at the receiver cannot reconstruct this MB very well.

This paper is organized as follows. In Section II, we introduce baseline rate shaping (BRS) and error concealment (EC) as the background. In Section III, “error concealment aware rate shaping (ECARS)” is proposed. Given any precoded video, ECARS first evaluates the gain considering a particular EC method used at the receiver. ECARS then performs a two-stage R-D optimization for rate adaptation under the current network condition, in terms of packet loss rate and bandwidth. In addition, we also introduce EC aware precoding where a MB prioritization scheme is presented. In Section IV, experiment results of ECARS together with EC aware precoding are shown. Concluding remarks are given in Section V.

II. BACKGROUND

We will give brief descriptions of baseline rate shaping (BRS) and error concealment (EC) in this section. BRS provides a simple illustration of what is involved in rate shaping for pre source- and channel- coded video. In addition, since the proposed ECARS takes into account error concealment for rate shaping, we also describe briefly error concealment techniques that may be used at the receiver.

A. *Baseline Rate Shaping (BRS)*

There are three stages to transmit the video from the sender to the receiver: (i) precoding, (ii) streaming with BRS, and (iii) decoding, as shown from Figure 1 to Figure 3.

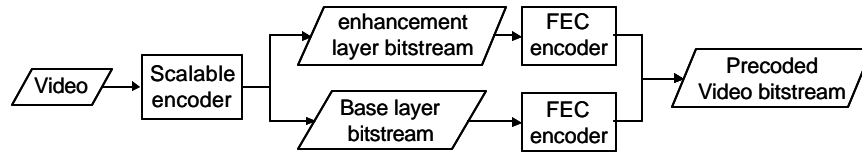


Figure 1. System diagram of the precoding process: scalable encoding followed by FEC encoding

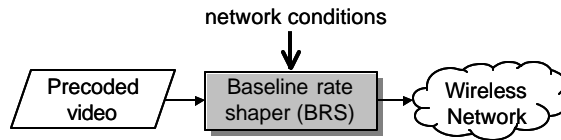


Figure 2. Transport of the precoded video with BRS

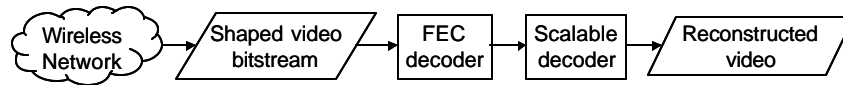


Figure 3. System diagram of the decoding process: FEC decoding followed by scalable decoding

BRS reduces the bit rate of each *decision unit* of the precoded video before it sends the precoded video to the wireless network. A decision unit can be a frame, a macroblock, etc., depending on the granularity of the decision. We use a frame as the decision unit herein. Let us consider the case in which the video sequence is scalable coded into two layers: one base layer and one enhancement layer. These two layers are FEC coded with unequal packet loss protection (UPP) capabilities. Therefore, there are four *segments* in the precoded video. The first segment consists of the bits of the base layer video bitstream (upper left segment of Figure 4 (a)). The second segment consists of the bits of the enhancement layer video bitstream (upper right segment of Figure 4 (a)). The third segment consists of the parity bits for the base layer video bitstream (lower left segment of Figure 4 (a)). The fourth segment consists of the parity bits for the enhancement layer video bitstream (lower right segment of Figure 4

(a). BRS decides a subset of the four segments to send. When the channel has abundant bandwidth, BRS will send with the configuration shown in Figure 4 (a). When the bandwidth is reduced, the second configuration shown in Figure 4 (b) is chosen. When the bandwidth is reduced even more, either Figure 4 (c) or Figure 4 (d) will be chosen depending on the wireless network condition. A rule of thumb is to choose parity bits to send instead of bits of the enhancement layer when the packet loss rate is high. In the extreme case where the bandwidth is so limited, none of the segments will be chosen to be sent as shown in Figure 4 (f). Interested readers can read more from [9], which consists of BRS by mode decision that we just describe and the discrete R-D combination.

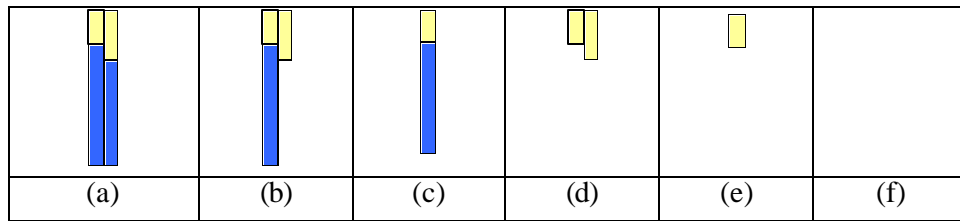


Figure 4. Six different combinations of subset of the four segments

B. Error Concealment (EC)

Error concealment relies on some a priori knowledge to reconstruct the lost video content. Such a priori knowledge can come from spatial or temporal neighbors. For example, we can assume that the pixel values are smooth across the boundary of the lost and retained regions. To recover lost data with the smoothness assumption, interpolation or optimization based on certain objective functions are often used. Figure 5 and Figure 6 show corrupted frames and the corresponding reconstructed frames. The black regions in Figure 5 (a) and Figure 6 (a) indicate losses of the video data. Figure 5 shows an error concealment method using spatial interpolation from the neighboring pixels. Figure 6 shows an error concealment method using temporal interpolation. That is, if some pixel values are lost, the decoder

copies the pixel values from the previous frame at the corresponding locations to the current frame. The error concealment method using temporal interpolation can be extended to copying the pixel values from the previous frame at the motion-compensated locations. The motion vectors used for motion compensation either are assumed error-free or can be estimated at the decoder [16][17].

We use the simple temporal interpolation method in this paper. Future extension includes using motion-compensated temporal interpolation, or more sophisticated error concealment methods as mentioned in [18].

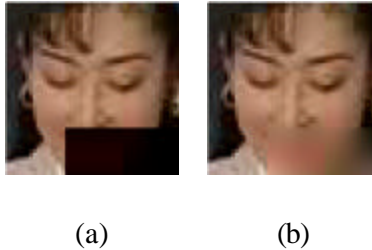


Figure 5. Error concealment example by spatial interpolation: (a) the corrupted frame without error concealment, and (b) the reconstructed frame with error concealment

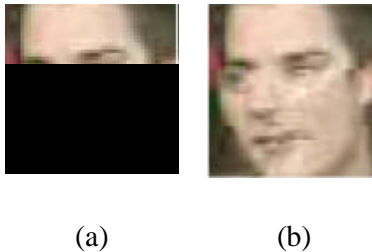


Figure 6. Error concealment example by temporal interpolation: (a) the corrupted frame without error concealment, and (b) the reconstructed frame with error concealment

III. ERROR CONCEALMENT AWARE RATE SHAPING (ECARS)

In this section, we will start from describing the wireless video transport system, including precoding, streaming with rate shaping, and decoding. We then propose the EC aware RS scheme (ECARS), that first evaluates the *gains*, which we will define formally, considering a particular EC method used at the receiver, then performs the two-stage R-D optimization. In addition, if the system allows for EC aware precoding, ECARS can take advantage of that. We will present an EC aware precoding process by means of MB prioritization.

