Real-Time Forward Error Correction for Video Transmission

Jimin Xiao*†, Tammam Tillo†, Chunyu Lin‡, Yao Zhao‡
* Dept. of Electrical Engineering and Electronics, University of Liverpool, Liverpool, UK
Email: jimin.xiao@liverpool.ac.uk
† Dept. of Electrical & Electronic Engineering Xi'an Jiaotong-Liverpool University, Suzhou, China
Email: tammam.tillo@xjtlu.edu.cn
‡ Institute of Information Science, Beijing Jiaotong University,
Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044
Email: {chunyulin,yzhao}@bjtu.edu.cn

***Abstract*—When the video streams are transmitted over the unreliable networks, forward error correction (FEC) codes are usually used to protect them. Reed-Solomon codes are block-based FEC codes. On one hand, enlarging the block size can enhance the performance of the Reed-Solomon codes. On the other hand, large Reed-Solomon block size leads to long delay which is not tolerable for real-time video applications. In this paper a novel approach is proposed to improve the performance of Reed-Solomon codes. With the proposed approach, more than one video frame are encompassed in the Reed-Solomon coding block yet no delay is introduced. Experimental results show that the proposed approach outperforms other real-time error resilient video coding technologies.**

***Index Terms*—H.264/AVC, real-time transmission, error resilience, Reed-Solomon code**

I. Introduction

Due to the unreliable underlying networks, the development of error resilient video coding techniques are a crucial requirement for video communication over lossy networks [1]. The current existed error resilient techniques include intra macroblock (MB) refreshment [2], Automatic Repeat reQuest (ARQ) and feedback-based Reference Picture Selection (RPS)[3], redundant picture coding with equal or lower quality [4][5], and Forward Error Correction (FEC) coding [6][7][8]. Among those error resilient approaches, intra macroblock refreshment and redundant picture coding cause no additional delay, making them suitable for delay constrained applications. However, for the Intra MB refreshment approach, since the coding efficiency of intra mode is typically several times lower than inter mode, the coding efficiency is compromised dramaticly. For the redundant picture coding, when the redundant version is used to replace the primary one, there would be mismatch error and mismatch error propagates all over the GOP. ARQ and RPS usually cause long delay because of the network round-trip time, and consequently they can not be employed for real-time applications. For the FEC approaches, the delay depends on the channel coding block size. In [6], the Reed-Solomon (RS) coding block includes the whole GOP, and one GOP of delay is caused. In [7], the RS coding block contains frames from one Sub-GOP, so the delay depends on the length of the Sub-GOP, and in [8], the RS code is encoded in frame level, no FEC coding delay is created. However, for the frame level FEC approach, usually the source packet number is not big enough to make the FEC code efficient.

The Reed-Solomon (RS) code has been widely used as FEC code to protect data packets against loss in packet erasure networks. In RS (N, K) code, for every K source packets, $(N - K)$ parity packets are introduced to make up a codeword of packets. As long as a client receives at least K out of the N packets, it can recover all the source packets. In general, for the same code rate K/N, increasing the value of K would enhance the performance of RS code. However, large K values lead to large RS coding blocks which causes long coding delays. For real-time applications like video conferences, this kind of delay is unacceptable.

In this paper, we propose to use systematic Reed-Solomon erasure code to protect the video packets in real-time fashion, while allowing to provide an error free version of the reference frame to stop the propagation error. In order to enlarge the RS coding block size, the Sub-GOP, which contains more than one frame, is used as RS coding block. On the encoder side, for the systematic RS code, the data is left unchanged and the parity packets are appended. Therefore, there is no encoding delay. Meanwhile, at the receiver end, to decode and display one frame in the Sub-GOP, the video decoder only needs packets belonging to this frame. If some packets of this frame get lost during transmission, error concealment is applied to conceal the lost packets. In this manner, the decoder does not need to wait for all the packets belonging to this Sub-GOP. Therefore, there is no delay on the decoder side. Later, when the transmission of all packets of this Sub-GOP is finished, the systematic RS decoder would try to recover the lost packets. If the lost packets of this Sub-GOP is less than $(N-K)$, the RS decoder is able to recover all the lost packets of this Sub-GOP, the video decoder will re-decode this Sub-GOP with all the received and recovered packets, updating the reference frame, so the concealment distortion would not propagate to later frames. Moreover, we propose a practical redundancy allocation mechanism.

