FLEXIBLE FEC AND FAIR SELECTION OF SCALABLE UNITS FOR HIGH QUALITY VOD STREAMING SERVICE OVER WIRELESS NETWORKS

Soo-young Jang, and Chae Y. Lee

Industrial and Systems Engineering, KAIST, Taejon, Korea, chae@ kaist.ac.kr

ABSTRACT

Providing high quality video on demand (VoD) streaming service over wireless networks is very challenging due to the limited capacity and error-proneness of the wireless environment. We propose a flexible forward error correction (FEC) and a fair selection scheme of scalable units that utilize a layered coding structure of H.264/SVC (scalable video coding). Three error-resilient techniques (e.g., unequal error protection, FEC, and retransmission) are adapted to minimize the total distortion of VoD streaming service. For flexible FEC, a rateless FEC code is adopted. The FEC code rates are based on the possible number of retransmission, the condition of the wireless channel and the layered coding structure of H.264/SVC for each packet. A theoretical study is performed to show how to utilize the possible number of retransmission for an adaptive FEC code rate. With fair selection, regular and retransmission-requested packets compete for resources without fixing the retry limit. Thus, excessive retransmission is prevented and the proposed scheme effectively provides capacity-limited and delay-constrained VoD streaming services. For this fair selection of scalable units, we formulate the problem using binary integer programming and propose an effective low complexity selection algorithm based on a priority index. The proposed algorithm prioritizes packets according to the priority index and the H.264/SVC structure. We show that the proposed scheme can minimize the total video distortion compared to other heuristic procedures. Other effects of the various factors are also considered for the performance of the new scheme.

KEY WORDS

flexible forward error correction (FEC), retransmission, scheduling window, unequal error protection (UEP), VoD streaming service.

1. INTRODUCTION

Recently, video traffic over wireless networks is increasing explosively with the proliferation of tablet PCs, smartphones and other mobile devices. Wireless networks, however, have intrinsically fundamental limitations including limited capacity, being prone to error and time-varying channel conditions.

To overcome the challenges of providing VoD streaming service over wireless networks, several technologies have been developed and standardized. H.264/SVC [1, 2] and error resilient techniques are representatives. H.264/SVC is a video compression standard that supports spatial, temporal and quality scalabilities. These scalabilities are implemented using a layered coding scheme (i.e., one base layer and one or more enhancement layers). Video playback is still possible without enhancement layers, but the cost is a decrease in video quality. Support of the scalabilities enables flexible adaptation to bandwidth fluctuation, bit errors and the heterogeneity of the devices used. We need to consider several aspects to fully take advantage of the H.264/SVC standard. Among them, the most typical two aspects are1) different transmission

DOI : 10.5121/ijmit.2018.10204

resource requirement and different impact on video distortion of each scalable unit (SU). Here, an SU is defined as a quality layer in each frameof each temporal layerin each Group of Pictures (GoP) 2) the occurrence of a packet error that affects not only the quality of the frame in which it belongs, but also that of subsequent frames due to inter-frame dependencies caused by the layered coding structure of H.264/SVC. Error resilient techniques are those that mitigate errors during transmissions that might be due to buffer overflow, or other reasons. Forward error correction (FEC), retransmission, and unequal error protection (UEP) are typical techniques to establish error resilience. To efficiently utilize FEC, retransmission, and UEP, how much FEC redundancy should be assigned, which packets should be retransmitted, and what degree of unequalness should be set, all need to be considered.

In this paper, we propose a flexible forward-error-correction (fFEC) and fairselection scheme of scalable units (fSSU) intended to minimize the total distortion of VoD streaming service. The proposed scheme fully utilizes the layered coding structure of H.264/SVC and adapts three error resilient techniques (i.e., FEC, retransmission, and UEP) to efficiently solve the problem. For flexible FEC the FEC code rates are based not only on bit error rate (BER) and distortion caused by the loss of SUs, but also on the possible number of retransmission. An analytical study is presented to show how to utilize the possible number of retransmission for an adaptive FEC code rate.

For fair selection the regular and the retransmission-requested packets compete for resources to minimize video distortion in each scheduling period. Using this new approach, we can prevent excessive retransmissions that may interrupt regular packets and thus increase video distortion. Throughout the paper, we use the term regular packets to describe packets that were never transmitted before, and the term retransmission-requested packets to describe packets for which earlier transmission attempts were unsuccessful.The fair selection of SUs isformulated asa binary integer programming (BIP) which is NP-hard. Thus we propose an effective low complexity fair selection algorithm, based on a proposed priority index considering both partial transmission and the possibility of retransmission.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 describes the system models, and the proposed fFEC-fSSU scheme is presented in Section 4. Section 5 provides evaluations of the simulation. Finally, the conclusions of the paper are presented in Section 6.

2. RELATED WORK

Error-resilient techniques to mitigate video distortions caused by propagation errors have been studied by many researchers. Joint source and channel coding (JSCC) schemes for the optimal partitioning of source and channel bits within limited capacity were considered in [3-5] using Lagrangian-based algorithms. These works may be successful in the environment without retransmissions. However, the rate-distortion curves and Lagrangian-based algorithms may increase the complexity of the optimization frameworks and make the JSCC infeasible for VoD streaming services. A new JSCC scheme in more general packet-based multimedia transmission model with low complexity hill-climbing method was proposed in [6]. UEP-FEC schemes with simple and accurate performance metrics to exploit the importance of prioritized video packets were proposed in [7, 8]. In two other cases, a UEP-FEC scheme utilizing a GA-based approach that considered two-dimensional scalabilities, and a layer-aware FEC (LA-FEC) that considered layered media codecs, respectively, were proposed [9, 10]. Whether to add a new packet to the transmission queue or increase the protection level of an existing packet in the queue according to the expected distortion gradient has also been proposed [11]. Another proposal included an FEC code-rate setting and layer-selection scheme using the virtual congestion levels providing feedback from network nodes [12]. Most of these workers did not consider the retransmission technique for error protection [4-12]. Without retransmission, the occurrence of burst packet losses caused by instantaneous channel-quality-degradation would severely degrade the video quality. Moreover, the possible number of retransmission was also not considered in setting the FEC code rate [3-12].

In other work, the picture type was used to set the maximum number of retransmission (i.e., I-, P-, and B- pictures were set to 5, 3, and 1, respectively)[13]. The importance of picture type (with the constraint, "Round-trip delay time (RTT)< display deadline") was used to determine which packets were retransmitted [14]. An optimal retry limit was proposed that was informed by analysis of the reasons for packet losses [15]. Scheduling metrics that considered the tradeoff between distortion and deadline were suggested [16], but many of these authors [13-16] did not consider the FEC. In this case, when deep channel fading occurs, packet retransmissions will be triggered frequently and as a result, a majority of resources will be tied up by the retransmission-requested packets. This may severely degrade video quality since it interrupts the transmission of regular packets. Furthermore, most of these workers [13-15] assumed that retransmission of a packet that failed delivery went on until the packet was delivered successfully or until the retransmission count reached the maximum number of retransmission allowed, and while the last worker [16] fixed the bandwidth for retransmission. However, these assumptions may cause great inefficiencies in VoD streaming service since the resources required for retransmission is huge and varies greatly due to the occurrence of burst errors.

Virtual channels that offered different levels of reliability and statistical loss guarantee were suggested [17], and a UEP-scheme that adopted hybrid automatic repeat request (HARQ) was proposed [18]. There was a proposal for a cross-layer architecture that would optimize perceptual quality for delay-constrained, scalable video transmission [19]. In these cases, only a few different levels of FEC code rates were allowed, and the retry limit was fixed [17-19].