A. Wireless Video Transport System

There are three stages to transmit the video from the sender to the receiver in a wireless video transport system: (i) precoding, (ii) streaming with rate shaping, and (iii) decoding, as shown from Figure 7 to Figure 9. In the precoding process (shown in Figure 7), video is encoded by both the source encoder and the FEC encoder. The precoding process is done before the time of delivery. The precoding process may be aware of the EC used at the receiver, which we will describe later. Notice that in this paper, the precoded video for ECARS is pre source-coded with a single layer. In the streaming stage (shown in Figure 8), ECARS takes the network conditions as the bandwidth and the packet loss rate into account to achieve the best video quality. The decoding process (shown in Figure 9) consists of FEC decoding followed by scalable decoding.

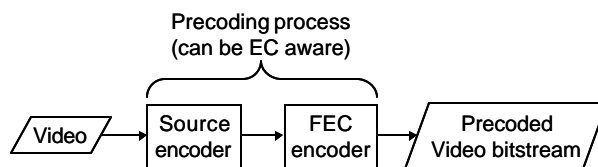


Figure 7. System diagram of the precoding process: source encoding (which can be EC aware) followed by FEC encoding

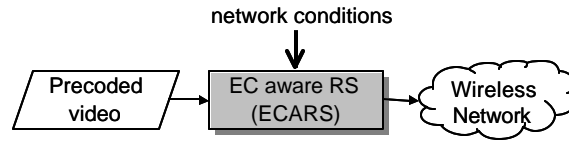


Figure 8. Transport of the precoded video with ECARS

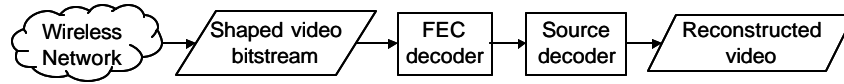


Figure 9. System diagram of the decoding process: FEC decoding followed by source decoding

B. R-D Optimization for ECARS

Given the precoded video, which is both source- and channel- coded, ECARS will perform bandwidth adaptation for streaming. We start from a simple example as an extension to BRS then give a more general ECARS.

Let us consider that the precoded video consists of two layers of video bitstream, namely, the base layer and the enhancement layer. Each layer is protected by parity bits from the FEC coding. The setting is shown earlier in Figure 4 (a). The rate shaper is extended to give a finer decision on how many symbols³ to send (or how many symbols to drop) for each layer, instead of deciding which segment(s) to drop as suggested by BRS. Since the rate shaper is aware of the EC method used at the receiver, it can evaluate how much distortion decrease it can get in if the rate shaper decides to send a certain amount of symbols for each layer. In general, the base layer can be reconstructed well with error concealment since the base layer consists of coarse information of the video that can be easily reconstructed. On the other

³ “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.

hand, the enhancement layer, which consists of fine details of the video, cannot be easily reconstructed. More distortion decrease could be obtained if the rate shaper decides to send the enhancement layer video. In this case, the EC aware rate shaper would assign a higher gain (distortion decrease) on sending symbols from the enhancement layer than the symbols from the base layer.

Note again that in this paper, the precoded video for ECARS is pre source-coded with a single layer. This single layer of video bitstream will be arranged into sublayers, which we will define shortly. The sublayers shall not be confused with the two-layered example given in the last paragraph for illustration purpose only.

Having understood how the gain of sending some part of the precoded video is determined considering the EC used at the receiver, we can now introduce a more general ECARS. Suppose ECARS is given the precoded video consisting of several *sublayers*. The sublayers are usually arranged in a way that the lower sublayers are more important in reconstructing the video quality than the higher sublayers are. That is, lower sublayers are associated with larger sublayer gains G_i 's, where i is the sublayer index; and higher sublayers are associated with smaller sublayer gains G_i 's. We will describe in more detail in Section III. C such a precoding process and definition of the sublayer gains. As shown in Figure 10 (a), the upper portion of each stripe consists of the symbols from source coding, and the lower portion of each stripe consists of the symbols from channel coding. The darken bars in Figure 10 (b) represent the symbols to be sent by ECARS.



(a)

(b)

Figure 10. (a) Precoded video in sublayers and (b) ECARS decision on which symbols to send

The problem formulation for ECARS 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, thus EC aware. G_i of Sublayer i is different for every frame. Since the precoded video is transmitted over error prone wireless networks, sublayers are subject to loss and have certain recovery rates given a particular rate shaping decision. The expected accumulated gain is then:

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

if each sublayer can be decoded independently⁴. v_i is the recovery rate of Sublayer i that is a function of r_i as shown later in (2). Using Reed-Solomon codes as the channel codes in this paper, Sublayer i is recoverable (or successfully decodable) if the number of erasures resulting from the lossy transmission is no more than $r_i - k_i$. k_i is the message (symbols from the source coding) size of Sublayer i and r_i is the number of symbols selected to be sent in Sublayer i . With Reed-Solomon codes used, $r_i \geq k_i$ with the exception of the last sublayer (not necessary the Sublayer h , can be the sublayer before that); and the whole sublayer is considered lost if the number of erasures is beyond the error-correction

⁴ If Sublayer i can be decoded only if Sublayer $i-1$ is decoded correctly, (1) can be modified to $G = \sum_{i=1}^h G_i \prod_{j=1}^i v_j$.

capability $r_i - k_i$. 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.

$$v_i = \sum_{l=0}^{r_i - k_i} \left[\binom{r_i}{l} (e_m)^l (1 - e_m)^{r_i - l} \right], i = 1 \sim h \quad (2)$$

where h is the number of sublayers of this frame in total and e_m is the symbol loss rate. The symbol loss rate can be derived from the packet loss rate as $e_m = 1 - (1 - e_p)^{m/s}$, where s is the packet size and m is the symbol size in bits. By choosing different combinations of the number of symbols for each sublayer, the expected accumulated gain will be different. The rate shaping problem can be formulated as follows:

$$\begin{aligned} & \text{maximize} && G = \sum_{i=1}^h G_i v_i \\ & \text{subject to} && \sum_{i=1}^h r_i \leq B \end{aligned} \quad (3)$$

where B is the bandwidth constraint this frame has to satisfy. To solve this problem, we propose a new two-stage R-D optimization approach. The two-stage R-D optimization first finds the near-optimal solution globally. The near-optimal global solution is then refined by a hill climbing approach. Prior work on R-D optimization includes [12]-[15]. The proposed two-stage R-D optimization is different from [12]-[15] in two folds. First, the model-based Stage 1 allows us to examine fewer samples from all the operational R-D states. Second, the proposed distortion measure (or “expected accumulated gain” in the terminology of this paper) accounts for the effects of packet loss as well as the channel codes by means of recovery rates.