The rest of the paper is organized as follows. In Section II, firstly the frame level Evenly FEC approach is introduced, this approach is used as a benchmark for the real-time FEC coding, later the proposed Sub-GOP FEC approach is presented in detail. In Sect. III some simulation results validating the proposed approach are given. Finally, some conclusions are drawn in Section IV.

II. Real-Time FEC Video Transmission Approaches

Since our objective is to design FEC video transmission system for real-time applications, no delay is allowed to be introduced in both video encoding and RS coding process. Therefore B-frame is not used here, and we use the IPPP... GOP structure. To make the RS coding efficient, fixed length each slice is used to create slices. The slice length is decided by the Maximum Transmission Unit (MTU) of underlying networks. In this paper, the term packet and slice are used interchangeably, as one packet per slice packetization method is adopted.

A. Frame-Level Evenly FEC

For real-time FEC video packet protection, one common approach is to perform RS coding in frame level, which means that the RS coding block contains data packets from the same video frame. Under this constraint, RS coding does not introduce any additional delay, as with the Intra MB refreshment and redundant picture coding. Let us assume the GOP length is L frames, for the ith frame in one GOP there are $K(i)$ source packets and $R(i)$ RS protection packets, $\mu = (N-K)/K$ is the redundant packet rate of RS coding.

$$R(i) = \begin{cases} \lceil \mu \cdot N(1) \rceil & \text{if } i == 1 \\ \left\lceil \mu \cdot \sum_{k=1}^{i} N(k) \right\rceil - \sum_{k=1}^{i-1} R(k) & \text{if } i > 1 \end{cases} \quad (1)$$

where operation $\lceil X \rceil$ is used to get the minimum integer number greater than or equal to X. For this approach, the RS protection packets are evenly allocated among all the frames, we name this approach as Evenly FEC.

B. Sub-GOP FEC

According to the law of large number, when the packet number is N, packet loss rate (PLR) is p, if $N \to \infty$, $N \cdot p$ packets would be lost, then the number of parity packet $N - K$ could be as small as $N \cdot p$ to recover all the lost packets. In practical situations, $K \to \infty$ is impossible, in this case with the same redundant packet rate $\mu = (N-K)/K$, the larger the value of K is, the higher the RS code performance can be. Motivated by this fact, we propose to encompass packets from a Sub-GOP of frames to one RS coding block to enlarge the value of K. Figure.1 shows one example of how to generate Sub-GOP of frames and allocate RS protection packets at the end of each Sub-GOP. On the other hand, in order to meet the

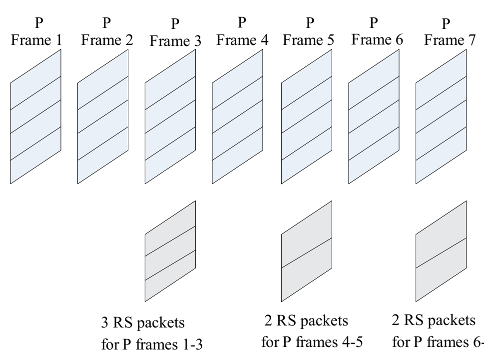


Figure 1. One example of RS protection packets allocation diagram for Sub-GOP FEC approach

real-time constraint, we use the systematic RS code, so the source packets are intact in the RS coding process. Therefore, at the receiver side, the video decoder only needs packets belonging to one frame to decode and display the frame. If some packets of this frame are lost during transmission, error concealment is used to conceal the lost packets.

