Hierarchical Optimization of Cascading Error Protection Scheme for H.264 Scalable Video Streaming

Wei-Chung Wen · Hsu-Feng Hsiao

Abstract Forward error correction (FEC) coding has been shown to offer a feasible solution to fulfill the need for Quality of Service for multimedia streaming over the fluctuant channels, especially in terms of the reduction of end-to-end delay. In this paper, we propose the Dynamic FEC-Distortion Optimization Algorithm to efficiently utilize the network bandwidth for better visual quality by means of hierarchical coding structure with the cascading error protection scheme. The optimization criteria are based on the unequal error protection by taking account of the error drifting problems from both temporal motion compensation and inter-layer prediction of the H.264/MPEG-4 AVC scalable video coding so that the priorities of each video components can be differentiated for the calculation of the distribution of parity packets. It is shown that the cascading error protection scheme makes the hierarchical structure of error erasure code more efficient. Also, the proposed algorithm works particularly well for fast motion videos and the performance does not depend on accurate estimation of packet loss rate.

Keywords Unequal error protection · Scalable video coding · FEC-distortion optimization · Cascading error protection

W.-C. Wen · H.-F. Hsiao (✉)
Department of Computer Science, Nation Chiao Tung University, Hsinchu, Taiwan
e-mail: hillhsiao@cs.nctu.edu.tw

W.-C. Wen
e-mail: wcwen@cs.nctu.edu.tw

1 Introduction

Personal, home, and mobile entertainment systems, such as DVB-H [1] and IPTV [2] which is being standardized by ITU-T, have been an emerging research and as well as an industrial emphasis due to the rapid progress of the network and multimedia signal processing technologies. However, it remains rather challenging for such entertainment systems to fulfill the needs for Quality of Service and Quality of Experience requirements in the mobile environments that might suffer from time-dependent channel fluctuation.

Besides the Automatic Repeat reQuest (ARQ) mechanism [3] which may possibly suffer from the intolerable end-to-end packet delay and exacerbated jitter, forward error correction codes have been shown to be a feasible solution for delay-sensitive multimedia applications. In DVB-H, Multi-Protocol Encapsulated Forward Error Correction (MPE-FEC) is implemented by interleaving the information packets and the protection packets resulting from Reed-Solomon error erasure code to deal with the burst error. The error protection strength in MPE-FEC is not really content-dependent though. On the other hand, rateless erasure codes (also known as fountain codes [4]), such as the raptor code [5, 6], can provide virtually infinite protection symbols. A modified version of such codes has been recently adopted in 3GPP [7]. However, unlike the Reed-Solomon error erasure code which shows the property of maximum distance separable, fountain codes generally have less coding efficiency.

In [8], Tan et al. proposed layered FEC for sub-band coded scalable video multicast in cooperation with the equation-based rate control algorithm. Adaptive FEC is adopted to recover lost packets so that the distortion function of video quality can be minimized by the optimized subscription of video and FEC layers, under the

assumption that different frames in a video layer shall have the same distortion.

In [9], an adaptive FEC scheme as a part of the reliable layered multimedia streaming over either unicast or multi-cast was proposed. The main objective of the FEC scheme is to maximize the streaming throughput while maintaining an upper bound of the error rate for each scalable video layer that FEC fails to decode. However, the upper bounds are pre-set manually according to the streaming applications.

The impact of packet loss and FEC overhead on a scalable bit-plane coded video in best-effort networks was analyzed in [10] and a similar optimization algorithm was proposed to allocate the bandwidth resource to FEC and video data.

In this paper, we propose a dynamic FEC-Distortion optimization algorithm that takes account of the error drifting problems resulting from both temporal motion compensation and inter-layer prediction of the H.264/ MPEG-4 AVC scalable video coding [11]. The content-dependent visual quality contribution of each video frame in a video layer is analyzed to achieve better quality of service at the same network resource. The proposed algorithm is based on the hierarchical coding structure with the cascading error protection scheme.

The rest of this paper is organized as follows. In Section 2 we present the related work. In Section 3 we describe the modification of the adaptive FEC protection scheme in the literature and also propose the dynamic FEC-distortion optimization algorithm to work with the H.264 scalable video coding, followed by Section 4, the simulation results and discussions. The concluding remarks are presented in Section 5.

2 Related Work

For one-to-many multimedia communications, scalable video coding is regarded as one of the promising coding schemes to deal with the time-dependent bandwidth fluctuation among heterogeneous receivers. H.264/MPEG-4 AVC is a video coding standard developed jointly by ITU-T and ISO. The MPEG-4 AVC scalable extension [11] is an amendment to the AVC and it is the state of the art of scalable video coding up to date.

The base layer of a H.264 SVC video stream can be configured to be MPEG-4 AVC compatible, and the enhancement layers of the stream are encoded by the scalability tools such as SNR, spatial, and temporal scalability. At the decoder side, in order to decode a video frame of the higher layer, all the lower layers of the coded stream of the same video frame are required. The more the accumulated video layers are received, the better the decoded video quality will be. Even though the reference

pictures can be arbitrarily assigned, the temporal scalability can be efficiently provided by the hierarchical-B motion compensation [11] as illustrated in Fig. 1. In the figure, a group of pictures (GOP) of 8 video frames is shown. The index in a picture symbol refers to its relative display order.

The spatial scalability is offered by encoding the video at different resolutions for different layers. During the inter layer prediction, inter-layer motion/residual prediction and intra prediction are utilized to facilitate the compression. With quality scalability, the substream provides the same spatio–temporal resolution as the complete bit stream, but with a lower video quality [11].

Scalable video coding in cooperation with congestion control algorithms offers scalability over bandwidth, among other scalabilities, to facilitate streaming over heteroge-neous networks. However, to overcome the impaired heterogeneous channels, proper link adaptation based on source or channel coding may become necessary. For the multimedia streaming services, there are some major metrics to evaluate a streaming system, including the limitation of end-to-end delay, jitter, packet loss, and system scalability. According to the ITU-T recommenda-tions [12], for voice conversation applications, one-way delay (OWD) of 0–150 ms indicates good interactivity, while OWD of more than 400 ms is intolerable. Jitter, defined as an estimate of the statistical variance of the packet inter-arrival time, of less than 50 ms is required, which is similar to a 2% VoIP packet loss. For packet loss requirements on VoIP applications, generally 1~2% is acceptable, which is roughly equivalent to 30 ms of speech loss out of 2 s duration. The forward error correction (FEC) codes [13, 14] or the acknowledgement (ACK) protocols such as ARQ, or Hybrid ARQ, are often used to protect the content of multimedia streaming. The end-to-end ARQ usually results in intolerable delay and it may also exacerbate jitters. Instead of relying solely on the ACK and retransmission mechanism as in ARQ, FEC inserts redundancy at transmitter side to protect occasional packet loss at the cost of shrinking throughput. Hybrid ARQ generally tries to find a balance between delay and throughput. More specifically, a cycle-based adaptive rate control algorithm was proposed to avoid the starvation of
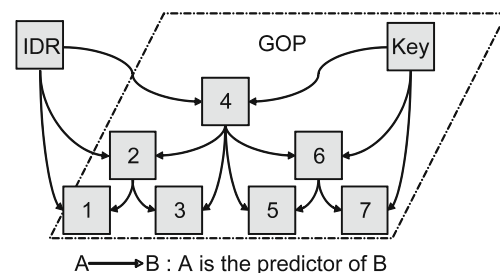
Fig. 1 Hierarchical prediction structure.

the receiver buffer with Hybrid ARQ scheme [15]. Also, based on channel estimation and rate distortion optimization, another adaptive FEC protected scalable audio streaming scheme with hybrid ARQ was proposed for wireless channels [16].