Furthermore, in this previous work related to retransmission [13-19] an assumption was made that the same FEC code rate would apply for retransmission requested packets, or that the modulation and coding scheme (MCS) level would just be lowered, independent of the channel conditions.
The contribution of our research is the following two aspects. First, we analytically show that even when the channel conditions do not change over consecutive transmissions, changing the FEC code rate for each transmission provides lower error probability than simply fixing the FEC code rate. Second, we propose a fair selection algorithm that considers the retransmission of erroneous packets by considering scheduling windows to enhance the performance of VoD streaming service. In each scheduling window, regular and retransmission-requested packets compete for resources without fixing the retry limit. Thus, excessive retransmission is prevented.

3. System Models

In this section we discuss the concepts of H.264/SVC, scheduling windows and retransmission.

3.1. OVERVIEW OF H.264/SVC

H.264/SVC supports three kinds of scalabilities (i.e., spatial, temporal, and quality scalability). For this reason, streaming service with various quality levels becomes possible using various combinations of these scalabilities. A video stream that contains several SUs related to spatial, temporal, and quality scalability may have enhanced resolution, frame rate, and frame quality, respectively. H.264/SVC supports scalabilities using a layered coding scheme. To successfully decode a layer of certain scalability, all of its lower layers must also be successfully decoded. The SUs of the lower layers have a greater impact on video distortion than do the SUs of the higher layers. The impact of SUs on video distortion also varies according to which scalabilities the SUs relate. This implies that each SU has a different impact on video distortion. The symbols required to transmit different SUs also differ. Moreover, the impact on video distortion and the symbols required for transmission are not proportional to each other. Thus the assumption that distortion is dependent on the amount of data received, which is inherent in the work [12, 20, 21], may cause inaccurate and inefficient results.

3.2. METRIC OF VIDEO QUALITY

Determining the peak signal-to-noise ratio (PSNR), which is the metric to determine the reconstructed video quality, requires relatively large amounts of calculation and time. However, scheduling should be done within a very short duration. For this reason, it is impractical to use the PSNR values in the scheduling algorithm. For this purpose, a low complexity, but efficient video-quality metric is required. For this purpose, we apply a simple and effective video-quality metric suggested previously [7, 8]. This metric approximates the effects of different SU losses on the reconstructed video quality by estimating the error propagation effect in regards to temporal layers, quality layers, and GoP orders.

$$D_{q,f,t,g} = \begin{cases} \dfrac{(G-g)^{C_3}}{(1+q)^{C_2}}\left(2^{(T-C_1 t)}-1\right), & if\ t=0 \\ \dfrac{1}{(1+q)^{C_2}}\left(2^{(T-C_1 t)}-1\right), & otherwise \end{cases} \tag{1}$$

s the estimated distortion caused by the loss of *SU(q,f,t,g)*.*SU(q,f,t,g)* represents the SU of q^{th} quality layer in f^{th} frame of t^{th} temporal layer in GoP #*g*. The parameters *q*, *f*, *t*, and *g* denote the quality layer id, frame number id, temporal layer id, and GoP order id, respectively. *T*represents the number of total temporal layers, and *G* is the number of GoPs in an intra-refresh period.C_1, C_2, and C_3 are the scaling factors related to the temporal layer id, quality layer id, and GoP order id, respectively, which are determined by numerical experimentations. Throughout the paper, we define the term 'the impact on video distortion' as the value $D_{q,f,t,g}$. The greater the impact on video distortion, the greater the quality degradation caused by the loss of certain SUs (i.e., these are more important than SUs with less impact on video distortion). Although the metric we used has low complexity and high effectiveness, the scheme to be proposed is general enough to accommodate any other metrics to measure the distortion of SUs.

3.3. SYSTEM ARCHITECTURE

Fig. 1 shows the system architecture. A video stream is encoded using the H.264/SVC video codec. We assume an H.264/SVC structure with three quality layers in each of four temporal layers and an intra-refresh period of three GoPs as shown; however, our scheme can be applied to other H.264/SVC structures. Each *SU(q,f,t,g)* is packetized into N packets since the size of SUs are too large to be transmitted whole. The number of fragments of an SU is determined by its size. We assume that an SU is the minimum decoding unit, and that all packets belong to an SU must be successfully transmitted to decode the SU. Each packet is encoded with FEC code (e.g., Raptor code) [22]. Since Raptor code is in the class of fountain codes (rate-less erasure codes), its code rate does not have to be one from the fixed-code-rate set. A symbol size is set to m bits. This packetization scheme lets a packet contain information symbols of only one SU and also, the same number of information symbols for each packet. This makes it more convenient to manage retransmission and to apply adaptive FEC. The scheduler only needs to know the IDs of the lost packets to see which and how many packets have been lost. Furthermore, when retransmitting a packet, we can adaptively set the FEC code-rate according to the characteristics of the SU to which that packet belongs (for its impact on video distortion and delay deadlines), as well as to the conditions of the wireless channel.

3.4. SCHEDULING WINDOW

A scheduling window is definedas a target set of SUs for scheduling in each scheduling period. A window consists of one or more GoPs, and scheduling window size, W, represents the number of GoPs included in the window. We have to decide which packets to select in each scheduling period within a scheduling window. The scheduling is done by considering only the packets constituting the GoPs in the window. The window shifts to the next GoP if the delay deadline of the least-numbered GoP in the current window is passed. Fig. 2 shows an example of how the scheduling window works. The size of the scheduling window and the buffering time are set to three GoPs and 160 ms, respectively. Utilizing the concept of scheduling windows helps video service to flexibly adapt to channel fluctuation, and prevent severe distortion even in the event of burst errors. If the scheduling window size is small, the occurrence of burst errors may severely degrade video quality, but if the scheduling window is too big, the complexity of scheduling may increase to the point that it becomes impractical to implement.

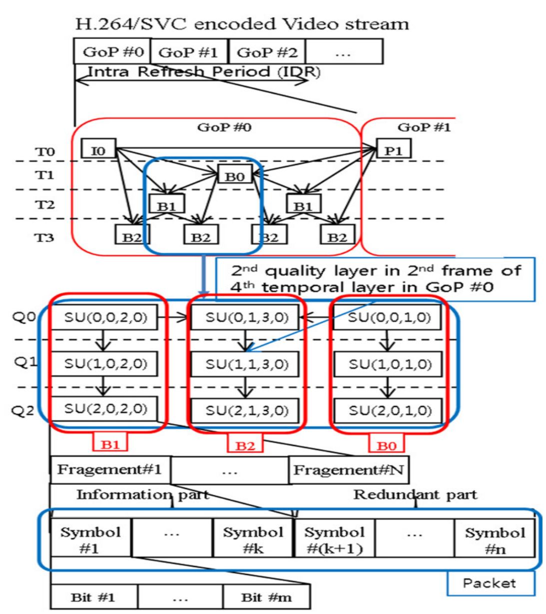

Fig. 1.System architecture

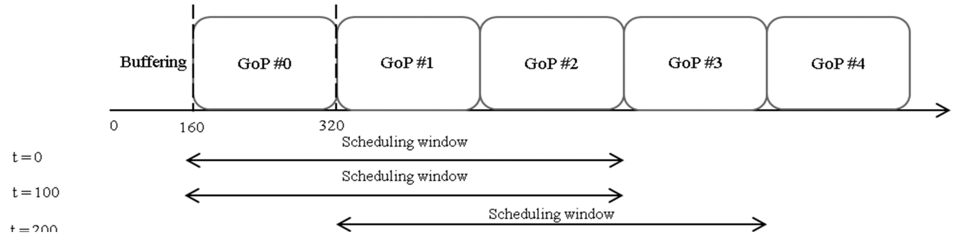

Fig. 2. Operation of scheduling window

3.5. RETRANSMISSION

Previous works that only consider FEC report that retransmission is not adequate to handle the delay for sensitive traffic such as video streaming. To handle this problem, we adopt the scheduling window to provide spare time until the delay deadlines of a packet. A few retransmissions may not violate the delay sensitivity of the video stream. This means that we can combine the FEC and retransmission techniques to more efficiently and effectively protect the packets from errors. In this manner we aim to minimize the distortion of VoD streaming without any problems. When we utilize the scheduling window concept, an issue arises. Packets have different possible numbers of retransmission depending on the time when the packets are scheduled, even if the deadlines of the packets are the same, so setting the maximum number of retransmission in advance for each packet may not be helpful. Therefore, we use the possible number of retransmission to set the FEC code rate. The analytical guidelines are shown in the following Propositions 1 - 5, Lemma 1, and Theorem 1. The proofs for these are given in the Appendix.