For easy illustration, let us take one example, suppose that one Sub-GOP contains 2 frames and each frame generates 5 packets. The RS coding redundant packet rate μ is 20%, which means for the 10 source packets of this Sub-GOP, $(12, 10)$ RS protection code is applied. In one Sub-GOP, when the first frame is encoded by the video encoder, immediately the encoded 5 packets are transmitted over the network. Due to network failure, let us assume 2 packets among these 5 packets are dropped. Then upon the 3 packets reaching the receiver side, the video decoder will decode and display this frame, and for the lost packets, error concealment is used. In this manner, no additional delay is introduced. Later, the second frame is encoded, generating another 5 source packets, meanwhile, as the second frame is the last frame in this Sub-GOP, 2 RS protection packets are generated base on the 10 source packets of this Sub-GOP. Suppose this time the 5 source packets and 2 RS protection packets successfully arrive at the receiver side without any loss. At the receiver side, the $(12, 10)$ RS decoder is able to recover 2 packet losses, then the video decoder will re-decode the first frame of this Sub-GOP with 3 received source packets and 2 recovered packets, and update the reference frame buffer. In this case there is no distortion in the second frame, and no error would propagate to the incoming frames. However, if the Evenly FEC approach is adopted, for each frame $(6, 5)$ RS code is used. The RS code would not be able to recover the 2 packets losses of the first frame, eventually the concealment distortion propagates to all the following frames, and severely degrades the video quality at the receiver side. In this example, both Evenly FEC and Sub-GOP use same amount of RS protection packets, and in both cases, no additional delay is introduced. However, the advantage of sub-GOP is obvious, because this approach is able to restrict concealment distortion in very few frames.

C. Optimal Sub-GOP Size and RS Packet Allocation

As described above, using the RS protection in Sub-GOP level could be better than in frame level, but the problem of how to divide frames into Sub-GOP and how to allocate the RS protection packets among all the Sub-GOPs is yet to be solved. On one hand, if the Sub-GOP includes too few frames, the value of K for the RS code is not large enough to make the RS code efficient. On the other hand, if the Sub-GOP includes too many frames, as the RS correction codes are not available until the last frame of this Sub-GOP, the quality of those frames before the last frame would degrade significantly. Consequently, the Sub-GOP length should be properly tuned to make the RS code efficient.

In general, I-frame generates much more bits than P-frame, and therefore more source packets are produced for I-frame. In our Sub-GOP FEC approach, for the I-frame we provide RS protection in frame level, the same as Evenly FEC approach, whereas for the P-frame we allocate RS protection packets in Sub-GOP level. Our objective is to optimally allocate Sub-GOP and RS protection packets and minimize the expected total distortion of this GOP. To do this we need to know the detailed information of this GOP, including the slice number in each frame, the concealment distortion caused by losing each slice, and how the distortion propagates. However, those information are not available for real-time on-the-fly transmission system. In light of such circumstance, we established a model to represent these information. The model parameters include the number of P-frames in one GOP, L, and the average number of slices in each P-frame, S. Since the value of S is usually stable, we can infer the value S from previous encoding experience. The expected concealment distortion of losing one packet is $\bar{d}$, the distortion in current frame propagates to the following frames, and the attenuation function of the distortion is $f(n)$. This means if the concealment distortion in current error is $\bar{d}$, it will propagate to the following frames and become $f(n) \cdot \bar{d}$ after n frames. For the sake of simplicity the function $f(n) = \alpha^{n-1}$ $(0 < \alpha < 1)$ is employed. The distortion caused by losing individual slice is independent, and we can get the total expected distortion for the whole GOP by summing up all the expected distortion caused by individual slices. For the P-frames in one GOP the total allocated RS protection packet number is $R = \mu \cdot S \cdot L$, here μ is the redundant packet rate of RS coding. We use $R(i)$ to denote the number of RS protection packet for P-frame i.

$$\sum_{i=1}^{L} R(i) \leq R \quad (2)$$

Figure.1 shows one example of how RS protection packets are allocated. We assume there are totally t positions where we insert RS protection packets, with frame number r_1, r_2,..., r_t, whereas other positions have no RS protection packets. The number of RS protection packets are $R(r_1)$,$R(r_2)$,..., $R(r_t)$. In the example of Figure.1, we allocate RS protection packets in 3 positions, $t = 3$, the 3 positions are $r_1 = 3$, $r_2 = 5$, $r_3 = 7$, and RS packet number is $R(r_1) = 3$, $R(r_2) = 2$, $R(r_3) = 2$. The RS protection packets allocated under frame r_{m+1} are used to protect the frames from $r_m + 1$ to r_{m+1}. Therefore, the parameter of RS (N, K) code for this Sub-GOP is $N = (r_{m+1} - r_m) \cdot S + R(r_{m+1})$ and $K = (r_{m+1} - r_m) \cdot S$. We use $\bar{D}(r_m + 1, r_{m+1})$ to denote the expected distortion caused by losing packets from frame $r_m + 1$ to r_{m+1}. It is important to note that $\bar{D}(r_m + 1, r_{m+1})$ does not only include the distortion in frames from $r_m + 1$ to r_{m+1}, but also the propagation distortion in later frames.