1) *Two-stage R-D Optimization: Stage 1*

We can see from (1) and (2) that the expected accumulated gain G is related to $\mathbf{r} = [r_1 \ r_2 \ \dots \ r_h]$ implicitly through the recovery rates $\mathbf{v} = [v_1 \ v_2 \ \dots \ v_h]$. We can instead find a model-based hyper-surface that explicitly relates \mathbf{r} and G . The model parameters can be trained from a set of training data (\mathbf{r}, G) , where \mathbf{r} values are chosen by the user and G values can be computed by (1) and (2). The optimal solution is the feasible solution within the intersection of the hyper-surface and the bandwidth constraint as illustrated in Figure 11. A complex model, with a lot of parameters, can be used to describe as close as possible the true distribution of the R-D states. The solution obtained from the intersection will be as close to optimal as possible. However, the number of (\mathbf{r}, G) pairs needed to train the model-based hyper-surface increases with the number of parameters.

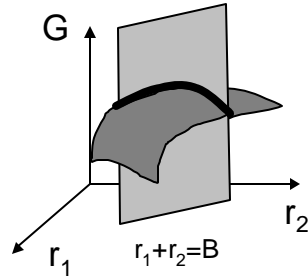


Figure 11. Intersection of the model-based hyper-surface (dark surface) and the bandwidth constraint (gray plane), illustrated with $h = 2$

In this paper, we use a quadratic equation to describe the relation between \mathbf{r} and G as follows:

$$G = \sum_{i=1}^h a_i r_i^2 + \sum_{i,j=1,i \neq j}^h b_{ij} r_i r_j + \sum_{i=1}^h c_i r_i + d \quad (4)$$

In this paper, the model parameters a_i , b_{ij} , c_i , and d are trained differently for each frame. They can be solved by surface fitting with a set of training data (\mathbf{r}, G) obtained from (1) and (2). For example, the parameters can be computed by:

$$\begin{pmatrix} a_i \text{'s} \\ b_{ij} \text{'s} \\ c_i \text{'s} \\ d \end{pmatrix} = (R^T R)^{-1} R^T \begin{pmatrix} {}^1G \\ {}^2G \\ \vdots \\ {}^\Xi G \end{pmatrix} \quad (5)$$

where the left super index of G is the index of the training data, R is a matrix consisting Ξ rows of $(r_i^2 \text{'s}, r_i r_j \text{'s}, r_i \text{'s}, 1)$. The complexity of computing a_i 's, b_{ij} 's, c_i 's, and d relates to the number of parameters $h^2 + h + 1$ and the number of training data Ξ , using (5). Note that the number of training data Ξ is in general much greater than the number of parameters $h^2 + h + 1$. Thus, a more complex model, such as a third-order model with $h^3 + h^2 + h + 1$ parameters, will not be suitable since it requires much more training data. In addition, Second-order Taylor expansion can approximate nicely in general every function. (4) can be seen as a second-order approximation to (1)(2). To reduce the computation complexity in reality, we can also choose a smaller h .

With (4), the near-optimal solution can be obtained by Lagrangian multiplier as follows.

$$J = \left(\sum_{i=1}^h a_i r_i^2 + \sum_{i,j=1, i \neq j}^h b_{ij} r_i r_j + \sum_{i=1}^h c_i r_i + d \right) + \mathbf{I} \left(\sum_{i=1}^h r_i - B \right) \quad (6)$$

By $\frac{\partial J}{\partial r_i} = 0$, we get:

$$r_i = \frac{-1}{2a_i} \left(\sum_{j=1, j \neq i}^h b_{ij} r_j + c_i + \mathbf{I} \right) \quad (7)$$

where \mathbf{I} is:

$$I = \frac{2B + \sum_{i=1}^h \frac{1}{a_i} \left(\sum_{j=1, j \neq i}^h b_{ij} r_j + c_i \right)}{- \sum_{i=1}^h \frac{1}{a_i}} \quad (8)$$

The near-optimal solution can be solved recursively using (7) and (8), starting from the initial condition

that all sublayers are allocated with equal number of symbols, $r_1 = r_2 = \dots = r_h = \frac{B}{h}$.

2) Two-stage R-D Optimization: Stage 2

Stage 1 of the two-stage R-D optimization gives a near-optimal solution. The solution can be refined by a hill-climbing based approach (Figure 12). The solution from Stage 1 is perturbed in order to yield a larger expected accumulated gain. The process can be iterated until the solution reaches a stopping criterion such as the convergence.

```

While (stop == false)
  zi = ri for all i=1~h
  For (j=1; j<=h; j++)
    For (k=1; k<=h; k++)
      zk = zk + delta for k==j //Increase sublayer j
      zk = zk - delta/(h-1) for k!=j //Decrease others
    End-for
    Evaluate Gj by equations (1) and (2)
  End-for
  Find the j* with the largest Gj*.
  For (i=1; i<=h; i++)
    ri = ri + delta for i==j*
    ri = ri - delta/(h-1) for i!=j*
  End-for
  Calculate the stop criterion.
End-while

```

Figure 12. Pseudocodes of hill-climbing algorithm

C. Error Concealment Aware Precoding

In addition to ECARS, the precoding process can be EC aware to prioritize the precoded video based on the *gain*. We present an example EC aware precoding process by means of macroblock (MB) prioritization. A MB in a frame is ranked according to its gain, which depends on how well this MB can be reconstructed by the EC method used at the receiver. The gain of sending a MB is large if the EC method used at the receiver cannot reconstruct this MB very well.

Let us consider that a simple temporal interpolation based EC method is adopted. Figure 13 provides us with an illustration of EC aware MB prioritization. If MB (1,1) is lost in Frame n , it cannot be well reconstructed by MB (1,1) from Frame $n-1$. On the other hand, if MB (0,3) is lost in Frame n , it can be well reconstructed by MB (0,3) from Frame $n-1$. Therefore, we should rank MB (1,1) with higher priority than MB (0,3).

