INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2017; 30: e2826
Published online 24 June 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.2826

Performance modeling of FEC-based unequal error protection for H.264/AVC video streaming over burst-loss channels

Chun-I Kuo[1], Ce-Kuen Shieh[1], Wen-Shyang Hwang[2] and Chih-Heng Ke[3,*,†]

[1]*Institute of Computer and Communication, Department of Electrical Engineering, National Cheng Kung University, Taiwan*
[2]*Department of Electrical Engineering, National Kaohsiung University of Applied Sciences, Taiwan*
[3]*Department of Computer Science and Information Engineering, National Quemoy University, Taiwan*

SUMMARY

Unequal error protection systems are a popular technique for video streaming. Forward error correction (FEC) is one of error control techniques to improve the quality of video streaming over lossy channels. Moreover, frame-level FEC techniques have been proposed for video streaming because of different priority video frames within the transmission rate constraint on a Bernoulli channel. However, various communication and storage systems are likely corrupted by bursts of noise in the current wireless behavior. If the burst losses go beyond the protection capacity of FEC, the efficacy of FEC can be degraded. Therefore, our proposed model allows an assessment of the perceived quality of H.264/AVC video streaming over bursty channels, and is validated by simulation experiments on the NS-2 network simulator at a given estimate of the packet loss ratio and average burst length. The results suggest a useful reference in designing the FEC scheme for video applications, and as the video coding and channel parameters are given, the proposed model can provide a more accurate evaluation tool for video streaming over bursty channels and help to evaluate the impact of FEC performance on different burst-loss parameters. Copyright © 2014 John Wiley & Sons, Ltd.

Received 17 February 2014; Revised 21 April 2014; Accepted 27 May 2014

KEY WORDS: forward error correction; burst-loss channels; H.264/AVC; video streaming

1. INTRODUCTION

Forward error correction (FEC) is a popular error control technique used to control the quality of delay-sensitive multimedia streaming to overcome packet losses and transmission errors in the current networks [1–4]. In the existing FEC technique, the sender side will generate some redundant packets via the FEC encoder, and they will be transmitted along with the source packets. If the receiver side can collect enough packets, the FEC decoder can reconstruct the source packets without error. Compared with FEC, the automatic repeat request (ARQ) has addition processing delay since the reliability of transmission is ensured by means of retransmitting lost data through an acknowledgment mechanism [5]. For delay constraints of video transmission applications [3, 4, 6], the FEC is a better choice than ARQ as it has lower end-to-end delay. Moreover, an alternative approach to achieving better video quality degradation in processing losses is to use unequal error protection (UEP), which differentiates the FEC protection level for prioritized video data [7]. The first UEP codes were proposed by Masnick and Wolf [8]. The idea behind UEP is that FEC allocates the stronger protection to more important data and the weaker one to less important data. The main goal of UEP technique is to achieve the maximum expected quality of video streaming. Based on the idea of UEP, the first frame in a group of pictures (GOP) has higher impact at the

*Correspondence to: Chih-Heng Ke, Department of Computer Science and Information Engineering, National Quemoy University, Taiwan.
†E-mail: smallko@gmail.com

video decoder side. Several researchers proposed the UEP solutions to ensure quality of service (QoS) for video streaming on the lossy channels [9–11]. Reviewing the literature, UEP can be applied to different FEC coding techniques. UEP with Reed-Solomon (RS) codes has been widely adopted in [12–14]. UEP with the low-density parity-check (LDPC) codes are also proposed for H.264/AVC scalable video coding [15, 16]. Since the use of a fixed code rate would lead to the bandwidth waste under the situation that the channel loss is overestimated, the rate-less codes such as Fountain codes and Raptor codes are proposed and their applications with UEP for video streaming can be found in [17, 18]. In this paper, we focus on RS codes due to its popularity in the traditional video transmission system.

In actual network loss behavior, the loss distributions of network congestion and/or wireless errors are bursty [19, 20]. If a packet is lost, the next packet exhibits finite temporal dependence and is also likely to be lost. Related works [21, 22] have shown that the FEC schemes become very ineffective over burst-loss transmission channels because the receiver side may not collect enough number of packets to reconstruct the original data. In video streaming, the packet loss probability, average burst length, and block loss distribution are very important parameters to estimate the efficiency of the existing FEC approaches [21–23].

With regard to evaluating the quality of video streaming, several analysis models of video streaming have been proposed to compute the expected quality on the part of distortions after FEC decoding [24, 25]. However, the characterizations of the random loss distribution in current existing researches are like a Bernoulli distribution. In contrast, in the burst channel, there are very little analytical works about estimating the expected video quality. In [26], Li *et al.* derived an analytical distortion model with FEC for burst channels without B frames and indicated that a sliding window algorithm proposed in [27] can reduce the computing complexity. However, this complexity-reduction algorithm is difficult to apply to video streaming with B frames, and the computing complexity of the distortion model creates a problem in the performance evaluation.

Considering the dependence of different types of H.264/AVC, several researchers proposed models of the decodable frame rate (DFR), which has lower complexity than the distortion model and is also a popular analysis model to evaluate the video quality [28]. Optimal FEC control algorithms have been proposed for streaming videos based on the expected DFR. In [29], Wu *et al.* create an analytical frame-level FEC model within a TCP-friendly rate constraint to obtain optimal reconstruction quality of a GOP. However, these existing models are still based on a Bernoulli channel and do not take the burst-loss patterns into consideration. In [30], the first DFR model for a bursty channel was proposed to evaluate how average burst length affects the DFR of video streaming at the receiver. This model just focuses on the impact of packet loss ratio and the burst-loss pattern distribution for different types of frames. It does not consider the transmission sequence in a GOP. Thus, the performance of analytical results does not match as increasing the GOP lengths or smaller video source packets.

For FEC-based video applications over burst-loss transmission channels, the FEC coding and UEP should be different from those given by the FEC model based on the assumption of Bernoulli channels. It is therefore necessary to consider the impact of the burst-loss pattern in building the UEP model of GOP. In this paper, our previous work in [30] is extended and the transmission sequence is further taken into consideration. We explore more accurate analysis model of UEP method for H.264/AVC using the standardized RS codes. Accordingly, this paper presents an analytical FEC model to compute an expected quality of a decoded video for the video streaming over burst-loss channels. Markov models (Gilbert models) are applied for many existing works [31–34], and a good approximation of the actual network transmission for a burst-loss channel. Assuming that the average packet loss rate and average burst loss length are available in the network, a Gilbert model is adopted as the burst packet loss process. In the experimental results, our proposed model can be validated by the simulation results and help observe the differences in DFR performance between the Bernoulli and the Gilbert channel. The proposed analytical model focuses on studying the impact of bursty-loss channels, and one of impact factors is average burst length. Finally, the simulations also show the computing complexity of different analytical models, and these results enable us to understand the overhead of computing complexity when considering a burst-loss channel.

DOI: 10.1002/dac

The remainder of this paper is organized as follows. Section 2 introduces the background to the analytical model in previous researches, while Section 3 formulates the DFR model to evaluate the video quality over burst-loss channels. Section 4 presents and discusses the experimental and simulation results. Finally, Section 5 provides some brief concluding remarks.

2. BACKGROUND

2.1. Channel model

The probability of a packet loss sequence for the Gilbert and Bernoulli channels is shown in Figure 1. Bernoulli channel is characterized by a packet loss probability P_B. Every transmitted packet is determined whether to be lost with the same probability. Thus, the loss probability depends on the amount of received and lost packets, and it will be easy to map them to the mathematical model.