The property of semi-reliable data transfer of FEC without the need for retransmission makes an appeal to the delay-sensitive applications such as the applications of live streaming and video conferencing. Popular FEC codes include the maximum distance separable (MDS) codes [17, 18] within the Reed Solomon code family and the rateless codes such as Raptor code [5]. For a systematic error erasure code, the term (n, k) is referred to the input of k message symbols and output of n coded symbols, including the original k message symbols.

In terms of the packetization of the FEC and SVC streams, a common way is to form blocks of packets (BOP) [19] where the encoded video streams are packed in horizontal packets and the systematic FEC code is applied across the video packets as shown in Fig. 2 to produce parity symbols. The parity symbols from different FEC coding are grouped into parity packets. Typically, the size of a symbol is one byte. Usually, those video packets and parity packets are placed upon RTP/UDP/IP protocol stack before transmission.

In [8], Tan et al. proposed a layered FEC algorithm for sub-band coded scalable video multicast. The algorithm adopts the equation-based rate control in the literature such that packet loss is one of the parameters to regulate the sending rate. The adaptive amount out of n symbols of the FEC coding is determined and transmitted to recover the lost packets so that the distortion can be reduced with the optimized subscription. A subscription is a vector which records the number of protected scalable video layers to be transmitted and the corresponding parity packets are grouped and partitioned into a number of FEC layers. However, the sub-band scalable video coding considered in [8] is assumed to be able to produce a number of components with equal importance. Most critically, those components shall have the ability to be decoded indepen-

dently. However, the inter-layer predictive coding in the scalable video coding is one of the most important tools to enhance the coding efficiency. It requires that both encoder and decoder shall have the same reconstruction of the reference pictures. Thus, when some parts of the video layer are lost and cannot be recovered by FEC decoding, the caused distortion will propagate to the other video scalable layers as to the other video frames, since the coding tool of predictive coding along the temporal prediction direction is also used frequently.

In [10], the authors investigated the effect of packet loss on the video quality of MPEG-4 Fine Granularity Scalability (FGS) [20]. However, MPEG-4 FGS does not possess the property of temporally predictive coding of enhancement layers and it is shown to be lack of coding efficiency severely.

3 The Proposed Optimization Algorithm

As stated earlier, the optimization algorithm in [8] is designed around the sub-band scalable video coding which is assumed to produce a number of components with equal importance that can be decoded independently. For the state-of-the-art MPEG-4 AVC scalable video coding, the assumptions are not true in general.

In this section, we first describe our modification of the optimization algorithm in [8] to be able to adapt to the MPEG-4 AVC scalable extension and the refined algorithm is named flat FEC-distortion optimization algorithm (FFDO) as the comparison basis. Then, a new error control coding scheme for unequal error protection is proposed. An integration algorithm for H.264 scalable video coding, dynamic FEC-distortion optimization algorithm (DFDO), is further constructed. In this paper, the definition of the video component stands for the Network Abstraction Layer unit of the encoded frame of a scalable layer.

3.1 Flat FEC-Distortion Optimization Algorithm

The main concept of FFDO is to find a subset of message packets and parity packets s^* so that the distortion caused by the error prone channels can be minimized. The parity packets are encoded by the Reed Solomon erasure code and the BOP packetization scheme illustrated in the earlier section is adopted. In the case of video streaming applications, the objective is equivalent to the maximization of the video quality defined as the peak signal to noise ratio (PSNR) as illustrated in (1).

$$s^* = \arg \max_{s \in M} \text{ PSNR}(s, p), \tag{1}$$
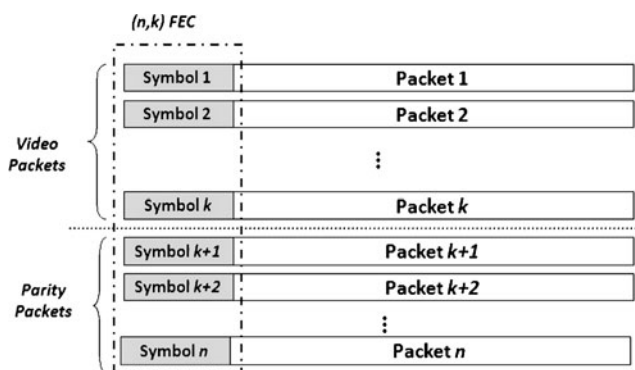
$$\text{s.t. } R(s) \leq B$$

Fig. 2 A block of packets.

where s is a subscription of scalable video layers and the corresponding parity packets for each video layer. M is the set of all possible subscriptions with the constraint that the required bitrates $R(s)$ of the subscription s is less than or equal to the available bandwidth B. The Network Abstraction Layer (NAL) units of Sequence Parameter Set and Picture Parameter Set [21]contain essential header information in order to decode video bit stream properly and thus they are assigned the strongest error correction code.

$PSNR(s,p)$ stands for the PSNR estimation of video quality of the subscription s in a GOP, given that the average packet loss rate is p. In contrast with the work in [8] where the PSNR of individual picture component is assumed to be equal and those components are also assumed to have the ability to be decoded independently, FFDO considers the cross relationship of inter-frame and inter-layer prediction and estimates the PSNR based on a GOP of video layers. $PSNR(s,p)$ can be expressed as the expectation of the video quality of possible received video layers as shown in (2).

$$PSNR(s, p) = \sum_{i=0}^{L-1} p_i \cdot PSNR_i, \tag{2}$$

where p_i is the FEC decodable probability of the first cumulative i layers (layer $0, 1, ..., i\text{-}1$) as shown in (3) and q_i in the equation is the FEC undecodable probability of layer i; $PSNR_i$ in (2) is the corresponding PSNR and L is the number of subscribed scalable video layers.

$$p_i = \begin{cases} q_i \prod\limits_{k=0}^{i-1} (1 - q_k) & if\ 0 \le i < L \\ \prod\limits_{k=0}^{L-1} (1 - q_k) & if\ i = L \end{cases}. \tag{3}$$

The FEC undecodable probability q_i of layer i is shown in (4) where the parameter pair (n, k) of the Reed Solomon code for layer i is $(M + k_i, M)$. p is the probability of packet error during the transmission.

$$q_i = \sum_{w=0}^{M-1} \binom{M + k_i}{w} (1 - p)^w p^{M+k_i-w}. \tag{4}$$

To reflect the quality degradation caused by the partially received and incompletely decoded video layers, the heuristic degradation factor α for FFDO is introduced into the calculation of $PSNR_i$ to better estimate the video quality, as shown in (5), where $PSNR'_i$ is the corresponding PSNR for only the first cumulative $i+1$ video layers (layer $0, 1, ..., i$) that are fully decoded.