Proposition 1: Goodput of FEC only and FEC with retransmission is the same (an error-occurred packet is retransmitted until success, or R times max and the FEC code rate is fixed for each retransmission).

Good put is the number of useful information symbols delivered per unit time (i.e., the FEC redundancy and retransmitted symbols are excluded.).

Differently from Shannon’s second theorem, Proposition 1 considers the retransmission of the erroneous packets. PER decrease and bandwidth increase for retransmission are considered in Proposition 1. Therefore, we should carefully select which packets to retransmit and how to set their FEC code rate for each retransmission. Otherwise, it is not worthwhile to combine FEC and retransmission.

Let $X \sim B(n,p)$ be the distribution of the number of error-occurred symbols when a packet of n symbols is transmitted over a wireless channel with symbol error probability p and $Pr(X \geq n-k+1)$ as (n, k)-FEC encoded packet error probability, where k represents the information unit size in symbol.

Proposition 2: Let $f(n) = Pr(X \geq n-k+1)$, where $X \sim B(n,p)$, $n \geq k$, $0 \leq p \leq 1$, and k and p are fixed variables. Then, $f(n)$ is the function that maps packet size to packet error probability.

Proposition 3: As n is large enough, $f(n)$ can be approximated to $h(n)$, where

$$h(n) = 1 - \Phi\left(\sqrt{\frac{n(1-p)}{p}} - \frac{k-1}{\sqrt{np(1-p)}}\right) \tag{2}$$

Proposition 4: Let I be the inflection point of $h(n)$.I is the real-number solution of (3).

$$A^3n^3 + A^2(B-A+1)n^2 + AB(3-B-3A)n - B^3 = 0, \quad where\ A = 1-p,\ and\ B = k-1 \tag{3}$$

Parameter I is the inflection point of the approximated function that maps packet size to packet error probability. If the packet size is in the range $[k,I)$, then the packet error probability decreases in concave fashion. Otherwise, in the range (I,∞), it decreases in convex fashion.

Lemma 1: The following inequality holds for $k \le n_1, \ldots, n_t \le I$, $\bar{n} = \frac{n_1 + \cdots + n_t}{t}$, $0 \le p \le 1$, and k and p are fixed variables. Assume that $\bar{n}$ is an integer. It can easily be done by adjusting $n_1, \ldots, n_t$ variables.

$$f(n_1) \times \cdots \times f(n_t) \le \{f(\bar{n})\}^t \tag{4}$$

We check that (4) still holds for most n variables in the range (I,∞) by assigning several different variables to the inequality.

Proposition 5:The probability of success of a packet delivery decreases as the possible number of retransmission becomes lower. In other words, the lower the possible number of retransmission, the higher the probability of packet error.

Theorem 1:Even when the channel conditions are the same for t transmissions, changing the FEC code rate for each transmission yields lower probability of error than fixing the FEC code rate with the same overhead, if the packet size is less than I after FEC channel coding.

According to Theorem 1, changing the FEC code rate for each retransmission reduces packet error rate, even when the channel conditions do not change. For this reason, it is better to consider additional parameters to set the FEC code rate. We utilize the possible number of retransmission, which changes over time. We give more protection (i.e., a lower FEC code rate) to the packets with a lower possible number of retransmission considering the Proposition 5. The details are shown in Section 4.2.

4. THE PROPOSED FFEC-FSSU SCHEME

4.1. OVERVIEW

High quality VoD streaming service over wireless networks is a multi-variable optimization problem. We need to consider video distortion model with respect to flexible FEC that reflects channel status, possible number of retransmissions of error packets, dependencies among scalable units (SUs), scheduling time of the packets in the same SU, and many others. These various aspects in the distortion model make the problem very challenging to solve jointly. Moreover, the possible number of retransmissions for the packets with different FEC code rates and the competition for bandwidth between regular and retransmission-requested packets within the same scheduling window should be considered to enhance the distortion performance. However, these aspects make the problem complex to solve jointly. Existing joint solution approaches [3-6] do not consider the possible number of retransmissions for the FEC code rates. Also, the approaches do not reflect competition for bandwidth within the same scheduling interval. Thus, to develop an effective low complexity procedure that is applicable to the real-world VoD service we divide the problem into two sub-problems: 1) what is reasonable FEC code rate of each packet? 2) which packets to select in each scheduling window?

Note that these two sub-problems are coupled due to the bandwidth limitation. Thus, in the first step we relax the capacity limitation and solve the FEC code rate computation. Then, in the second step we solve the SU selection with capacity limitation considering the FEC code rates in the first step. In the two-step procedure two typical tradeoffs need to be examined related to the FEC code rate computation and SU selection.The tradeoff related to FEC code rate is that as we increase the amount of redundant symbols assigned to a packet, the error robustness increases. However, increased redundancy may delay transmissions of other packets. The tradeoff related to SU selection is whether to select the SUs that have higher impact on video distortion or those that have closer delay deadlines.

These tradeoffs are handled in the following two steps and specific procedures are provided in Section IV.B and IV.C. The first step involves a flexible computation of the FEC code rate. We need to decide how much redundancy to allow for each packet.Input variables are PER thresholds, the impact on video distortion, BER, and the possible number of retransmission. The output variables are the packet size, which is the sum of information symbols and FEC redundant symbols. The second step involves the fair selection of the SUs. In this step, we need to decide which SUs to transmit considering the capacity limit, delay constraints, and the H.264/SVC structure.

In this manner, we prevent two possible inefficiencies that could have arisen from selecting the SUs to transmit first and then, adding FEC redundant symbols to the SUs to minimize the video distortion within a limited capacity. These two possible inefficiencies are: 1) SUs may be overprotected, especially under good channel conditions. Redundant symbols are likely to be useless. 2) Some SUs may not be protected enough, especially under bad channel conditions due to the capacity limit, thus transmission errors may occur frequently. To manage these, we first decide how many redundant symbols to put into each SU; then we select the SUs to transmit. This guarantees an appropriate FEC code rate and maximizes the efficiency ofresource usage.

4.2. FLEXIBLE FEC CODE RATE COMPUTATION

$S_{q,f,t,g}$ is the packet size (information symbols + parity symbols) of $SU(q,f,t,g)$. $S_{q,f,t,g}$ is calculated by considering the impact on video distortion, BER, and the possible number of retransmission.

To consider the impact on video distortion, we adopt the concept of UEP (i.e., more protection to the important data). To do so we first set the PER threshold of $SU(q,f,t,g)$, $P_{th}(q,f,t,g)$, proportional to the impact of the $SU(q,f,t,g)$ on the video distortion. Then, we impose restrictions on the error probabilities of packets constituting the $SU(q,f,t,g)$ using $P_{th}(q,f,t,g)$. This implies that the greater the impact on the video distortion, the stronger the protection applied to the $SU(q,f,t,g)$ should be.