However, in a Gilbert channel, packet losses exhibit dependencies over time. In this study, the burst packet losses over the network are modeled by the Gilbert model (i.e., two-state Markov model). As shown in Figure 1, the transition probability from Good State to Bad State is defined by p, and the probability of the opposite transition is denoted by q. At the Good state, a packet is dropped with probability 0 whereas a packet is dropped with probability 1 at the bad state. Accordingly, the one-step transition matrix is defined as

$$\varphi = \begin{bmatrix} 1-p & p \\ q & 1-q \end{bmatrix} \tag{1}$$

In the two-state Markov model, the average packet loss probability is expressed as

$$P_B = \frac{q}{p+q} \tag{2}$$

Then the average burst length, L_B, is the average number of consecutive packet losses:

$$L_B = \frac{1}{p} \tag{3}$$

As shown in Figure 1, it is clear that the loss probability between the Gilbert (two-state Markov) and Bernoulli channels are different in the same cases. Hence, when calculating the loss probability

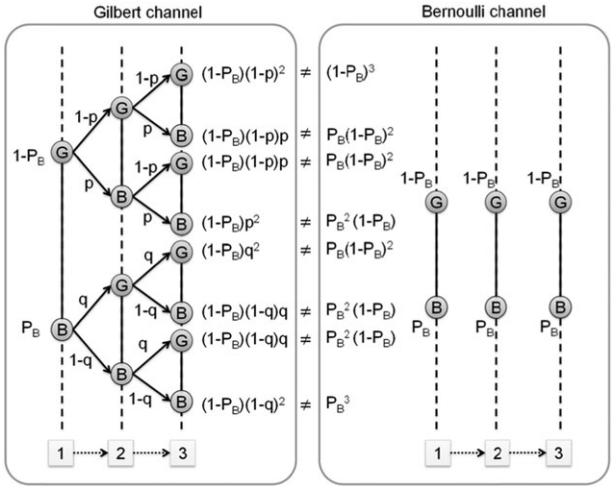


Figure 1. The probability of packet loss sequence.

DOI: 10.1002/dac

for a two-state Markov channel, the loss dependencies should be considered, and the calculating cases of the Gilbert channel are more complex than Bernoulli one. When calculating the probability of n-symbols, all 2^n cases represent a rather elaborate calculation process.

In this paper, the Gilbert model is still used as the burst-loss process, and the loss dependencies would be analyzed numerically. Consider the loss pattern for a transmitted n-length symbol sequence as an n-bit binary random variable $L_n = \{H_i\}_{i=1}^{n}$. The random variable $H_i = 0/1$ indicates the correct or loss receipt of the i-th symbol. Then, the total number of all possible cases of L_n is 2^n. An ordered set can be defined as $\mathbf{I}_n = \{l_n^r | 1 \le r \le 2^n\}$. The elements of $\mathbf{I}_n$ represent n-bit binary numbers and can be defined as

$$l_n^r = \begin{cases} 0^n & \text{for } r = 1 \\ 1 + l_n^{r-1} & \text{for } 1 < r \le 2^n \end{cases} \tag{4}$$

where 0^n represents n successive symbols. Therefore, the r-th loss pattern of an n-length symbol sequence can be defined as l_n^r.

For different loss channel models, the same loss pattern, l_n^r, is computed, and the probabilities of $P(l_n^r)$ will be different in different channels. $P(l_n^r)$ can be taken recursively. From the definition of $\{l_n^r\}$, given $P(l_n^r),\ r = 1, \ldots,\ 2^{n-1}$, the loss pattern probabilities for the two-state Markov model can be written as

$$\begin{cases} P(l_1^1) = 1 - P_B \\ P(l_1^2) = P_B \\ P(l_n^{4w-3}) = (1-p)P(l_{n-1}^{2w-1}) \\ P(l_n^{4w-2}) = pP(l_{n-1}^{2w-1}) \\ P(l_n^{4w-1}) = qP(l_{n-1}^{2w}) \\ P(l_n^{4w}) = (1-q)P(l_{n-1}^{2w}) \\ \qquad\qquad \textit{for } w = 1, \ldots, 2^{n-2} \end{cases} \tag{5}$$

2.2. Decodable frame rate for Bernoulli channel

In H.264/AVC, a video can be divided into several GOPs, and each GOP contains three types of frames in a periodic sequence. Thus, the raw video data of H.264/AVC video are encoded into Intra-coded (I), Predictive (P), and Bidirectional (B) video frames. Each GOP begins with an I frame, which is an independently coded picture. A P frame is encoded based on motion differences from the previous I frame or P frame, and should be in relation to previous I frame or P frame. B frames are encoded based on the motion differences from the immediate past and future I or P frames. There is a descending order of importance among their three types of pictures as I, P, and B frames. The organization of frames in a typical GOP is arranged as follows:

$$\mathrm{I\,B_{0,0}\cdots B_{0,N_{BP}-1}P_1\cdots P_mB_{m,0}\cdots B_{m,N_{BP}-1}P_{m+1}\cdots P_{N_P}B_{N_P,0}\cdots B_{N_P}},\ {}^{P}\mathrm{N_{BP}} - 1, \tag{6}$$

where $\mathrm{N_P}$ is the number of P frames, $\mathrm{N_B}$ is the number of B frames in the GOP, and $\mathrm{N_{BP}}$ is the number of B frames between an I and a P frame or two P frames. Let R_F be the encoding frame rate per second. The effective GOP transmission rate G is given by

$$G = \frac{R_F}{1 + N_P + N_B} \tag{7}$$

DOI: 10.1002/dac

Therefore, the probabilities of successful transmission for three frame types are obtained from

$$\begin{aligned} Q_I &= B(S_I + S_{IF}, S_I, P_B), \\ Q_P &= B(S_P + S_{PF}, S_P, P_B), \\ Q_B &= B(S_B + S_{BF}, S_B, P_B). \end{aligned} \tag{8}$$

where the operation $B(n, k, p)$ is the successful transmission of n packets, of which at least k packets will be successful at the receiver as the failure probability of an event is p. Q_I, Q_P, and Q_B are the probability of successful transmission of an I, P, or B frame, respectively; S_I, S_P, and S_B are the I, P, and B frame sizes (in packets); S_{IF}, S_{PF}, and S_{BF} are the numbers of FEC packets for I, P, and B frames.

The DFR is a good measure of video streaming performance in a lossy network. The DFR is defined as the expected number of decodable frames at the receiver, and can be calculated according to [23], as shown in Equation (9).

$$R = G{\cdot}Q_I \left[\begin{array}{c} 1 + \dfrac{Q_P - Q_P^{N_P+1}}{1 - Q_P} + \\ N_{BP}{\cdot}Q_B \left(\dfrac{Q_P - Q_P^{N_P+1}}{1 - Q_P} + Q_I Q_P^{N_P} \right) \end{array} \right] \tag{9}$$

where R is the expected number of decodable frames received error-free in 1 s.

3. ANALYTICAL MODEL

The proposed analytical model aims at providing the reference performance of FEC on H.264/AVC video over burst-loss channels. In Section 2.2, the relationship between the burst loss and DFR is not addressed. In this section, the case of I and P frames would be discussed first, and then the B frames will be considered in our proposed model in the next section.