The degradation factor α will be further discussed in the later section.

$$PSNR_i = PSNR'_{i-1} + \alpha \times \left(PSNR'_i - PSNR'_{i-1} \right)\ 0 \le \alpha < 1. \tag{5}$$

3.2 Dynamic FEC-Distortion Optimization Algorithm

FFDO described in the earlier section is based on the assumption that different frames in the same video layer exhibit constant quality distortion. However, this is usually not the case for the real H.264/MPEG-4 AVC Scalable videos. The quality distortion (or video quality, on the other hand) depends on the content of each video frame as well as the quantization parameter used in each macro block. Due to the error propagation effect that can result from not only the prediction coding across the video layers but also the temporal motion compensation coding in each individual video layer, the quality distortion caused by different frames of a video layer can also vary. As a result, it shall offer more incentive in terms of the improved PSNR performance for the global optimal bit allocation of the H.264 SVC and FEC to take this factor into consideration.

The main objective of the proposed Dynamic FEC-Distortion Optimization (DFDO) algorithm is to increase the decodable probability of important pictures when a video layer of a GOP cannot be completely received and decoded. The bit allocation process follows coarse-to-fine principle. At first, DFDO determines the best parameter configuration that indicates the number of video layers to be protected as well as the protection strength for each video layer within a GOP by the optimization process similar to FFDO, and then the second tier of the optimization processes of DFDO is applied to each subscribed layer to reallocate the protection packets within the video layer. DFDO classifies a video layer into a number of clusters which have different importance, and then the algorithm reallocates the protection packets to clusters according to their importance in order to minimize the quality distortion of whole layer (or to maximize the PSNR of that video layer).

For example shown in Fig. 3, the number of pictures in a GOP is 16 and the hierarchical prediction structure [11]
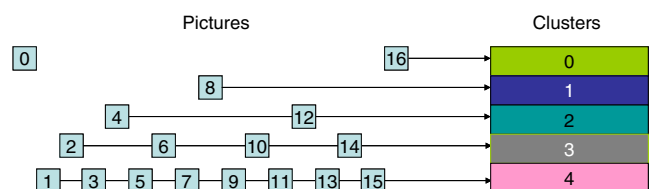

Fig. 3 An example of video classification according to its temporal scalability.

recommended by JVT for the temporal scalability is used due to its high coding efficiency.

The cluster of a video frame is classified by their temporal level. When the hierarchical prediction structure is applied, each video layer can be further decomposed into a number of temporal levels and the predictor of a video frame is formed by two reconstruction frames of lower temporal levels. Therefore, when the video frame of the lower temporal level cannot be reconstructed correctly, video frames of higher temporal levels will also suffer. In other words, video frames that belong to lower temporal levels are in general more important than those of higher temporal levels in terms of the contribution to the video quality.

The proposed unequal protection scheme of DFDO within a video layer forms a cascade of FEC codes with increasing protection coverage as shown in Fig. 4 where the blue segments are the clusters of the video bit stream and the green segments stand for the parity packets from the encoding of the source clusters indicated within the brackets. K_i represents the number of packets in video cluster i, and M_i denotes the number of parity packets for the video clusters (cluster 0, 1,…, to cluster i) by performing $(n, k)_i = \left(\sum_{k=0}^{i} (K_i + M_i), \sum_{k=0}^{i} K_i \right)$ FEC encoding of the Reed Solomon erasure code. The summation of M_i shall be the same as the number of parity packets allocated to this layer by Flat FEC-Distortion Optimization algorithm mentioned earlier. The more important a video cluster is, the more times it will be coded to produce the parity packets.

The objective of this stage is to find the best allocation patterns of parity symbols such that the video distortion of the whole video layer can be minimized (or the PSNR of the video layer can be maximized) as illustrated in (6).

$$s'' = \arg \max_{s \in M^*} PSNR^*(s, p). \tag{6}$$

Fig. 4 An example of DFDO cascading error protection scheme of a video layer.

0	K_0
1	K_1
2	K_2
3	K_3
4	K_4
(0,1,2,3,4)	M_4
(0,1,2,3)	M_3
(0,1,2)	M_2
(0,1)	M_1
(0)	M_0
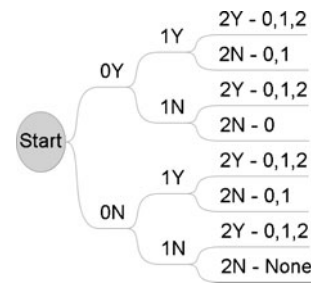
Fig. 5 A binary decoding tree for a 3-cluter video.

In (6), an allocation pattern s of parity packets is a vector where each element determines the number of protection symbols for the corresponding video clusters. M^* is the set of all possible allocation patterns. $PSNR^*(s,p)$ is the corresponding PSNR when the allocation pattern s is applied to a video layer over the channel with average packet loss rate p. The PSNR of the whole video layer can be estimated as the summation of PSNR contributed by each and every video cluster and it is illustrated in (7), where p_i^* represents the decodable probability of only the first i video clusters (video cluster 0 to i-1), $PSNR_i^*$ is the corresponding PSNR contribution, and C is the total number of video clusters.

$$PSNR^*(s,p) = \sum_{i=1}^{C} p_i^* \cdot PSNR_i^*. \tag{7}$$

To demonstrate the calculation of the decodable probability p_i^*, we construct a binary tree of a 3-cluster video as shown in Fig. 5. Each path that starts from the root node and ends at the leaf node forms a decoding path. An Y-edge denotes that the corresponding FEC coding is decodable, and an N-edge means that the corresponding FEC coding is not decodable. For example, Path 0Y1N2Y denotes the event that the first $(n, k)_o = \left(\sum_{k=0}^{0} (K_i + M_i), \sum_{k=0}^{0} K_i \right)$ code is decodable, followed by the undecodable $(n, k)_1$ code, and further followed by the decodable $(n, k)_2$ code. Since the $(n, k)_2$ code protects video clusters 0 to 2 at the same time, all the video clusters will be decoded correctly at the end of the path 0Y1N2Y. The decodable probability of the only first i video clusters, p_i^*, can be derived by such binary trees. In Fig. 5, p_3^* is the sum of the probabilities of the decoding paths 0Y1Y2Y, 0Y1N2Y, 0N1Y2Y, and 0N1N2Y. And for p_2^*, its value is the sum of the probabilities of decoding paths 0Y1Y2N and 0N1Y2N.