$$\bar{D}(r_m + 1, r_{m+1}) = \bar{D}_i(r_m + 1, r_{m+1} - 1) + \bar{D}_p(r_{m+1}) \quad (3)$$

As described in Equation (3), the distortion $\bar{D}(r_m + 1, r_{m+1})$ is caused by two parts: $\bar{D}_i(r_m + 1, r_{m+1} - 1)$ is the expected distortion within frames $r_m + 1$ to $r_{m+1} - 1$. For those frames, the recover capability of RS code cannot be used by the time when those frames are decoded and displayed. The subscript i means that $\bar{D}_i(r_m + 1, r_{m+1} - 1)$ only accounts the $internal$ distortion within frames $r_m + 1$ to $r_{m+1} - 1$, error propagation to frame r_{m+1} and later frames is not accounted in this term. $\bar{D}_p(r_{m+1})$ is the sum of expected distortion in frame r_{m+1} and it also account for the propagated distortion in the coming frames. The subscript p denotes that this term includes the $propagation$ distortion. Note that by the time of decoding and displaying frame r_{m+1}, the RS code would try to recover the lost source packets within this Sub-GOP, but when the packet loss number in this Sub-GOP is beyond the recovery capability of RS code, the RS code would not be able to recover the lost packets. For the term $\bar{D}_i(r_m+1, r_{m+1}-1)$, as from frame r_m+1 to $r_{m+1}-1$ the expected packet loss number of each frame is $p \cdot S$, taking the error propagation into consideration, we get

$$\bar{D}_i(r_m + 1, r_{m+1} - 1) = \sum_{i=1}^{r_{m+1}-r_m-1} \phi(i) \cdot p \cdot S \cdot \bar{d} \quad (4)$$

where $\phi(n) = \sum_{i=0}^{n-1} f(i)$. Similarly, for the term $\bar{D}_p(r_{m+1})$, from frame $r_m + 1$ to frame r_{m+1} there are $(r_{m+1} - r_m) \cdot S$ source packets, while the RS protection packet number is $R(r_{m+1})$. Let us use $p_s(k)$ to denote the probability of losing k source packets, and $p_r(k)$ to denote the probability of losing k RS protection packets, both before the RS code is decoded. We use $p_{rs}(k)$ to denote the probability of losing k source packets after RS code correction.

$$p_s(k) = \binom{(r_{m+1} - r_m) \cdot S}{k} (1-p)^{(r_{m+1}-r_m) \cdot S - k} p^k \quad (5)$$

$$p_r(k) = \binom{R(r_{m+1})}{k} (1-p)^{R(r_{m+1})-k} p^k \quad (6)$$

Let us use $P_r(k)$ to denote no less than k RS protection packets are lost

$$P_r(k) = \sum_{i=k}^{R(r_{m+1})} p_r(i) \quad (7)$$

Since losing k source packets after RS code correction is caused by losing k source packets, and at the same time losing more than $R(r_{m+1}) - k$ RS protection packets. So the probability of losing k source packets after RS correction is

$$p_{rs}(k) = \begin{cases} p_s(k) \cdot P_r(R(r_{m+1}) - k + 1) & k \le R(r_{m+1}) \\ p_s(k) & k > R(r_{m+1}) \end{cases} \quad (8)$$

The expected packet loss number among frame r_m+1 to frame r_{m+1} after RS code correction is

$$\bar{\Gamma} = \sum_{k=1}^{(r_{m+1}-r_m) \cdot S} k \cdot p_{rs}(k) \quad (9)$$

Therefore, the expected packet loss number in each frame after RS code correction is $\bar{\Gamma}/(r_{m+1} - r_m)$, and we can get the expected distortion in frame r_{m+1}, taking into account error propagation from previous frames inside this Sub-GOP.

