Optimal FEC code concatenation for unequal error protection in video streaming applications

Lukasz Kondrad[a], Imed Bouazizi[b], Vinod Kumar Malamal Vadakital[b], and Moncef Gabbouj[a]

[a]Tampere University of Technology, Tampere, Finland;
[b]Nokia Research Center, Tampere, Finland

ABSTRACT

In this paper, we propose a novel algorithm for constructing an Unequal Error Protection (UEP) FEC code targeted towards video streaming applications. A concatenation of a set of parallel outer block codes followed by a packet interleaver and an inner block code is presented. The algorithm calculates on the fly the optimal allocation of the code rates of the inner and outer codes. When applied to video streaming applications using H.264, the discussed UEP framework achieves gains of up to 5dB in video quality compared to equal error protection (EEP) FEC at the same code rate.

Keywords: FEC, channel coding, video streaming, H.264

1. INTRODUCTION

Multimedia streaming to mobile handsets is a topic of fervent research, both in academia and industry. International standard bodies, such as the Digital Video Broadcasting Organization (DVB) and the Third Generation Partnership Project (3GPP), have specified open standards for delivery of multimedia services to mobile terminals. DVB's Digital Video Broadcasting - Handheld (DVB-H),[1] and 3GPP's Multimedia Broadcast Multicast Service (MBMS)[2] and Packet-switched Streaming Service (PSS)[3] are some of those open standards.

Mobile wireless channels are prone to bursty time-varying errors due to radio channel characteristics. Compressed video, an important part of multimedia data, is sensitive to those errors due to the extensive use of predictive coding during compression. Therefore, to enable robust transmission of multimedia data, channel coding with forward error correcting (FEC) codes are frequently used. Common examples of FEC codes are Reed-Solomon (RS) codes[4] (used e.g. in DVB-H) and Raptor codes[5] (used e.g. in MBMS).

Channel coding solutions are typically media un-aware. Consequently, the same level of protection is applied to all bits/packets (assuming equal size packets) of a source block. On the other hand, compressed video data is composed of data parts that are of unequal importance to the decoding and presentation of the video. In these cases, unequal error protection (UEP) has the potential to provide improved rate-distortion performance.

Based on the literature study presented in Section 2, we acknowledge that application layer UEP schemes are mainly focused on how to assign priorities to data and algorithms to split protection between levels of importance. The problem of how to assign priorities is a valid question but will not be addressed in this paper. On the other hand, we focus on what is the optimal way to split available protection between data. We investigate a new approach to use concatenated codes for UEP, where an outer code works as a traditional UEP code (divide data to different levels of protection) and an inner code with EEP, which gives additional protection for all the packets produced by outer code. We propose an algorithm for an optimal split of repair data between outer codes and an inner code and between levels of protection in outer codes for an estimated packet loss rate.

This paper is organized as follows. Section 2 gives some background on the original UEP framework, distortion optimisation and concatenated codes. Section 3 describes in detail the proposed algorithm. In Section 4, simulations comparing the proposed UEP algorithm to the basic EEP algorithm are presented. The results serve to illustrate that the proposed UEP framework effectively provides graceful degradation. Finally, Section 5 contains our conclusions and future work.

Email addresses: lukasz.kondrad@tut.fi, imed.bouazizi@nokia.com, vinod.malamal-vadakital@nokia.com, moncef.gabbouj@tut.fi.

2. RELATED WORK

Many contributions have been made in the literature to address the issue of robustness against packet loss in multimedia data transmission. The pioneering work by Albanese et al[6] introduces media aware packet handling in the so-called Priority Encoding Transmission scheme (PET). PET was one of the first packet based data protection schemes, which operates by discarding less important media packets while retaining the more important ones when the transmission channel degrades. The PET scheme does not give methods to estimate the media packet priorities but instead assumes pre-allocated priorities. However, priority assignment to media packets is critical for any UEP scheme. An inaccurate priority assignment may lead to a detrimental impact on the efficiency of the UEP algorithm. Therefore, numerous priority assignment schemes were researched and their results reported.

Leicher[7] proposed a simple coarse three-level classification scheme. The priority specifies the percentage of packets necessary to recover the message fragment contained in the current packet. Applied to MPEG-2 compressed video, this reflects the fact that I-frames are required to decode P-frames and that B-frames are dependent on I- and P-frames.