3.1. Performance model with bursty channel

The coding dependence is discussed in Section 2.2. However, the transmission sequence is different with coding dependence, and the transmission sequence is an important issue in discussing the burst channel model. As shown in Figure 2, the general transmission sequence of video streaming in a GOP accounts for the first and last GOP. There are two special issues of transmission sequences that should be considered: (a) because of the coding dependence, the part of the B frames in the previous GOP will be sent after the current I frame, but these B frames will not affect the results of the current DFR; (b) another I frame in the next GOP will also be sent before the last group of B frames, but this I frame will affect the results of calculation of the current DFR. In Figure 2, the part of the B frames following the leading I frame will not affect the DFR of the current GOP. Therefore, $\mathbf{K}$ is defined as an ordered sequence set of source packets, and is given by

$$\begin{aligned} \mathbf{K} &= \left\{ \begin{array}{c} k_0 = S_I, k_1 = S_P, k_2 = S_B, \ldots, \\ k_{(N_B+1)+1} = S_P, \ldots, k_{N_P(N_B+1)+1} = S_I, \ldots \\ , k_{(N_P+1)(N_B+1)} = S_B \end{array} \right\} \\ &= \left\{ \left. \begin{array}{l} k_x = S_I, \\ k_{y(N_B+1)+1} = S_P, \\ k_{y(N_B+1)+z+1} = S_B \end{array} \right| \begin{array}{l} x = 0, \\ x = N_P(N_B+1)+1, \\ 0 \le y \le N_P, \\ 0 < z \le N_B \end{array} \right\} \end{aligned} \tag{10}$$

DOI: 10.1002/dac

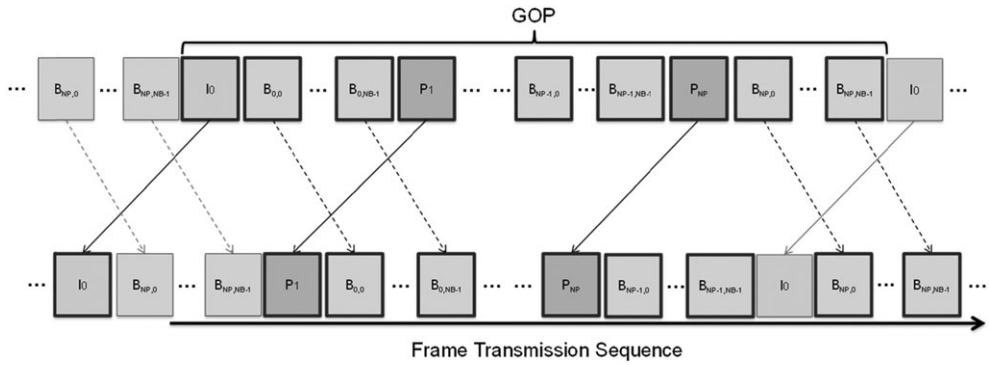


Figure 2. The transmission sequence of video streaming.

Moreover, in order to simplify the discussion of redundant packets, the redundant packets of the P and B frames are set to S_{PF} and S_{BF} for all P and B frames in a GOP. **H** is defined as an ordered set of redundant packets, and can be shown as follows:

$$\begin{aligned}\mathbf{H} &= \left\{ \begin{array}{c} h_0 = S_{IF}, h_1 = S_{PF}, h_2 = S_{BF}, \ldots, \\ h_{(N_B+1)+1} = S_{PF}, \ldots, h_{N_P(N_B+1)+1} = S_{IF}, \\ \ldots, h_{(N_P+1)(N_B+1)} = S_{BF} \end{array} \right\} \\ &= \left\{ \begin{array}{l|l} h_x = S_{IF}, & x = 0, \\ h_{y(N_B+1)+1} = S_{PF}, & x = N_P(N_B+1)+1, \\ h_{y(N_B+1)+z+1} = S_{BF} & 0 \le y \le N_P, \\ & 0 < z \le N_B \end{array} \right\} \end{aligned} \tag{11}$$

According to the coding behavior of video streaming, I frame and P frames are classified as interfering frames. If one of the interfering frames is lost, it will interfere with the decoding of other frames. Therefore, all of the B frames will be classified as non-interfering frames as they do not interfere with other frames. In the interfering frames, the first interfering frame is an I frame, and the other interfering frames are P frames. The total number of interfering frames is N_P+2 because I frame of the next GOP is also considered. If the previous interfering frame is not decodable, it will affect the decodability of the following frames. Let g_i be the loss patterns of the i-th decodable interfering frame following a lost interfering frame, that is, $g_i = \{0^i 1 | 0 < i \le N_P + 1\}$, where 0^i denotes consecutively decodable frames. If all of the interfering frames can be successfully decoded, let G_i be the loss pattern of consecutive N_P+2 decodable interfering frames, that is, $G_i = \{0^i | i = N_P + 2\}$. Therefore, the loss patterns of the i-th decodable interfering frames ($\mathbf{d}_i$) can be denoted as

$$\mathbf{d}_i = \left\{ g_i = \left\{ 0^i 1 \middle| 0 < i \le N_P + 1 \right\} G_i = \left\{ 0^i | i = N_P + 2 \right\} \right. \tag{12}$$

Next, in the non-interfering frames, because the loss of a non-interfering frame generally has no bearing on other frames, all B frames can be easily grouped to a sequence without considering I or P frames. Therefore, when the i-th interfering frame is decodable, the loss patterns of B frames ($\mathbf{b}_i$) are shown by

$$\mathbf{b}_i = \left\{ I^r_{i \cdot N_B} \middle| \begin{array}{l} 0 < i \le N_P + 1, \\ 1 \le r \le 2^{i \cdot N_B} \end{array} \right\} \tag{13}$$

DOI: 10.1002/dac

Finally, Equations (12) and (13) can be merged into a whole GOP pattern. Given the loss pattern for the video frame in the GOP, the set $\mathbf{D}_i^{N_B}$ is defined as the decodable loss pattern of the i-th interfering frames in the GOP.

$$\begin{aligned}\mathbf{D}_i^{N_B} &= \{merge(\mathbf{d}_i, \mathbf{b}_{i-1}, N_B) | 0 < i \leq N_P + 1\} \\ &= \begin{cases} \{merge(g_i, \mathbf{b}_{i-1}, N_B) | 0 < i \leq N_P + 1\} \\ \{merge(G_i, \mathbf{b}_{i-1}, N_B) | i = N_P + 2\} \end{cases}\end{aligned} \tag{14}$$

where the bit operator $merge(x, y, s)$ first selects s bits from y and then inserts them into a suitable location, which is behind the second bit of x. Moreover, the bit operator would select the next s bits from y and insert them into the next suitable location, which is behind the third bit of x, until all bits of y are spired to x. For instance, $merge(00001, 110001, 2) = 00110000011$.

The n-bit binary random variable can be set as A_j to indicate the loss pattern of the j-th frame before FEC recovery. Correspondingly, a k-bit binary random variable B_j is used to represent the successful receiving of the k source packets in the j-th frame after FEC recovery. If the receiver cannot successfully receive the k packets to be decoded, these loss patterns can be denoted as $\overline{B}_j$. As some packets could be lost because of a lossy channel, the FEC decoder would convert A_j into B_j, denoted as $F(A_j) = B_j$. Similarly, the $\overline{B}_j$ can be denoted as an n-bit binary random variable $\overline{A}_j$. For each GOP, given the loss pattern for the video frame, $\mathbf{D}_i^{N_B}$, the loss pattern of a video frame can be converted to the loss pattern of the video packets associated with the frame. Each bit of $\mathbf{D}_i^{N_B}$ will be extended to a k-bit B_j, and then the probability of the video packets loss pattern $\mathbf{D}_i^{N_B}$ can be written as

$$\begin{aligned}P\left(\mathbf{D}_i^{N_B}\right) &= \begin{cases} \Pr\left\{B_0, \ldots, \overline{B}_{i+(i-1)N_B}\right\} & for \quad 0 < i \leq N_P + 1 \\ \Pr\left\{B_0, \ldots, B_{(i-1)(N_B+1)}\right\} & for \quad i = N_P + 2 \end{cases} \\ &= \begin{cases} \Pr\left\{F(A_0), \ldots, F\left(\overline{A}_{i+(i-1)N_B}\right)\right\} \\ \qquad\qquad\qquad for \quad 0 < i \leq N_P + 1 \\ \Pr\left\{F(A_0), \ldots, F\left(A_{(i-1)(N_B+1)}\right)\right\} \\ \qquad\qquad\qquad for \quad i = N_P + 2 \end{cases}\end{aligned} \tag{15}$$

Then, the set of loss patterns $\mathbf{A}_j$ indicates the loss patterns when there is no packet lost in the j-th frame after FEC recovery. $\mathbf{A}_j$ can be given by

$$\mathbf{A}_j = \left\{A_j \middle| num\left(A_j\right) \leq h_j, h_j \in \mathbf{H}\right\} \tag{16}$$

where $num(x)$ denotes the number of lost packets in binary pattern x. In Equation (16), this means that less than h_j packets are lost in the j-th frame, and this frame is decodable by the FEC decoder.