BER can be determined by ARQ feedback, by CQI (channel quality indicator) subchannel, or by both. It allows adaptive setting of the FEC code rates according to BS-MS channel conditions. The higher the BER is, the lower should be the FEC code rate assigned. As in [23], we assume the video streaming service receivers are likely to move at relatively slow speed and hence the channel conditions tend to change slowly. With 3 kmph user speed and 3 ms feedback delay, 15 ms lifetime of CQI values can be assumed to avoid erroneous prediction [24]. For video streaming services in LTE downlink, QoS for users can be maintained with at least 0.2 CQI report rate per transmission time interval [25]. Also, according to the latest 3GPP LTE technical specification [26], the reporting periodicity of LTE system can be 2, 5, 10 ms and so on. Hence, if the ARQ feedbacks and/or CQI reporting are carried by either PUCCH or PUSCH for at least every 5 ms, it will have minimal impact on the prediction of the channel conditions with the presence of the feedback delay.

The possible number of retransmission is determined as in (5) using the parameters in [27]:

$$r = \left\lfloor \frac{T_{remaining} - T_{forward}}{T_{forward} + T_{ACK/NACK}} \right\rfloor \tag{5}$$

$$T_{forward} = T_{fFEC-fSSU} + T_{S-enc/dec} + T_{C-enc/dec} + T_{net} \tag{6}$$

$T_{remaining}$ is the remaining time until decoding deadline expiration and $T_{forward}$ is the one-way end-to-end delay. If $T_{remaining} - T_{forward} < 0$, then the packet should be dropped due to the decoding deadline expiration. $T_{ACK/NACK}$ is ACK/NACK delay. $T_{forward}$ is sum of the algorithm computation time $T_{fFEC-fSSU}$, the source encoding and decoding delay $T_{S-enc/dec}$, the channel encoding and decoding delay $T_{C-enc/dec}$, and the network delay T_{net}.

Since SUs have their own decoding deadlines, the possible number of retransmission for SUs in the same scheduling window are different. The fewer the possible number of retransmissions, the lower the FEC code rate assigned.

From the above, we can determine $S_{q,f,t,g}$ with (7)–(13). Table I shows the notations and their descriptions used in the equations.

First, we define the probability of symbol error, p, by

$$p = 1 - (1 - b)^m \tag{7}$$

Second, we calculate the (n,k)-FEC encoded packet error probability as

Table I : Parameter Description

Notation	Description
n	Packet size in symbol. ($S_{q,f,t,g}$)
k	Information unit size in symbol
m	Symbol size in bit
r	Possible number of retransmissions
b	BER
$P_{FEC}(n,k,p)$	(n,k)-FEC encoded packet error probability
$P_{FEC/Ret}(r,n,k,p$	(n,k)-FEC encoded packet error probability with r possible number of retransmission

$$\begin{aligned} P_{FEC}(n,k,p) &= 1 - \Pr(X \le n-k),\ X \sim B(n,p) \\ &= 1 - \sum_{i=0}^{n-k} \binom{n}{i} p^i (1-p)^{(n-i)} \end{aligned} \tag{8}$$

We apply normal approximation to the binomial distribution using de Moivre-Laplace theorem to reduce the computational complexity of (8).

$$\begin{aligned} P_{FEC}(n,k,p) &= 1-\Pr(X \le n-k),\ X \sim B(n,p) \\ &\cong 1-\Pr(Z \le \frac{n-k+0.5-np}{\sqrt{npq}}),\ Z \sim N(0,1) \end{aligned} \tag{9}$$

Then, we can define the PER considering the possible number of retransmission as

$$P_{FEC/\mathrm{Ret}}(r,n,k,p) = \left\{P_{FEC}(n,k,p)\right\}^{(r+1)} \tag{10}$$

Finally, we determine the $S_{q,f,t,g}$ by finding the minimum n that satisfies (11) by (12) and (13).

$$P_{FEC/\mathrm{Ret}}(r,n,k,p) \le P_{th}(q,f,t,g) \tag{11}$$

$$S_{q,f,t,g} = \arg\min_{n \ge k} d(n) \tag{12}$$

Where

$$\begin{aligned} d(n) &= \begin{cases} P_{th}(\cdot) - P_{FEC/\mathrm{Ret}}(\cdot), & if\ \ P_{th}(\cdot) \ge P_{FEC/\mathrm{Ret}}(\cdot) \\ \infty & , otherwise \end{cases} \\ (P_{th}(\cdot) &= P_{th}(q,f,t,g), P_{FEC/\mathrm{Ret}}(\cdot) = P_{FEC/\mathrm{Ret}}(r,n,k,p)) \end{aligned} \tag{13}$$

4.3. Fair Selection Of The Scalable Units

4.3.1. Binary Integer Programming (BIP) Formulation

We first formulate the problem of fair selection of the scalable unit using BIP. Since an SU is the minimum decoding unit, and the whole packets constituting the SU should be transmitted successfully to decode the SU, it is hard to determine the impact on video distortion of each packet. For example, if only one out of ten packets is successfully transmitted, or if nine out of ten packets are successfully transmitted, they both lead to the same result, the decoding failure of the SU and thus, the same amount of distortion occurs. That's why we do not consider partial transmission in the formulation.

The following notations are used in the BIP formulation:

- C: Capacity in symbols.
- S_w: A set of SUs within scheduling window, W.
- $D_{q,f,t,g}$: Distortion caused by the loss of $SU(q,f,t,g)$.
- $N_{q,f,t,g}$: Number of packets remaining to successfully transmit $SU(q,f,t,g)$.
- $S_{q,f,t,g}$: Number of symbols required to transmit a packet constituting $SU(q,f,t,g)$.
- $x_{q,f,t,g}$: Binary variable that represents the selection of $SU(q,f,t,g)$.

C, S_w, $D_{q,f,t,g}$, $N_{q,f,t,g}$, $S_{q,f,t,g}$ are the input variables and $x_{q,f,t,g}$ is the decision variable.
The objective of (14) is to minimize the total distortion.

$$\min \sum\nolimits_{SU(q,f,t,g)\in S_w} \left(D_{q,f,t,g} \cdot (1-x_{q,f,t,g})\right) \tag{14}$$

Note that the objective function (14) selects SUs that minimize the distortion within the capacity limit. BIP thus does not allow partial transmission of packets of an SU in each scheduling interval.

The constraint set forth in (15) is the capacity limit constraint. The number of symbols required to transmit selected SUs cannot exceed the given capacity, *C*.

$$\sum_{SU(q,f,t,g)\in S_w}\left(S_{q,f,t,g}\cdot N_{q,f,t,g}\cdot x_{q,f,t,g}\right)\le C \tag{15}$$

The constraint set forth in (16) is the dependency constraint. We define the expression '*(q',f',t',q')<(q,t,f,g)*', which indicates that *SU(q,f,t,g)* references *SU(q',f',t',g')*.

$$x_{q,f,t,g}\le x_{q',f',t',g'},\forall(q',f',t',g')<(q,f,t,g),\forall(q,f,t,g) \tag{16}$$

Finally, the constraint set forth in (17) is the binary integer constraint (the variable x should be 0 or 1).

$$x_{q,f,t,g}\in\{0,1\} \tag{17}$$

Thus, the problem becomes

$$\begin{aligned}&\text{BIP1:}\\&\text{Objetive (14)}\\&\text{Subject to (15),(16),(17)}\end{aligned} \tag{18}$$

BIP1 is the 0-1 knapsack problem with the dependency constraints which is NP-hard [28]. Since a scheduling period is usually very short (i.e., few ms), we need to suggest a heuristic algorithm with low time complexity for practical usage. Besides the low complexity, we consider partial transmission of the SUs and the possibility of retransmission of SUs to enhance the performance of the algorithm.