Another approach for media data partitioning was proposed by Boyce[8] as the High Priority Partitioning (HiPP) scheme, which uses the data partitioning tool in MPEG-2 to provide unequal loss protection for MPEG-2 compressed video. In MPEG-2 data partitioning the stream data is split into high and low priority partitions. The critical data, such as headers, motion vectors, and low frequency discrete cosine transform (DCT) coefficients, are placed in the high priority partition, while the remaining high frequency coefficients are placed in the low priority partition. A priority breakpoint specifies a cutoff that decides whether the coefficients should go into the high or low priority partition.

In,[9] another unequal loss protection (ULP) framework, which applies unequal amounts of forward error correction (FEC) to embedded data to provide graceful degradation of picture quality in the presence of increasing packet loss was presented. In 2005 Goshi et al[10] applied this ULP framework to H.263 compressed video streams, investigating re-orderings of the bitstream in order to transform it into an embedded bitstream. An embedded bitstream is arranged in a descending order of importance to the decoding process.

Further work in this area was done by Horn et al[11] . The authors propose a scheme similar to PET with details on the practical implementation and application to spatially scalable video codec.

UEP application to video streaming is not a new topic and many studies have been conducted since Albanese's work. In spite of that, many researchers still work in the area and recent contributions can be found in literature, e.g.[12] or.[13] It can be related to special compression methods,[14] scalable video streaming,[15] or dedicated to concrete applications.[16]

The UEP schemes presented above mainly aim at determining how to assign priorities to the data in order to achieve graceful degradation. In this paper we propose a novel approach of splitting protection data, we examine usage of concatenated FEC codes for UEP. The coding technique where a data stream is encoded with an outer code, and subsequently, the coded data is further encoded with an inner code to further improve transmission performance has been known for many years and was first proposed by Forney.[17] Though the deployment of concatenated codes is most common in physical layer, in literature use of the concatenated codes on other layers can be found. In,[18] for instance, the authors proposed concatenated codes for robust packet video transmission where a Reed-Solomon outer code was used to protect against packet losses and a rate-compatible punctured convolutional (RCPC) code as inner code for protection against bit errors.

3. FEC CODE CONSTRUCTION ALGORITHM

In order to achieve a good error protection performance an accurate estimate of the channel conditions and the perceived residual packet loss rate after FEC decoding is critical. We present an analysis of the expected residual packet loss rate and the resulting distortion starting from an estimation of the channel packet loss rate. We then develop an algorithm for constructing a concatenated FEC code to achieve higher protection for important media packets, and therefore, improved average quality at the receiver.

The media data is treated as FEC source blocks. A FEC source block is a set of media packets to be delivered over a given channel with a given level of protection. The level of protection is expressed by the FEC code rate which is used to deduce the amount of repair data that is allowed for a given source block. Based on the importance of each media packet for the decoding process (and hence also for the overall quality), the source block is partitioned into m sub-blocks. Each sub-block is then assigned an error protection level depending on its priority. The optimal number of sub-blocks m is determined by the algorithm.

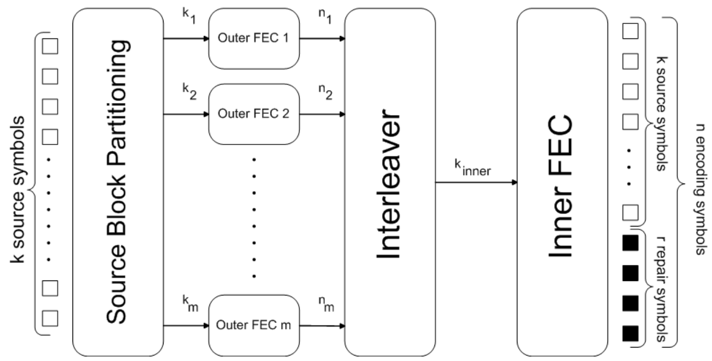


Figure 1. FEC code concatenation for UEP

In the algorithm, we assume that packet loss over the communication channel is an i.i.d process with a packet loss probability p for all packets (independent of their size). However, wireless channel errors tend to occur in bursts. To overcome this problem, interleaving has been employed to re-distribute the bursty errors so they become random errors. The FEC concatenation scheme with the interleaver is presented in Figure 1. The algorithm used for interleaving packets from all sub-blocks is presented below. This algorithm assumes that the number of packets in each sub-block follows the inequality $n_1 \geq n_2 \geq ... \geq n_m$, where n_i represents the number of packets in the i^{th} sub-block.