In other words, $\overline{A}_j$ signifies that at least h_j of the transmitted packets are lost in the j-th frame. In this situation, the set $\overline{\mathbf{A}}_j$ represents all cases and it can be given by

$$\overline{\mathbf{A}}_j = \left\{\overline{A}_j \middle| num\left(A_j\right) > h_j, h_j \in \mathbf{H}\right\} \tag{17}$$

DOI: 10.1002/dac

From Equations (15), (16), and (17), the $P\left(\mathbf{D}_{\boldsymbol{i}}^{N_B}\right)$ can be written as

$$P\left(\mathbf{D}_{\boldsymbol{i}}^{N_B}\right) = \begin{cases} \Pr\left\{\mathbf{A}_0, \ldots, \overline{\mathbf{A}}_{i+(i-1)N_B}\right\} & \text{for } \ 0 < i \leq N_P + 1 \\ \Pr\left\{\mathbf{A}_0, \ldots, \mathbf{A}_{(i-1)(N_B+1)}\right\} & \text{for } \ i = N_P + 2 \end{cases} \tag{18}$$

Let s be a series bit binary variable for the loss pattern of GOP. A set $\mathbf{S}_{\boldsymbol{i}}$ is defined as the i-th decodable interfering frame. Equation (19) specifies the packet loss pattern set $\mathbf{S}_{\boldsymbol{i}}$. The bitwise operator $shl(x, n)$ appends n bits of successive zero to the binary pattern x.

$$\begin{aligned} &\mathbf{S}_{\boldsymbol{i}} \\ &= \begin{cases} \left\{ s \left| \begin{array}{l} s = \sum_{j=0}^{(i-1)(N_B+1)} shl\left(A_j, \sum_{m=j+1}^{i}(k_m + h_m)\right) + \overline{A}_{i+(i-1)N_B}, \\ k_m \in \mathbf{K}, h_m \in \mathbf{H}, A_j \in \mathbf{A}_{\boldsymbol{j}}, \overline{A}_{i+(i-1)N_B} \in \overline{\mathbf{A}}_{\boldsymbol{i}} \end{array} \right. \right\} & \textit{for } 0 < i \leq N_P + 1 \\ \left\{ s \left| s = \sum_{j=0}^{(i-1)(N_B+1)} shl\left(A_j, \sum_{m=j+1}^{i}(k_m + h_m)\right), k_m \in \mathbf{K}, h_m \in \mathbf{H}, A_j \in \mathbf{A}_{\boldsymbol{j}} \right. \right\} & \textit{for } i = N_P + 2 \end{cases} \end{aligned} \tag{19}$$

This operation concatenates on the packet loss pattern in each frame and constitutes a binary sequence of the packet loss pattern in the GOP. As a result, $s \in \mathbf{S}_{\boldsymbol{i}}$ bespeaks the joint sequence from $A_1 \in \mathbf{A}_1, \ldots, A_j \in \mathbf{A}_{\boldsymbol{j}}$ in a sequence. Let $m(i)$ be packet length before FEC recovery when the loss pattern is $\mathbf{D}_{\boldsymbol{i}}^{N_B}$; it can be given by

$$m(i) = \begin{cases} \sum_{j=0}^{(i-1)(N_B+1)+1} \left(k_j + h_j\right) & \textit{for } 0 < i \leq N_P + 1 \\ \sum_{j=0}^{(i-1)(N_B+1)} \left(k_j + h_j\right) & \textit{for } i = N_P + 2 \end{cases} \tag{20}$$

Then, let t be the index of loss pattern, and the set $\mathbf{T}_{\boldsymbol{i}}$ can be defined for the decimal number representation of a binary pattern:

$$\mathbf{T}_{\boldsymbol{i}} = \{t | t = 1 + dec(s), s \in \mathbf{S}_{\boldsymbol{i}}\} \tag{21}$$

where the operator $dec(x)$ transforms binary number x into decimal number. $l_{m(i)}^{t}$ is characterized as the t-th packet loss pattern with the packet length of $m(i)$. Equation (18) can be rewritten as

$$P\left(\mathbf{D}_{\boldsymbol{i}}^{N_B}\right) = \sum_{s \in \mathbf{S}_i} P(s) = \sum_{t \in \mathbf{T}_i} P\left(l_{m(i)}^{t}\right) \tag{22}$$

Finally, the cumulative expected DFR of a GOP, denoted by *DFR*, can be computed using

$$DFR = \sum_{i=1}^{N_P+1} \left[\overline{num}\left(\mathbf{D}_{\boldsymbol{i}}^{N_B}\right) \cdot P\left(\mathbf{D}_{\boldsymbol{i}}^{N_B}\right)\right] + \left[\overline{num}\left(\mathbf{D}_{N_P+2}^{N_B}\right) - 1\right] \cdot P\left(\mathbf{D}_{N_P+1}^{N_B}\right) \tag{23}$$

where $\overline{num}\ (x)$ computes the number of 0's in the binary pattern x.

DOI: 10.1002/dac

The formula in Equation (23) is the general form of the loss patterns model of the DFR in a GOP. In our proposed analysis model, a block of B frames of the previous GOP are not taken into consideration because these frames would not affect the DFR of the current GOP. However, the probability with a transmission sequence should be taken into consideration. The one-step matrix can be extended to an $M=(N_B \cdot S_B)$-step transition matrix, and then the M-step transition matrix can be written as

$$\varphi_M = \varphi^M \tag{24}$$

According to Equation (24), the transition probabilities of the Gilbert model between the (S_I+S_{IF})-th and $(S_I+S_{IF}+1)$-th packets are $(p_M, q_M)=((\varphi_M)_{1,2},(\varphi_M)_{2,1})$, and the loss pattern probabilities of Equation (5) should be extended to

$$\begin{cases} P\left(l_1^1\right) = 1 - P_B \\ P\left(l_1^2\right) = P_B \\ P\left(l_n^{4w-3}\right) = \begin{cases} (1-p)P\left(l_{n-1}^{2w-1}\right) & \text{for otherwise} \\ (\varphi_M)_{1,1} \cdot P\left(l_{n-1}^{2w-1}\right) & \\ \qquad \text{for } n = S_{GOP} - (S_I + S_{IF}) & \end{cases} \\ P\left(l_n^{4w-2}\right) = \begin{cases} pP\left(l_{n-1}^{2w-1}\right) & \text{for otherwise} \\ (\varphi_M)_{1,2} \cdot P\left(l_{n-1}^{2w-1}\right) & \\ \qquad \text{for } n = S_{GOP} - (S_I + S_{IF}) & \end{cases} \\ P\left(l_n^{4w-1}\right) = \begin{cases} qP\left(l_{n-1}^{2w}\right) & \text{for otherwise} \\ (\varphi_M)_{2,1} \cdot P\left(l_{n-1}^{2w}\right) & \\ \qquad \text{for } n = S_{GOP} - (S_I + S_{IF}) & \end{cases} \\ P\left(l_n^{4w}\right) = \begin{cases} (1-q)P\left(l_{n-1}^{2w}\right) & \text{for otherwise} \\ (\varphi_M)_{2,2} \cdot P\left(l_{n-1}^{2w}\right) & \\ \qquad \text{for } n = S_{GOP} - (S_I + S_{IF}) & \end{cases} \\ \qquad \text{for } w = 1, \ldots, 2^{(n-2)} \end{cases} \tag{25}$$

where S_{GOP} is the packets number of a GOP including source and redundant packets. The *DFR* can be obtained similarly over a burst channel for FEC by following Equations (10)–(25).