4.3.2. The Proposed Fair Selection Algorithm

Since partial transmission is allowed, we have to decide not only which SUs, but also how many packets that belong to the SUs, to transmit within the scheduling interval. For the decision criteria, we propose the metric called the priority index.

The priority index is the metric that measures the efficiency of an SU. It is calculated by dividing the impact of the SU on video distortion, by the resources required to transmit the SU. We give higher priority (i.e., more scheduling opportunities) to those SUs which have high impacts on video distortion and that require small amounts of resource for transmission. Furthermore, error resilient techniques are considered in the priority index. Thus, the symbols required to transmit an SU differ for every scheduling interval due to allowances for partial transmission, transmission errors, retransmission, and adaptive FEC. Equation (19) shows the priority index.

$$\begin{aligned}PI_{q,f,t,g}&=\frac{\text{Distortion caused by the loss of SU(q,f,t,g)}}{\text{Required symbols to transmit SU(q,f,t,g)}}\\&=\frac{D_{q,f,t,g}}{N_{q,f,t,g}\times S_{q,f,t,g}}\end{aligned} \tag{19}$$

Where *PIq,f,t,g*, *Dq,f,t,g*, *Nq,f,t,g*, and *Sq,f,t,g* denote the priority index of *SU(q,f,t,g)*, distortion caused by the loss of *SU(q,f,t,g)*, the number of remaining packets to successfully transmit *SU(q,f,t,g)* and the number of required symbols to transmit a packet constituting *SU(q,f,t,g)*(i.e., packet size in symbols), respectively.

To calculate the priority index, we should determine $D_{q,f,t,g}$, $N_{q,f,t,g}$, and $S_{q,f,t,g}$. $D_{q,f,t,g}$ is determined by (1), $S_{q,f,t,g}$ is obtained from the FEC code rate computation and $N_{q,f,t,g}$ can easily be determined using ARQ feedback (i.e., ACK or NACK) information and the sequence number from the RTP/NAL headers [2].

The priority index matches well with fair selection as it does not consider whether the packet is regular or retransmission-requested. Instead, it considers the number of remaining packets needed to successfully transmit an SU, which depends on the original size of the SU, transmission errors, and retransmissions. It helps to select the SUs with higher efficiencies.

To minimize the possibility of unnecessary packet transmissions, we consider the possibility of retransmission and H.264/SVC structure. We adopt the Rule 1 to accommodate the H.264/SVC structure.

Rule 1. To allocate resources to *SU(q,f,t,g)*, all the SUs that *SU(q,f,t,g)* references, should have been successfully transmitted or have been allocated resources in the same scheduling interval.

The pseudo-code of the proposed fair selection algorithm is represented in Table II. The input variables are k, C, S_w, $D_{q,f,t,g}$, $N_{q,f,t,g}$, $S_{q,f,t,g}$, and $r_{q,f,t,g}$, and the output variables are the number of packets to transmit from *SU(q,f,t,g)*, $x_{q,f,t,g}$. First, we calculate the priority indices of SUs within the scheduling window (Lines 1-3).

Then, the remaining parts of the algorithm (Lines 5-25) iterate until the scheduling window set is empty, or the capacity has been exhausted. We extract *SU(q,f,t,g)*, which has the highest priority index among the SUs, satisfying Rule 1 (Line 5). If the capacity, C, is greater than the required symbols to transmit the *SU(q,f,t,g)*, $N_{q,f,t,g} \times S_{q,f,t,g}$, then select the *SU(q,f,t,g)* and delete the *SU(q,f,t,g)* from the S_w (Lines 6-9).

If C is lower than $N_{q,f,t,g} \times S_{q,f,t,g}$, then check the possible number of retransmission of the *SU(q,f,t,g)*. If the retransmission is possible, then select as manypackets as possible from the *SU(q,f,t,g)*, since the remaining packets of the *SU(q,f,t,g)* can be selected in the next scheduling period. If the retransmission is not possible, then discard the *SU(q,f,t,g)* and all the SUs that reference it from S_w (Lines 10-20). If the capacity has been exhausted, then terminate the algorithm (Lines 22-24).

Table II : Pseudo-Code Of The Proposed Fair Selection Algorithm

Input:	k, C, S_w, $D_{q,f,t,g}$, $N_{q,f,t,g}$, $S_{q,f,t,g}$, $r_{q,f,t,g}$
Outut:	$x_{q,f,t,g}$

```
for ∀SU ∈ S_w do
      PI_{q,f,t,g} ← D_{q,f,t,g} /(N_{q,f,t,g} × S_{q,f,t,g})
end for
while S_w ≠ ∅ do
    SU_{q,f,t,g} ← extract the highest priority index satisfying Rule 1 from S_w
      if C − N_{q,f,t,g} × S_{q,f,t,g} ≥ 0 do
                  x_{q,f,t,g} ← N_{q,f,t,g}
                  C ← C − N_{q,f,t,g} × S_{q,f,t,g}
                  S_w ← S_w − {SU_{q,f,t,g}}
      else if C − N_{q,f,t,g} × S_{q,f,t,g} < 0 do
            if r_{q,f,t,g} > 0 do
                  x_{q,f,t,g} ← ⌈C/S_{q,f,t,g}⌉
                  C ← C − N_{q,f,t,g} × S_{q,f,t,g}
                  S_w ← S_w − {SU_{q,f,t,g}}
                  if C < k do
                        break
                  end if
            else if r_{q,f,t,g} = 0 do
            discard SU_{q,f,t,g} and all the SUs that reference it from S_w
            end if
      end if
      if C < k do
            break
      end if
      S_w ← S_w − {SU_{q,f,t,g}}
end while
```

Table III: Simulation Parameters

Parameter	Value
Symbol size in bit, m	8
Information unit size in symbol, k	100
The number of GoPs in an IDR	3
The number of temporal layers	4
The number of quality layers	3
Scheduling window size, W	3
Scheduling interval (ms)	160

5. SIMULATION RESULTS

In this section, we present the results from a simulation of the proposed fFEC-fSSU scheme. Table III shows the simulation parameter settings. We use Foreman video sequence settings for the simulation.

The values of C_1, C_2, and C_3 in (1) are set as 0.35, 3.9, and 1.37, respectively [8]. The scheduling window size and the capacity are set to three and 10,240 symbols respectively. The PER threshold which is inversely proportional to the impact of distortion is set from 0.01 to 0.0001 unless otherwise mentioned.

To investigate the performance of the proposed fFEC-fSSU scheme, three different algorithms are used.

1) BIP: The FEC code rate computation works as in the proposed scheme. 'BIP' implements the optimal solution obtained by solving the BIP1 instead of implementing the proposed fair selection algorithm.
2) LHC (local hill-climbing): This is the variation of the algorithm proposed in [7]. From this work, the FEC redundant symbols are allocated to the SUs using LHC-based scheme. SU selection process worksin the same manner as the fFEC-fSSU scheme.
3) DHC (double hill-climbing): This is the joint source and channel coding scheme proposed in [6]. DHC is applied for FEC code rate assignment and SU selection process. For both LHC and DHC, the selection process is done by considering the regular and the retransmission-requested packets as in the proposed method.
4) DRL (deadline-based retry limit): This is used in [13, 14] instead of the proposed fair selection algorithm. DRL first selects all the packets among the retransmission-requested ones, if their deadlines are not over and retry limits are not reached; then allocates resources to them. After that, among the regular packets, the packets for transmission are selected according to their impacts on video distortion within the remaining resources. The other parts work in the same manner as the fFEC-fSSU scheme.