1. set $i = 1$
2. set $number\_of\_symbols\_interleaved = n_1$
3. set $i = i + 1$
4. calculate $put\_every\_x\_symbols = floor(number\_of\_symbols\_interleaved/n_i)$
5. calculate $how\_many\_symbols\_to\_put = ceil(n_i/number\_of\_symbols\_interleaved)$
6. set $number\_of\_symbols\_interleaved = number\_of\_symbols\_interleaved + n_i$
7. if $i < m$ go to step 3

In a system where an equal error protection (n,k) maximum distance separable (MDS) block code such as Reed-Solomon is in use, the residual error rate P_{EEP} is calculated as follows:

$$P_{EEP} = \sum_{l=n-k+1}^{n} \frac{l}{n} \times P\left(l|p\right) \quad (1)$$

$$where$$

$$P\left(l|p\right) = \binom{n}{l} \left(p\right)^{l} \left(1-p\right)^{n-l}$$

The result may be interpreted as the probability that a given packet is lost along with at least $n-k-1$ other packets, which would mean that it is not recoverable using the EEP FEC.

The objective of the algorithm is to optimally partition the source block and allocate the share of repair data to each of the resulting sub-blocks as well as calculate optimal split of repair data between outer codes and the inner code. Let k_i be the number of packets in partition $i|i \in \{1..m\}$. Let Pri_i represent the priority of partition i.

The resulting concatenated code minimizes the overall expected distortion D_{total} as defined by the following equation:

$$D_{total} = \sum_{i=1}^{m} Pri_i \times P_{loss}(i) \tag{2}$$

$P_{loss}(i)$ represents the residual packet loss rate for packets that belong to partition i. $P_{loss}(i)$ depends on the code rate of the outer code of partition i, $\frac{k_i}{n_i}$, and of the code rate of the common inner code $\frac{\sum_{i=1}^{m} n_i}{n}$.

The residual packet loss rate $P_{loss}(i)$ for partition class i is calculated according to the following equations:

$$P_{loss}(i) = \sum_{l=n_i-k_i+1}^{n_i} \frac{l}{n_i} \times P\left(l|P_{EEP_{inner}}\right) \tag{3}$$

$$where$$

$$P_{EEP_{inner}} = \sum_{l=n-k_{inner}+1}^{n} \frac{l}{n} \times P(l|p)$$

$$and$$

$$k_{inner} = \sum_{i=1}^{m} n_i$$

Based on the previous residual packet loss calculations, we propose the following low-complexity heuristical algorithm for exploring the optimal code construction.

1. set $m = 1$ and $P_{total_{min}} = \infty$
2. set $k_i = \frac{k}{m} \forall i \in 1..m$
3. set Pri_i as the average priority of all packets of partition i
4. set $k_{inner} = k + 1$
5. allocate n_i for each partition i so that $\sum_{i=1}^{m} n_i = k_{inner}$ and n_i proportional to Pri_i
6. calcualte $P_{loss}(i) \forall i \in 1..m$
7. calcualte $D_{total} = \sum_{i=1}^{m} Pri_i \times P_{loss}(i)$
8. store D_{total} if smaller than $D_{total_{min}}$
9. set $k_{inner} \leftarrow k_{inner} + 1$
10. if $k_{inner} = n$ set $m \leftarrow m + 1$ and goto step 2
11. else goto step 5

The algorithm checks all practical constructions from a single partition to m_{max} partitions. For each construction, the amount of repair data assigned to each partition is proportional to the average priority of that partition. Finally, the estimated overall distortion for the construction is calculated and the construction is marked as interesting if the resulting distortion is below all prior constructions. As a result, the construction with the minimal distortion expectation is determined. This algorithm is of low complexity and may run in real-time scenarios for each source block without significant additional delay.