We can use square sum of the pixel differences between the original MB and the EC-reconstructed MB as the measure for priority. The larger the square sum is, the larger the gain for this MB is, thus, the higher the priority of this MB is. Assuming that the neighboring MB of the MB considered are decoded without errors, the MB gain g_j is defined as follows:

$$g_j = \sum_{u=0}^{255} (c_{ju} - p_{ju} - s_{ju})^2, \quad j=1 \sim \text{number of MB in a frame} \quad (9)$$

where u ⁵ is the coefficient index in a MB, c_{ju} is the coefficient of the EC-reconstructed MB, p_{ju} is the prediction value of this MB, and s_{ju} is the residue value of this MB. $p_{ju} + s_{ju}$ is the ideal value without any transmission error or rate adaptation by rate shaping. $c_{ju} - (p_{ju} + s_{ju})$ is to see how far the

⁵ We consider only the Y components in the MB without loss of generality. Thus, there are four 8×8 blocks or 256 coefficients inside.

EC value is from the ideal value. The assumption that the neighboring MB are decoded without errors is valid if the packet losses are not too bursty.

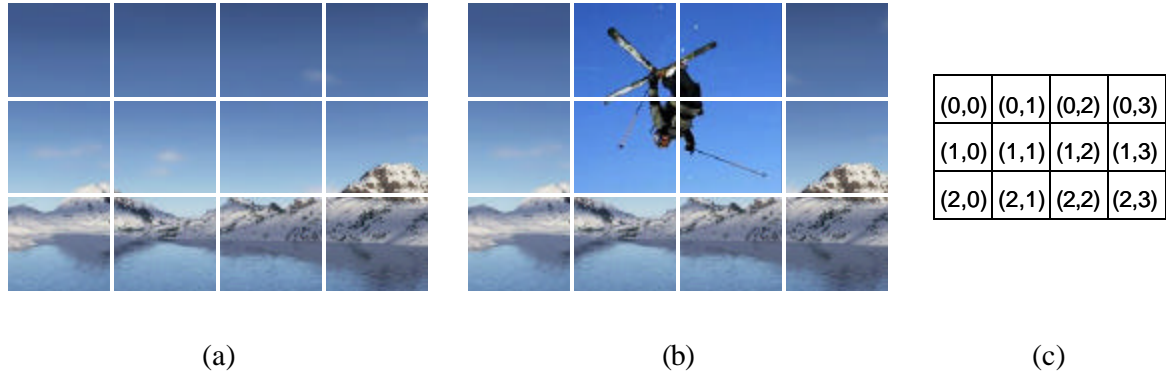


Figure 13. (a) Frame $n - 1$, (b) Frame n , and (c) MB indices. EC aware MB prioritization— MB (1,1) has higher priority than MB (0, 3)

An observation to make is that the conventional video coding can be considered as a special case of the proposed EC aware MB prioritization. Let us consider the case where no motion vector is used in video coding. The MB with large residues is encoded and transmitted, while the MB with small residues does not need to be transmitted since the small residues will become zero after quantization. This case translates to the case of EC aware MB prioritization using temporal interpolation with zero motion vectors. Let us consider another case where motion vectors are included in video coding. This then translates to the case of EC aware MB prioritization using temporal interpolation with motion vectors. We can see that the proposed EC aware MB prioritization is more general since it is not limited to any specific error concealment method.

The source-coded bitstream with EC aware MB prioritization can be appended with parity bits from the FEC coding. First, the bits of the highest priority MB is placed followed by the bits of the second highest priority MB and so on, as shown in Figure 14 (a). To label the MB after the MB are

ordered by their priorities, 446 bytes of complementary information of the MB labels are needed if the video is in common intermediate format (CIF). The bits are then divided into sublayers as shown in Figure 14 (b). Sublayer $i+1$ has more bits than Sublayer i since we want to achieve UPP for the sublayers when appended with the parity bits. For example, we can let Sublayer 1 consists of bits from the first 10 highest priority MB, Sublayer 2 consists of bits from the following 20 highest priority MB, and so on. Each sublayer is then appended with parity bits from the FEC coding as shown in Figure 14 (c).

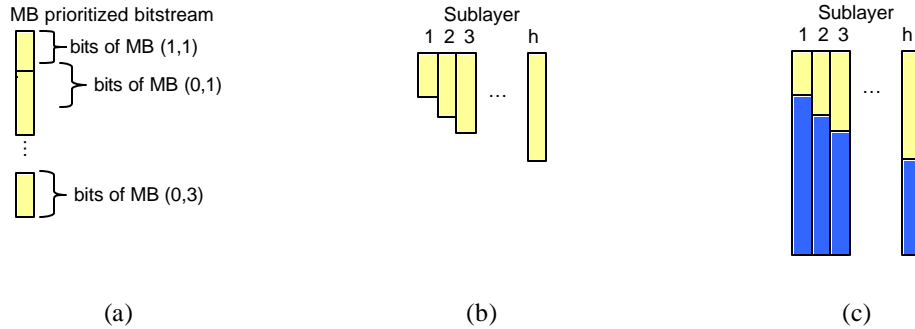


Figure 14. Precoded video: (a) MB prioritized bitstream, (b) MB prioritized bitstream in sublayers, and (c) FEC coded MB prioritized bitstream

Also, with the MB gain defined, we can define the sublayer gain correspondingly as:

$$G_i = \sum_{j \in \{\text{indices of MB that belong to Sublayer } i\}} g_j, \quad i = 1 \sim \text{number of sublayers in a frame} \quad (10)$$

Note again that ECARS can perform rate adaptation with or without EC aware precoding as long as the precoded video is provided with sublayer gains.

To summarize, the proposed ECARS with EC aware precoding utilizes the MB gains considering the EC method used at the receiver. The expected accumulated gain used in the later R-D

optimization is not only based on the MB gains but also on the current network condition. A two-stage R-D optimization approach is then proposed for finding the optimal solution.

IV. EXPERIMENT

In the experiment, we will show results of the proposed ECARS together with EC aware precoding, compared with the naïve rate shaping method “unequal error protection rate shaping (UPPRS)” described in Figure 15. 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 $i + 1$.