$$\bar{D}_l(r_{m+1}) = \frac{\bar{\Gamma}}{r_{m+1} - r_m} \cdot \phi(r_{m+1} - r_m) \cdot \bar{d} \quad (10)$$

The distortion in frame r_{m+1} will propagate to the end of this GOP, therefore

$$\bar{D}_p(r_{m+1}) = \bar{D}_l(r_{m+1}) \cdot \phi(L - r_{m+1} + 1) \quad (11)$$

The total expected distortion for the whole GOP is $\bar{D}_{total}$, for RS protection allocation like in Figure.1, $\bar{D}_{total}$ can be expressed as follow

$$\bar{D}_{total} = \bar{D}(1, i_1) + \sum_{m=1}^{t-1} \bar{D}(r_m + 1, r_{m+1}) \quad (12)$$

At this point, the optimization problem can be formulated as the following constrained minimization:

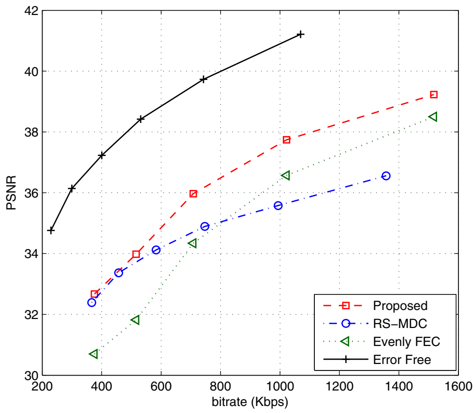
$$\begin{cases} \text{min} & \bar{D}_{total} \\ \text{subject to} & \sum_{i=1}^{N} R(i) \le R \end{cases} \quad (13)$$

III. Simulation Results

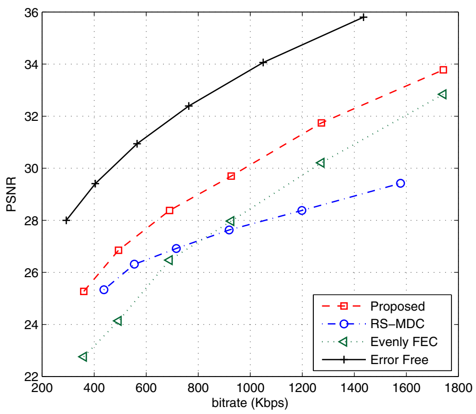
Our simulation setting is built on the JM14.0 H.264 codec. CIF video sequence Foreman and Bus are used for the simulation. The GOP structure is IPPP..., and GOP length is 30 frames. The reference frame number is 1, in other words, only the previous frame is used for prediction. One slice is transmitted in one packet, taking the MTU of wireless network into account, we set the target slice length as 400 byte. For the distortion attenuation function $f(n) = \alpha^{n-1}$, we use $\alpha = 1$. In order to have a fair comparison, we compare our Sub-GOP approach with RS-MDC [4] and Evenly FEC approach, because all those approaches meet the real-time constraint and cause no additional delay. Figure.2 and Figure.3 show the PNSR for different bit-rate for random packet loss rate 5% and 10%, respetively. Obviously, the proposed Sub-FEC approach always outperforms RS-MDC and Evenly FEC in all the simulation environments. For both the Foreman and Bus sequences, the gain over Evenly FEC can be more than 2dB in low bit-rate when PLR is 5%, and the gain over RS-MDC could be over 5dB in high bit-rate when PLR is 10%. It is very interesting to note that the gap between Sub-GOP FEC and Evenly FEC is larger in low bit-rate than in high bit-rate. This is because in high bit-rate more packets are generated for each frame, and consequently, for the frame level Evenly FEC approach, the value of K is relatively large to make the RS coding efficient.