The decodable probability p_i^* can be expressed by (8) to (11). Equation (9) corresponds to the decodable probability of the paths which start from edge ID-Y and there are pvK packets recovered from the $(n, k)_0$ code to the $(n, k)_{ID-1}$ code. Equation (10) corresponds to the undecodable probability of the paths which start from edge ID-N and there are pvK packets recovered from the $(n, k)_0$ code to the

$(n, k)_{ID-1}$ code. Equation (11) determines the minimal number of packets that are required to decode video clusters 0 to ID. C is the number of video clusters in a video layer, p is the average packet loss rate, K_i is the number of packets in video cluster i, and M_i denotes the number of parity packets for the video clusters (cluster 0, 1,…, to cluster i) by performing the $(n, k)_i$ FEC encoding.

$$p_i^* = Y(0, i, 0) + N(0, i, 0). \tag{8}$$

$$Y(ID, t, pvK)$$
$$= \begin{cases} 0, if\ ID \geq t \\ \sum\limits_{i=0}^{K_{ID}} \sum\limits_{j=K(ID)-pvK-i}^{M_{ID}} \begin{matrix} C_i^{K_{ID}} C_j^{M_{ID}} p^{K_{ID}+M_{ID}-i-j}(1-p)^{i+j} \times \\ \begin{bmatrix} Y(ID+1, t, K(ID))+ \\ N(ID+1, t, K(ID)) \end{bmatrix} \end{matrix}, otherwise \end{cases} \tag{9}$$

$$N(ID, t, pvK) = \begin{cases} 0, if\ ID = t-1 \\ \sum\limits_{i=0}^{K_{ID}-1-pvK} \sum\limits_{j=0}^{K(ID)-1-pvK-i} \begin{matrix} C_i^{K_{ID}} C_j^{M_{ID}} p^{K_{ID}+M_{ID}-i-j}(1-p)^{i+j} \times \\ \begin{bmatrix} Y(ID+1, t, pvK+i)+ \\ N(ID+1, t, pvK+i) \end{bmatrix} \end{matrix}, otherwise \end{cases} \tag{10}$$

$$K(ID) = \sum_{i=0}^{ID} K_i. \tag{11}$$

The video quality measurement $PSNR_i^*$ in (7) of only the first i video clusters being fully decodable can be estimated by (12), where $PSNR_j^C$ is the summation of PSNR contributed by pictures in the video cluster j. The parameter β is a leaky factor to reflect the quality degradation caused by temporal prediction to an unavailable and error concealed picture. The exponential form of β is due to the hierarchical prediction structure over the temporal direction.

$$PSNR_i^* = \sum_{j=0}^{i-1} PSNR_j^C + \sum_{j=i}^{C-1} PSNR_j^C \times \beta^{j-i+1}, \ 0 \leq \beta < 1. \tag{12}$$

3.3 The Degradation Factors

The heuristic degradation factor α for FFDO is introduced into the calculation of $PSNR_i$ to reflect the quality degradation caused by the partially received and incompletely decoded video layers, as shown in (5). The variation tendency of the degradation factor α is studied through two

Table 1 Encoding parameters and the corresponding PSNR.

Layer	Spatial Resolution	Accumulated Bit-rate (kbs)	PSNR(dB)	
			mobile 30 fps	crew 30 fps
0	QCIF	200	21.51	29.60
1	QCIF	400	21.91	30.34
2	CIF	600	28.04	31.59
3	CIF	800	29.65	32.15
4	4CIF	1000	30.74	33.50

video sequences, *mobile* and *crew*, with the coding parameters and PSNRs listed in Table 1. A screenshot of the video sequences can be found in Fig. 10. The PSNR is calculated between the source video (4CIF) and the up-sampled reconstructed video layers.

For each given packet loss rate, a protected video stream is simulated over the lossy channels 100 times randomly in order to obtain the average PSNR for each degradation factor α. The results with best degradation factor α are shown in Fig. 6. It can be observed from the figure that α decreases as the average packet loss rate increases. This is reasonable since the portion of the partially received and incompletely decoded video layers shall increase with the packet loss rate. From Figs. 7 and 8, the degradation factor α is not very sensitive with respect to the different packet loss rates. In the following simulations, we will choose the median among the best degradation factors at different packet loss rates shown in Fig. 6.

The leaky factor β in DFDO to compensate the quality degradation caused by temporal prediction that refers to an unavailable and error concealed picture can be determined in a similar fashion. Two video sequences, *ice* and *city*
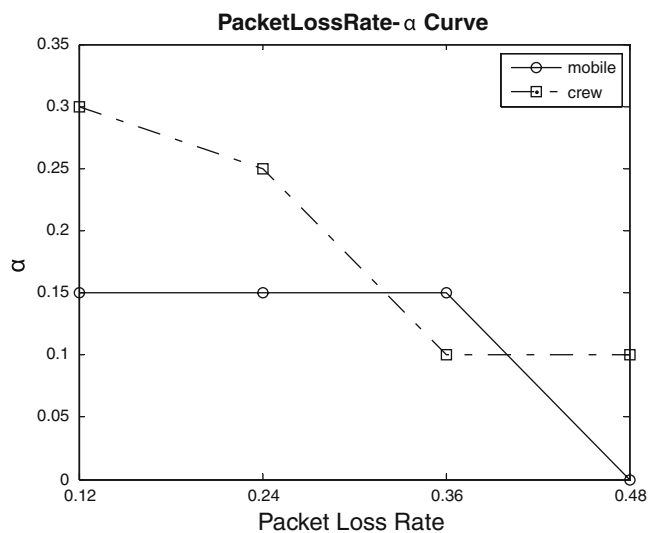
Fig. 6 Packet loss rate-α curve.