The performance of the proposed scheme and the four different algorithms are evaluated in terms of the total distortion, defined as the sum of the video quality metric, $D_{q,f,t,g}$, in (1), from the beginning to the end of the Foreman video sequence. Average results of 10 experiments are presented in most simulation except Fig. 4 and 6 where channel condition is given with typical BER.

Fig.3(a) and 3(b) show the total distortion with various capacities and BERs, respectively. In Fig.3(a), the BER randomly changes from 0.00 to 0.05 for each scheduling interval, while the capacity is fixed at 2048, 4096, 6144, 8192, and 10,240 symbols. In Fig.3(b), the capacity is fixed at 10,240 symbols with BER 0, 0.01, 0.02, 0.03, 0.04 and 0.05. In all cases, the proposed scheme performs better than the others. In Fig. 3(a) the average improvements of total distortion by the proposed procedure are 25%, 28%, and 31% respectively compared to the LHC, DHC, and DRL. The improvements in Fig. 3(b)are 37%, 38%, and 40% respectively. Note that the performance of 'BIP' is worse than the proposed scheme. This is mainly due to the partial transmission adopted in the fair selection algorithm of the proposed method as stated in Section IV.C.

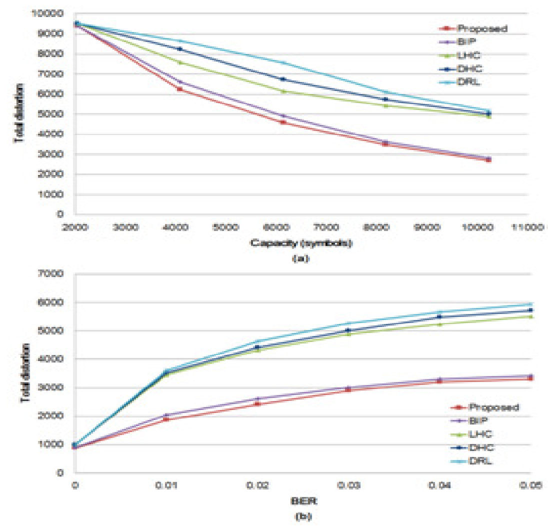

Fig. 3.Performance comparison

In Fig. 4, we compare the proposed algorithm with DHC. We trace thedistortion of each GoP in the video sequence. The BER is changing as in the figure. It is clear that the proposed algorithm shows lower distortion than DHC throughout the video sequence. The proposed algorithm demonstrates 35% less distortion in average. The partial transmission of SUs selected by the proposed procedure allows transmitting part of SUs in the selected scheduling interval. The remaining packets are transmitted in the following scheduling interval. Therefore, big SUs are free from the disadvantage in obtaining transmission opportunities. Moreover, the proposed algorithm reflects the possible number of retransmissions in the computation of FEC code rates.

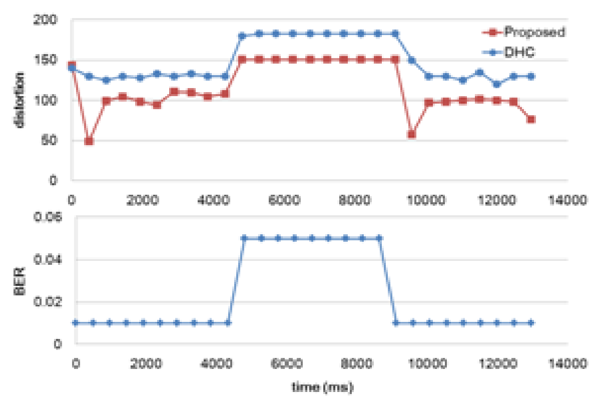

Fig. 4.Performance comparison with deep channel fading

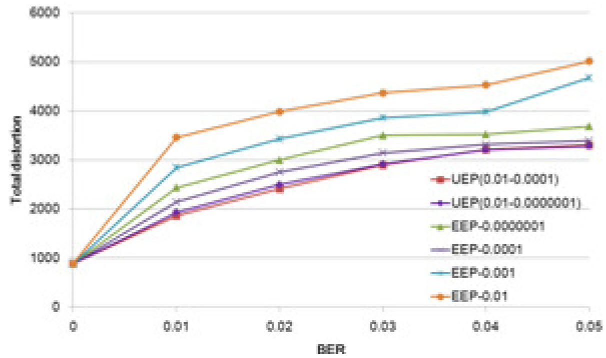

Fig. 5.The effect of PER threshold

Fig.5 shows the effect of the UEP adopted in the proposed scheme by comparing it with the different equal error protection (EEP) at various BERs. We set the PER threshold, $P_{th}(q,f,t,g)$, of UEP(0.01-0.0001) and UEP(0.01-0.000001) inversely proportional to the impact on distortion, from 0.01 to 0.0001 and from 0.01 to 0.000001, respectively. The greater the SU impacts on distortion, the lower the UEP threshold is set. Whereas, for the EEPs in Fig.5, we set the same PER threshold for all SUs. The thresholds for EEP-0.000001, EEP-0.0001, EEP-0.001, and EEP-0.01 schemes are set to 0.000001, 0.0001, 0.001, and 0.01, respectively. The other parts work the same as the proposed scheme. From this figure, we can also clearly see that the proposed scheme, with UEP, performs better than the ones with EEP in all cases. In addition, by comparing EEP-10^{-7} and EEP-10^{-4}, we see that EEP-10^{-7} performs worse than EEP-10^{-4}. When the PER threshold is too small, the bandwidth is wasted due to too much redundancy, and as a result, the performance gets worse.Notice that with PER threshold of 10^{-7}, the amount of FEC redundancy per packet increases up to 25% compared to the scheme with PER threshold of 10^{-4} using (13).

In Fig.6, we can judge the effect of the scheduling window size on distortion when deep channel fading occurs. We trace the distortion of each GoP in the video sequence. The BER settings are the same as in Fig. 4. The simulation is performed with scheduling window sizes 1, 3, and 5. We can see that the distortion increment, as well as the distortion itself, tends to be smaller as the scheduling window size gets bigger when deep channel fading occurs. The reason for this is that the proposed scheme preferentially allocates resources to the temporal and quality base layers of the GoPs in the scheduling window, which have the greatest impact on video distortion. The bigger the scheduling window, the more base layers are included. Since more base layers are transmitted, even with deep channel fading, the proposed scheme with bigger scheduling windows has a better chance of successfully transmitting the base layers than it would with smaller scheduling windows. That's why the distortion of the proposed scheme with bigger scheduling windows increases more smoothly than it does with smaller scheduling windows. Additionally from the figure, we can see that when the channel is in good state and steady, the distortion of larger scheduling windows is a little bit greater than the smaller ones. This is because most of the resources are used to transmit the base layers, thus suppressing the transmission of enhancement layers. Recall that the capacity limit exists in each scheduling interval. From the above result, we can observe that when the channel is in good and steady condition, smaller scheduling windows may perform well. However, when deep channel fading occurs frequently, the larger scheduling windows are likely to perform better.