IV. Conclusions

In this paper, a real-time FEC video transmission approach is proposed. We first present the general idea of this approach, then the theoretical model for allocating Sub-GOP and RS protection packets are given. With this model, the allocation problem becomes

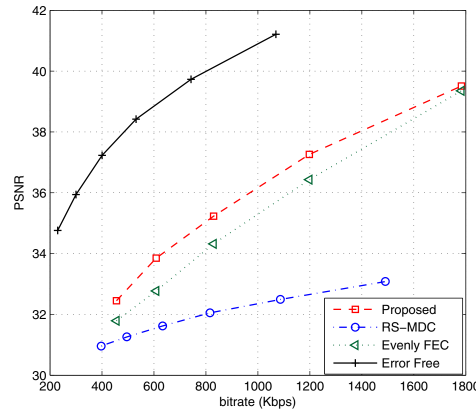


a. Foreman

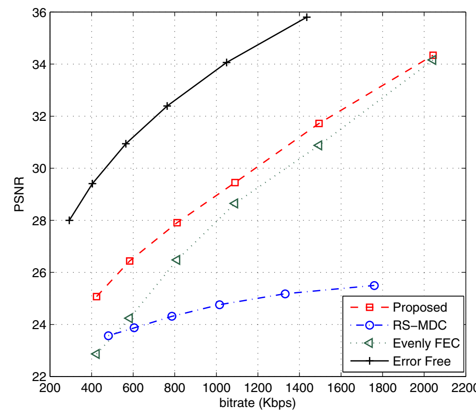


b. Bus

Figure 2. PSNR versus bitrate for the three approaches. The network PLR is 5% and the redundant packet rate μ is 20% . (a) Foreman sequence (b) Bus sequence



a. Foreman



b. Bus

Figure 3. PSNR versus bitrate for the three approaches. The network PLR is 10% and the redundant packet rate μ is 40% . (a) Foreman sequence (b) Bus sequence

a constrained optimization problem. In order to validate the proposed approach, the performances of the proposed approach are compared with other real-time error resiliency approaches. Experimental results demonstrate the proposed approach has considerable practical value for for real-time applications.

The computational complexity of the Sub-GOP size and RS packet allocation is tremendous, currently we are working on some low complexity algorithms. And in this paper, for simplicity, the video packets are considered either as received or lost. However, in practical situations, all the received packets have different delays, and those packets arrive at the destination beyond the maximum delay are usually considered as lost packets. Currently, we are working on a new FEC system, which takes into account the network delays.

Acknowledgment

This work was supported by National Natural Science Foundation of China (No.60972085, No.60903066), Sino-Singapore JRP (No.2010DFA11010) and National Science Foundation of China for Distinguished Young Scholars (No.61025013). The authors would like to thank professor Michele ELIA in Politecnico di Torino for guidance in Reed-Solomon code, and Sihang Yu in Xian Jiaotong-Liverpool University for English language correction.

References

[1] Y. Wang, S. Wenger, J. Wen, and A. Katsaggelos, “Error resilient video coding techniques,” *Signal Processing Magazine, IEEE*, vol. 17, pp. 61 –82, July 2000.

[2] R. Zhang, S. Regunathan, and K. Rose, “Video coding with optimal inter/intra-mode switching for packet loss resilience,” *Selected Areas in Communications, IEEE Journal on*, vol. 18, pp. 966 –976, June 2000.

[3] S. Lin, S. Mao, Y. Wang, and S. Panwar, “A reference picture selection scheme for video transmission over ad-hoc networks using multiple paths,” in *Multimedia and Expo, 2001. ICME 2001. IEEE International Conference on*, pp. 96 – 99, 2001.

[4] T. Tillo, M. Grangetto, and G. Olmo, “Redundant slice optimal allocation for h.264 multiple description coding,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 1, pp. 59 –70, 2008.

[5] C. Zhu, Y.-K. Wang, M. Hannuksela, and H. Li, “Error resilient video coding using redundant pictures,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 1, pp. 3 –14, 2009.

[6] E. Baccaglini, T. Tillo, and G. Olmo, “Slice sorting for unequal loss protection of video streams,” *Signal Processing Letters, IEEE*, 2008.

[7] X. Yang, C. Zhu, Z. G. Li, X. Lin, and N. Ling, “An unequal packet loss resilience scheme for video over the internet,” *Multimedia, IEEE Transactions on*, vol. 7, no. 4, pp. 753 – 765, 2005.

[8] N. Thomos, S. Argyropoulos, N. Boulgouris, and M. Strintzis, “Robust transmission of h.264/avc video using adaptive slice grouping and unequal error protection,” in *Multimedia and Expo, 2006 IEEE International Conference on*, pp. 593 –596, 2006.