Fig. 7 α-PSNR curves of video "*mobile*" under different packet loss rates.
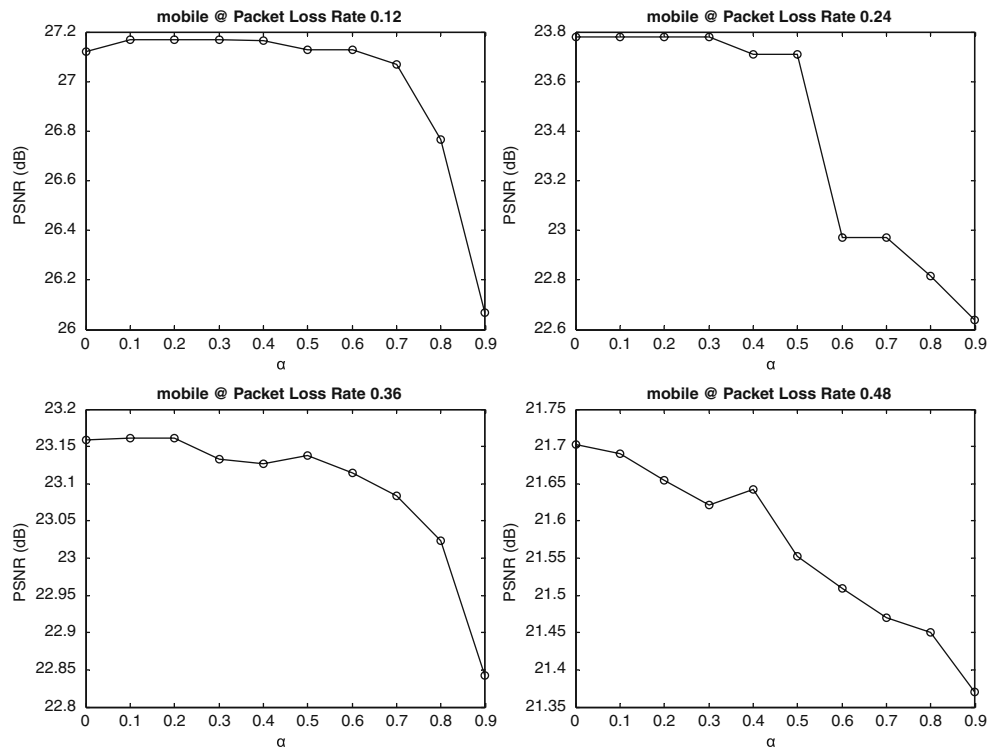

(with screenshot shown in Fig. 10), are encoded with the parameters listed in Table 2. Each video stream goes through 100 channels randomly with average packet loss rate 0.286. The results of the average PSNR are shown in Fig. 9 and it reveals that DFDO demonstrates stable and better performance when β is between 0.6 and 0.95.

4 Simulations

FFDO is extended from the optimization algorithm in [8] to remove the assumption about the equal importance of video components so that MPEG-4 AVC scalable video coding can be applied. DFDO is proposed to differentiate the

Fig. 8 α-PSNR curves of video "*crew*" under different packet loss rates.
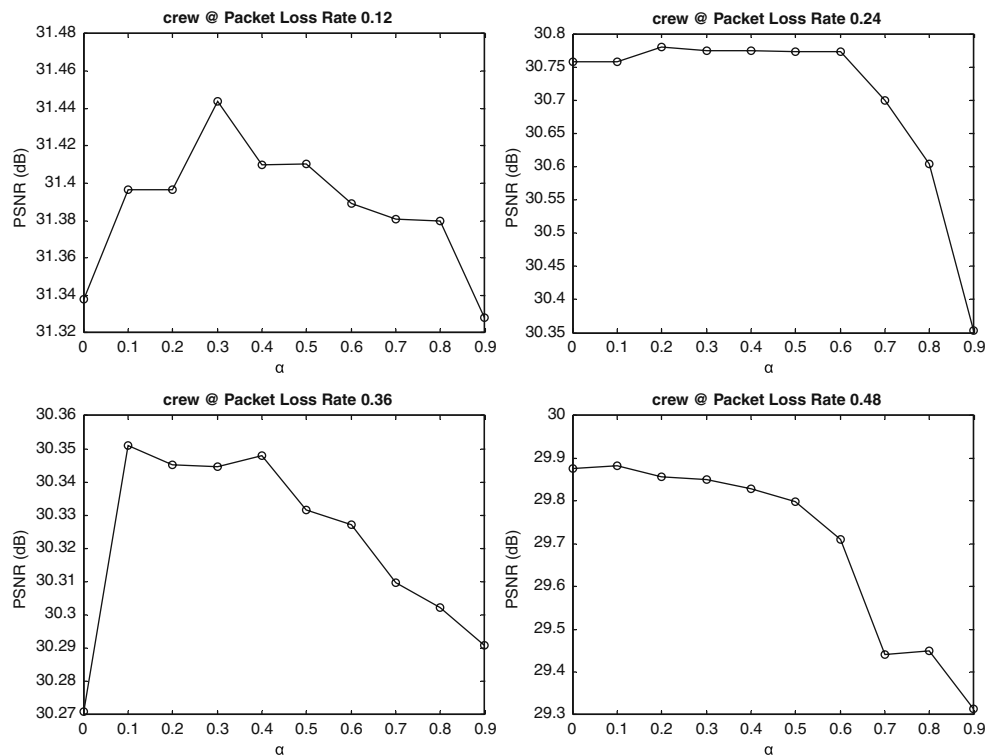
Table 2 Encoding parameters and the corresponding PSNR.

Layer	Spatial Resolution	Accumulated Bit-rate (kbs)	PSNR(dB)	
			ice 30 fps	city 30 fps
0	QCIF	200	30.14	25.07
1	QCIF	400	30.50	25.27
2	CIF	600	32.25	26.75
3	CIF	800	32.69	27.21
4	4CIF	1000	36.30	30.91

protection priorities of both inter-layer and inter-temporal prediction structures. In this section, a series of simulations are conducted to show the performance comparison of both algorithms.

4.1 Performance Comparison of Both Algorithms

Two video sequences, *football* and *soccer* shown in Fig. 10, are encoded by MPEG-4 AVC scalable video coding with parameters listed in Table 3 at 30 frames per second and the average packet loss rate of the error-prone channel is 0.2857. The degradation factor α in FFDO and DFDO is 0.15 and the leaky factor β in DFDO is 0.75, as described in the section earlier. The end-to-end bandwidth distributions over time (in terms of GOP index) are shown in Fig. 11 for the simulations of the video sequences "*football*" and "*soccer*", respectively.

The PSNR performances of both algorithms are shown in Fig. 12. For the convenience of comparison, the PSNR difference between DFDO and FFDO is shown in Fig. 13 where the positive values in Fig. 13 indicate the amount by that DFDO outperforms FFDO. The average PSNR for streaming the video sequence *football* with DFDO is 28.82 dB and it is 28.41 dB for FFDO. For the sequence *soccer*, the performance of DFDO is better than FFDO by 0.67 dB in average (8dB, maximum). From Fig. 13, it is obvious that DFDO performs especially well when the available bandwidth is not sufficient (between frames 65–97 and frames 193–225).
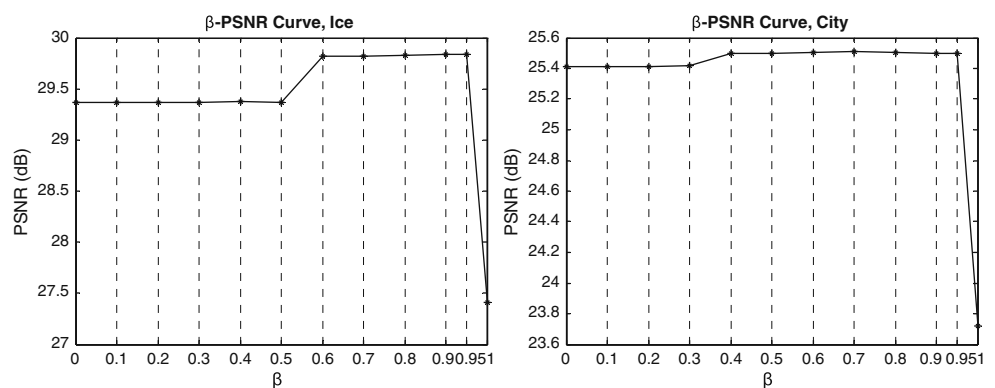
4.2 Performance at Higher Motion