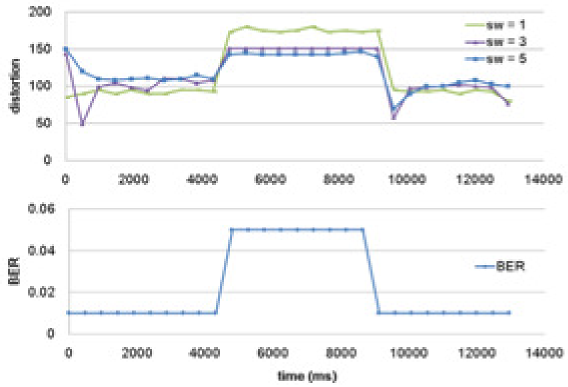

Fig. 6.The effect of scheduling window size with deep channel fading

5. CONCLUSION

In this paper, we consider a flexible FEC and fair selection of scalable units (fFEC-fSSU) that exploits H.264/SVC structure and three error resilient techniques – unequal error protection (UEP), forward error correction (FEC), and retransmission, to efficiently tackle the distortion problem. We solve the problem into two steps: flexible FEC code rate computation and fair selection of the SUs. Different FEC code rates are applied flexibly among the SUs considering their impact on video distortion and the two time-varying factors: wireless channel condition and the possible number of retransmission. This mechanism is based on the result of thetheoretical study to minimize the probability of packet errors. Then, in each scheduling period, those SUs to transmit are fairly selected considering the proposed priority index, as well as the delay deadline and H.264/SVC structure. Using this approach, we are able to prevent excessive retransmissions that may interrupt regular packets with substantial impacts on video distortion.

From the simulation results we conclude that it is better to set unequal PER thresholds according to the impact on video distortion of SUs. This can effectively protect the SUs with higher impact on video distortion within limited capacity. Moreover, we can cope with the various wireless channel conditions by utilizing the concept of scheduling windows. When deep channel fading occurs frequently, it is preferable to increase the size of the scheduling windows. By doing so, we increase the chances of successfully transmitting the base layers which greatly impact video distortion. We also observe that allocating the dedicated resources for retransmission or setting the maximum retry limit in advance, degrades the performance of the algorithm in the case of H.264/SVC VoD streaming.

Appendix

A. Proof of Proposition 1

Let $Goodput_{FEC}$ be the goodput of FEC only, then

$$Goodput_{FEC} = \left\lfloor \frac{C}{n \times m} \right\rfloor \times k \times m \times (1 - P_{FEC}) = \left\lfloor \frac{C}{n \times m} \right\rfloor \times k \times m \times \{1 - \Pr(X \geq n-k+1)\}$$

Also, let $Goodput_{FEC/Ret}$ be the goodput of FEC with retransmission, then

$$Goodput_{FEC/Ret} = \left\lfloor \frac{C}{n \times m} \right\rfloor \times k \times m \times \frac{1}{r_{FEC/Ret}} \times (1 - P_{FEC/Ret}) = \left\lfloor \frac{C}{n \times m} \right\rfloor \times k \times m \times \{1 - \Pr(X \geq n-k+1)\}$$

because, a) if a packet is retransmitted until success,

$$\overline{r_{FEC/Ret}} = \frac{1}{1-\Pr(X \ge n-k+1)} \text{ and } P_{FEC/Ret} = 0$$

b) if a packet is retransmitted at most R times,

$$\overline{r_{FEC/Ret}} = \frac{1-\{\Pr(X \ge n-k+1)\}^{R+1}}{1-\Pr(X \ge n-k+1)} \text{ and } P_{FEC/Ret} = 1-\{\Pr(X \ge n-k+1)\}^{R+1}$$

Therefore, $Goodput_{FEC} = Goodput_{FEC/Ret}$.

B. Proof of Proposition 2

If $X\sim B(n,p)$, $n\geq k$, $0\leq p\leq 1$, and k and p are fixed, then $\Pr(X \ge n-k+1) = \sum_{i=n-k+1}^{n} \binom{n}{i} p^i (1-p)^{n-i}$, which is dependent only on the variable n. Thus, the probabilitycan be written as*f(n)* which satisfies the condition of the function with domain $[k,\infty]$, and codomain (0,1). As n is the packet size and $Pr(X\geq n-k+1)$ is the packet error probability, *f(n)* is the function that maps the packet size to the packet error probability.

C. Proof of Proposition 3

As $n$becomes large enough, it is reasonable to approximate binomial distribution to normal distribution.

$$\Pr\left(\mathrm{X} \ge \mathrm{n}-\mathrm{k}+1\right) = \Pr\left(\frac{X-np}{\sqrt{np(1-p)}} \ge \frac{n-k+1-np}{\sqrt{np(1-p)}}\right) = 1-\Phi\left(\frac{\sqrt{n(1-p)}}{\sqrt{p}} - \frac{k-1}{\sqrt{np(1-p)}}\right)$$

D. Proof of Proposition 4

Since the function $h(n)$ is twice differentiable at $n = I$ and $h''(n)$ changes sign at $n = I$, the point (I, $f(I)$) is an inflection point of the graph of $h(n)$. Also, $h(n)$ is concave in the range $[k,I)$ and convex in the range (I,∞), because $h''(n) < 0$ for $[k,I)$ and $h''(n) > 0$ for (I,∞). Let

$$F(n) = \int_{-\infty}^{n} f(t)\,dt,\ where\ f(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} \text{ and } Q(n) = \frac{\sqrt{n(1-p)}}{\sqrt{p}} - \frac{k-1}{\sqrt{np(1-p)}}$$

then we have

$$h(n) = 1-F\left(Q(n)\right),\ h'(n) = -f\left(Q(n)\right)\times Q'(n)$$

And $h''(n) = -\left\{f'\left(Q(n)\right)\times Q'(n)\times Q'(n) + f\left(Q(n)\right)\times Q''(n)\right\}$

$$h''(n) = \frac{1}{\sqrt{2\pi}} e^{-\frac{Q(n)^2}{2}} \times \frac{A^3n^3 + A^2(B-A+1)n^2 + AB(3-B-3A)n - B^3}{4\sqrt{n^7A^3(1-A)^3}}$$

$$where\ A = 1-p, and\ B = k-1$$

Therefore, n that satisfies $h''(n) = 0$ is the real number solution of the following equation:

$$A^3n^3 + A^2(B-A+1)n^2 + AB(3-B-3A)n - B^3 = 0$$

$$where\ A = 1-p,\ and\ B = k-1$$

E. Proof of Lemma 1

From Proposition 2, *f(n)* can be approximated to *h(n)*. Hence, we compare $\prod_{i=1}^{t} h(n_i)$ and $\left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}^t$ instead of $\prod_{i=1}^{t} f(n_i)$ and $\left\{f\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}^t$. The logarithm function is monotonically increasing and concave, if we assume the base is higher than one. Thus, we need to compare $\log\left\{\prod_{i=1}^{t} h(n_i)\right\}$ and $\log\left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}^t$.

Since $\log\left\{\prod_{i=1}^{t} h(n_i)\right\} = \sum_{i=1}^{t} \log(h(n_i))$ and $\log\left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}^t = t \times \log\left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}$

from the Jensen's inequality we have $\frac{\sum_{i=1}^{t} \log\{h(n_i)\}}{t} \le \log\left\{\frac{\sum_{i=1}^{t} h(n_i)}{t}\right\}$

Now, *h(n)* is concave over the range $k \le n \le I$. Again, from the Jensen's inequality,

$\frac{\sum_{i=1}^{t} h(n_i)}{t} \le h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)$ and $\sum_{i=1}^{t} \log\{h(n_i)\} \le \mathrm{t} \times \log\left\{\frac{\sum_{i=1}^{t} h(n_i)}{t}\right\} \le \mathrm{t} \times \log\left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}$

We thus have $\prod_{i=1}^{t} h(n_i) \le \left\{h\left(\frac{\sum_{i=1}^{t} n_i}{t}\right)\right\}^t$

F. Proof of Theorem 1

The FEC code rate varies as the packet size, *n*, changes because the size of information part is fixed to *k*. Thus, the left and right-hand side of the inequality (4) in Lemma 1 can respectivelybe interpreted as changing and fixing the FEC code rate for each transmission. The overhead due to the FEC redundancy for both sides of (4) is equal,because $n_1 + \cdots + n_t = \bar{n} \times t$. Therefore, Theorem 1 is valid from Lemma 1.