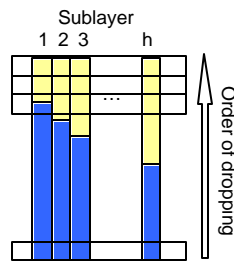


Figure 15. UPPRS illustration

Wireless networks are generally with time-varying packet loss rate and fluctuating bandwidth. The packet loss rate and bandwidth vary at each time interval. The time interval of our simulation is the frame interval (33 ms for a frame rate of 30 frames/sec). We simulate random bandwidth fluctuation and use a two-state Markov-chain [19][20] (Figure 16) to simulate the bursty bit errors. Example traces of simulated bandwidth and packet loss rate are shown in Figure 17. In reality, through standards such as the real-time control protocol (RTCP, part of the real-time transport protocol (RTP)) [21], rate shaper can obtain network condition information. The delay of such network condition information is in general less than a frame interval given the one-way transmission time described in [22].

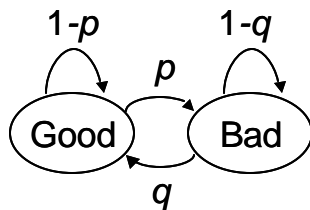


Figure 16. Two-state Markov chain for bit error simulation

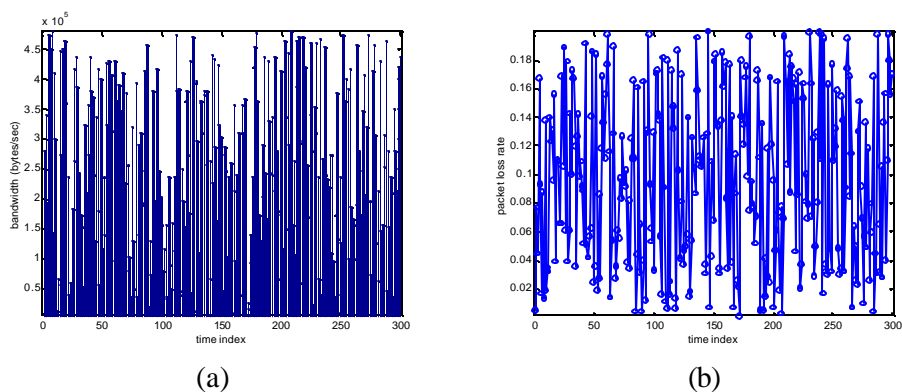


Figure 17. Traces of: (a) Bandwidth and (b) packet loss rate

The test video sequences are “akiyo”, “foreman”, and “stefan” in common intermediate format (CIF) (Figure 18 (a)-(c)). We use H.263 [10] for video encoding. Results in the following are shown for the luminance Y components only.



(a) (b) (c)

Figure 18. Test video sequences in CIF: (a) akiyo, (b) foreman, and (c) stefan

Figure 19 to Figure 21 show the EC aware precoding by MB prioritization. A MB is more important than the others are, if its square sum of the pixel differences between the original MB and the EC-reconstructed MB (that is the MB of the previous frame in this paper) is larger. The brighter the MB is, the larger the MB gain is, and hence the higher the MB priority is. In Figure 19, the only scene variation is from the anchor, mostly in the head and mouth regions. In Figure 20, most of the scene variations are from the head of the foreman. In Figure 21, the scene variations are from the movements of the tennis player and the camera moves. EC-reconstructed MB differs more from the original MB in those regions with more scene variations. Thus, the MB in those regions is shown with brighter intensity.

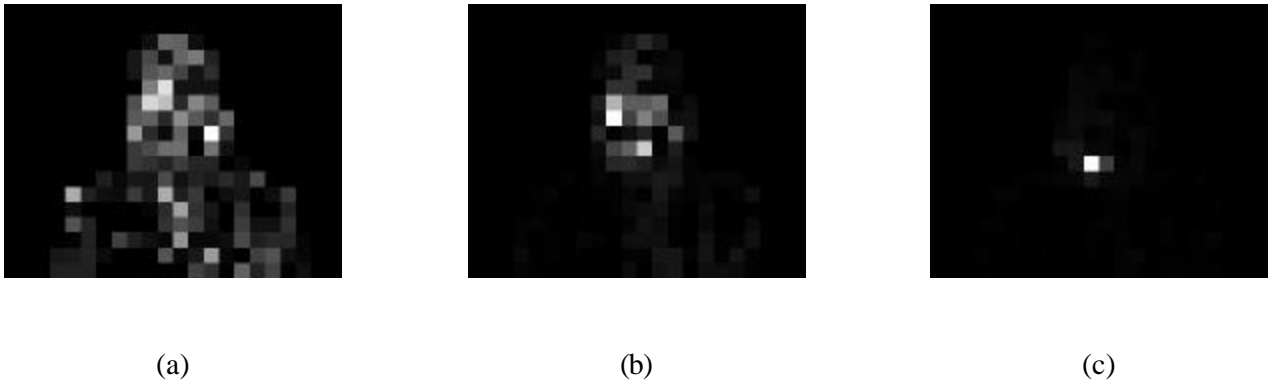
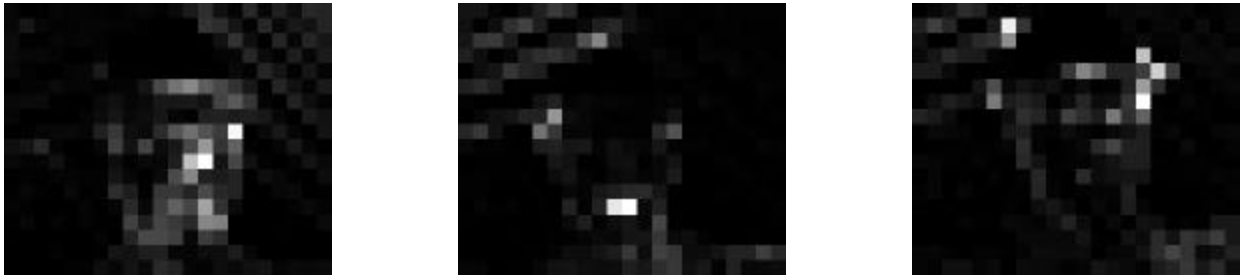


Figure 19. EC aware MB prioritization of Sequence “akiyo” in (a) Frame 2, (b) Frame 32, and (c) Frame 122



(a)

(b)

(c)

Figure 20. EC aware MB prioritization of Sequence “foreman” in (a) Frame 2, (b) Frame 32, and (c) Frame 122



(a)