4. EVALUATION

Simulations were conducted to compare the proposed UEP scheme to conventional linear block code EEP. Four widely used video test sequences were used: Glasgow@QCIF, News@CIF, Costguard@CIF, and Salesman@CIF. All sequences contained 300 frames and were coded using baseline profile of H.264/AVC. The bitrate was set to 384 kbps with the maximum slice size set to 1200 bytes. Intra Decoder Refresh frames were inserted every 30 pictures and one non-reference 'p' picture was inserted between any two reference pictures. For simplicity the H.264 decoder used a simple copy error concealment technique, where a lost region is replaced by co-located data from the last correctly reconstructed frame. When an entire picture is lost, a copy of the previous picture in presentation order is used as concealment for the lost picture.

For accurate evaluation of the algorithm, we assume that priority of the packet is proportional to the distortion it causes when the packet is lost. In order to estimate the distortion caused by a packet loss, a-priori simulations were conducted to measure the difference in quality between correctly received sequence and sequence with a single packet loss affecting the packet of which the priority is to be determined. If the packet is received correctly and in time, the distortion measured will reflect the encoding distortion. If, however, the packet is lost, the decoder will conceal the lost region of the picture using older data. As a measure for the video quality, we consider using the widely accepted peak signal-to-noise ratio (PSNR).

To reflect the bursty nature of errors on wireless channel a Gilbert-Eliot model was used for the simulations. Each sequence was tested using seven different error patterns, with average packet error rates equal 1%, 5%, 10%, 15%, 20%, 25%, and 30% and average burst error length equal 5 for each error pattern.

For comparative statistics, simulations using EEP were also carried out under exactly the same channel conditions. The algorithm presented in Section 3 was used for generating unequal error protection for a specified error rate p.

The graphs plotted in Figures 2, 3, 4, and 5 show the distortion for the simulated Glasgow, News, Salesman, and Coastguard sequences respectively. Each sub-figure represents results for different $\frac{k}{n}$ code rate: a) $\frac{2}{3}$, b) $\frac{3}{4}$, c) $\frac{5}{6}$. These results reflect the case where the actual channel loss rate is equal to the estimated packet loss rate used during the FEC code construction. As such, this applies to unicast streaming applications, where the sender receives actual feedback about the channel conditions. The results show that the use of the proposed UEP algorithm outperforms the EEP based approach especially at higher error rates.

Since in wireless channels the channel error rate is not constant and varies with time, location, and mobility pattern of the receiver, the UEP algorithm was also evaluated for actual packet loss rates different from the estimated packet loss rate used by the algorithm. This also covers the case of video multicast applications, where the terminal population experiences a wide range of packet loss rates and the algorithm cannot address the channel conditions experienced by all receivers. In Tables 1, 2, 3, and 4 the difference between UEP and EEP PSNR values is presented for the tested sequences. The columns correspond to the estimated channel error rates used by the algorithm while rows correspond to the actual packet loss rates used in the simulations.

Table 1. Glasgow code rate 5/6

	1%	5%	10%	15%	20%	25%	30%
1%	0.00	0.00	0.00	-0.10	-0.10	-0.16	-0.16
5%	0.00	-0.17	-0.34	-0.19	0.18	-0.43	0.02
10%	-0.35	-0.37	-0.20	1.14	0.70	-0.53	0.66
15%	-0.35	-0.36	-0.29	2.89	1.54	1.02	0.96
20%	0.00	-0.03	-0.67	3.88	3.22	3.18	2.16
25%	0.00	-1.07	-0.62	4.56	4.00	5.46	3.73
30%	0.00	-0.34	-0.24	4.60	3.67	5.37	5.21

Table 2. News code rate 5/6

	1%	5%	10%	15%	20%	25%	30%
1%	0.00	0.00	0.00	-0.07	-0.13	-0.08	-0.14
5%	-0.18	0.00	0.00	0.32	0.00	-0.98	-0.37
10%	0.00	-0.39	-0.22	1.26	0.41	-0.04	-0.01
15%	-0.22	-0.26	-0.86	1.25	1.99	1.41	1.00
20%	0.00	-0.43	-1.17	3.34	3.70	2.45	3.35
25%	0.00	-0.79	-0.99	5.04	4.80	3.85	4.14
30%	0.00	0.00	-0.18	5.00	4.04	4.74	2.98