3.2. Computing complexity

In this section, the computing complexity of different schemes will be estimated. The complexity of calculating the time depends on the FEC cases discussed. In the FEC coding technique, a greater number of redundant packets will increase the cases of loss distribution in an FEC block. If there are K source packets and H redundant packets in an FEC block, the total number of decodable cases is $\sum_{i=0}^{H}\binom{K+H}{i}$ per block. For each case, $(\mathrm{K}+\mathrm{H}-1)$ multiplications and one summation are used to calculate the decodable probability in an FEC block. However, the number of summations is significantly smaller than the number of multiplications, and then the summations would be ignored in our analysis. For the I, P, and B frames, similar frame type has the same source packets, S_I, S_P, and S_B, and the same redundant packets, S_{IF}, S_{IP}, and S_{IB}. For an I frame, the total number of

DOI: 10.1002/dac

decodable cases can be defined by $B_I = \sum_{i=0}^{S_{IF}} \binom{S_I + S_{IF}}{i}$, and this also can be applied for P and B frames. Moreover, for an I frame, the total number of non-decodable cases is defined as $\overline{B}_I = \sum_{i=S_{IF}+1}^{S_I+S_{IF}} \binom{S_I + S_{IF}}{i}$.

An analysis model had been proposed for burst channels in [25]. This model does not take the transmission sequence into consideration in a GOP; it only considers the distribution of burst loss in a FEC block. Therefore, the total FEC cases can be given by

$$C_a = B_I + B_P + B_B \tag{26}$$

Finally, the DFR would be calculated by a video-dependent formula. The total number of multiplications can be written as

$$M_a = (n_I - 1)B_I + (n_P - 1)B_P + (n_B - 1)B_B + (N_P + 2)(N_P + 1) \tag{27}$$

In [26], Li *et al.* derive an analytical FEC model for burst channels without B frames, and discuss all possible decodable cases in a GOP. One of the assumptions is that the I frame would be decodable forever. Therefore, if the B frames are taken into consideration in Li *et al.* model [26], the discussed cases can be given by

$$C_b = B_I \cdot T_I {T_P}^{N_P} {T_B}^{(N_P+1)N_B} \tag{28}$$

where T_I, T_P, and T_B are the total numbers of distribution cases for I, P, and B frames, which means that $T_I = B_I + \overline{B}_I$, $T_P = B_P + \overline{B}_P$, and $T_B = B_B + \overline{B}_B$. Thus, the sum of multiplications can be written as

$$M_b = (2n_I + N_P \cdot n_P + (N_P + 1)n_B - 1) \cdot C_b \tag{29}$$

In order to reduce the computational complexity, Li *et al.* stated that the sliding window algorithm proposed in [27] can be employed. However, this algorithm is based on the distortion model, and hardly applied to IPB … patterns because the dependence of IPB … is more complex than IP … patterns.

In our proposed model, we not only take the transmission sequence into consideration but also discuss the decodable cases. Therefore, the total decodable cases can be given by

$$C_c = B_I \begin{bmatrix} \overline{B}_P \sum_{i=0}^{N_P-1} \left({B_P}^i {T_B}^{i \cdot N_B} \right) \\ + \overline{B}_I {B_P}^{N_P} {T_B}^{N_P N_B} + B_I {B_P}^{N_P} {T_B}^{(N_P+1)N_B} \end{bmatrix} \tag{30}$$

There are $N_P + 2$ different lengths in our analysis model. Therefore, the total number of multiplications in our proposed model can be defined by

DOI: 10.1002/dac

$$M_c = B_I \left\{ \begin{array}{l} \overline{B}_P \sum_{i=0}^{N_P-1} \left\{ \begin{bmatrix} n_I + n_P \\ +i\begin{pmatrix} n_P \\ +n_B N_B \end{pmatrix} \\ -1 \end{bmatrix} \left({B_P}^i {T_B}^{i \cdot N_B} \right) \right\} \\ + \begin{bmatrix} 2n_I \\ +N_P \begin{pmatrix} n_P \\ +n_B N_B \end{pmatrix} - 1 \end{bmatrix} \overline{B}_I {B_P}^{N_P} {T_B}^{N_P N_B} \\ + \begin{bmatrix} 2n_I \\ +N_P \begin{pmatrix} n_P \\ +n_B N_B \end{pmatrix} \\ +n_B N_B - 1 \end{bmatrix} B_I {B_P}^{N_P} {T_B}^{(N_P+1)N_B} \end{array} \right\} \tag{31}$$

The summations are ignored because the multiplications are larger than the summations. Obviously, compared with [26], our proposed analysis model simplifies the total decodable FEC cases. More comparisons of computing complexity will be discussed in the next section.

4. PERFORMANCE ANALYSIS

This section presents the analytical results for DFR models based on the proposed model, respectively. We further conduct NS-2 simulation experiments to observe the video streaming performance under a realistic network environment. The value of packet size was set to 1024 bytes. In the video sender, the test video clip ('Foreman') was encoded in the Quarter Common Intermediate Format (176×144). The encoder employed the frame pattern IPPP … and IPB … for the content and a constant quantization parameter QP = 2. The number of frame in this video clip is 400, and the GOP patterns are defined as GOP(x, y), where x denotes the distance between two anchor frames (I or P), and y denotes the distance between two full images (I frames). For the example GOP(3, 12), the GOP structure is IBBPBBPBBPBB. For all experiments, two different FEC UEP settings were adopted to determine the differentiated degrees of protection at the video frame level. At the stronger degree of protection, more low-priority frame types received FEC redundancies. In our experiments, two different protection cases are discussed. First, the stronger degree of protection had the redundant packets of 4, 1 for the I, P frames, respectively. Second, the weaker degree of protection just assigns fewer redundant packets in a GOP, and the redundant packets of I frame is set to 2. Briefly, two redundant packets are assigned to the I frame, and the others are not protected. All results were presented in DFR.

4.1. Analysis results

In the first set of experiments, the measured DFR, the estimation of the proposed model, [28], and [30] are all compared. The analysis model of [28] is defined in uniform lossy channels, and [30] does not take the transmission sequence into consideration. Figures 3 and 4 show the DFR results when the average burst length varies from 1 to 10 and the packet loss rate varies from 1% to 10%. The simulation results are also plotted to observe the DFR behavior in a realistic network environment. The results of the GOP structure without B frames (IPPPPPPPP) are shown in Figure 3, and the other GOP structure (IBBPBBPBBPBB) is shown in Figure 4. In GOP(1, 10), the mean size of I and P frame are 10084.25 and 4214.225 bytes. In the analysis of this case, the number of I frame packets are set to 10 packets, and the number of P frame packets are set to 5. On the other hand,