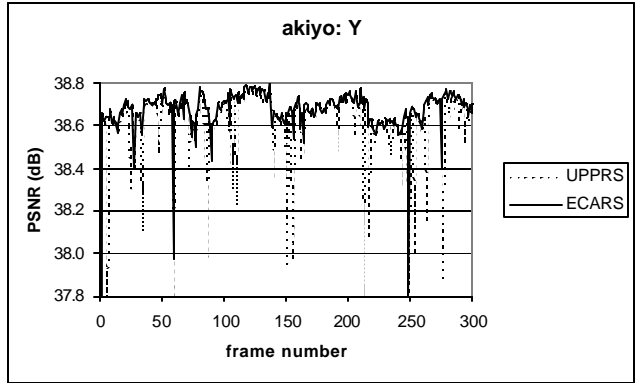
(b)

(c)

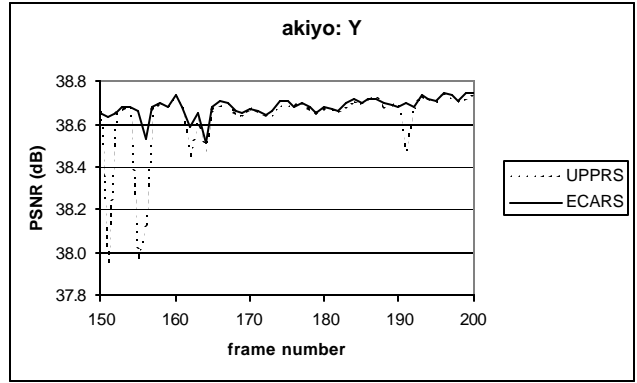
Figure 21. EC aware MB prioritization of Sequence “stefan” in (a) Frame 2, (b) Frame 32, and (c) Frame 122

Frame by frame PSNR results for Sequence “akiyo”, “foreman” and “stefan” are shown in Figure 22, Figure 23, and Figure 24, respectively. The overall PSNR performance for all three test sequences is shown in Figure 25. We can see that the proposed ECARS performs better than UPPRS. The improvement of ECARS over UPPRS is the most significant in Sequence “stefan” followed by Sequence “foreman” and “akiyo”. Sequence “stefan” is difficult to be reconstructed well by error concealment if the video data is lost during the transmission. It is more crucial to send the right

combination of symbols that is aware of the EC method at the receiver. Therefore, the performance improvement of ECARS over UPPRS is more prominent.

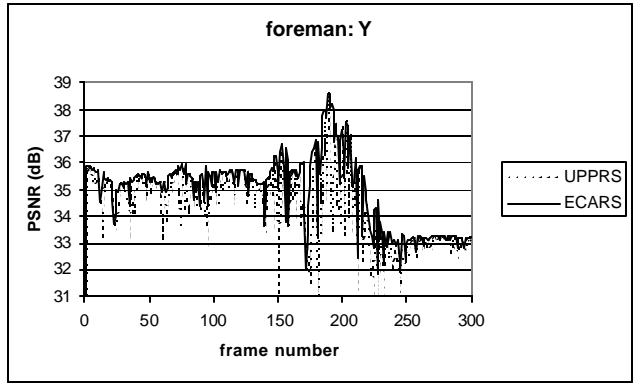


(a)

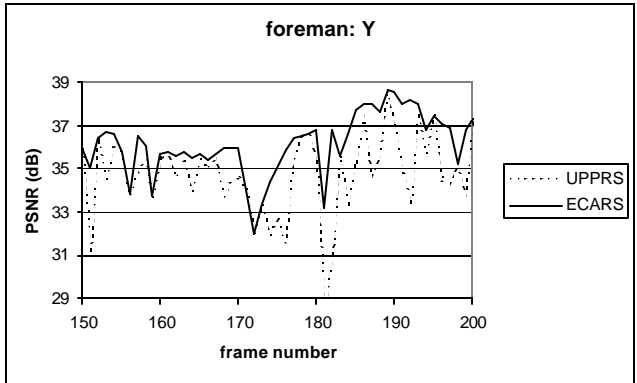


(b)

Figure 22. Frame by frame PSNR of UPPRS and ECARS with Sequence “akiyo”: (a) result from Frame 1 to Frame 300, (b) zoomed result from Frame 150 to Frame 200



(a)



(b)

Figure 23. Frame by frame PSNR of UPPRS and ECARS with Sequence “foreman”: (a) result from Frame 1 to Frame 300, (b) zoomed result from Frame 150 to Frame 200

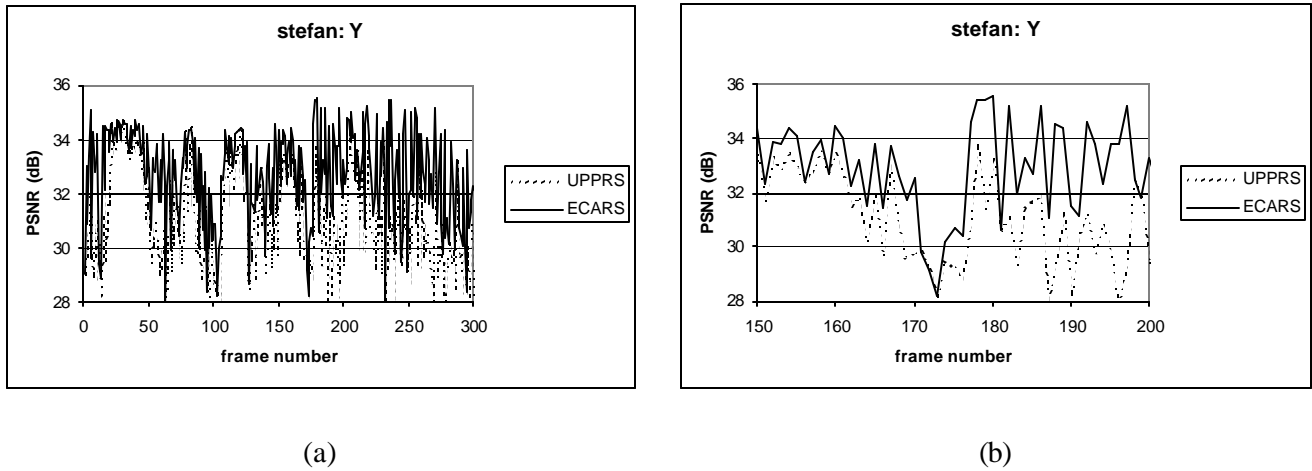


Figure 24. Frame by frame PSNR of UPPRS and ECARS with Sequence “stefan”: (a) result from Frame 1 to Frame 300, (b) zoomed result from Frame 150 to Frame 200