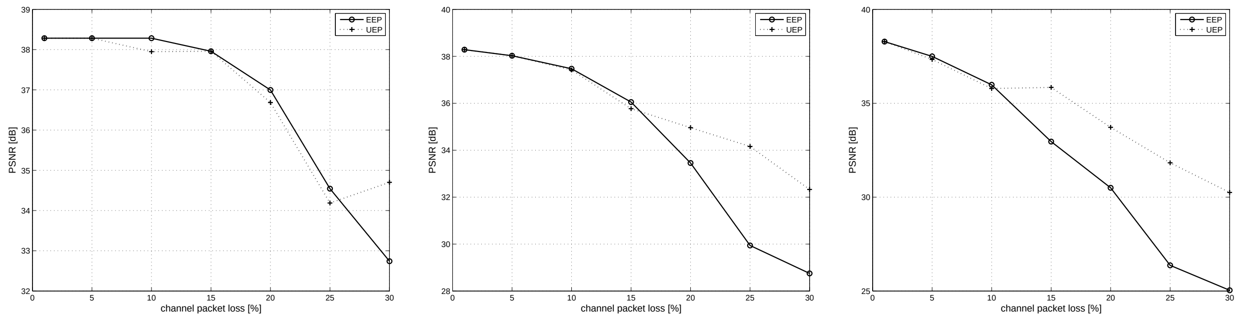
Table 3. Salesman code rate 5/6

	1%	5%	10%	15%	20%	25%	30%
1%	0.00	0.00	0.00	-0.03	-0.03	-0.04	-0.03
5%	0.00	-0.08	-0.14	0.14	0.19	-0.07	-0.14
10%	-0.09	-0.20	-0.27	0.66	0.43	0.37	0.23
15%	-0.12	-0.13	-0.52	1.37	1.21	1.02	0.92
20%	0.00	-0.22	-0.37	2.18	1.89	1.77	1.89
25%	0.00	-0.21	-0.51	3.10	3.25	2.94	3.39
30%	-0.04	0.00	-0.40	2.18	3.33	3.65	3.05

Table 4. Coastguard code rate 5/6

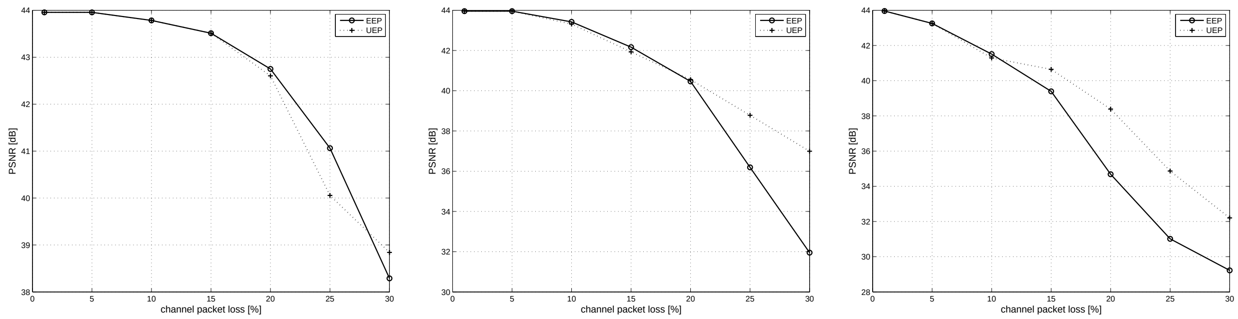
	1%	5%	10%	15%	20%	25%	30%
1%	0.00	0.00	0.00	-0.12	-0.09	-0.06	-0.08
5%	0.00	0.00	-0.20	-0.13	-0.45	-0.33	-0.61
10%	0.00	-0.35	-0.34	-0.12	0.25	-0.01	0.10
15%	0.00	-0.23	-0.69	1.73	1.54	0.90	0.80
20%	0.00	-0.25	-0.45	1.62	1.72	1.79	1.89
25%	0.00	-0.09	-0.09	3.43	1.91	3.18	1.95
30%	0.00	-0.23	-0.29	2.81	2.11	2.73	2.55

The results show that the penalty of wrong estimation of the actual channel packet loss rate is not significant (it amounts to a maximum of -1.17db, table 2). On the other hand, the lower right corner of each table exhibits significant improvements in the video quality against EEP for relatively high error rates and medium accuracy of the channel loss rate estimation. Hence, in a broadcast streaming application, the system can be configured to use a relatively high estimate of the packet loss rate. This will benefit users at the edge of broadcasting cell with relatively poor channel conditions, while not impacting receivers with good channel conditions.