DOI: 10.1002/dac

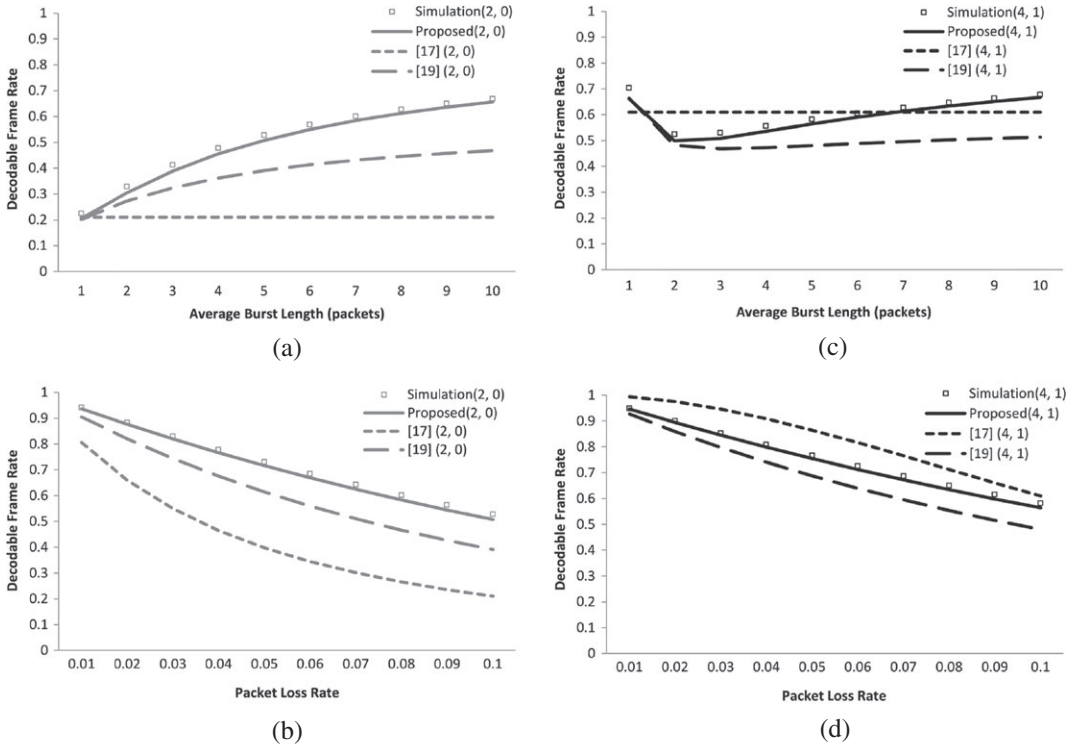


Figure 3. Comparisons of decodable frame rate for video streaming with GOP(1, 10).

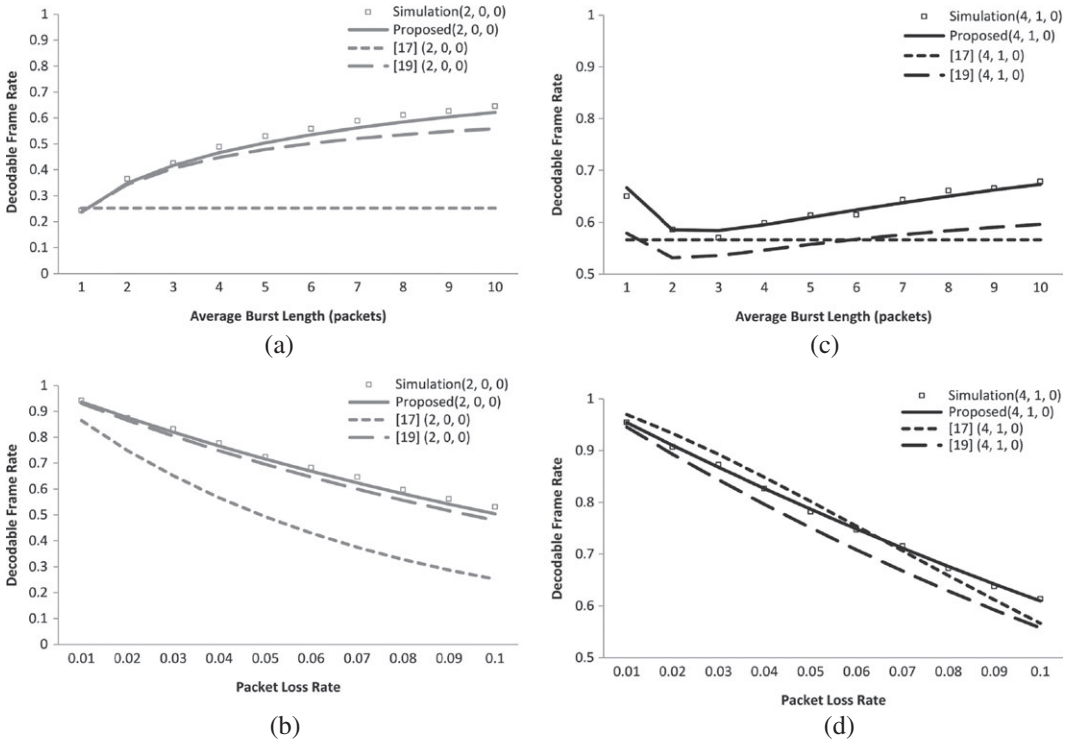


Figure 4. Comparisons of decodable frame rate for video streaming with GOP(3, 12).

DOI: 10.1002/dac

for GOP(3, 12), the mean size of I, P, and B frame are 10134.85, 5354.59, and 3307.55 bytes, respectively. Therefore, the number of I, P, and B frame packets are set to 10, 6, and 4. For all analytical and simulation cases, the average packet loss rate is set to 10% in both Figure 3(a, c) and Figure 4(a, c), and the average burst length is set to 5 in both Figure 3(b, d) and Figure 4(b, d). We examine the performances on two different FEC protection cases by adjusting the number of corresponding FEC redundant packets. For each set of experiments, the results of experiments with weaker protection degree are shown in Figure 3(a, b) and Figure 4(a, b), whereas those with stronger protection degree are shown in Figure 3(c, d) and Figure(c, d).

In our simulation, the results are plotted with a block symbol, and 95% confidence intervals are estimated from 2500 experiments. However, the standard deviation is too small (about $<0.5\%$) to make the distribution of simulation results close to a single point. As shown in Figures 3 and 4, it can be seen that the model of [28] is irresponsive to the changes in the burst packet loss length and presents the differentiated DFR values for different FEC UEP settings. In Figure 3(a), and (c), we can observe that firstly, the case with a weaker degree of UEP, FEC (2, 0), has increased DFR values as the average burst-loss length increases in both of our proposed analysis model and [30] and secondly, the case with a higher degree of UEP, FEC (4, 1), has decreased DFR values as the average burst-loss length is increased. Moreover, the DFR values will decrease with increasing average burst-loss length, but will increase when the average burst length is larger than 2 because of the concentrated loss. Comparing with the other models, our proposed model is close to the simulation results (the relative error $<3\%$). In the analysis of experiment, the simulation results do not completely match numerical results because of the different numbers of packets in a GOP. In the simulation, all frames are assigned to the target redundant packets, but the frame size in a video trace varies. In our proposed numerical model, the number of packets for I, P, and B frame is assigned to the average number of packets in the video trace.

The situations described previously can also be shown in Figure 4, obviously. The GOP structure of Figure 4 has eight B frames. Whether there is a stronger or weaker degree of UEP, the B frame will not be assigned any redundant packet, and the protection of other frames is the same as in Figure 3. Comparing Figure 3 with Figure 4, the DFR values of the stronger degree of UEP are still higher than those obtained by the weaker degree of UEP. The differences between [30] and the simulation results in Figure 4 are less than those of Figure 3. It is because the impact of corresponding frame in GOP(1, 10) is large than that in GOP(3, 12). Moreover, it is noted that the longer burst packet loss length produces a lower packet loss rate because more packet losses easily aggregate into a long burst packet loss length with a fixed packet loss rate. For the cases with a weaker degree of UEP, the long burst-loss length thus causes increased DFR values. As to the cases with stronger degrees of UEP, it is observed that the longer burst-loss length easily decreases the FEC efficacy because the receiver may not receive a sufficient number of packets with the FEC-encoded P and B frames to reconstruct the original video frames. However, when the burst loss length increases to be beyond a certain value, the DFR curve rises instead. This is because the losses easily concentrate within a frame in the case of long burst loss length.

For [28], all DFR curves in Figure 3(a, c) are the same as those in Figure 4(a, c) because the uniform-DFR model is unaware of burst packet loss. From Figures 3 and 4, the uniform-DFR model underestimates the DFR for the cases with a lower degree of UEP, while it overestimates the DFR for the cases with a higher degree of UEP. Although the burst channel model is applied for [30], the results do not match because the transmission sequence is not taken into consideration.