G. Proof of Proposition 5

Let *r*be the possible number of retransmission, $P_{suc}(r)$ be the packet delivery success probability, $P_{err}(t)$ be the packet error probability at time period t, and *α*be the scheduling interval, then

$$P_{suc}(r) = (1 - p_{err}(t)) + p_{err}(t) \cdot (1 - p_{err}(t+\alpha)) + \cdots + p_{err}(t) \cdots p_{err}(t + (n-1) \cdot \alpha) \cdot (1 - p_{err}(t + r \cdot \alpha))$$

Since $0 \le P_{err}(t) \le 1$, all the expressions in the right-hand side of the equation (40)are positive. Thus, as *r* increases, $P_{suc}(r)$ increases. Therefore,

$$P_{suc}(r) \ge P_{suc}(r'), \forall r \ge r'$$

REFERENCES

[1] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard,” IEEE Trans. Circuits Syst. Video Technology, vol. 17, no. 9, pp. 1103-1120, 2007.

[2] P. Amon, T. Rathgen, and D. Singer, “File Format for Scalable Video Coding,” IEEE Trans. Circuits Syst. Video Technology, vol. 17, no. 9, pp. 1174-1185, 2007.

[3] F. Zhai, Y. Eisenberg, T. Pappas, R. Berry, and A. Katsaggelos, “Rate-distortion optimized hybrid error control for real-time packetized video transmission,” IEEE Trans. Image Processing, vol. 15, no. 1, pp. 40–53, Jan. 2006.

[4] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao, “Joint source-channel rate-distortion optimization for H.264 video coding over errorprone networks,” IEEE Trans. Multimedia, vol. 9, no. 3, pp. 445-454, 2007.

[5] M. Stoufs, A. Munteanu, J. Cornelis, and P. Schelkens, “Scalable joint source-channel coding for the scalable extension of H.264/AVC,” IEEE Trans. Circuits Syst. Video Technology, vol. 18, no. 12, pp. 1657-1670. 2008.

[6] C. Chen, Y. Zhang, Y. Fu, and Z. Yang, "A new joint source and channel coding scheme for packet-based scalable multimedia streams", in Proc. IEEE GLOBECOM Workshops, pp. 954-959, 2010.

[7] H. Ha, and C. Yim, “Layer-weighted unequal error protection for scalable video coding extension of H.264/AVC,” IEEE Trans. Consumer Electronics, vol. 54, no. 2, pp. 736-744, 2008.

[8] C. Chi, Y. Zhang, and L. Wang, “Joint Power Control and FEC Unequal Error Protection for Scalable H.264 Video Transmission over Wireless Fading Channels,”in Proc.Global Telecommunications Conference, pp. 1-6, 2009.

[9] Y. Wang, T. Fang, L. P. Chau, and K. H. Yap, “Two-Dimensional Channel Coding Scheme for MCTF-Based Scalable Video Coding,” IEEE Trans. Multimedia, vol. 9, no. 1, pp. 37-45, 2007.

[10] C. Hellge, D. Gomez-Barquero, T. Schierl, and T. Wiegand, “Layer-Aware Forward Error Correction for Mobile Broadcast of Layered Media,” IEEE Trans. Multimedia, vol. 13, no. 3, pp. 551-562, 2011.

[11] E. Maani, and A. K. Katsaggelos, “Unequal Error Protection for Robust Streaming of Scalable Video Over Packet Lossy Networks,” IEEE Trans. Circuits Syst. Video Technology, vol. 20, no. 3, pp. 407-416, 2010.

[12] X. Zhu, R. Pan, N. Dukkipati, V. Subramanian, and F. Bonomi, “Layered Internet Video Engineering (LIVE): Network-Assisted Bandwidth Sharing and Transient Loss Protection for Scalable Video Streaming,” in Proc. IEEE Infocom, pp. 1-5, Mar. 2010.

[13] R. Razavi, M. Fleury, and M. Ghanbari, “Buffer size resilience for Bluetooth video streams from adaptive ARQ with active discard,” Electronics Letters, vol. 43, no. 20, pp. 1075-1077, 2007.

[14] Z. Ma, J..Zhou, and T. Wu, “A cross-layer QoS scheme for MPEG-4 streams,” in Proc. IEEE WCNIS, pp. 392-396, June 2010.

[15] Q. Li, and M. van der Schaar, “Providing adaptive QoS to layered video over wireless local area networks through real-time retry limit adaptation,” IEEE Trans. Multimedia, vol. 6, no. 2, pp. 278-290, 2004.

[16] P. Bucciol, E. Masala, and J. C. De Martin, “Perceptual ARQ for H.264 video streaming over 3G wireless networks,” in Proc. IEEE ICC, vol. 3, pp. 1288-1292, June 2004.

[17] R. Chakravorty, S. Banerjee, and S. Ganguly, “MobiStream: Error-Resilient Video Streaming in Wireless WANs Using Virtual Channels,” in Proc. IEEE Infocom, pp. 1-14, Apr. 2006.

[18] X. Qin, Z. Jiang, D. Sui, X. Zhang, and D. Yang, “Reliable Transmission of H.264 Video over

Wireless Network," in Proc. IEEE VTC, pp. 2789-2793, May 2008.

[19] A. A. Khalek, C. Caramanis, and R. W. Heath, "A Cross-Layer Design for Perceptual Optimization Of H.264/SVC with Unequal Error Protection," IEEE Journal on Selected Areas in Communications, vol. 30, no. 7, pp. 1157-1171, 2012.

[20] D. Wu, Y. T. Hou, B. Li, W. Zhu, Y.-Q. Zhang, and H. J. Chao, "An end-to-end approach for optimal mode selection in Internet video communication: theory and application," IEEE Journal on Selected Areas in Communications, vol. 18, no. 6, pp. 977-995, 2000.

[21] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao, "Joint Source-Channel Rate-Distortion Optimization for H.264 Video Coding Over Error-Prone Networks," IEEE Trans. Multimedia, vol. 9, no. 3, pp. 445-454, 2007.

[22] A. Shokrollahi, "Raptor codes," IEEE Trans. Information Theory, vol. 52, no. 6, pp. 2551-2567, 2006.

[23] G. Cheung and A. Zakhor, "Bit allocation for joint source/channel coding of scalable video," IEEE Trans. Image Processing, vol. 9, no. 3, pp. 340-356, 2000.

[24]G. D. Gonzalez, M. Garcia-Lozano, S. R. Boque, and J. Olmos, "Improving channel state information feedback for static intercell interference coordination in LTE," in Proc. IEEE ICC, pp. 4893-4898, 2012.

[25] R. Basukala, H. A. M. Ramli, K. Sandrasegaran, and L. Chen, "Impact of CQI feedback rate/delay on scheduling video streaming services in LTE downlink," in Proc. IEEE ICCT, pp. 1349-1352, 2010.

[26] 3GPP TS 36.213 V11.3.0 technical specification, "Physical layer procedures (Release 11)".

[27] A. Argyriou, "Cross-Layer Error Control for Multimedia Streaming in Wireless/Wireline Packet Networks," IEEE Trans. Multimedia, vol. 10, no. 6, pp. 1121-1127, 2008.

[28] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Company, 1979.