(a) $\frac{2}{3}$ (b) $\frac{3}{4}$ (c) $\frac{5}{6}$

Figure 2. Video quality for Glasgow sequence at code rates (a), (b) and (c)

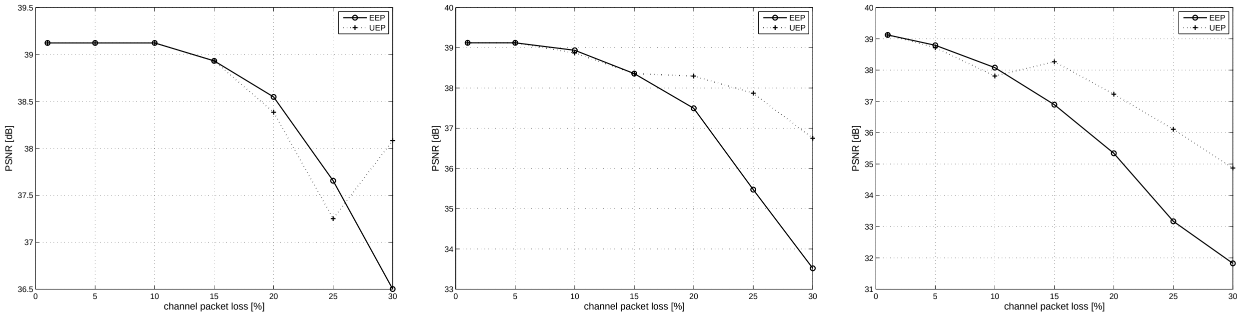


(a) $\frac{2}{3}$ (b) $\frac{3}{4}$ (c) $\frac{5}{6}$

Figure 3. Video quality for News sequence at code rates (a), (b) and (c)

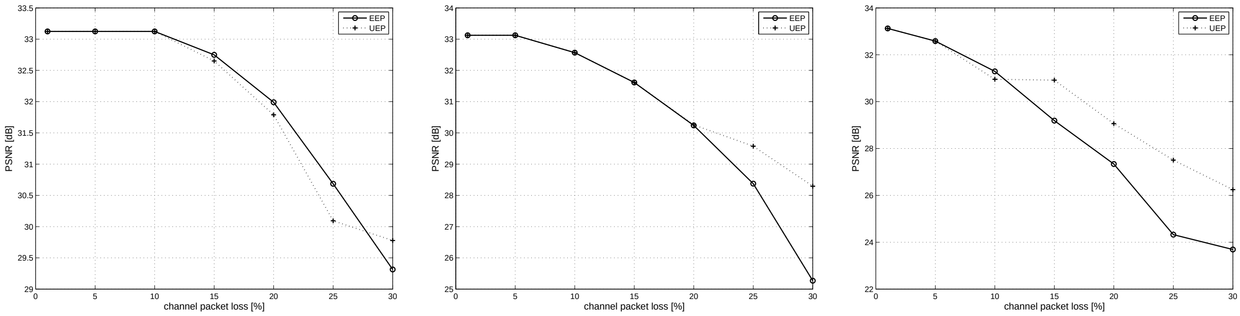
5. CONCLUSION

This paper described an algorithm for constructing a concatenated forward error correction (FEC) code. The algorithm calculates the optimal split of repair data between outer codes and the inner code for any given estimate of packet loss rate. It achieves a good protection against packet losses by exploiting the knowledge of the media data priorities. Simulation results, using a prior priority assignment for an H.264/AVC encoded bit stream, show significant improvements of the perceived video quality against equal error protection (EEP) approaches. Both unicast as well as multicast/broadcast streaming applications were addressed in the evaluation of the algorithm. Recommendations for the configuration of the algorithm for the multicast/broadcast scenario were also provided