To better examine the performance of the proposed algorithm when a video sequence contains contents of larger motion, frame rate conversion is carried out to down-sample the same video sequences, *football* and *soccer*, from 30 fps to 15 fps. Those two sequences are then encoded by MPEG-4 AVC scalable video coding with parameters listed in Table 3. The bandwidth distributions versus time in terms of GOP index for both sequences are shown in Fig. 14 and the difference of PSNR performance of FFDO and DFDO is shown in Fig. 15. In the case of *football* sequence, the difference is almost twice as compared with the case at 30 fps.

4.3 Misestimation of Packet Loss Rate

The average packet loss rate shall be estimated for the FFDO and DFDO to calculate the protection distribution as stated earlier. To investigate the behavior of the algorithms at the situation of misestimating the average packet loss rate, we fix the estimate of the packet loss rate to be 0.2857 while the real packet loss rate is between 0.24 and 0.32 for the video sequence *football* at 15 and 30 fps. The PSNR difference between DFDO and FFDO is shown in Fig. 16 which shows that the performance of DFDO exceeds the performance of FFDO by 0.25 to 0.75 dB in average even when the packet loss rate is misjudged.

4.4 Reallocating Parity Packets Without Cascading Protection Structure

In this section we present another comparison basis to show the effectiveness of the cascading error protection scheme in the proposed DFDO algorithm. The FFDO algorithm is modified so that after FFDO determines the number of video layers and also the corresponding parity packets, each video cluster described in Section 3.2 is encoded with the error control codes separately, instead of with the cascading error protection scheme. The mechanism of this modified

Fig. 9 β-PSNR curves for sequence "*ice*" (*left*) and "*city*" (*right*).

Fig. 10 Video sequences used in the simulations.

Sequence: *mobile* Sequence: *crew* Sequence: *city*
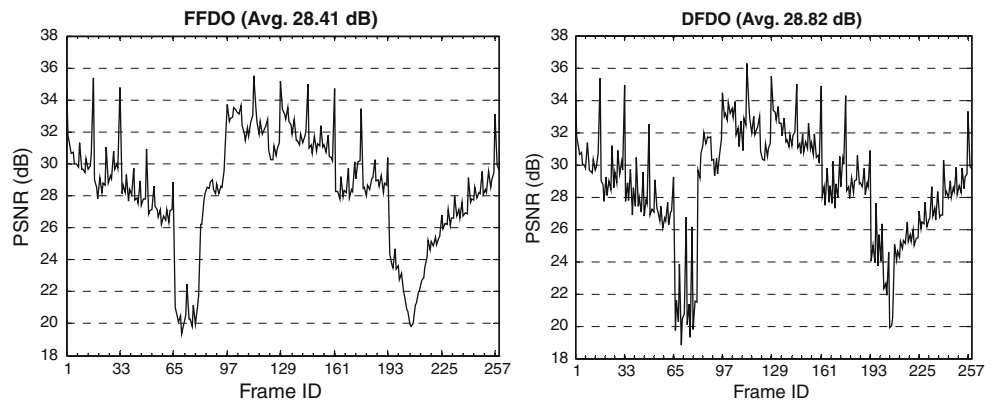
Sequence: *football* Sequence: *soccer* Sequence: *ice*

Table 3 Encoding parameters and the corresponding PSNR.

| Layer	Spatial Resolution	Accumulated Bit-rate (kbs)	PSNR(dB)			
			football 30 fps	soccer 30 fps	football 15 fps	soccer 15 fps
0	QCIF	200	27.54	28.32	28.43	28.68
1	QCIF	400	28.96	28.92	29.93	29.18
2	CIF	600	31.08	30.01	32.71	30.38
3	CIF	800	32.12	30.46	34.25	30.85
4	4CIF	1000	32.32	32.11	34.41	32.96

Fig. 11 Bandwidth distributions for football at 30 fps (*left*) and soccer at 30 fps (*right*).
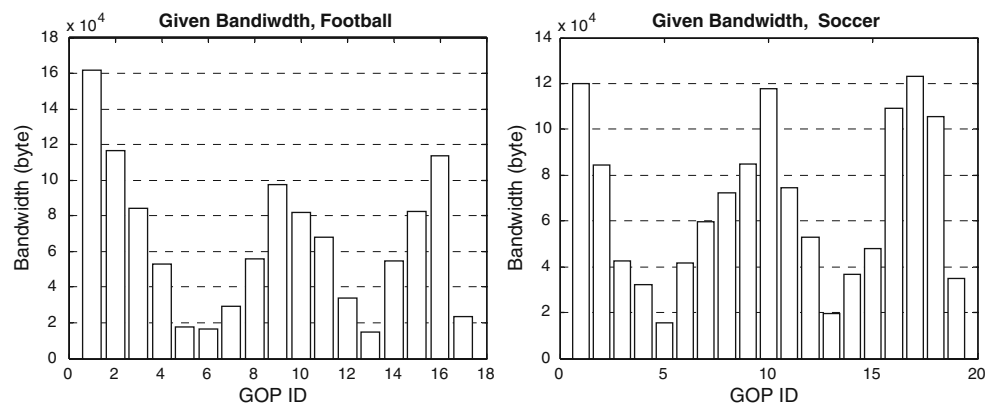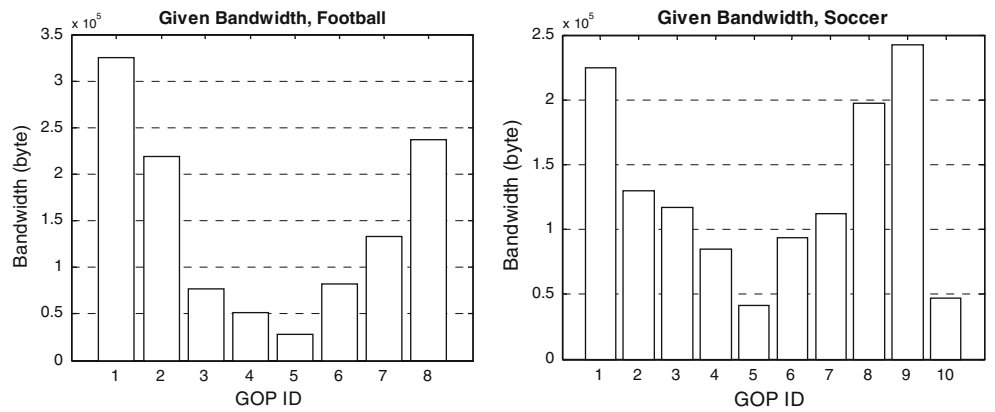
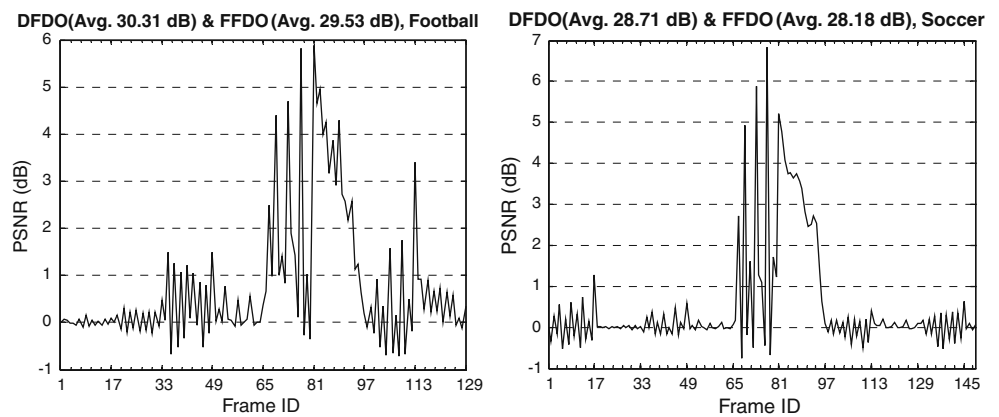**Fig. 12** PSNR performance of FFDO and DFDO for the video sequence *football*.