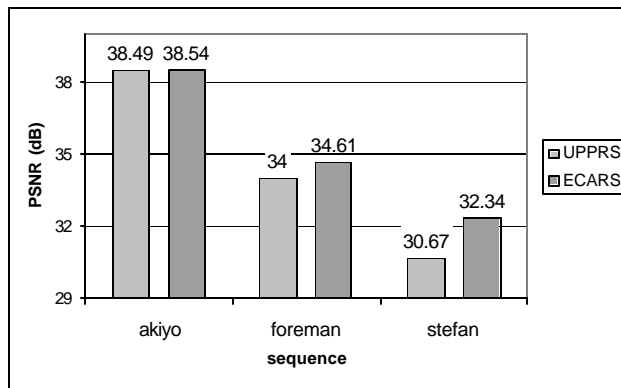


Figure 25. Overall PSNR of UPPRS and FGRS with sequences “akiyo”, “foreman”, and “stefan”

Some sample frames are shown in Figure 26, Figure 27, and Figure 28 for the three test sequences. These three examples show the cases where UPPRS does not perform as well as ECARS. In Figure 26, UPPRS does not protect the MB in the eye regions well enough as ECARS. The MB in the eye regions are thus corrupted. Error concealment reconstructs the corrupted MB with the pixel values of the previous frame. The current frame has the eyes closed while the previous frame has the eyes open. On the other hand, ECARS protects the MB in the eye regions well enough and thus does not

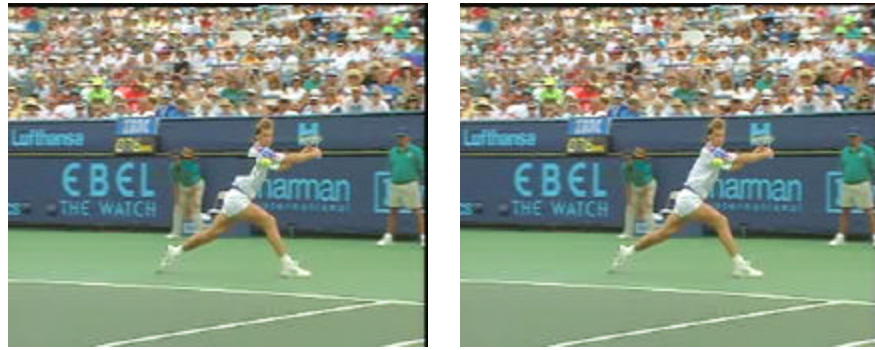
result in corrupted MB in the eye regions. Similarly, Figure 27 and Figure 28 show that the MB in the hat and body regions, respectively, are protected better by ECARS than UPPRS.



Figure 26. Example decoded frame, Frame 5, of Sequence “akiyo” with (a) UPPRS and (b) ECARS



Figure 27. Example decoded frame, Frame 150, of Sequence “foreman” with (a) UPPRS and (b) ECARS



(a)

(b)

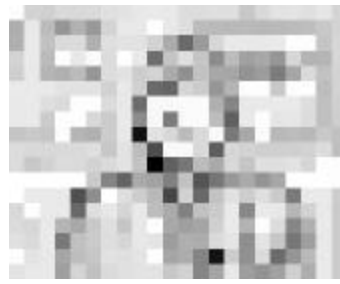
Figure 28. Example decoded frame, Frame 181, of Sequence “stefan” with (a) UPPRS and (b) ECARS

To examine how ECARS outperforms UPPRS, we look at the MB recovery rates of all the MB in three sample frames, Frame 2, Frame 32, and Frame 122. With the Reed-Solomon codes used in this paper, the MB recovery rates can be computed given the R-D optimization result $\mathbf{r} = [r_1 \ r_2 \ \dots \ r_h]$ of the frame examined. We can verify the validity of the proposed rate shaping algorithm if the MB that is harder to be reconstructed well by error concealment has higher recovery rate.

Figure 29 and Figure 30 show the MB recovery rates of Sequence “akiyo”, Figure 31 and Figure 32 show the MB recovery rates of Sequence Sequence “foreman”, and Figure 33 and Figure 34 show the MB recovery rates of Sequence “stefan”. Figure 29, Figure 31, Figure 33 are the results by UPPRS while Figure 30, Figure 32, and Figure 34 are the results by ECARS. The brighter the MB is, the higher the probability it can be received without errors. The recovery rate is determined by the video transport scheme, that is, either UPPRS or ECARS. We can see that Figure 30 resembles Figure 19 more than Figure 29 does. Similarly, Figure 32 resembles Figure 20 more than Figure 31 does; and Figure 34 resembles Figure 21 more than Figure 33 does. With ECARS, the MB that is with higher priority indeed gets higher recovery rate.



(a)

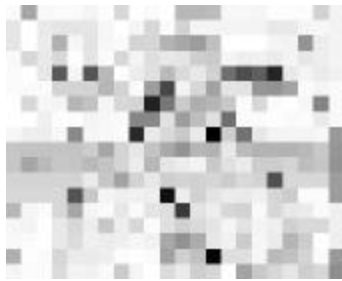


(b)



(c)

Figure 29. MB loss recovery rates of Sequence “akiyo” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using UPPRS



(d)



(e)



(f)

Figure 30. MB loss recovery rates of Sequence “akiyo” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using ECARS



Figure 31. MB loss recovery rates of Sequence “foreman” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using UPPRS

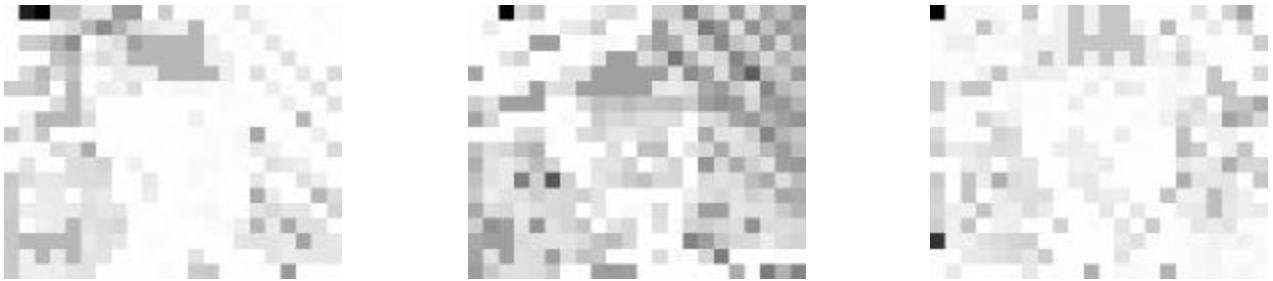


Figure 32. MB loss recovery rates of Sequence “foreman” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using ECARS