(a) $\frac{2}{3}$ (b) $\frac{3}{4}$ (c) $\frac{5}{6}$

Figure 4. Video quality for Salesman sequence at code rates (a), (b) and (c)



(a) $\frac{2}{3}$ (b) $\frac{3}{4}$ (c) $\frac{5}{6}$

Figure 5. Video quality for Coastguard sequence at code rates (a), (b) and (c)

Though, the results presented in the paper are quite promising, there are still some open questions which should be considered in future work. First, the complexity of assigning priorities to media units is not yet fully solved. In this paper, this procedure has been simplified by running a pre-computation phase that must be run on each video sequence to be transmitted. This turns out to be a significant limiting factor to the applicability of the proposed algorithm in live multimedia streaming. However, in a use case where stored multimedia content is streamed, pre-computation results could be stored as side information. We are also aware of the fact that PSNR does not necessarily reflect the perceived quality of video. Therefore to exactly evaluate the algorithm, subjective quality test may also need to be considered. Furthermore, an exhaustive evaluation of the proposed method with other unequal error protection (UEP) methods should be conducted.

REFERENCES

[1] EN 302 304 V1.1.1, "Digital Video Broadcasting (DVB): Transmission systems for handheld terminals."

[2] 3GPP TS 26.346, "Multimedia Broadcast/Multicats Service (MBMS); Protocols and codecs (Release 7)."

[3] 3GPP TS 26.234, "Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs(Release 7)."

[4] Rizzo, L., "Effective erasure codes for reliable computer communication protocols," *ACM Computer Communication Review* **27**, 24–36 (April 1997).

[5] Shokrollahi, A., "Raptor codes," *IEEE Transactions on Information Theory* **52**, 2551–2567 (June 2006).

[6] Albanese, A., Blomer, J., Edmonds, J., Luby, M., and Sudan, M., "Priority encoding transmission," in [*IEEE Symposium on Foundations of Computer Science*], 604–612 (November 1994).

[7] Leicher, C., "Hierarchical encoding of MPEG sequences using priority encoding transmission (PET)," Tech. Rep. TR-94-058, Berkeley, CA (November 1994).

[8] Boyce, J. M., "Packet loss resilient transmission of MPEG video over the internet," in [*Signal Processing: Image Communication*], 7–24 (September 1999).

[9] Mohr, A., Riskin, E., and Ladner, R., "Unequal loss protection: Graceful degradation of image quality over packet erasure channels through forward error correction," (1999).

[10] Goshi, J., Mohr, A. E., and Ladner, R. E., "Unequal loss protection for H.263 compressed video," *IEEE Transaction on Circuits and Systems for Video Technology* **15**, 412–419 (March 2005).

[11] Horn, U., Stuhlmuller, K., Link, M., and Girod, B., "Robust internet video transmission based on scalable coding and unequal error protection," *Image Communication* **15**, 77–94 (September 1999).

[12] Nafaa, A., Ahmed, T., and Mehaoua, A., "Unequal and interleaved fec protocol for robust mpeg-4 multicasting over wireless lans," *IEEE International Conference on Communications* **3**, 1431–1435 (June 2004).

[13] Qu, Q., Pei, Y., and Modestino, J. W., "An adaptive motion-based unequal error protection approach for real-time video transport over wireless IP networks," *IEEE Transactions on Multimedia* **8**, 1033–1044 (October 2006).

[14] Cheng, L., Zhang, W., , and Chen, L., "Rate-distortion optimized unequal loss protection for fgs compressed video," *IEEE Transactions on Broadcasting* **50**, 126–131 (June 2004).

[15] Shi, Y., Wu, C., and Du, J., "A novel unequal loss protection approach for scalable video streaming over wireless," *IEEE Transactions on Consumer Electronics* **53**, 363–368 (May 2007).

[16] Vadakital, V. K. M., Hannuksela, M. M., Rezaei, M., and Gabbouj, M., "Method for unequal error protection in DVB-H for mobile television," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications* , 1–5 (September 2006).

[17] Forney, G. D., "Concatenated codes," *M.I.T Press* (1966).

[18] Pei, Y. and Modestino, J., "Use of concatenated FEC coding for real-time packet video over heterogeneous wired-to-wireless IP networks," *International Symposium on Circuits and Systems* **2**, 840–843 (May 2003).