Fig. 13 PSNR difference of FFDO and DFDO for the video sequence *football* (left) and *soccer* (right) at 30 fps.
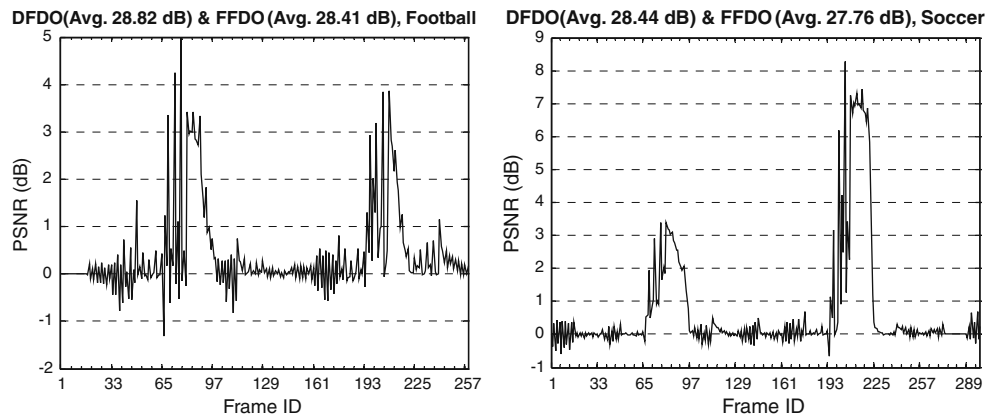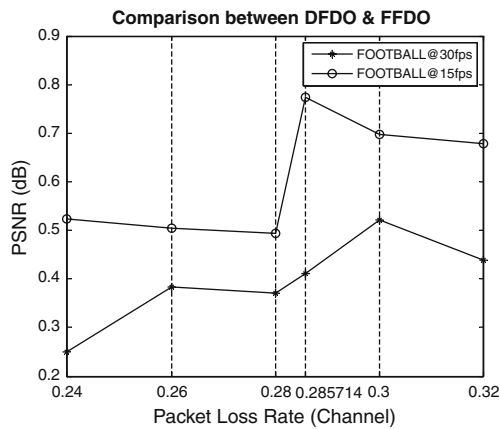

Fig. 14 Bandwidth distributions for football at 15 fps (left) and soccer at 15 fps (right).
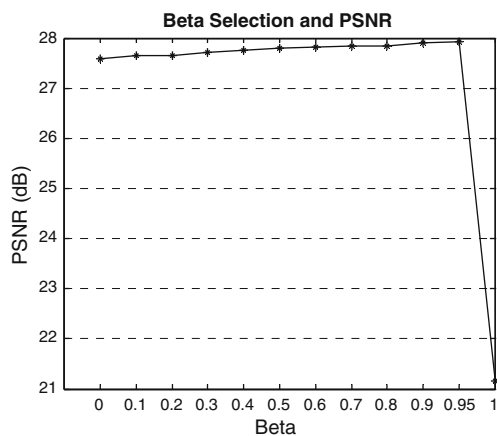

Fig. 15 PSNR difference of FFDO and DFDO for the video sequence *football* (left) and *soccer* (right) at 15 fps.

Fig. 16 PSNR difference of DFDO and FFDO for sequence *football* at 15 and 30 fps.

Fig. 17 β selection for the modified FFDO.

FFDO to find the best distribution pattern of the parity packets within the same video layer follows (6) and (7) where p^*_i can be obtained by (3) and (4) and the video quality measurement $PSNR^*_i$ is shown in (12).

Video sequence *ice* is used to redo the determination of the leaky factor β and the result is shown in Fig. 17. Video sequences *football* and *soccer* at 30 fps over the error-prone channel with average packet loss rate 0.2857 are protected by the original FFDO and the modified FFDO where β equals 0.95. The simulation results are shown in Fig. 18. The performance of the modified FFDO is actually worse than the original FFDO. The reason is that the video clusters that are encoded separately will reduce the coding efficiency of the error control coding. Thus, the proposed cascading error protection scheme can optimize the distribution of the unequal priorities across and within video layers better while maintaining the coding efficiency of the error control coding.

5 Conclusions

In a multimedia streaming system, the forward error correction code is a useful technique to facilitate reliable data transfer so that the retransmission of lost packets that may lead to unacceptable delay can be avoided at the cost of extra bandwidth. Thus, for a video streaming service with limited network resource over the error-prone channels, the quality of service and also the quality of experience will be enhanced if the network resource can be spent wisely.