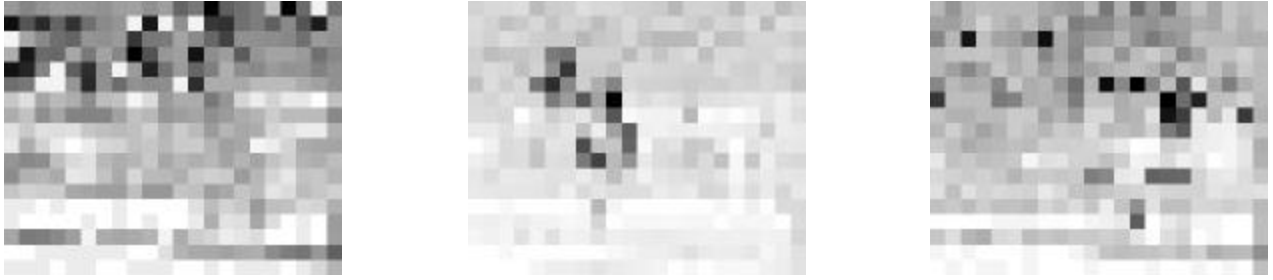


Figure 33. MB loss recovery rates of Sequence “stefan” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using UPPRS



Figure 34. MB loss recovery rates of Sequence “stefan” in (a) Frame 2, (b) Frame 32, and (c) Frame 122 using ECARS

V. CONCLUSION

We proposed in this paper error concealment aware rate shaping (ECARS) for video transport over wireless networks. ECARS is applied to pre source- and channel- coded video. ECARS first evaluates the gain of sending the MB of the precoded video, as opposed to not sending it but reconstructing it by EC. Then given a certain packet loss rate, the expected accumulated gain can be derived and be included

in the R-D optimization problem formulation. Finally, ECARS performs R-D optimization by the proposed two-stage R-D optimization approach. The proposed two-stage R-D optimization approach first obtains the near-optimal solution by finding the intersection of the model-based hyper-surface and the bandwidth constraint, and refines the solution from Stage 1 by a hill-climbing based approach. Furthermore, the precoding process can be EC aware to prioritize the precoded video based on the MB gains. The proposed ECARS outperforms the naïve UPPRS approach in the experiment.

The expected accumulated gain discussed in this paper is defined within each frame. All the frames are intra-coded and the decision made by the rate shaper will not affect the frames that follow. Future work includes extending ECARS for video with frame dependency, e.g. inter-coded video. Some discussions can be found in [23]. Feedback information, such as which MB is corrupted and the mean of the corrupted MB, is used by ECARS with frame dependency consideration.

The way the MB are grouped into sublayers in this paper is fixed and is not part of the ECARS R-D optimization, since how MB are grouped should be considered in the precoding process but not in the rate shaping stage. In the future, we can consider R-D optimization on the way MB are grouped into sublayers (that is, the number of source-coded symbols that go to each sublayer) given the rate shaping problem is solved.

VI. REFERENCE

- [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] A. Eleftheriadis and D. Anastassiou, "Meeting Arbitrary QoS Constraints using Dynamic Rate Shaping of Coded Digital Video", *NOSSDAV 1995*, pp. 96-106, Durham, New Hampshire, April 1995.
- [4] A. Eleftheriadis and D. Anastassiou, "Constrained and General Dynamic Rate Shaping of Compressed Digital Video", *ICIP 1995*, vol. 3, pp. 396-399, Washington D.C., October 1995.
- [5] S. Jacobs and A. Eleftheriadis, "Streaming Video Using Dynamic Rate Shaping and TCP Congestion Control", *Journal of Visual Communication and Image Representation*, 9(3), 1998, pp. 211-222.

- [6] 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.
- [7] W. Zeng and B. Liu, "Geometric-Structure-Based Error Concealment with Novel Applications in Block-Based Low-Bit-Rate Coding", *IEEE Transactions on Circuits and Systems for Video Technology*, 9(4), June 1999, pp. 648-665.
- [8] S. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, 1995.
- [9] T. P.-C. Chen and T. Chen, "Adaptive Joint Source-Channel Coding using Rate Shaping", *ICASSP 2002*, Orlando, FL, U.S.A., May 2002.
- [10] 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.
- [11] Motion Pictures Experts Group, "Overview of the MPEG-4 Standard", ISO/IEC JTC1/SC29/WG11 N2459, 1998.
- [12] Y. Shoham and A. Gersho, "Efficient Bit Allocation for an Arbitrary Set of Quantizers", *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 36(9), September 1988, pp. 1445-1453.
- [13] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit Allocation for Dependent Quantization with Applications to Multiresolution and MPEG Video Coders", *IEEE Transactions on Image Processing*, 3(5), September 1994, pp. 533-545.
- [14] A. Ortega and K. Ramchandran, "Rate-Distortion Methods for Image and Video Compression". *IEEE Signal Processing Magazine*, 15(6), November 1998, pp. 23-50.
- [15] P. A. Chou and Z. Miao, "Rate-Distortion Optimized Streaming of Packetized Media", submitted to *IEEE Transactions on Multimedia*, February 2001.
- [16] W. M. Lam, A. Reibman, and B. Liu, "Recovery of Lost or Erroneously Received Motion Vectors", *ICASSP 1993*, vol. 5, pp. 417-420.
- [17] M. E. Al-Mualla, N. Canagarajah, D. R. Bull, "Multiple-reference temporal error concealment", *ISCAS 2001*, vol. 5, pp. 149-152.
- [18] Trista Pei-chun Chen and Tsuhan Chen, "Second-Generation Error Concealment for Video Transport over Error Prone Channels", *Wireless Communications and Mobile Computing*, Special Issue on Multimedia over Mobile IP, October 2002.
- [19] F. Alajaji and T. Fuja, "A Communication Channel Modeled on Contagion", *IEEE Transactions on Information Theory*, 40(6), pp. 2035-2041, 1994.
- [20] M. Yajnik, S. Moon, J. Kurose, D. Towsley, "Measurement and Modeling of the Temporal Dependence in Packet Loss," *INFOCOM 1999*, pp. 345-52, March 1999.
- [21] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson: "RTP: A Transport Protocol for Real-Time Applications", RFC1889, Jan. 1996. <ftp://ftp.isi.edu/in-notes/rfc1990.txt>.
- [22] ITU-T Recommendation G. 114, "One-Way Transmission Time", May 2000.
- [23] Trista Pei-chun Chen and Tsuhan Chen, "Rate Shaping for Video with Frame Dependency", *ICME 2003*, Baltimore, MD, July 2003.