In Figure 3(b, d), the curves of all schemes are close to a linear equation, and the curves of our proposed model are still matched with the simulation results. With the weaker degree of UEP, the results of [28] deviate significantly because of the uniform loss model. Although [30] takes the burst-loss model into consideration, the analysis model of [30] just focuses on the impact of burst loss in one frame. This analysis model does not consider the impact factor of burst loss between different frames. With the stronger degree of UEP, the results of [28] seriously deviate from the simulation results. In Figure 4, as the B frames are taken into consideration, with the weaker degree of UEP, the curves of [28] and [30] are close to each other because the impact between different frames is more important. On the contrary, with the stronger degree, the slop of a curve of [30] shares the same trend as that of our simulation. In summary, the observations of Figure 3 are the

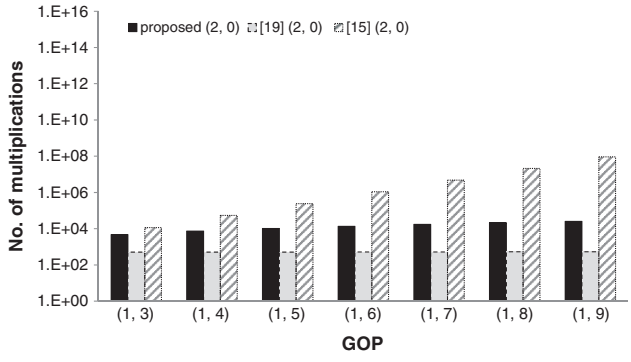
DOI: 10.1002/dac

same as Figure 4. The performance of [28] still deviates significantly. Although the slope of a curve of [30] is close to the simulation, the error coefficient is still too high to evaluate in different network environments. In addition, whatever the results of different cases shown, the DFR curves of the simulation and the proposed model are close to each other.

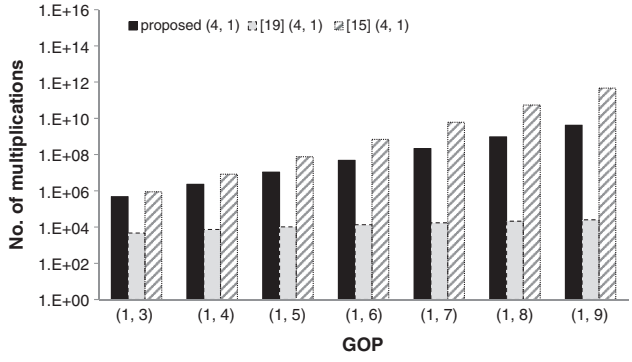
4.2. Computing complexity

In the second set of experiments, the measured computing complexity, the estimation of the proposed model, [30], and [26] are all compared. Figures 5 and 6 show the comparisons of a number of multiplications with different GOP structures and FEC coding rates. In the source packets of the video clip ('Foreman'), the average source packets can be set as eight packets for an I frame, two packets for a P frame, and one packet for a B frame. The results of the GOP structure without B frames are shown in Figure 5, and further GOP structures with B frames are shown in Figure 6.

As is well known, the computing complexity will increase with increasing GOP lengths, but the curves of complexity at different GOP lengths in [30] are close to a horizontal line. Obviously, the computing complexity of [30] has lower multiplications than the others because the transmission 4sequence is not taken into consideration. However, the analysis results of [30] do not match the simulation results in Figures 3 and 4. Moreover, the analysis model of [26] is designed for the distortion analysis in video streaming without B frames, and it calculates all of the encoding and decoding cases. In Figures 5 and 6, the computational complexity should be discussed, and do not take the calculated video distortion model into account. As shown in Figures 5 and 6, the computing complexity of [26] is the highest in all of the situations. The difference between our proposed and [26] is exponential growth, and the growth rate is about 2. The complexity of this model has been extended to video streaming with B frames in this paper. However, with the increasing of GOP lengths, although the analysis model of [26] is good at evaluating the quality of video streaming, the complexity is too high to be evaluated in real network environments because of the higher computing complexity.



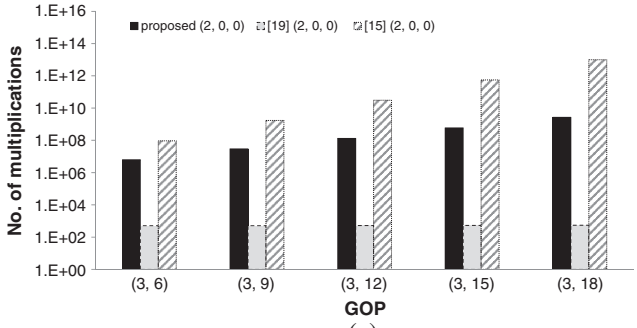
(a)



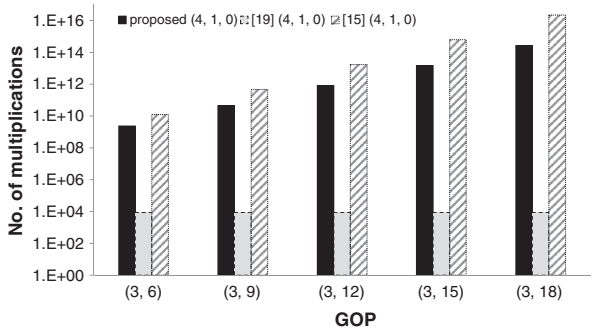
(b)

Figure 5. Comparisons of number of multiplications with GOP(1, 9).

DOI: 10.1002/dac



(c)



(d)

Figure 6. Comparisons of number of multiplications with GOP(3, 12).

In Figure 5, the complexity of our proposed analysis model is higher than that in [30] because of the transmission sequence. There is an interesting result in Figure 5(a). In this case, there are no redundant packets in P frames, and the complexities of our proposed model are close to each other at different GOP lengths. However, when some redundant packets are added to the P frames for protection, as in Figure 5(b), the complexity increases exponentially. In Figure 6(a, b), when B frames are taken into consideration, the complexity of our proposed model is much higher than in [30], but still lower than in [26]. In Figure 6(a), even if there are no redundant packets in the P and B frames, the complexities of our proposed model will increase because the GOP structures included several B frames. However, when one redundant packet is added into the P frame in Figure 6 (b), the complexity of our proposed model increases. Therefore, our proposed analysis model can provide more accurate results than the model of [30] and can also give a lower complexity analysis than the model of [26].

5. CONCLUSIONS

In this paper, an analytical model is derived to evaluate the H.264/AVC video delivery performance of a frame-level FEC scheme over burst-loss channels. The analytical results show that the burst-loss affects the FEC efficacy, and the DFR model based on the uniform loss process easily leads to a performance bias under the burst-loss condition. Through a series of NS-2 simulation experiments, it is shown that our proposed model can provide a more accurate analysis than [30] and lower complexity than [26]. However, the complexity of our model is still too high to apply to high-quality video because of the inclusion of a larger data source. Future works will include more FEC techniques, such as interleaving, in the proposed model, and will use an n-state Markov model as the packet loss process to adapt the proposed model to the dynamically varying network situation in wireless mobile communications. Moreover, the impact model of different FEC codes, such as low-density parity-check and rateless codes, in bursty channel will be an interesting future work to be discussed.

DOI: 10.1002/dac

ACKNOWLEDGEMENTS

The authors would like to thank the National Science Council, Taiwan, R.O.C., for financially supporting this research under Contact No. <102-2221-E-507-003>.