In this paper, the flat FEC-distortion optimization (FFDO) algorithm is based on the modification of the work in [5] not only to be able to adapt to the scalable video streams encoded by the H.264/MPEG-4 AVC scalable extension standard but also to take account of the inter-layer prediction. Then the dynamic FEC-distortion optimization (DFDO) algorithm is proposed, which further improves FFDO so that the pictures within the same video layer of a GOP are coded according to their priorities by the cascading error protection scheme in terms of the contribu-

Fig. 18 Simulation results for *football* (left) and *soccer* (right).
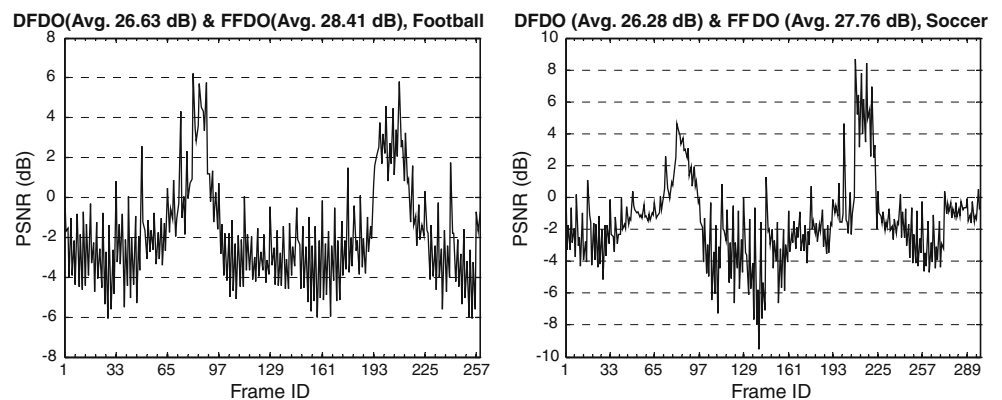
tion to the video quality. Thus, when a video layer of GOP cannot be completely received, the proposed algorithm will have more chance to recover important pictures.

The simulation results show that the average PSNR of DFDO outperforms FFDO, especially when the motion of the video is more significant. It is also indicated in the simulations that it is worth considering the optimization of the distribution of parity packets across and within video layers when the network resource is unfortunately limited. The estimate of the average packet loss rate is helpful to improve the streaming performance but it is not a sensitive parameter as demonstrated in the simulations for the DFDO algorithm.

References

 1. Reimers, U. H. (2006). DVB—The family of international standards for digital video broadcasting. *Proceedings of the IEEE, 94*, 173–182.
 2. Lee, C.-S. (2007). IPTV over next generation networks in ITU-T. in *2nd IEEE/IFIP International Workshop on Broadband Convergence Networks*, 1–18.
 3. Fairhurst, G. & Wood, L. (2002). RFC 3366: Advice to link designers on link Automatic Repeat reQuest (ARQ). *Internet Engineer Task Force.*
 4. Byers, J. W., Luby, M., & Mitzenmacher, M. (2002). A digital fountain approach to asynchronous reliable multicast. *IEEE Journal on Selected Areas in Communications, 20*, 1528–1540.
 5. Shokrollahi, A. (2006). Raptor codes. *IEEE Transactions on Information Theory, 52*, 2551–2567.
 6. Luby, M., Watson, M., Gasiba, T., Stockhammer, T., & Xu, W. (2006). Raptor codes for reliable download delivery in wireless broadcast systems. in *3rd IEEE Consumer Communications and Networking Conference*, 192–197.
 7. 3GPP (2005). Specification Text for Systematic Raptor Forward Error Correction. *3GPP TSG SA WG4 S4-AHP205.*
 8. Tan, W.-T., & Zakhor, A. (2001). Video multicast using layered FEC and scalable compression. *IEEE Transactions on Circuits and Systems for Video Technology, 11*, 373–386.
 9. Hsiao, H.-F., Chindapol, A., Ritcey, J. A., & Hwang, J.-N. (2005). Adaptive FEC Scheme For Layered Multimedia Streaming over Wired/Wireless Channels. in *IEEE 7th Workshop on Multimedia Signal Processing*. 1–4.
10. Kang, S.-R., & Loguinov, D. (2007). Modeling best-effort and FEC streaming of scalable video in lossy network channels. *IEEE/ACM Transactions on Networking, 15*, 187–200.
11. Schwarz, H., Marpe, D., & Wiegand, T. (2007). Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology, 17*, 1103–1120.
12. ITU-T (1996). Recommendation G. 114. Feb 6, 1996.
13. Luby, M., Vicisano, L., Gemmell, J., Rizzo, L., Handley, M., & Crowcroft, J. (2002). RFC 3452: Forward Error Correction (FEC) building block. *Internet Engineer Task Force,* December 2002.
14. Luby, M., Vicisano, L., Gemmell, J., Rizzo, L., Handley, M., & Crowcroft, J. (2002). RFC 3453: The Use of Forward Error Correction (FEC) in reliable multicast. *Internet Engineer Task Force,* December 2002.
15. Mohamed, H., Luigi, A., & Marwan, K. (2004). Video transport over wireless channels: a cycle-based approach for rate control. in *Proceedings of the 12th annual ACM international conference on Multimedia* New York, NY, USA: ACM, 2004.
16. Zhang, Q., Wang, G., Xiong, Z., Zhou, J., & Zhu, W. (2004). Error robust scalable audio streaming over wireless IP Networks. *IEEE Transactions on Multimedia,* December 2004.
17. Lacan, J., & Fimes, J. (2004). Systematic MDS erasure codes based on vandermonde matrices. *IEEE Communications Letters, 8*, 570–572.
18. Rizzo, L. (1997). Effective erasure codes for reliable computer communication protocols. *SIGCOMM Comput Commun Rev, 27*, 24–36.
19. Horn, U., Stuhlmüller, K., Link, M., & Girod, B. (1999). Robust Internet video transmission based on scalable coding and unequal error protection. *Signal Processing-Image Communication, 15*, 77–94.
20. Li, W. (2001). Overview of fine granularity scalability in MPEG-4 video standard. *IEEE Transactions on Circuits and Systems for Video Technology, 11*, 301–317.
21. Advanced Video Coding for Generic Audiovisual Services. (2005). *ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG4-AVC),* Version 1: May 2003, Version 2: Jan. 2004, Version 3: Sep. 2004, Version 4: July 2005.

Wei-Chung Wen received the B.S. degree and the M.S. degree both in computer science from the National Chiao Tung University, HsinChu, Taiwan, R.O.C. in 2005 and 2008, respectively. Since 2009, he has worked for Skytel Technology Co., Kaohsiung, Taiwan.

Hsu-Feng Hsiao received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, R.O.C. in 1995, the M.S. degree in electrical engineering from the National Chiao Tung University, Hsinchu, Taiwan, R.O.C. in 1997, and the

Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, USA in 2005.

He was an engineering officer in the Communication Research Laboratory of the Ministry of National Defense, Taiwan from 1997 to 1999. From 2000 to 2001, he was a software engineer at HomeMeeting, Redmond, WA. He had been then a Research Assistant in the department of Electrical Engineering, University of Washington till 2005. Dr. Hsiao has been an Assistant Professor in the department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan since 2005. His research interests include multimedia signal processing and wired/wireless communications.