REFERENCES

1. Nafaa A, Taleb T, Murphy L. Forward error correction strategies for media streaming over wireless networks. *IEEE Communications Magazine* 2007; **46**(1):72–79. DOI: 10.1109/MCOM.2008.4427233.
2. Kang SR, Loguinov D. Modeling best-effort and FEC streaming of scalable video in lossy network channels. *IEEE/ACM Transactions on Networking* 2007; **15**(1):187–200. DOI: 10.1109/TNET.2006.890110.
3. Liang Z, Dong X, Aaron Gulliver T, Liao X. Performance of transmitted reference pulse cluster ultra-wideband systems with forward error correction. *International Journal of Communication Systems* 2014; **27**(2):265–276. DOI: 10.1002/dac.2359.
4. Bouras C, Kanakis N, Kokkinos V, Papazois A. Application layer forward error correction multicast streaming over LTE networks. *International Journal of Communication Systems* 2013; **26**(11):1459–1474. DOI: 10.1002/dac.2321.
5. Kliazovich D, Redana S, Granelli F. Cross-layer error recovery in wireless access networks: the ARQ proxy approach. *International Journal of Communication Systems* 2012; **25**(4):461–477. DOI: 10.1002/dac.1271.
6. Inoie A. Audio quality in lossy networks for media-specific forward error correction networks. *International Journal of Communication Systems* 2014; **27**(2):289–302. DOI: 10.1002/dac.2361.
7. Mohr AE, Riskin EA, Ladner RE. Unequal loss protection: graceful degradation of image quality over packet erasure channels through forward error correction. *IEEE Journal on Selected Areas in Communications* 2000; **18**(6):819–828. DOI: 10.1109/49.848236.
8. Masnick B, Wolf J. On linear unequal error protection codes. *IEEE Transactions on Information Theory* 1967; **13**(4):600–607. DOI: 10.1109/TIT.1967.1054054.
9. Baccaglini E, Marchetto G, Tillo T, Olmo G. Efficient slice-aware H.264/AVC video transmission over time-driven priority networks. *International Journal of Communication Systems* 2013. DOI: 10.1002/dac.2578.
10. Liu X, Yang LT, Zhu W, Sohn K. Efficient temporal error concealment algorithm for H.264/AVC inter frame decoding. *International Journal of Communication Systems* 2011; **24**(10):1282–1297. DOI: 10.1002/dac.1193.
11. Chen JL, Liu SW, Wu SL, Chen MC. Cross-layer and cognitive QoS management system for next-generation networking. *International Journal of Communication Systems* 2011; **24**(9):1150–1162. DOI: 10.1002/dac.1218.
12. Yang X, Zhu C, Li GZ, Lin X, Ling N. An unequal packet loss resilience scheme for video over the internet. *IEEE Transactions on Multimedia* 2005; **7**(4):753–765. DOI: 1109/TMM.2005.846782.
13. Zhang Y, Qin S, He Z. Transmission distortion-optimized unequal loss protection for video transmission over packet erasure channels. *Proceedings of the IEEE International Conference on Multimedia and Expo*, Barcelona, Spain, 2011; 1–6. DOI: 10.1109/ICME.2011.6011931.
14. Tillo T, Baccaglini E, Olmo G. Unequal protection of video data according to slice relevance. *IEEE Transactions on Image Processing* 2011; **20**(6):1572–1582. DOI: 10.1109/TIP.2010.2095865.
15. Liu Y, Qaisar SB, Radha H, Men A. On unequal error protection with low density parity check codes in scalable video coding. *Proceedings of the Information Sciences and System*, Baltimore, MD, 2009; 793–798. DOI: 10.1109/CISS.2009.5054826.
16. Kondrad L, Bouazizi I, Gabbouj M. LDPC FEE code extension for unequal error protection in DVB-T2 system: design and evaluation. *International Journal of Digital Multimedia Broadcasting* 2012. DOI: 10.1155/2012/834924.
17. Ahmad S, Hamzaoui R, Al-Akaidi MM. Unequal error protection using fountain codes with application to video communication. *IEEE Transactions on Multimedia* 2011; **13**(1):92–101. DOI: 10.1109/TMM.2010.2093511.
18. Lou Z, Song L, Zheng S, Ling N. Raptor codes based unequal protection for compressed video according to packet priority. *IEEE Transactions on Multimedia* 2013; **15**(8):2208–2213. DOI: 10.1109/TMM.2013.2280561.
19. Loguinov D, Radha H. End-to-end internet video traffic dynamic: statistical study and analysis. *Proceedings of the IEEE INFOCOM* 2002; **2**:723–732. DOI: 10.1109/INFCOM.2002.1019318.
20. Bolot JC. End-to-end packet delay and loss behavior in the Internet. *Proceedings of the ACM SIGCOMM* 1993; **23**(4):289–298. DOI: 10.1145/167954.166265.
21. Kang K, Shin H. Reduced data rates for energy efficient Reed-Solomon FEC on fading channels. *IEEE Transactions on Vehicular Technology* 2009; **58**(1):176–187. DOI: 10.1109/TVT.2008.923671.
22. Lai CC, Tiang PW, Abdullah MK, Ali BM, Mahdi MA. FEC performance analysis based on Poisson and bursty error patterns for SDH and OTN systems. *Photonic Network Communications* 2006; **11**(3):265–270. DOI: 10.1007/s11107-005-7353-5.
23. Yu X, Modestino JW, Kurceren R, Chan YS. A model-based approach to evaluation of the efficacy of FEC coding in combating network packet losses. *IEEE/ACM Transactions on Networking* 2008; **16**(3):628–641. DOI: 10.1009/TNET.2007.900416.
24. Chakareski J, Chou PA. Application layer error-correction coding for rate-distortion optimized streaming to wireless clients. *IEEE Transactions on Communication* 2004; **52**(10):1657–1687. DOI: 10.1109/ICASSP.2002.5745158.
25. Tan WT, Zakhor A. Video multicast using layered FEC and scalable compression. *IEEE Transactions on circuits and Systems for Video Technology* 2001; **11**(3):373–386. DOI: 10.1109/76.911162.

26. Li Z, Chakareski J, Shen L, Wang L. Video quality in transmission over burst-loss channels: a forward error correction perspective. *IEEE Communications Letters* 2011; **15**(2):238–240. DOI: 10.1109/LCOMM.2011.122810.101931.
27. Li Z, Chakareski J, Niu X, Zhang Y, Gu W. Modeling and analysis of distortion caused by Markov-model burst packet losses in video transmission. *IEEE Transactions on Circuits and Systems for Video Technology* 2009; **19**(7):917–931. DOI: 10.1109/TCSVT.2009.2022806.
28. Lin CH, Chilamkurti NK, Zeadally S, Shieh CK. Performance modeling of MPEG4 video streaming over IEEE 802.11 using distribution coordination function. *Wireless Communication and Mobile Computing* 2010; **11**(9):1312–1322. DOI: 10.1002/wcm.938.
29. Wu H, Claypool M, Kinicki R. Adjusting forward error correction with temporal scaling for TCP-friendly streaming MPEG. *ACM Transactions on Multimedia Computing, Communication and Applications (TOMCCAP)* 2005; **1**(4):315–337. DOI: 10.1145/1111604.1111605.
30. Kuo CI, Shih CH, Shieh CK, Hwang WS, Ke CH. Modeling and analysis of frame-level forward error correction for MPEG video over burst-loss channels. *Applied Mathematics & Information Sciences* 2014; **8**(4):1845–1853. DOI: 10.12785/amis/080442.
31. Bolot JC, Vega-Garcia A. The case for FEC-based error control for packet audio in the Internet. *ACM Multimedia Systems*, 1997.
32. Sanneck H, Carle G. A framework model for packet loss metrics based on loss run length. *Proceedings of the SPIE/ACM SIGMM Multimedia Computing and Networking*, San Jose, CA, USA, 2000; 177–187. DOI: 10.1117/12.373520.
33. Vuectic B, Du J. Channel modeling and simulation in satellite mobile communication systems. *IEEE Journal Selected Areas Communications* 1992; **10**(8):1209–1218. DOI: 10.1109/49.166746.
34. Tan CC, Beaulieu NC. On first-order Markov modeling for the Rayleigh fading channel. *IEEE Transactions on Communications* 2000; **48**(12):2032–2040. DOI: 10.1109/26.891214.